Sketch4Match – Content-based Image Retrieval System Using Sketches

B. Szántó, P. Pozsegovics, Z. Vámossy, Sz. Sergyán
Óbuda University/Institute of Software Technology, Budapest, Hungary
{szanto90balazs, pozsee1st}@gmail.com, {vamossy.zoltan, sergyan.szabolcs}@nik.uni-obuda.hu

***Abstract*— The content based image retrieval (CBIR) is one of the most popular, rising research areas of the digital image processing. Most of the available image search tools, such as Google Images and Yahoo! Image search, are based on textual annotation of images. In these tools, images are manually annotated with keywords and then retrieved using text-based search methods. The performances of these systems are not satisfactory. The goal of CBIR is to extract visual content of an image automatically, like color, texture, or shape.**

This paper aims to introduce the problems and challenges concerned with the design and the creation of CBIR systems, which is based on a free hand sketch (Sketch based image retrieval – SBIR). With the help of the existing methods, describe a possible solution how to design and implement a task spesific descriptor, which can handle the informational gap between a sketch and a colored image, making an opportunity for the efficient search hereby. The used descriptor is constructed after such special sequence of preprocessing steps that the transformed full color image and the sketch can be compared. We have studied EHD, HOG and SIFT. Experimental results on two sample databases showed good results. Overall, the results show that the sketch based system allows users an intuitive access to search-tools.

The SBIR technology can be used in several applications such as digital libraries, crime prevention, photo sharing sites. Such a system has great value in apprehending suspects and indentifying victims in forensics and law enforcement. A possible application is matching a forensic sketch to a gallery of mug shot images. The area of retrieve images based on the visual content of the query picture intensified recently, which demands on the quite wide methodology spectrum on the area of the image processing.

I. Introduction

Before the spreading of information technology a huge number of data had to be managed, processed and stored. It was also textual and visual information. Parallelly of the appearance and quick evolution of computers an increasing measure of data had to be managed. The growing of data storages and revolution of internet had changed the world. The efficiency of searching in information set is a very important point of view. In case of texts we can search flexibly using keywords, but if we use images, we cannot apply dynamic methods. Two questions can come up. The first is who yields the keywords. And the second is an image can be well represented by keywords.

In many cases if we want to search efficiently some data have to be recalled. The human is able to recall visual information more easily using for example the shape of an object, or arrangement of colors and objects. Since the human is visual type, we look for images using other images, and follow this approach also at the categorizing. In this case we search using some features of images, and these features are the keywords. At this moment unfortunately there are not frequently used retrieval systems, which retrieve images using the non-textual information of a sample image. What can be the reason? One reason may be that the text is a human abstraction of the image. To give some unique and identifiable information to a text is not too difficult. At the images the huge number of data and the management of those cause the problem. The processing space is enormous.

Our purpose is to develop a content based image retrieval system, which can retrieve using sketches in frequently used databases. The user has a drawing area where he can draw those sketches, which are the base of the retrieval method.

Using a sketch based system can be very important and efficient in many areas of the life. In some cases we can recall our minds with the help of figures or drawing. In the following paragraph some application possibilities are analyzed.

The CBIR systems have a big significance in the criminal investigation. The identificaton of unsubstantial images, tattoos and graffities can be supported by these systems. Similar applications are implemented in [9], [10], [11].

Another possible application area of sketch based information retrieval is the searching of analog circuit graphs from a big database [7]. The user has to make a sketch of the analog circuit, and the system can provide many similar circuits from the database.

The Sketch-based image retrieval (SBIR) was introduced in QBIC [6] and VisualSEEK [17] systems. In these systems the user draws color sketches and blobs on the drawing area. The images were divided into grids, and the color and texture features were determined in these grids. The applications of grids were also used in other algorithms, for example in the edge histogram descriptor (EHD) method [4]. The disadvantage of these methods is that they are not invariant opposite rotation, scaling and translation. Lately the development of difficult and robust descriptors was emphasized. Another research approach is the application of fuzzy logic or neural networks. In these cases the purpose of the investment is the determination of suitable weights of image features [15].

II. Our Project

In this section the goal and the global structure of our system is presented. The components and their communications

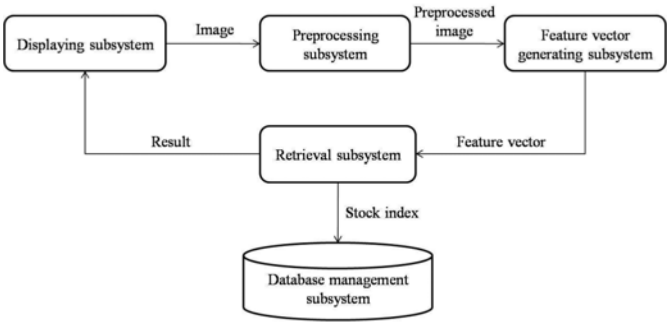

Fig. 1. The global structure of the system.

are introduced, and the functionality of subsystems and the algorithms are shown.

A. The Purpose of the System

Even though the measure of research in sketch-based image retrieval increases, there is no widely used SBIR system. Our goal is to develop a content-based associative search engine, which databases are available for anyone looking back to freehand drawing. The user has a drawing area, where he can draw all shapes and moments, which are expected to occur in the given location and with a given size. The retrieval results are grouped by color for better clarity. Our most important task is to bridge the information gap between the drawing and the picture, which is helped by own preprocessing transformation process. In our system the iteration of the utilization process is possible, by the current results looking again, thus increasing the precision.

B. The Global Structure of Our System

The system building blocks include a preprocessing subsystem, which eliminates the problems caused by the diversity of images. Using the feature vector generating subsystem our image can be represented by numbers considering a given property. The database management subsystem provides an interface between the database and the program. Based on the feature vectors and the sample image the retrieval subsystem provides the response list for the user using the displaying subsystem (GUI). The global structure of the system is shown in Fig. 1.

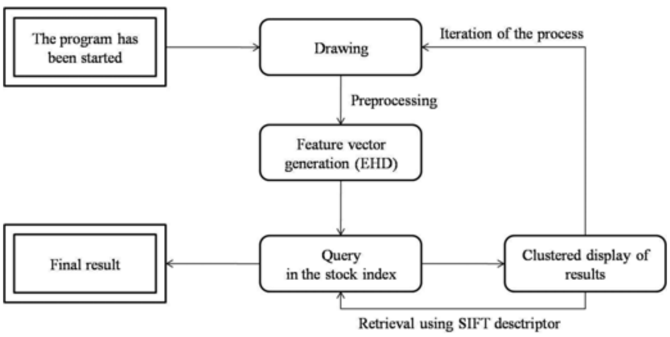

Fig. 2. The data flow model of the system from the user's point of view.

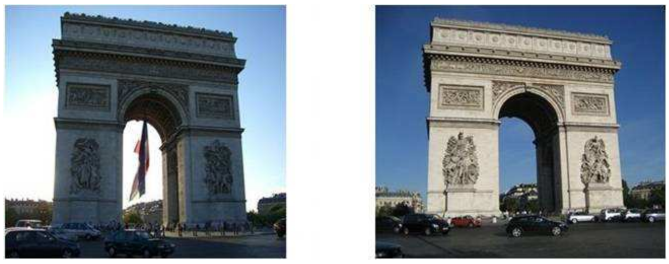

Fig. 3. The retrieval has to be robust in contrast of illumination and difference of point of view.

The content-based retrieval as a process can be divided into two main phases. The first is the database construction phase, in which the data of preprocessed images is stored in the form of feature vectors – this is the off-line part of the program. This part carries out the computation intensive tasks, which has to be done before the program actual use. The other phase is the retrieval process, which is the on-line unit of the program.

Examine the data flow model of the system from the user's point of view. It is shown in Fig. 2. First the user draws a sketch or loads an image. When the drawing has been finished or the appropriate representative has been loaded, the retrieval process is started. The retrieved image first is preprocessed. After that the feature vector is generated, then using the retrieval subsystem a search is executed in the previously indexed database. As a result of searching a result set is raised, which appears in the user interface on a systematic form. Based on the result set we can again retrieve using another descriptor with different nature. This represents one using loop.

C. The Preprocessing Subsystem

The system was designed for databases containing relatively simple images, but even in such cases large differences can occur among images in file size or resolution. In addition, some images may be noisier, the extent and direction of illumination may vary (see Fig. 3), and so the feature vectors cannot be effectively compared. In order to avoid it, a multi-step preprocessing mechanism precedes the generation of descriptors.

The input of the preprocessing subsystem is one image, and the output is the respective processed result set (see Fig. 4).

The main problem during preprocessing of the color images

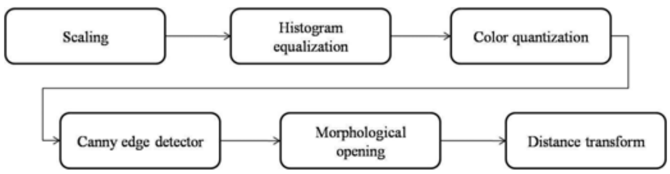

Fig. 4. The steps of preprocessing.

of real situations is that the background containing several textures and changes generate unnecessary and variable-length edges [12]. As a possible solution texture filters were analyzed, for example the entropy calculation based filter. It gives very valuable results, if a textured object of little color stands in a homogenous background.

Therefore, the classification of the image pixel intensities minimizes the number of the displayed colors. If only some intensity values represent the images, then according to our experience, the color based classification of result images can also be easily implemented. As an approximate method the uniform and minimum variance quantization [19] were used. After the transformation step edges are detected, of which the smaller ones are filtered by morphological opening filter.

D. The Feature Vector Preparation Subsystem

In this subsystem the descriptor vectors representing the content of images are made. Basically three different methods were used, namely the edge histogram descriptor (EHD) [4], the histogram of oriented gradients (HOG) [2] and the scale invariant feature transform (SIFT) [16].

Our system works with databases containing simple images. But even in such cases, problems can occur, which must be handled. If the description method does not provide perfect error handling, that is expected to be robust to the image rotation, scaling and translation. Our task is to increase this safety.

Another problem was encountered during the development and testing. Since own hand-drawn images are retrieved, an information gap arises between retrieved sketch and color images of database. While an image is rich of information, in contrast at a binary edge image only implicit content and explicit location of pixels can be known.

How could we allow a comparison between the two extremes, so that we keep only the relevant information of both? This transformation step has to be incorporated into the method, or to be made during the preprocessing. As we wrote in the previous subsection, the images of database were transformed into edge images, so information was lost, however. In order to discover the implicit content the 2-dimensional distance transform [5] was used.

E. The Retrieval Subsystem

As the feature vectors are ready, the retrieval can start. For the retrieval the distance based search was used with Minkowski distance [13], and the classification-based retrieval [14].

F. The Database Management Subsystem

The images and their descriptors are stored and the necessary mechanism for subsequent processing is provided. This is the database management subsystem, which consists of three parts, the storage, the retrieval, and the data manipulation modules [3].

The storage module provides images, information and the associated feature vectors are uploaded to the database. The file name, size and format of the image are attached. The information related to the preparation is gathered, as the maker's name, creation date, image title, the brand and type of recording unit. In addition, we may need more information of color depth, resolution, image dimension, vertical and horizontal resolution, possibly the origin of the image, so we take care of their storage. For storage the large images are reduced. The data is stored in a global, not scattered place in the hard disk.

The retrieval results are obtained by usage of query module. The retrieval subsystem contacts the database, which provides the descriptors. For optimization it is already loaded at startup to a variable, data structure. If we have the result of retrieval, the database retrieves the result image using the primary key. In addition, statistics can be taken due to a variety of criteria.

G. The Displaying Subsystem

Because drawings are the basis of the retrieval, thus a drawing surface is provided, where they can be produced. Also a database is needed for search, which also must be set before the search. In case of large result set the systematic arrangement of search results makes much easier the overviews, so it is guaranteed. The methods in our system cannot work without parameters, and therefore an opportunity is provided to set these as well.

The number of results to show in the user interface is an important aspect. Prima facie the first n pieces of results can be displayed, which conveniently can be placed in the user interface. This number depends on the resolution of the monitor, and forasmuch the large resolution monitors are widely used, so this number can move between 20 and 40. Another approach is to define the maximum number of results (n), but we also observe that how the goodness of individual results can vary. If the retrieval effectiveness is worse by only a given ratio, the image can be included in the display list.

In our system the possible results are classified, and the

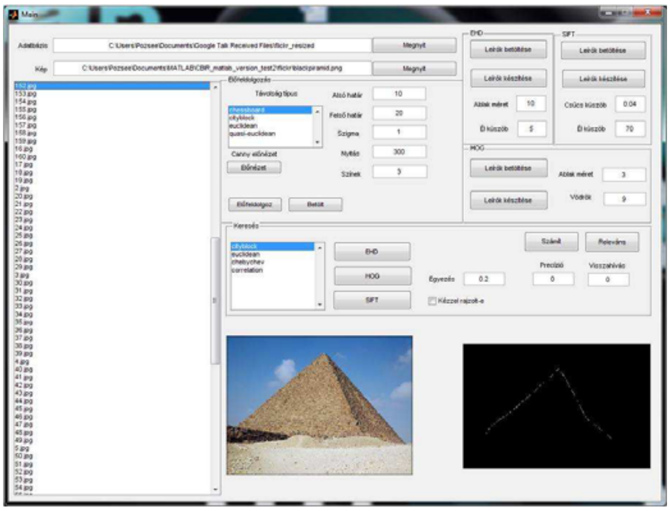

Fig. 5. The implemented user interface.

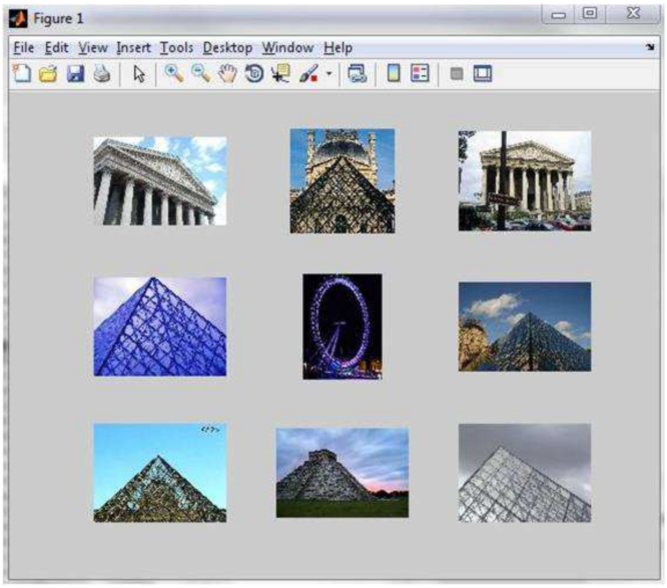

Fig. 6. The first nine results can be seen in a separate window.

obtained clusters are displayed. Hence the solution set is more ordered and transparent. By default the results are displayed by relevance, but false-positive results can be occurred, which worsen the retrieval results. If the results are reclassified in according to some criterion, then the number of false-positive results decreases. Thus the user perception is better. Since the color-based clustering for us is the best solution, so our choice was the k-means clustering method [1], which is perfectly suited for this purpose.

The implemented user interface can be seen in Fig. 5 and Fig. 6. Our program has been writen in MATLAB, and during the implementation some new idea of [18] was considered.

III. Tests and Results

A. Used Test Databases

The system was tested with more than one sample database to obtain a more extensive description of its positive and negative properties. The Microsoft Research Cambridge Object

Fig. 7. Some sample images of the Microsoft Research Cambridge Object Recognition Image Database.

Fig. 8. Some sample images of Flickr 160 database.

Recognition Image Database was used, which contains 209 realistic objects. All objects have been taken from 14 different orientations with 450×450 resolution. The images are stored in TIF format with 24 bits. This database is most often used in computer and psychology studies. Some images of this database can be seen in Fig. 7.

Another test database was the Flickr 160. This database was used before for measuring of a dictionary-based retrieval system [8]. 160 pieces of general-themed pictures have sorted from the photo sharing website called Flickr. The images can be classified into 5 classes based on their shape. A lot of images contain the same building and moments. The database is accompanied by examples, which is based on the retrieval. Since the test result are documented and the retrieved sketches are also available, so the two systems can be compared with each other. Some images of Flickr 160 database can be seen in Fig. 8.

Wang is the third used database, which contains 1000 images from the Corel image database. The images can be divided into 10 classes based on their content, namely Africa, beaches, moments, buses, food, dinosaurs, elephants, flowers, horses and mountains. Using this database color-based grouping of our system can be tried (see Fig. 9).

At the tests used sketch images can be seen in Fig. 10.

Fig. 9. Some images of Wang database clustered by color content.

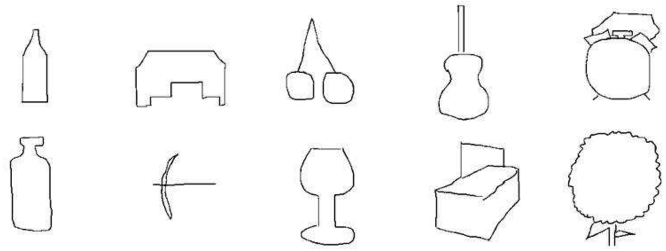

Fig. 10. Sketch images, which was used at the tests.

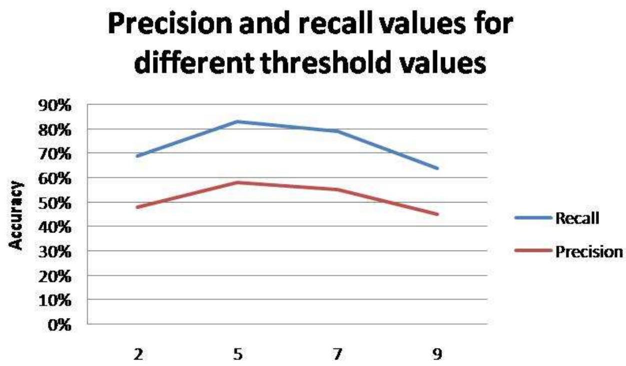

Fig. 12. Effect of threshold value change using EHD method. The block size is constant 10.

B. Testing Aspects, Used Metrics

We can evaluate the effectiveness of the system forming methods, and compare the different applied methods, if we define metrics. Thus, we can determine which method works effectively in what circumstances, and when not.

Let be a test database containing N pieces images, P length retrieval list, from which Q pieces matter as relevant results, and Z denotes the number of expected relevant hits. If we know this information, the following metrics can be calculated.

$$precision = \frac{relevant\ hits\ (Q)}{all\ hits\ (P)}, \tag{1}$$

where the $precision$ gives information about the relative effectiveness of the system.

$$recall = \frac{relevant\ hits\ (Q)}{expected\ hits\ (Z)}, \tag{2}$$

where the $recall$ gives information about the absolute accuracy of the system.

The number of all and expected hits is determined in each case of testing methods. The impact of multi-level retrieval to the efficiency of retrieval is measured, which confirms the importance of multi-level search. In addition, the ROC curves plot the true and false positive hit rate. The area under the curve reflects the efficiency of the method.

When the Object Databank database was used by EHD the provided precision and recallu values can be seen in Fig. 11 using different block size valus, and in Fig. 12 using different threshold values.

In Fig. 13 and 14 similar result graphs can be seen in that case when the HOG method was tested.

Our system was compared with other systems. If we focus on the average precision value, we can find our system better than some systems before (see Table I). So our system is more effective than the examined other systems.

IV. Conclusions

Among the objectives of this paper performed to design, implement and test a sketch-based image retrieval system. Two main aspects were taken into account. The retrieval process has to be unconventional and highly interactive. The robustness of the method is essential in some degree of noise, which might also be in case of simple images.

The drawn image without modification can not be compared with color image, or its edge representation. Alternatively a distance transform step was introduced. The simple smoothing and edge detection based method was improved, which had a similar importance as the previous step.

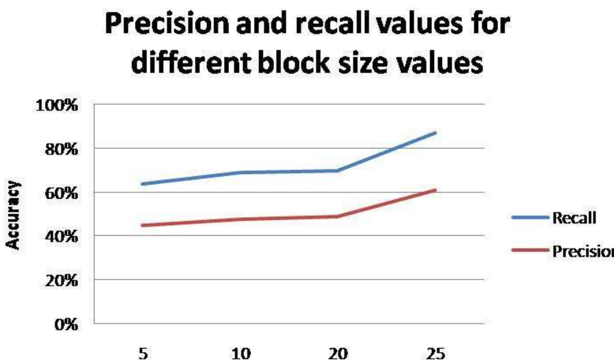

Fig. 11. Effect of block size change using EHD method. The threshold is constant 2.

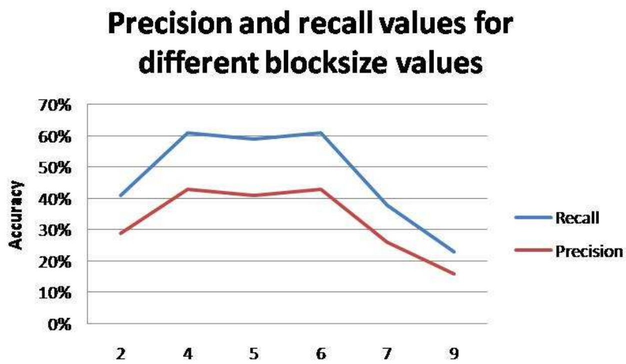

Fig. 13. Effect of block size change using HOG method. The number of bins is constant 9.

TABLE I

THE PERFORMANCE OF USED METHODS IN SKETCH-BASED SYSTEMS.

Method	HOG (with gradient map)	HOG (without gradient map)	SIFT	EHD (own)	HOG (own)
Average precision	54%	42%	41%	43%	44%

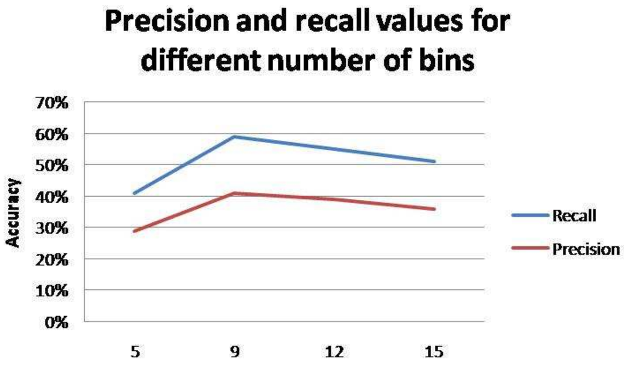

Fig. 14. Effect of number of bins change using HOG method. The block size is constant 5.

At the tests the effectiveness of EHD and the dynamically parameterized HOG implementation was compared. It was examined with more databases. In our experience the HOG in more cases was much better than the EHD based retrieval. However, the situation is not so simple. The edge histogram descriptor can mainly look better for information-poor sketches, while in other case better results can be achieved for more detailed. This is due to the sliding window solution of HOG. Using the SIFT-based multi-level solution the search result list is refined. With the categorization of retrieval response a bigger decision possibility was given to the user on that way, he can choose from more groups of results.

REFERENCES

[1] D. Comaniciu, and P. Meer, "Robust analysis of feature spaces: color image segmentation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 750–755, June 1997.

[2] N. Dalal, and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 886–893, July 2005.

[3] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: an experimental comparison," *Information Retrieval*, vol. 11, pp. 77–107, December 2007.

[4] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "An evaluation of descriptors for large-scale image retrieval from sketched feature lines," *Computers and Graphics*, vol. 34, pp. 482–498, October 2010.

[5] R. Fabbri, L.D.F. Costa, J.C. Torelli, and O.M. Bruno, "2D Euclidean distance transform algorithms: a comparative survey," *ACM Computing Surveys*, vol. 44, pp. 1–44, February 2008.

[6] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Hiang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: the QBIC system," *IEEE Computer*, vol. 28, pp. 23–32, 2002.

[7] Gy. Györök, "Embedded hybrid controller with programmable analog circuit," *IEEE 14th International Conference on Intelligent Systems* pp. 59.1–59.4, May 2010.

[8] R. Hu, M. Barnard, and J. Collomosse, "Gradient field descriptor for sketch based image retrieval and localization," *International Conference on Image Processing*, pp. 1–4, 2010.

[9] A.K. Jain, J.E. Lee, and R. Jin, "Sketch to photo matching: a feature-based approach," *Proc. SPIE, Biometric Technology for Human Identification VII*, vol. 7667, pp. 766702–766702, 2010.

[10] A.K. Jain, J.E. Lee, R. Jin, and N. Gregg, "Graffiti-ID: matching retrieval of graffiti images," *ACM MM, MiFor'09*, pp. 1–6, 2009.

[11] A.K. Jain, J.E. Lee, R. Jin, and N. Gregg, "Content based image retrieval: an application to tattoo images," *IEEE International Conference on Image Processing*, pp. 2745–2748, November 2009

[12] T. Hashimoto, A. Rövid, G. Ohashi, Y. Ogura, H. Nakahara, and A.R. Várkonyi-Kóczy, "Edge detection based image retrieval method by sketches," *Proc. of the Internatonal Symposium on Flexible Automation*, pp. 1–4, 2006.

[13] J.B. Kruskal, "Nonmetric multidimensional scaling: a numerical method," *Psychometrika*, vol. 29, pp. 115–129, 1964.

[14] Y. Liu, and F. Dellaert, "A classification based similarity metric for 3D image retrieval," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 800–805, June 1998.

[15] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[16] D.G. Lowe, "Object Recognition from Local Scale-Invariant Features," *IEEE International Conference on Computer Vision*, vol. 2, p. 1150, 1999.

[17] J.R. Smith, and S.F. Chang, "VisualSEEK: a fully automated content-based image query system," *ACM Multimedia '96*, pp. 97–98, 1996.

[18] J. Tick, and J. Fodor, "Some classes of binary operations in approximate reasoning," *Studies in Informatics and Control*, vol. 15, pp. 259–270, 2006.

[19] S.J. Wan, P. Prusinkiewicz, and S.K.M. Wong, "Variance-based color image quantization for frame buffer display," *Color Research and Application*, vol. 15, pp. 52–58, February 1990.