A Content Based Image Retrieval Approach Based On Document Queries

M. Ilie[1]
[1]Department Name, "Dunarea de Jos" University of Galati, Faculty of Automatic Control, Computers, Electrical and Electronics Engineering, Galati, Romania

Abstract - *This paper presents a new content based image retrieval (CBIR) approach, which makes use of descriptors originating in the local and global search spaces. The algorithms extracts four colour descriptors, one texture descriptor and two local descriptors which are used to train the corresponding classifiers, based on neural networks. Subsequently, the classifiers are grouped in two weighted majority voting modules, for local and global characteristics. The system is tested on regular images and on document scans obtained from two datasets used a benchmarks in previous conferences, in order to verify the architecture robustness. The experimental results demonstrate the effectiveness of the proposed model.*

Keywords: CBIR, neural networks, image descriptors, weighted majority voting

1 Introduction

The necessity of the content based image retrieval phenomenon was imposed by the problems encountered in different areas. Initially, the image classification was done based on text labels, which was proven to be very time consuming and error prone. Starting from this problem, the image processing techniques have been improved, combined and extended across a vast number of fields, like duplicate detection and copyright, creating image collections, medical applications, video surveillance and security, document analysis, face and print recognition, industrial, military and so on. The term of "content" implies that the images are deconstructed into descriptors, which are analyzed and interpreted as image features, as opposed to image metadata, like annotations, geo-tags, file name or camera properties (flash light on/off, exposure etc.)

The traditional CBIR approaches try to solve this problem by extracting a set of characteristics from one image and comparing it with another one, representing a different image. The results obtained until now are promising but still far from covering all the requirements risen by a real world scenario. Also, the current approaches target specific problems in the image processing context. Because of that, most of the CBIR implementations work in a rather similar way, on homogenous data. This causes significant performance drops whenever the test data originates from a different area than the training set.

This paper proposes a CBIR architecture model with descriptors originating in different search areas. In order to be able to classify images originating in document scans, we have added an extra module, responsible for the document image segmentation stage. The user is offered the possibility of querying the engine with both document scans and regular images in order to retrieve the best N matches.

During the implementation stage we have faced multiple problems, as specified below:

- image preprocessing;
- extraction of characteristics from various spaces; implementation of a supervised machine learning module;
- document image segmentation;
- benchmarking the overall performance.

We have reached the conclusion that a CBIR engine can obtain better results in the presence of multiple sets of descriptors, from different search spaces or from the same one, even if the test images originate in very different areas.

2 Related work

The CBIR engines are trying to mimic the human behaviour when executing a classification process. This task is very difficult to accomplish due to a large series of factors.

The CBIR queries may take place at different levels [1]:

- feature level (find images with X% red and Y% blue);
- semantic level (find images with churches);
- affective level (find images where a certain mood prevails). There is no complete solution for the affective queries.

All the CBIR implementations use a vector of (global or local) characteristics which originate in different search spaces - colour, texture or shape.

In the colour space there are many models but recently the focus is set on various normalizations of the RGB one in the attempt of obtaining invariant elements. Two of the most interesting ones are c1c2c3 (which eliminates both shadows and highlight areas) and l1l2l3 (which eliminates only the shadows, but keeps the highlight areas) [2].

There are 4 large categories for determining the texture descriptors [3]:

- statistical (Tamura features, Haralick's co-occurrence matrices);
- geometrical (Voronoi tesselation features);
- spectral (wavelets [4], Gabor and ICA filters);
- model based (MRFs [5], fractals).

One of the most widely embraced approach is to use local binary patterns [6].

In what regards the local descriptors, probably the most famous algorithm (scale invariant feature transform SIFT) was introduced by David Lowe [7]. Since then, many approaches have been developed. Some of the most popular ones are based on speeded un robust feature (SURF) [8], histogram of oriented gradients HOG [9], gradient location and orientation histogram (GLOH) [10] or local energy based shape histogram (LESH) [11].

3 Our approach

The proposed approach targets to classify a mixed set of images, containing real world scenes and document scans. The system mainly follows the standard CBIR architecture as it can be seen in the image below. It is composed of two interconnected submodules:

- the training and learning module;
- the document classification module.

A valid use case scenario contains the below stages:

- the system is trained on a set of images;
- each image is analyzed and decomposed in relevant descriptors;
- the descriptors are provided as input to a machine learning module, which is in charge of setting the class boundaries;
- each new regular image (not document) is decomposed and classified accordingly;
- each new document scan is preprocessed and segmented. The extracted images are then classified;
- the system extracts the 10 most relevant results and provides them as an answer to the user query.

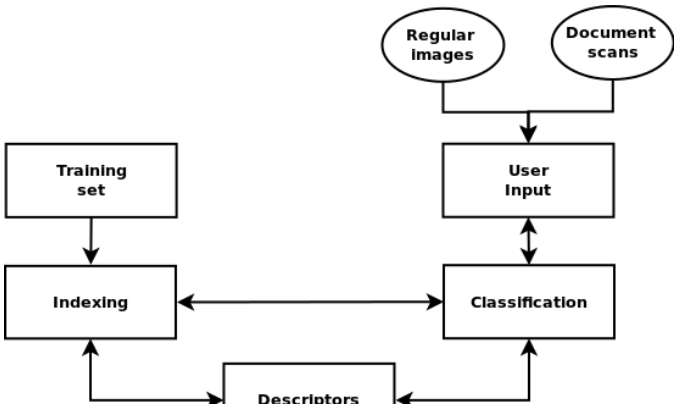

Figure 1. The basic system architecture

The indexing process is based on supervised machine learning and is conducted on regular images. The user is allowed to enter queries based on both image types.

We are using a mixed set of image characteristics:

- different colour spaces;
- texture space;
- local descriptors.

We have not used any shape descriptors, as the preliminary tests showed that in this area these do not produce a noticeable improvement. The main problem was caused by the fact that the objects contained in the images may be affected by problems like occlusion or clutter.

In the colour descriptors area, we have used 4 sets of characteristics, as it follows:

- c1c2c3 and l1l2l3. As explained above, these colour spaces are very useful when applied on real world images. The coordinates are described by the equations below:

$$c_1 = arctg \frac{R}{\max(G,B)}; \quad (1)$$

$$c_2 = arctg \frac{G}{\max(R,B)}; \quad (2)$$

$$c_3 = arctg \frac{B}{\max(G,R)} \quad (3)$$

$$l1(R,G,B) = \frac{(R-G)^2}{(R-G)^2 + (R-B)^2 + (B-G)^2} \quad (4)$$

$$l2(R,G,B) = \frac{(R-B)^2}{(R-G)^2 + (R-B)^2 + (B-G)^2} \quad (5)$$

$$l3(R,G,B) = \frac{(B-G)^2}{(R-G)^2 + (R-B)^2 + (B-G)^2} \quad (6)$$

- the whole image in RGB coordinates;
- the RGB histogram, with 256 bins.

Each of the four sets of characteristics is used as an input for a standard feed forward/back propagation neural network. The neural networks' outputs are then collected by a simplified weighted majority voting module, as it can be observed in the image below.

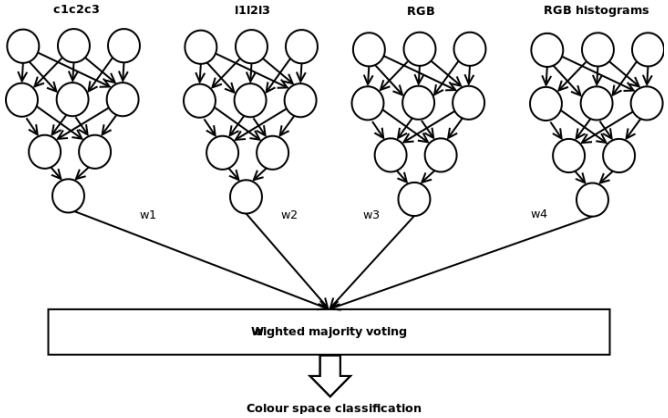

Figure 2. Colour space classification

The weighted module works according to the below algorithm:

- let n be the number of accepted classes and k be the number of classifiers;
- each neural network will produce on the final layer a vector Cx={c1, c2, ... cn}, where 1<=x<=k;
- the weight associated to the output layer will be W={ w1, w1 ... wk};
- the weighted result will be provided by the wiCi sum, as specified below, where R ϵ [1, n], max(C) represents the maximum value obtained for a certain class, and idx represents the position of this class in the final vector

$$R = idx\left(max\left(\sum_{i=1}^{k} w_i C_i\right)\right) \qquad (7)$$

In the texture space area we have chosen an approach based on local binary pattern descriptors, mainly because of their invariant properties for colour or rotation.

For the local descriptors we have chosen two sets of characteristics, based on scale invariant feature transform (SIFT) and histogram of oriented gradients (HOG). Traditionally, the HOG descriptors are used in order to train an SVM classifier, but since we are dealing with a multiple classification problem, we have used neural networks in the learning stage for both descriptors. The two types of local descriptors produced similar results during the tests, therefore the combined classifier for SIFT and HOG uses equal weights of 50%.

The final classifier includes an additional weighted majority voting module, as shown by the image below.

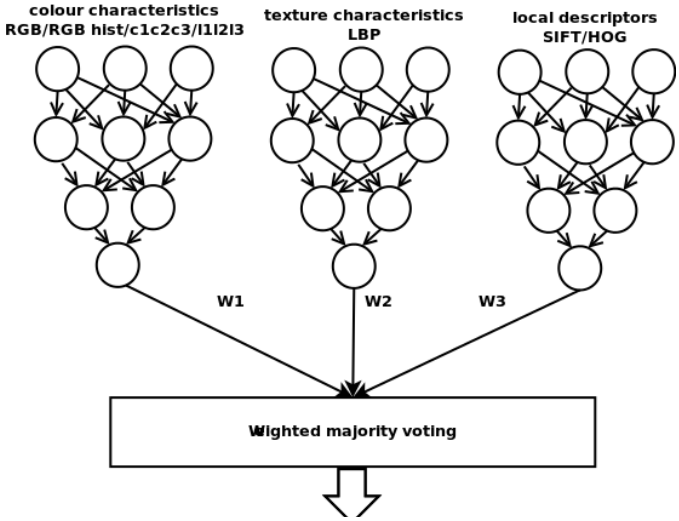

Figure 3. The final classification

Since the aim is to classify document scans as well as regular images, we have also included an additional document analysis module. Its purpose is to process the scans and extract the images included in the document, in order to pass them subsequently to the module in charge of extraction of descriptors and to classify them accordingly. We are not interested in text segmentation, therefore this module will only binarize the document and go through a bottom-up [12] image segmentation stage, based on the below steps:

- text filtering, implemented as a simplified XY axis projection module [13];
- the document is split in tiles, which are analyzed according to their average intensity and variance. The decision criteria is that in a particular tile, an image tends to be more uniform than the text;
- the remaining tiles are clustered through a K-Means algorithm, which uses as a decision metric the Euclidean distance;
- the clusters are filtered according to their connectivity and scarcity scores in order to eliminate tiles containing text areas with different fonts, affected by noise/poor illumination or by page curvature;
- the final clusters are exposed to a reconstruction stage and merged into a single image, which is provided as an input to the modules in charge of descriptors extraction/classification.

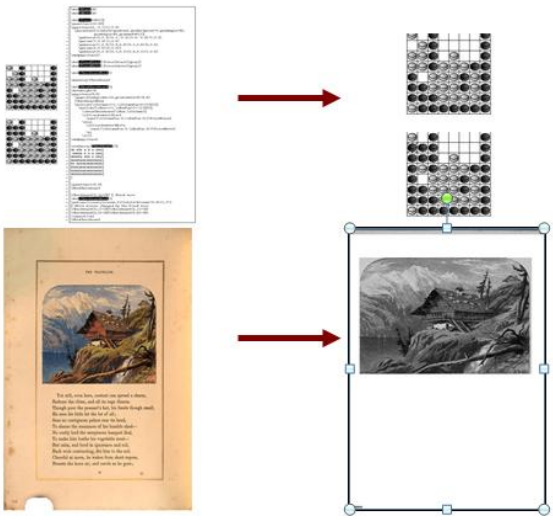

Figure 4: Document image segmentation results

All the neural networks have the same structure. The transfer function is sigmoid and the images in the training set have been split in three groups:

- 60% for training;
- 20% for cross-validation;
- 20% for testing.

In order to validate the neural network progress, we have used gradient checking on the cost function specified below:

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K}\left(y_k^{(i)}\log\left(h_\theta\left(x^{(i)}\right)\right)_k + \left(1 - y_k^{(i)}\right)\log\left(1 - h_\theta\left(x^{(i)}\right)\right)_k\right)\right] + \frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left(\theta_{ji}^{(l)}\right)^2 \quad (8)$$

with the following notations:

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\} - \textit{the } (input, output) \textit{ vector}$$

$$h_\theta(x) \in R^K - \textit{the hypothesis function}$$

L - the number of layers

sl - the number of neurons in a specific layer l

K - the number of classes, with $y \in R^K$

Θ - the matrix which stores the weights for each layer

4 Experimental setup

We aimed to create a scalable application, portable between different architectures and operating systems. Therefore, in what regards the programming language, we have chosen python over Matlab or Octave, for practical reasons, especially for the libraries which facilitate the user interface generation, socket management and data processing. The operating system is a 12.04 LTS 32 bit Ubuntu, running on a machine with two cores with hyper threading and 4 GB of RAM. The software architecture is modular to facilitate any subsequent refactoring; each sub-module is implemented in a class. In order to make better use of the hardware, the modules that require resources intensively use multi-threading and multi-processing techniques. The data is stored in a MySQL database, based on MyISAM. The application follows the standard client-server architecture, in order to facilitate the exposure of functionalities to multiple users at once. So far, we have disregarded the user management problems.

We have restricted the number of recognized classes at 10 so far. The training was conducted on a CIFAR data set, provided by [14]; it includes 60000 small (32x32 pixels) colour images. The author's tests involved feed forward neural networks as well, with performances revolving around 87%.

The document scans data consisted of 1380 images, obtained from 2 sources:

- scans of old, degraded documents, used as a benchmark in the ICDAR 2007 conference [15];
- high quality copies, containing mostly manuals and documentation for the Ubuntu 12.04 operating system. In order to be able to use them, we have previously converted them from the pdf format to the jpeg one.

Initially we have tried replicating the CIFAR benchmark results. We have also used a neural network approach, based on RGB descriptors only; we obtained similar performances (85%). However, when we tried to classify an image originating from a different image set (document scans), the accuracy dropped significantly, by more than 10%. Therefore, we started experimenting with various combinations of RGB/c1c2c3/l1l2l3 descriptors. The results are described by the table below:

Table 1. Colour space experiments

Combined descriptors	**Results**
RGB+l1l2l3	82%
RGB+c1c2c3	**84%**
RGB+RGB histograms	69%
RGB	71%

As we can observe, the presence of the c1c2c3 and l1l2l3 colour spaces produces an improvement of over 10%, leading to the below conclusions:

- the experiments conducted on real world images confirm the necessity of additional colour space descriptors;
- c1c2c3 produces the most solid performance boost. After analyzing the images in the data set, we have observed the presence of many pictures with shady areas, which shows that this colour space is adequate for these conditions. The image below shows the effects of the c1c2c3 normalization on an image containing highlight and shadow areas;
- introducing the RGB histograms as a global descriptor actually produced a performance drop, as two different images may have very similar colour histograms, yet a very different content. This was

mainly caused by the surrounding conditions in which the picture of a certain object was taken. Also, the presence of the shadow areas affect the histograms and implicitly the classification result.

After experimenting with various colour space weight values we have chosen the below values:

- the RGB histograms weights have been set to w4=10%, which improved the overall performance. We have kept this set of descriptors for situations where the colour plays a more important role in the classification process. In this case, the user will be able to adjust this value accordingly;
- the rest of the weights have been set to w[1:3]=30% (for the RGB/c1c2c3/l1l2l3 descriptors). This lead to a combined overall performance of 86%. As we mentioned before, the UI offers the user the possibility of manually adjusting the global colour relevance (associated to the RGB histograms) in the final result. As an example, the user can choose a combination like w1=20%, w2=20%, w3=20% and w4=40%.

The next set of experiments was conducted in the texture space, with the LBP descriptor. The main problems in this area were related to choosing the cell shape and size, along with the number of pixels which compose the final descriptor. After a series of tests we concluded the below:

- choosing radial cells over square cells produces an overall performance increase of over 10%, going over 95%;
- the execution times are larger when using radial cells, especially due to the trigonometric calculus;
- over-increasing the cell size leads to performance drops, as the small textures are ignored.

The conclusion was that we will use square cells of 3x3 pixels and 8 pixels to compose the local texture descriptor.

Subsequently, we have started experimenting with the rest of the descriptors. For the HOG and SIFT algorithms we have used the authors' implementations. The tests involved the usage of singular descriptors and combining all of them together; the final weights have been set as it follows:

- W1=30% for the colour space;
- W2=30% for the texture space;
- W3=40% for the local descriptors. These have been considered more representative than the global descriptors.

The results are presented in the table below:

Table 2. Experiments involving all descriptor spaces

Descriptor type	**Results**
Colour space	86%
Texture space (LBP)	85%
SIFT	82%
HOG	85%
All of the above	**92%**

The results show that combining multiple types of descriptors from multiple search spaces leads to performance improvements. On the above mentioned data set, the results are promising and show an increase of over 5%. Also, the proposed architecture is able to correctly classify images obtained from document scans as well as regular images.

5 Future research

In the future we would like to continue the research conducted so far. There are many areas which can be improved and also, upon refactoring they can provide new functionalities:

- we intend to add a module in charge of collecting an user score, which can be used later for altering the default weights in the majority voting module. This way, the CBIR engine will be able to provide more representative result for the user. Also, this feature can be combined with a user management module so that the system can recall the user's preferences;
- we also intend to insert a module that can analyze a certain image and compute how many shadow areas it contains. This module would help in deciding which colour space is more adequate for each particular situation;
- the number of characteristics is very large; only the SIFT descriptors may go over 100000 for 640x480 images. In order to be able to compute the results much faster, we considering the possibility of adding a module in charge of reducing the dimensionality;
- in the document processing area, the system is currently cropping out the images from a scan. In order to have more accurate results, we intend to add an OCR module, which can extract the text content as well. The text can be later on reduced to keywords, which can be used in the classification and retrieval process.

6 Acknowlwdgements

The authors would like to thank the Project SOP HRD /107/1.5/S/76822 - TOP ACADEMIC, of University “Dunarea de Jos” of Galati, Romania.

7 References

[1] Sebe, N. Feature extraction & content description - DELOS - MUSCLE Summer School on Multimedia digital libraries, Machine learning and cross-modal technologies for access and retrieval. www.videolectures.net. [Online] 02 25, 2007. www.videolectures.net/dmss06_sebe_fecd/.

[2] Colour-based object recognition. Gevers, T., Smeulders, A.W. s.l. : Pattern Recognition, 1999, Vol. 32.

[3] Vassilieva, Natalia. RuSSIR - Russian Summer School in Information Retrieval . [Online] 2012. http://videolectures.net/russir08_vassilieva_cbir/.
[4] Illumination invariant extraction for face recognition using neighboring wavelet coefficients. X. Cao, W. Shen, L.G. Yu, Y.L. Wang, J.Y. Yang, Z.W. Zhang. 2012, Pattern Recognition.
[5] Range map superresolution-inpainting, and reconstruction from sparse data. Arnav V. Bhavsar, Ambasamudram N. Rajagopalan. 2012, Computer Vision and Image Understanding.
[6] Wikipedia. Local binary patterns. www.wikipedia.org. [Online] 11 08, 2011. http://en.wikipedia.org/wiki/Local_binary_patterns.
[7] Object recognition from local scale-invariant features. Lowe, David. s.l. : International Conference on Computer Vision, 1999.
[8] Herbert Bay, Tinne Tuytelaars, Luc Van Gool. Speeded-Up Robust Features (SURF). Zurich, Leuven, Belgia : s.n., 2008.
[9] Histograms of Oriented Gradients for Human Detection. Navneet Dalal, Bill Triggs. 2005, International Conference on Computer Vision \& Pattern Recognition, pp. 886-893.
[10] Krystian Mikolajczyk, Cordelia Schmid. A Performance Evaluation of Local Descriptors. IEEE transactions on pattern analysis and machine intelligence. 2005.
[11] Head Pose Estimation in Face Recognition across Pose Scenarios. Saquib Sarfraz, Olaf Hellwich. Madeira, Portugal : s.n., 2008. Proceedings of VISAPP 2008, Int. conference on Computer Vision Theory and Applications. pp. 235-242.
[12] A survey of document image classification: problem statement, classifier architecture and performance evaluation. Nawei Chen, Dorothea Blostein. 2007, International Journal of Document Analysis and Recognition (IJDAR), p. Volume 10.
[13] Recursive X-Y Cut using Bounding Boxes of Connected Components. Jaekyu Ha, Robert M. Haralick.
[14] Krizhevsky, Alex. Learning Multiple Layers of Features from Tiny Images. 2009.
[15] A. Antonacopoulos, D. Bridson, C. Papadopoulos. ICDAR 2007 Page Segmentation Competition. ICDAR. [Online] 2007. http://www.primaresearch.org/ICDAR2007_competition/.