

Automatic System for Extraction of Content-Based Characteristics from Digital Images

Gonzalo León, José Luis Delgado, Covadonga Rodrigo, Fernando López, Valentín Sama
Dpto. Lenguajes y Sistemas Informáticos - E.T.S.I. Informática - U.N.E.D.
gleon1@alumno.uned.es , {jdelgado,covadonga,flopez,vsama}@lsi.uned.es

Abstract

In this paper we expose the development of a CBIR system (Content-based Image Retrieval) that is able to retrieve images from a corpus based upon the image content. In order to obtain such functionality, the system establishes a set of characteristics which will be automatically generated. This allows the system to univocally identify each image from the collection. The sort of characteristics is diverse and they are related to concepts such as entropy, Gabor filters and image size. After the calculation of characteristics of each image, a calibration process is performed whereby the system estimates the best weight for each characteristic. This estimation makes use of a calibration algorithm and a set of experiments, and the result is the influence of each characteristic in the main function that is used for the retrieval process. The calibration process starts in an equally balanced situation (all the characteristics have the same influence in the main function), and after several iterations the weight for each characteristic is fixed. The following task is the image validation, where the modifications to the main function are verified so as to ensure that the new function is better than the previous one. Finally, the image retrieval process is performed according to the *ImageCLEFmed rules*. The retrieval results have not been the expected ones, but we must say they are a good starting point that makes us establish several work lines for the future.

Categories and Subject Descriptors

I.4 [Image Processing and Computer Vision]: I.4.0 General [Query Languages]; I.4.5 Reconstruction [Transform Methods]; I.4.6 Segmentation [Pixel Classification]; I.4.7 Feature Measurement; I.4.9 Applications; I.5 [Pattern Recognition]: I.5.2 Design Methodology; I.5.3 Clustering; I.5.5 Implementation

General Terms

Algorithms, Experimentation, Measurement, Performance, Verification

Keywords

CBIR, Calibration, Validation, JAI

1 Introduction

Image characterisation and the process of finding links amongst a collection of images is a well-known challenge that has been studied for a long time. This task can be addressed in two different ways: textual (based upon a set of textual metadatas related to each image) and graphical (based upon the image observation and its properties). The former requires an initial tagging process; hence the results will depend on the goodness of fit. The latter is a more objective way of characterising an image, but requires some complex mathematical procedures which increase the difficulty of obtaining results for its later analysis.

Focused on the second way and, more specifically in the context of digital images, the work entails the extraction of significant data from these images, being this collection of attributes sufficient for identifying the images. This significant data (from now on we will call them only characteristics) is a set of short-scope attributes, allowing the identification, or at least the characterisation, within a certain group. The sort of characteristics is very diverse, covering the range from the properties such as height and width until the statistical data of the transformed image. The reason for extracting the characteristics instead of working with the original images lies in the big size that images usually have, and implies a very high time-consuming task that cannot be afforded. Besides, the characteristic has to be a tiny piece of information since its usefulness lies in the ability of making very often comparisons with other equivalent characteristics within the same image or from other ones.

Currently there are several image processing systems which pursue this aim. For instance, the GIFT (*GNU Image-Finding Tool*) is an image retrieval system based upon the content (a.k.a. *CBIR – Content-based Image Retrieval*). Given a certain image, the system returns the most similar images ordered by relevance. GIFT is intended as a client-server application.

This work is aimed at developing a system that is able to extract, from collection of images, a set of characteristics from each of them which will be further used for cataloguing and comparing purposes. The characteristics are only based upon the image content and not upon other data (textual metadata).

2 An approach to the system

In the previous section we introduced the concept of *characteristic* as an attribute that allows, alone or in combination with other attributes, establishing the membership of an image to a certain group. Obtaining characteristics and its later use in the images identification has been applied in many fields, mainly in the biometric identification as it is exposed in [2, 3], with great successful results.

The first question that appears when developing the system is to know which the best option is amongst all the possibilities, and what has already been developed for this aim. Consequently there is a need in investigating the characteristics which are likely to be extracted from an image, and to what extent they are suitable for their comparison.

Another goal is to plan the system in a scalable way in terms of the characteristics extraction software. The system should not have a close architecture when the purpose is to add new characteristics. On the contrary, the addition of new characteristics should be easy as far as possible, avoiding the modification of the code since while developing the system, the goodness of the characteristics' results developed will be unknown.

In the same way, the system that is going to be developed should be able to:

- Store a collection of images (*corpus*) and their related characteristics.
- Store a collection of classes/groups or characteristics which can be applied to images and their influence in later comparisons.
- Compare the image's characteristics with those belonging to the images within the corpus.
- Combine the comparison's results provided by the different characteristics when obtaining a unified measure of similarity, for its later sorting.

Amongst all the existing options when developing our system, we have chosen a Java development starting from scratch, using the Java 1.5 Standard Edition (J2SE), focused on the program execution in different platforms (operating systems), with no need for recompiling, making the application to be executed in several places only moving the files from one machine to another.

A further reason justifying this choice is the great number of libraries which are available for Java, covering many different fields like digital image processing. In this sense, we can find JAI (Java Advanced Imaging), that is an optional library (not included in the standard distribution J2SE) allowing the reading and processing of images for extracting their characteristics. Furthermore, the use of the concurrent paradigm by means of threads was also considered to choose Java as the selected technology.

3 Selection of characteristics

The number and source/nature of the characteristics available for this study was very diverse, so there are some characteristics more connected to the image's source (for instance, the number of bits describing the color depth per pixel), that allow a direct mathematical and easier management, whereas there are some other characteristic obtained through a mathematical algorithm (for instance, Gabor Filters) upon the image and its later processing requires a better knowledge and produces an indirect application. Hence it is necessary to perform an analysis of these characteristics and evaluate whether they are useful or not, through a justified selection.

On the other hand it seems to be a logical thought the fact that the influence of these characteristics is not the the same in the final result. A priori, characteristics related to the image's dimension (for instance, height and width) should have a lower weight than other characteristics, such as the image's average power; nevertheless, this needs to be accepted in a relative way. For this reason a calibration process is required to ensure to what extent each characteristic impacts in the final value of the comparison. The calibration process will be described in Section 4.

After performing some experimental assessments and addressing some indications shown in several articles (see [7, 5, 1, 4]), the final number of characteristics was reduced to 9. The comparison procedure to be used is the one giving a value of 0 for those images with a very low similarity, whereas it gives a value close to 1 when the similarity is higher. In all the cases, the *JAI (Java Advanced Imaging)* library has been used for the image reading and other related information. The different characteristics, grouped by the method to extract them, are shown in the next subsection.

Characteristics related to the Histogram

Euclidean distance between two histograms

Given the histograms from two images, one measure of the similarity between the images could be the Euclidean distance between both histograms, transforming these histograms in a n-dimensional vector, where n is the number of colours. Let x and y be the histograms from the two images. The euclidean distance is defined as:

$$D = \sqrt{\sum_{i=0}^n (x_i - y_i)^2}$$

However, there is a problem in directly applying this formule. The chances for the pixel value in images with different sizes can be very variable and, at last, they can be the same image (if the images are not normalised in their size). For this reason the comparison is not performed with that value but the weight of such value. It is used the probability for such value, i.e. the numer of chances of each pixel's value divided by the total number of pixels. Since the similarity between images is inversely proportional to the euclidean distance, it is provided as the value for the similarity: $1 - D$.

The histogram presents a problem while working with images in a grey-scale or a colour-scale. The source of this problem lies in the number of histograms required to define the image. In the grey-scale images only one histogram is needed whereas the colour-scale images need three histograms (one per band colour: red, green and blue). In this way, what it is executed is a comparison between histograms in a colour-scale image and, if a comparison between grey and colour images is performed, firstly there is a transformation of the colour histograms into a grey-scale one, so that the comparison can be established at the same level.

Arithmetic average from the Histogram

Another interesting value to be considered is the arithmetic average of the pixels' values, easily calculated from the histogram. The value for the similarity between two images is obtained through the next formule $1 - |x_1 - x_2|/x_1 + x_2$, where $|x_1 - x_2|$ is the absolute value of the difference between both averages.

Entropy

There is another characteristic related to the histogram and obtained from it, named entropy. It is defined as follows:

$$H = - \sum_{i=1}^n p(x_i) \cdot \log_2 p(x_i)$$

where n is the number of different values that each pixel can have, and $p(x_i)$ is the chance for the value at the position i . The entropy returns a unique real value for each image. In order to compare two images, the similarity value comes from the next formule:

$$S_H = 1 - \frac{|E_1 - E_2|}{E_1 + E_2}$$

where E_1 y E_2 are the entropies of each image. As it can be stated, the result returned by the formule, the less difference between the images, the closer value to 1. However, this formule does not take into account the case of entropies with different sign, leading to a non significant value, since the addition of both entropies is not the addition of their magnitudes. To avoid this problem, in case of a negative entropy, the less value of both entropies is added to the negative one. This makes one of them to get the value of 0 while the other is necessarily positive.

Local entropy

Since the entropy is a unique valor for the entire image, there is a lack of local information about the image, i.e. it does not define the image in its different areas. To be more accurate at this point, a new characteristic was added, named "*local entropy*". The procedure involves the division of the image in 36 areas of the same size, and the calculation of the entropy for each of these areas. The result is a vector used as a characteristic of the digital image.

To calculate the similarity between two local entropies vectors, the process consists of getting the local entropies for the same area within the two images, calculated as $1 - x/VM$, where x is the difference between the entropies and VM is the maximum value. Empirically was set to 10. Finally the arithmetic average of the similarities was selected as the similarity between the two images.

Characteristics related to the Gabor's filters

Gabor's filters are used in a digitalised image and provide information about its texture. They are commonly parametrised with the orientation and the spatial frequency, as it is stated in [8].

This transformation lies in the application of a kernel created from the Gabor's function described over the digital image. For our purposes it is convenient to study the image in different orientations, wave lengths and differences. Eight orientations have been used, from 0 to $(7/8)\pi$, with variations of $(1/8)\pi$. For the wave lengths four options were selected: 2.73, 5.47, 8.20, 10.93. For the differences, two options were taken into account: 0 and $-(1/2)\pi$. From each transformed image, the arithmetic average is obtained as well as the typical deviation; they are two different characteristics.

The whole similarity between two vectors (both arithmetic averages as well as typical deviations) is calculated through the arithmetic average (in absolute value) between the elements at each vector's position. The similarity S_1 is calculated as $1 - x/VM$, where x is the arithmetic average of the differences and VM is the maximum value empirically set to 100.

The final similarity value is calculated as $0.1 \cdot S_1 + 0.9 \cdot S_2$, where 0.1 and 0.9 are the selected values coming from several experiments which were carried out about this filter.

Other characteristics

Other easier characteristics have been included in the system and are described as follows:

- *Colour Bits*: Number of bits used to represent the image. In general, 8 and 16 bits are used for images in grey-scale and 24 bits for the full-colour images. The similarity is $1 - |bc_1 - bc_2| / bc_1 - bc_2$, where bc_1 y bc_2 are the bits of the first and second images respectively.
- *Aspect Ratio*: Proportion between the width and the height of each image. The similarity value is $1 - |a_1 - a_2| / \max(a_1, a_2)$, where a_1 and a_2 are the aspect ratios for the first and the second images respectively.
- *Tiny Image*: Direct comparison between images reduced to a size of 40x30 pixels. This characteristic, despite of being very simple, is highly effective when comparing images with a similar shape, as it is stated in [6]. The similarity value is calculated by the formule $1 - De / VM$, where De is the euclidean distance and VM is the maximum value set to 12000.
- *Size*: This is the number of pixels. The similarity is returned by the next formule $\min(t_1, t_2) / \max(t_1, t_2)$, where t_1 and t_2 are the sizes for the first and the second images respectively.

4 Calibration and Validation

As we forementioned in the previous sections, the *calibration* process was the one that leads to the modification of each characteristics' weight so as to increase the hit ratio in the group's prediction. It is possible for one characteristic to be better than any other one to classify an image, but this could not happen when working with any other different image. For this reason there is a need in finding an average weight of influence in terms of final similarity that allows an appropriate classification for the greater number of images.

The calibration process is a non trivial task. It can be supposed that, through the entire collection of working images, it is selected a better weight that improves our classification process for the same collection of images. This could be understood like "*cheating*" and distorting the results since the perfect classification is used in advance. In other words, if the weights are modified to let the Image X, that is known in advance it belongs to the Group Y, to be classified within the Group Y, it would not be honest to state that the classification has been carried out in the right way (the information to solve the problem has been used in advance and it should be unknown at that moment). This could be compared to asking someone to foretell a card that has been previously shown.

For this reason, the calibration requires a later process named *validation*. The validation allows the system to decide whether the modification of the weights enclosed in the calibration enhances or not the results. To achieve this goal, the validation process utilises a collection of images not used in the calibration, and a sort of validity is obtained for the weights calculated in the previous step.

To carry out the calibration process it was used the *ImageCLEF* information from the previous years. Specifically the database used was *casimage* (<http://pubimage.hcuge.ch>) that was divided in disjoint groups. A total of 27 groups were extracted from the 57, discarding those with a few number of images or those very similar to existing ones.

The goal was to obtain the characteristics' weights that make the similarity value, for each image within the a group, as high as possible. As a secondary objective, the similarity value for a certain image belonging to a group should be as low as possible when comparing to images belonging to other groups; this objective prevents the system classifying in a wrong way. These goals were achieved comparing all the images within the corpus amongst them, obtaining a collection of sorted descendant lists by similarity. For each list, it was calculated the prediction capacity to the group it belonged to, having the greatest capacity prediction those located in the first positions of the list. If it was possible to infer, from such value for the prediction, that an image belonged to the group it really belonged to, then it was asserted that there had been a hit in the prediction.

At this point of the process, the weight of a characteristic was slightly increased and the step above was performed again. If the number of hits was stable or increased, the characteristic's weight was increased as well and the number of hits was re-calculated. If the number of hits decreased, the previous weight was restored and the process tried to increase the weight of other characteristic. This process was executing until covering the whole set of characteristics. We call

this “*Calibration Algorithm*”.

The calibration algorithm is a very time-consuming task. Considering a corpus that consists of 5000 images, the system took 5 days in executing the algorithm. However, the algorithm has demonstrated to be the most suitable to obtain the best results compared to other algorithms evaluated. For a collection of images outside the corpus to calibrate the system, the hit ratio was increased from the 56% (that gave the same weight to all the characteristics) to the 86% with the weights coming from the algorithm.

In any case, and even being the computational cost very high, the calibration process is required since it is reasonable that the system improves future classifications when the weights for the right classifications are previously extracted (feedback). To start the study of the calibration process, some initial analysis were carried out taking as inputs the results of the comparisons between a group of images and the images within the repository (corpus). To manage this, the similarity values were extracted in order to include them in a database with the objective of graphically displaying this result. Figure 1 shows the similarity distribution between images within the same group and images belonging to different groups, for the characteristic of local entropy.



Figure 1 – Similarity Distribution for Local Entropy

It should be visually noted that the similarity amongst images within the same group (Groups 1-57) are placed, in average, above the similarities amongst different groups. This is a logical circumstance because it asserts that the similarity amongst images within the same group is higher than the one coming from different groups.

Table 1 illustrates the values obtained for the similarities once the algorithm was executed through several iterations:

5 Analysis of the results and conclusions

In our first participation with the ImageCLEF we can say we have not achieved the expected results. Compared to the rest of the groups, and analysing the metrics provided by the organisation, we have got the last position in most of the cases. The reason for this position lies on the number of extracted images for each topic, always under the number of images provided by the rest of participants (we have provided dozens while some other teams have provided thousands).

CHARACTERISTICS	WEIGHT
Euclidean Distance between Histograms	0.04474
Entropy	0.04517
Local Entropy	0.17401
Tiny Image	0.30959
Colour Bits	0.06239
Aspect Ratio	0.04784
Size	0.08883
Typical Desviation of Gabor Transform (0)	0.03907
Typical Desviation of Gabor Transform $(-\frac{1}{2}\pi)$	0.04099
Arithmetic Averages of Gabor Transform (0)	0.03919
Arithmetic Averages of Gabor Transform $(-\frac{1}{2}\pi)$	0.05582
Arithmetic Averages of the Histogram	0.05232

Table 1 – Calibration Algorithm results

This fact can be explained from the point of view of our system. We defined some criteria to decide whether an image was relevant or not, by means of a procedure that is described as follows: for each topic, a list of images from the corpus was generated, filtered by similarity and sorted descendantly. Afterwards, for each image within the list, the typical deviation of each image and its previous ten similar ones was calculated. When this value was under 0,01, i.e. when the similarity of this image with the previous ten was very close, the list was "cut" in that point, and the images provided were the previous ones. The reason for selecting this method lies on the observation of the similarities in such ordered list and the fact that the similarity within the images at the first position decreased significantly, and after these first position the way it decreased was softer.

Probably the use of a lower value would have been more suitable. Above all, perhaps we have not taken account of a very important factor, like the fact of that some "good" images, due to the calibration, may fall to the latest position of the list and be discarded. Besides, it seems to be logical that scanning the entire list, the number of images retrieved would be bigger; this possibility was discarded due to the lack of efficiency and speed of the overall system. In any case, it will be a very relevant issue to consider for the future.

In the same way, the characteristics that have been used in the development should be revised (probably the addition of anyone else should be desirable and even required), as well as a significant improvement of the efficiency in the calibration process so as to get the best fits in less time; and finally, to make the process to be adapted depending on the source of the images, making the algoritm execute in a different and optimum fashion.

References

- [1] X. Wei D. Raccoceanu D. Le Thi Hoang R. Teodorescu N. Vuillemot C. Lacoste, J. Lim. Ipal knowledge-base medical image retrieval in imageclefmed 2006. IPAL French-Singaporean Joint Lab, 2006.
- [2] R. Sánchez-Reillo (2) C. Sánchez-Ávila (1). Sistemas de identificación biométrica mediante patrón de iris utilizando representación multiescala e información de fase. (1) Departamento de Matemática Aplicada a las Tecnologías de la Información. E.T.S.I. Telecomunicación - Universidad Politécnica de Madrid (UPM) (2) Departamento de Ingeniería Eléctrica, Electrónica y Automática. Universidad Carlos III de Madrid (UC3M).
- [3] F. J. Suárez López F. Martín Rodríguez. Identificación dactilar basada en filtros de gabor. Departamento de Teoría de la Señal y Comunicaciones. Universidad de Vigo.
- [4] B. C. Desai P. Bhattacharya M. M. Rahman, V. Sood. Cindi at imageclef2006. image retrieval & annotation tasks for the general photographic and medical image collections. Dept. of Computer Science & Software Engineering, Concorcudia University, 2006.

- [5] M. Sanderson P. Clough, H. Müller. The clef 2004 cross-language image retrieval track. *CLEF 2004*, LNCS 3491:597–613, 2004.
- [6] D. Heesch S. Rüger P. Howarth, A. Yavlinski. Visual features for content-based medical image retrieval. Multimedia Information Retrieval Team, Imperial College London.
- [7] C. Mollet R. Besançon. Using text and image retrieval systems: Lic2m experiements at image-clef 2006. *Grupo CEA-LIST/LI2CM en Fontenay-aux-Roses - CEDEX*, 0:0, 2006.
- [8] P. Kruizinga S. E. Grigorescu, N. Petkov. Comparison of texture featres based on gabor filters. *IEEE Transaction on Image Processing*, 11:10, 2002.