Using Local Binary Patterns and Gaussian Mixture Models to Bridge the Semantic Gap in Content-Based Image Retrieval

Oussama AIADI*, Belal KHALDI and Mohammed Lamine KHERFI
Univ. Ouargla, Faculté des Nouvelles Technologies de l'Information et de la Communication, Laboratoire LAGE, Ouargla 30000, Algérie.
*Email: o_aiadi@hotmail.com

Abstract: Content-Based Image Retrieval (CBIR) engines are systems aiming at using the visual features of images in order to find their relevant. Despite the significant efforts that have been made by researchers to develop CBIR systems, they still suffer from the semantic gap between low level image features and high level user concepts. In this paper, we propose a fully automatic learning-based method to bridge this gap. Our method uses a Gaussian Mixture Model (GMM) as a visual model for each concept, where each component within it group images having the same visual appearance. Our method presents a multitude of advantages: 1) allows user to naturally express their needs using a textual query; 2) permit to retrieve images from unlabeled collections using a textual query; 3) It is fully automatic, as it doesn't require any human intervention. Experimental results show the efficiency of our method and a high accuracy in retrieval has been achieved.
Keywords: CBIR, Semantic gap, Supervised learning, GMM.

1. Introduction

With the noticeable prevalence of the image acquisition devices such as smart phone cameras, there has been an explosive growth of image collections size. Hence, quickly and accurately finding targeted images have become a challenging task. Image search engines have been appeared to accomplish this task and satisfy user's needs. Currently, there are two basic methods being adopted in these engines. One is content-based image retrieval (CBIR) and the other is text-based image retrieval (TBIR).

In TBIR, user is allowed to naturally express their needs using a textual query, and then desired images are retrieved based on matching the query with image annotations. Therefore, TBIR performance is totally dependent on the availability of such annotations [1]. In order to overcome TBIR drawbacks, many CBIR systems have been proposed [2 - 9]. In CBIR, user supplies a query image which is supposed to be similar to the desired ones; Afterwards relevant images are identified based on low level image features such as color, texture, shape. However, despite that CBIR systems have attracted much attention from researchers, their performance is still unsatisfactory. One of the main issues behind this limitation is the semantic gap between low level image features and high level concepts. The semantic gap occurs because of the possible contradiction between two interpretations, one by user and the other by the machine, that may be given to the same image [10, 11].

In this paper, we propose a method to bridge the semantic gap in CBIR by leaning a visual model for each concept using machine learning techniques. In particular, we model each concept with a Gaussian Mixture Model (GMM), where each component represents a possible appearance of the concept. During retrieval, user can naturally express their needs using a textual query; relevant images are the ones maximizing the likelihood to the GMM corresponding to the query. Our method has the advantage of retrieving images using a textual query in spite of the lack of textual descriptions with those images. In addition, it is fully automatic, as it avoids any kind of human intervention such as the relevance feedback used by the methods presented in [12, 13]. Experimental results given at the end of this paper prove the effectiveness of our method.

2. Proposed method

The aim of our method is to bridge the semantic gap in CBIR using machine learning techniques. In particular, we learn a visual model of each concept using a set of images that are labeled with this

concept. Those images are called training images, we denote the training set of a concept *C* by T_C. The steps of our method are the following:

- Features extraction: in this step, three features namely color histogram, Hu moments and Local Binary Patterns (LBP) are extracted from each T_C.
- Clustering: After features having extracted, we cluster T_C into three clusters, we get the set of clusters $Clust_{T_C} = \{C_k, k = 1, \dots, 3\}$. This process is done in order to assemble visually similar images in the same cluster i.e., each cluster C_k represents one appearance for the concept.
- Modeling: probability densities of clusters within each $Clust_{T_C}$ are fused in a GMM representing the concept's visual model. Parameters of the GMM are firstly initialized using the k-means algorithm and then estimated using Expectation- Maximization (EM) algorithm.
- Retrieval: after a textual query is supplied by user, relevant images are the ones having obtained the highest probability scores by evaluating the GMM associated with the query concept.

More details about our method are given in the sub-sections below:

2.1. Features extraction

Determination of the best feature combination that permits to effectively distinguish relevant images from irrelevant ones during retrieval is a critical task. That's because of the overlap between visual characteristics of these images. In this work, three features are employed which are: color histogram, Hu moments and Local Binary Patterns (LBP).

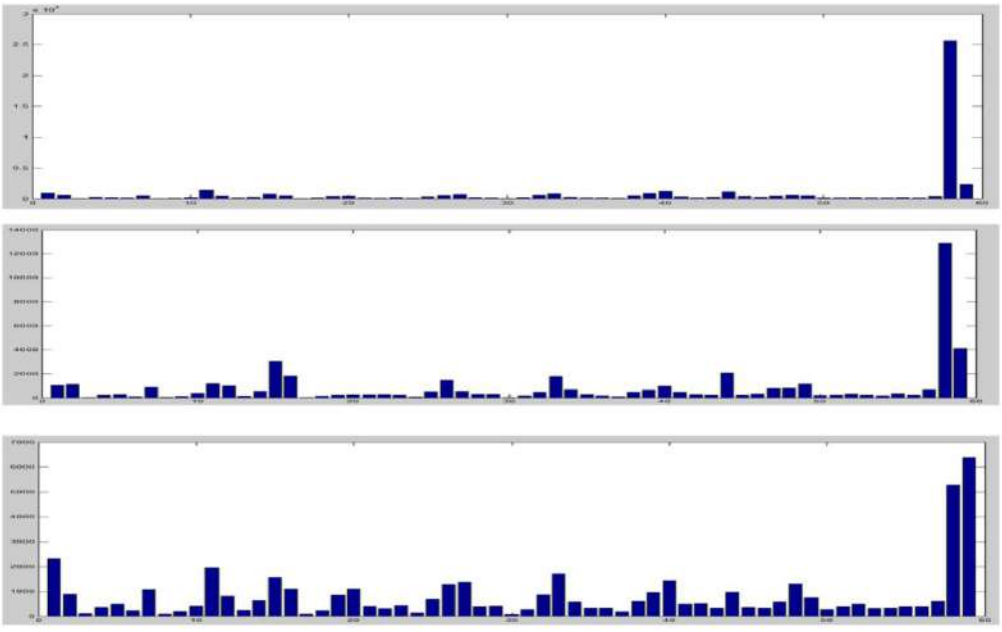


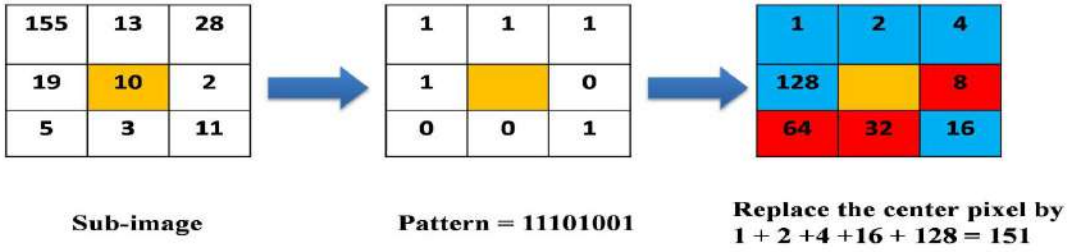
Figure 1 LBP histograms of three images from different classes

2.1.1. Color histogram

Due to its simplicity, efficiency, ease to be implemented and invariance to image size, RGB based color histogram has been chosen as a color descriptor. For storage and time consuming reasons, a quantization of RGB color space, which consists in subdividing each of the three RGB channels into three bins, is done. A feature vector, which represents the twenty seven bins of the histogram is obtained by quantization, is extracted from each training image. Each bin in the histogram represents the percentage of the related color within the image.

2.1.2. Local Binary Patterns (LBP):

Using color histogram only as a feature is practically insufficient because images could contain some objects with the same color but with different shape and texture. Therefore, and so as to assure achieving a maximum performance by our method, we combine the color histogram with the LBP which describes image texture. LBP has proven its ability as a powerful feature in many applications including texture classification [14] and face recognition [15]. The basic LBP operator [14] is computed by assigning each pixel within the image with a binary code of eight bits. This code is computed by considering the center pixel of a 3*3 neighborhood as a threshold value. The decimal value that corresponds to the binary one is then used to replace the original value of the center pixel. The appearance frequency of each code, called pattern, in the image is then computed to construct a histogram of 256 (2^8) dimensions. Figure 1 shows an example of the LBP calculation. According to [14], there exist 58 amongst the 256 pattern that provide more information



than others, which make possible to use only a small subset of patterns to describe image texture. These patterns are called uniform and contain at most two contiguous of bit suits. Later on, LBP was extended to neighborhood with different sizes [16]. Figure 2 shows the LBP histogram of three images from different classes.

Figure 2 Example of LBP calculation

2.1.3. Hu moments:

Hu moments [17] are shape features which are invariant to scale, position and rotation. The geometric moment of (p+q) order of an image I_{xy} is given by

$$m_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} x^p * y^q * I_{xy} \quad (1)$$

Where M and N are the image dimensions and I_{xy} denotes the value of the pixel has the coordinates x and y.

The center of mass of I_{xy} is defined as

$$(\bar{x}, \bar{y}) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (2)$$

The centralized moments of I_{xy} are invariant under position. Those moments are given by

$$\mu_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} (x - \bar{x})^p * (y - \bar{y})^q * I_{xy} \quad (3)$$

The normalized central moments are scale invariant, they are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \text{ Where } \gamma = \frac{p+q}{2}, \forall\, p + q \geq 2 \quad (4)$$

The seven moments of Hu are given with

$$I_1 = \eta_{20} + \eta_{02} \quad (5)$$

$$I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (6)$$

$$I_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (7)$$

$$I_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \quad (8)$$

$$I_5 = (\eta_{30} + 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{21} - \eta_{03})^2 + (\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (9)$$

$$I_6 = (\eta_{20} + \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})] \quad (10)$$

$$I_7 = (3\eta_{12} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (11)$$

2.2. Clustering

A training image set associated with a given concept includes images that have different visual appearances. In order to discover the concept's visual appearances and assemble visually similar images together, each set *Tc* is clustered into three clusters based on the features previously extracted. Each of those clusters represents images that have the same visual appearance. We use the well-known k-means algorithm for that purpose; it is an iterative algorithm which consists in two main steps:

1- Initialization: set *K* points as centers of clusters.
2- Assign each point to the closest cluster center.
3- After assignment having reached, clusters centers are updated.
4- Repeat the steps 2 and 3 until convergence.

At the end of this step, for each *Tc*, we get the set of clusters $Clust_{T_C} = \{C_k, k = 1, \dots, 3\}$ where each cluster includes images that have the same visual appearance.

2.3. Concepts modeling using GMM and EM

We model each cluster within $Clust_{T_C}$ with the probability density function of M-dimensional Gaussian distribution which is given by:

$$G(x|\theta_{C_k}) = \frac{1}{\sqrt{|\Sigma_k|(2\pi)^M}} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma^{-1}(x-\mu_k)}, \quad k=1, \dots, 3 \quad (12)$$

where x is a M-dimensional data point, $and\ \theta_{C_k}$ denotes the parameters of the Gaussian distribution that corresponds to the k^{th} cluster, these parameters are:

μ_k: The mean vector of the points belonging to the cluster.
$\sum_k$: The covariance matrix.

Probability density functions (pdf) that correspond to the set of clusters $Clust_{T_C} = \{C_k, k = 1, \dots, 3\}$

of a concept C, were incorporated in a Gaussian Mixture Model (GMM) representing the visual model of this concept. This GMM is given by:

$$P_C(x|\theta_C) = \sum_{k=1}^{3} w_{C_k} G(x/\theta_{C_k}), \theta_C = \{\theta_{C_k}, k = 1, \dots, 3\}, \quad (13)$$

where w_{C_k} denotes the weight of the k^{th} distribution of a concept *C* and θ_C the parameters of Gaussian component densities of the mixture P_C.

The likelihood of the data that belong to $Clust_{T_C}$ is given by:

$$L = \prod_{i=1}^{N_V} P_c\,(x|\theta_C), \tag{14}$$

where N_V denotes the number of data points that belong to $Clust_{T_C}$.
Expectation-Maximization (EM) algorithm is then used to maximize L and estimate the parameters of θ_C.

1. Expectation step:

$$y_{ij} = \frac{\frac{w_j}{\sqrt{|\Sigma|\,(2\pi)^M}}\, e^{-\frac{1}{2}(x_i-\mu_j)^T \Sigma^{-1}(x_i-\mu_j)}}{\sum_{i=1}^{N_V} y_{ij}} \tag{15}$$

2. Maximization step:

$$w_j = \frac{\sum_{i=1}^{N_V} y_{ij}}{N_V}, \tag{16}$$

$$\mu_j = \frac{\sum_{i=1}^{N_V} x_i\, y_{ij}}{\sum_{i=1}^{N_V} y_{ij}}, \tag{17}$$

$$\Sigma_j = \frac{\sum_{i=1}^{N_V} y_{ij}\,(x_i-\mu_j)(x_i-\mu_j)^T}{\sum_{i=1}^{N_V} y_{ij}}, \tag{18}$$

These two steps are repeated for $i = 1,\ldots, N_V$ and $j = 1,\ldots, 3$ until convergence.

2.4. Image retrieval

The previous steps represent the learning process which is accomplished offline. Then, in order to retrieve desired images, user supplies a textual query which represents a concept previously learnt. The retrieval procedure is carry out as follow, each test image I is assigned with a probability score obtained by evaluating the GMM associated with the query concept, where the feature vector $\mathbf{x}_i$ extracted from I is taken as parameter. Images have the highest probability scores are considered relevant to the query. In other words, relevant images are the ones for which the likelihood of belonging to the GMM is maximal.

3. Experimental setup

This section is devoted to provide details about settings of our experiments.

- Dataset

  In this work, experiments were conducted on the well-known Columbia University Image Library (COIL-100) dataset [18]. It is composed of 7200 images from 100 different object classes (72 images per class). Figure 3 shows some representative images from COIL-100. Sixty (60) images from each class were used for leaning purposes, whereas, the remaining (12) is intended for testing.

- Performance metrics

  We use two metrics to measure the performance of our method. These metrics are the precision and the recall, which are defined as:

$$\text{Precision} = \frac{\textit{Number of relevant retrieved images}}{\textit{total number of retreived images}}$$

$$\text{Recall} = \frac{\textit{Number of relevant retrieved images}}{\textit{total number of relevant images in the dataset}}$$

Figure 3 some representative images from different classes in COIL-100 dataset

4. Experimental results

In order to test the effectiveness of our method, we conduct a retrieval experiment on the COIL-100 dataset. The aim of this experiment is to assess the performance of our method in terms of precision and recall. After user supplied a textual query, our method is asked to retrieve 10 relevant images to the query. To do so, we perform the retrieval as described in the previous section. As a second aim for this experiment, we investigate the performance of each of the features separately. Figure 4 summarizes the obtained results.

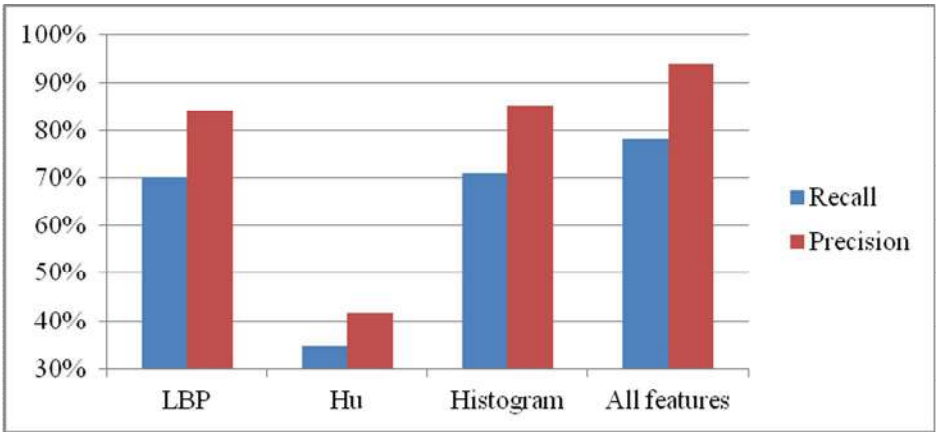


Figure 4 precision and recall yielded by our method

From Figure 4 we notice that LBP and color histogram have roughly an equal performance. In addition, Hu moments have yielded the lowest precision because of the high resemblance between

the images from the different classes in terms of shape. However, by combining the three features, our method has achieved a high precision and recall with 93.80% and 78.17% respectively.

5. Conclusion

The performance of CBIR systems is still limited because of the semantic gap between low level image features and high level user semantics. In this paper, we have presented a method to bridge the semantic gap in CBIR by leaning a visual model for each concept using machine learning techniques. Our method allows user to retrieve images from unlabeled collections using a textual query. Moreover, it doesn't require any human intervention. Experimental results conducted using the well-known COIL-100 dataset have demonstrated the effectiveness of our method.

References

[1] L. Wu, R. Jin and A. K. Jain; "Tag completion for image retrieval", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. **35**, No. 3, pp. 716-727, (2013).

[2] V. E. Ogle and M. Stonebraker; "Chabot: Retrieval from a relational database of images", *IEEE Computer*, Vol. **28**, No. 9, pp. 40–48, (1995).

[3] A. Pentland, R. Picard, and S. Sclaroff; "Photobook: Content-Based Manipulation of Image Databases", *International Journal of Computer Vision*, Vol. **18**, No. 3, pp. 233-254, (1996).

[4] J. Z. Wang, J. Li, and G. Wiederhold; "SIMPLIcity: Semantics-sensitive integrated matching for picture libraries", *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. **23**, No. 9, pp. 947–963, (2001).

[5] Y. Rui, T. S. Huang, and S. Mehrotra; "Content-based image retrieval with relevance feedback in MARS", in Proc. *IEEE Int. Conf. Image Processing*, Santa Barbara, CA, pp. 815–818, (1997).

[6] I. J. Cox, M. L. Miller, S. M. Omohundro, and P. N. Yianilos; "PicHunter: Bayesian relevance feedback for image retrieval", inProc. *Int. Conf. Pattern Recognition*, Vienna, Austria, pp. 361–369, (1996).

[7] J. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Gorowitz, R. Humphrey, R. Jain, and C. Shu; "Virage image search engine: An open framework for image management", inProc. *SPIE Conf. Storage and Retrieval for Image and Video Databases IV*, San Jose, CA, (1996).

[8] M. L. Kherfi, D. Ziou, and A. Bernardi; "Combining positive and negative examples in relevance feedback for content-based image retrieval", *J. Vis. Commun. Image Represent.*, Vol. **14**, No. 4, pp.428–457, (2003).

[9] M. Flickner, H. Sawhney, W. Niblack, J. Sahley, Q. Huang, 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**, No. 9, pp. 23–32, (1995).

[10] Y. Liu, D. Zhang and G. Lu ; "A Survey of Content-based Image Retrieval with High-level Semantics", *Pattern Recognition*, Vol. **40**, No. 1, pp.262-282, (2007).

[11] A.W.M. Smeulders, M. Worring, A. Gupta, R. Jain; "Content-based image retrieval at the end of the early years", *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. **22**, No. 12, pp. 1349–1380, (2000).

[12] A. Saju, I. T. B. Mary , A. Vasuki , P. S. Lakshmi; "Reduction of semantic gap using relevance feedback technique in image retrieval system", international conference on Applications of Digital Information and Web Technologies, pp.148-153 (2014).

[13] K. K. Kumar. and T. V. Gopal; "Multilevel and multiple approaches for Feature Reweighting to reduce semantic gap using relevance feedback", International conference on Contemporary Computing and Informatics (IC3I), pp-13-18 (2014).

[14] M. Topi, O. Timo, P. Matti, S. Maricor; "Robust texture classification by subsets of local binary patterns", in Proc. of *International Conference on Pattern Recognition*, Vol. **3**, pp. 935–938, (2000)

[15] T. Ahonen, A. Hadid, M. Pietikäinen; “Face Description with Local Binary Patterns: Application to Face Recognition”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. **28**, No. 12, pp. 2037–2041, (2006).

[16] T. Ojala, M. Pietikäinen, T. Mäenpää; “Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. **24**, No. 7, pp.971–987, (2002).

[17] H. Ming-Kuei; "Visual pattern recognition by moment invariants", *Information Theory, IRE Transactions*, Vol. **8**, pp. 179-187, (1962).

[18] S. A. Nene, S. K. Nayar and H. Murase; "Columbia Object Image Library (COIL-100)", *Technical Report CUCS-006-96*, (1996).