Labeled Images Verification Using Gaussian Mixture Models

Micheal Baechler
University of Fribourg
Department of Informatics
Bd de Pérolles 90
1700 Fribourg, Switzerland
micheal.el-betjali@unifr.ch

Jean-Luc Bloechle
University of Fribourg
Department of Informatics
Bd de Perolles 90
1700 Fribourg, Switzerland
jean-luc.bloechle@unifr.ch

Jean Hennebert*
University of Applied Science
Institute of Business
Information Systems
TechnoArk 3
3960 Sierre, Switzerland
jean.hennebert@hevs.ch

ABSTRACT

We are proposing in this paper an automated system to verify that images are correctly associated to labels. The novelty of the system is in the use of Gaussian Mixture Models (GMMs) as statistical modeling scheme as well as in several improvements introduced specifically for the verification task. Our approach is evaluated using the Caltech 101 database. Starting from an initial baseline system providing an equal error rate of 27.4%, we show that the rate of errors can be reduced down to 13% by introducing several optimizations of the system. The advantage of the approach lies in the fact that basically any object can be generically and blindly modeled with limited supervision. A potential target application could be a post-filtering of images returned by search engines to prune out or reorder less relevant images.

Categories and Subject Descriptors

I.5.2 [**PATTERN RECOGNITION**]: Design Methodology—*Feature evaluation and selection, Pattern analysis.*

General Terms

Algorithms, Design, Experimentation.

Keywords

Image Processing, Gaussian Mixture Model, Likelihood Ratio Detector.

1. INTRODUCTION

Automatic recognition of objects in images is a task that has been widely studied in the past decades. Applications are numerous in different fields such as medical image analysis, supply/processing chain supervision or traffic control [9, 4, 10]. Generally speaking, a recognition task can be in identification or verification mode. In identification mode, the system answers the following question: which object amongst a set of N objects is present in a given image (1:N classification). In verification mode, the system verify the presence of a claimed object, then answering a yes/no question (1:2 classification).

In our work, we are interested in improving the performances of web image search engines using image modeling methodologies. Usually the images that are returned by images search services correspond loosely to the entered keyword. We performed experiments on a dozen keywords using publicly available indexation services and we roughly measured that the quantity of unrelevant images can be as high as 50%. The reason for such bad performances is probably linked to the fact that these services use only the text surrounding the image to build the indexation tables.

To improve this situation, we are proposing to use a post-filtering method applied to the set of images returned by the search engine. The filtering is based on the real content of the image by computing a matching score between the image and a model associated to the keyword. The system is therefore performing a verification for each image returned by the search engine, answering a yes/no question about the matching of the image with the keyword used for the search. In our approach, this keyword is supposed to indicate an object or a pattern that is present in the image. We also make the assumption that a set of T images is available for training our models. The learning phase is therefore supervised, but we keep as target to use technologies that could potentially be used in an unsupervised manner.

The novelty of our approach is in the use and improvement of a generic system based on Gaussian Mixture Models (GMM) that are used to compute the likelihood value of a set of observations given a model (associated to the keyword). Such models are versatile and have been proposed in several pattern recognition tasks such as speech, handwriting, biometrics and even image recognition [14, 7, 1]. The GMMs are fed by a feature extraction module computing Discrete Cosine Transform features using a sliding window on top of the image. More specifically, we are proposing in this paper several improvements that are specific to the post-filtering task explained above.

This paper is organized as follows. In Section 2, an overview

[*] Jean Hennebert is also affiliated with the University of Fribourg.

SAC'09 March 8-12, 2009, Honolulu, Hawaii, U.S.A.

of several approaches of detection and recognition of objects in image is given. Section 3 describes the fundamental principles of our feature extraction and modeling scheme. We present in section 4 the database and the protocol which are used to evaluate our system. Section 5 presents several improvements that are specific to the task of automatic indexation of images. Finally, conclusions and future work are presented.

2. RELATED WORK

Automatic recognition of objects in images is a task that has been widely studied in the past and the applications are numerous. We are interested here in modelling schemes that are generic, i.e., that are not specific to a given shape, illumination and size of object in an image. In this direction, several machine learning methods have already been proposed and studied.

Some approaches are based on the automatic detection of spatial configuration of local features [13]. The configurations that are occurring frequently on the object are retained to build the model. This approach can be actually seen as an intermediate processing layer able to filter out the large amount of features and hence to facilitate the recognition task.

The recognition in itself can be performed using well known pattern matching algorithms such as Artificial Neural Networks (ANN), K-nearest neighbors (KNN), Support Vector Machines (SVM) [11, 3, 2]. Hybrid approaches have also been proposed to take advantages of the strengths of different approaches. For example, in [18], a hybrid approach is proposed where a KNN system is used as a first step before feeding a more precise SVM system. The benefit of this approach is to reduce the time needed for a classification. Some other approaches are specifically crafted to learn models from few training images using, as prior information, models previously learnt [6].

The use of GMMs has been extensively studied in some pattern recognition tasks, such as in speaker recognition [14] and signature modelling [7]. Quite closely related to our application, GMMs have also been proposed for face verification systems [1].

3. SYSTEM DESCRIPTION

The general functioning of our system is illustrated in Fig. 1. It is composed of two distinct phases: the training (A) and testing (B) phases. The training phase aims at computing the different models that are going to be associated to the respective keywords. The training is supervised and, for a given keyword, the subset of corresponding images is selected from the database to train the model. The testing phase allows us to evaluate the effectiveness of the approach by performing the verification on a set of unseen images. The system is tested on a set of *true* images where the label corresponds effectively to the content of the image. We also test the ability of the system to reject *incorrect* images where the claimed label does not correspond to the content of the image. Training and testing phases share the same frontend composed of a preprocessing step used to normalize the images and a feature extraction block that transform the image into a set of features adequate for classification. The last block implements the GMMs training and testing.

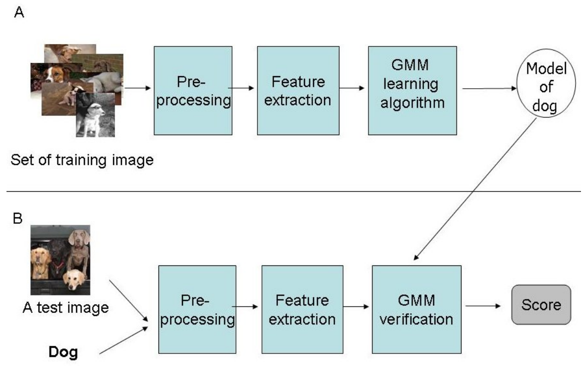

Figure 1: Training (A) and testing (B) phases.

3.1 Preprocessing

As our system is supposed to work on any images indexed on the web, the objective of the preprocessing step is to homogenize some basic characteristics. First, the size of the image is normalized to a constant value of 200 × 200 pixels. The value of 200 pixels has been chosen to keep the CPU load in acceptable ranges while conserving most of the visual information about the objects that are modeled. Second, the image is converted into gray scale coded on 8 bits. While the loss of colors is a loss of information, this conversion makes the system compatible with all input images and reduce the feature variabilities (hence reducing the needs of larger training set). Working in gray-scale is also a way to reduce the CPU load. Finally, a histogram normalization is performed to reduce the impact of intensity and illumination variation.

3.2 Feature Extraction

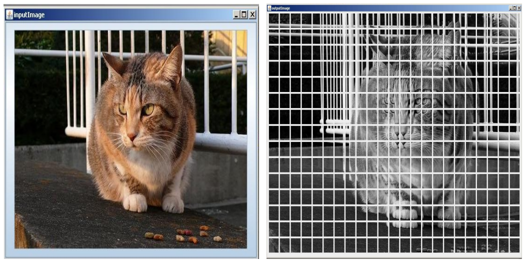

Figure 2: Decomposition in sub images.

The first step of our feature extraction block is to decompose the image in a set of smaller overlapping sub images by sliding a window in the X and Y directions. We used a sliding window of $P \times P$ pixels shifted of $P/2$ pixels and, in most of our experiments, we set $P = 16$. This decomposition is illustrated in Fig. 2. The next step is to compute Discrete Cosine Transform (DCT) coefficients on each sub image. Given a sub image $f(x, y)$, a transformation in terms of orthogonal basis function is applied:

$$C(u, v) = \alpha(v) * \alpha(u) \sum_{y=0}^{P-1} \sum_{x=0}^{P-1} f(y, x) * \beta(y, x, v, u)$$

for $u, v = 0, \ldots, P-1$ and where

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{P}}, & \text{for } u = 0 \\ \sqrt{\frac{2}{P}}, & \text{for } u = 1, 2, \ldots, P-1 \end{cases}$$

$$\beta(y, x, v, u) = cos\frac{(2y+1)u\pi}{2P}cos\frac{(2x+1)v\pi}{2P}$$

The DCT is actually used to express a sequence of data as a sum of cosine functions oscillating at different frequencies. Fig. 3 (left) illustrates the basis functions $\beta(y, x, v, u)$. The DCT matrix is then flattened in a vector by ordering the coefficients according to a zig-zag scheme illustrated in Fig. 3 (right). The M first coefficients are kept to obtain the vector $x_n = (c_0^n, c_1^n, \ldots, c_{M-1}^n)$ where n represents the index of the sub image in the set of N sub images. The DCT is well suited as feature extraction for image recognition as it allows measuring and distinguishing periodical patterns in images. The DCT is also widely used in signal processing to compress sequences of information as, for example, in JPEG [17]. A drawback of the DCT lies in its sensitivity to variations of illuminations, at least in the first DCT coefficients. To reduce this sensitivity, we use the *DCTmod2* which is a modified version of DCT where the first coefficients are replaced by differential values computed on the adjacent X and Y windows as explained in [16, 15].

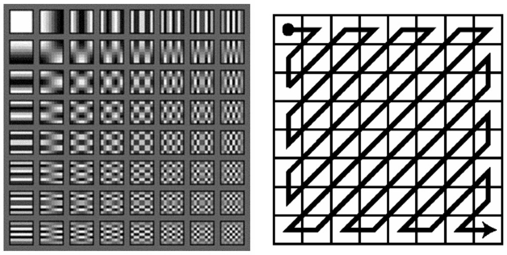

Figure 3: DCT basis functions and zig-zag pattern

3.3 Modeling by GMM

As result of the feature extraction, an input image is then represented by a sequence of DCTmod2 vectors $X = \{x_1, \ldots, x_N\}$. For a given image category c, our objective is to build a statistical model λ_c able to estimate the probability density function or *likelihood* of each sub image $p(x_i|\lambda_c)$. This can be achieved using GMMs where the likelihood is estimated as a weighted sum of multivariate Gaussian densities (see e.g. [14, 16]):

$$p(x_n|\lambda_c) = \sum_{i=1}^{I} w_i \mathcal{N}(x_n, \mu_i, \Sigma_i)$$

in which I is the number of mixtures, w_i is the weight for mixture i and the Gaussian densities $\mathcal{N}$ are parameterized by a $M \times 1$ mean vector μ_i, and a $M \times M$ covariance matrix Σ_i. The mixture weights w_i also satisfy the constraint $\sum_{i=1}^{I} w_i = 1$.

To reduce the CPU load (and also the number of parameters), we make the hypothesis that the DCT coefficients are uncorrelated which allows us to use diagonal covariance matrices. By making the hypothesis of sub image independence, the global *likelihood* score for $X = \{x_1, \ldots, x_N\}$ is simply computed with a product of the likelihoods:

$$S_c = p(X|\lambda_c) = \prod_{n=1}^{N} p(x_n|\lambda_c)$$

We also compute the likelihood score S_w of the hypothesis that X is **not** from the given category using a so-called world model λ_w trained by pooling a large number of images of unrelated categories. The likelihood S_w is computed in a similar way, by using a weighted sum of Gaussian mixtures. The optimal decision whether to reject or to accept the claimed belonging to a category is performed comparing the ratio of both scores S_c and S_w against a global threshold value θ. The ratio is often computed in the log-domain to handle the computation with very small numbers:

$$R_c = \log(S_c) - \log(S_w)$$

The training of the world models λ_w is performed with the Expectation-Maximization (EM) algorithm [12] that iteratively refines the component weights, means and variances to monotonically increase the likelihood of the training feature vectors. We also apply a simple binary splitting procedure to increase the number of Gaussian components to a predefined value. The training of the category model λ_c is performed by adapting the world model parameters λ_w using the Maximum A Posteriori algorithm [14].

The verification decision is then taken according to :

$$\begin{cases} X \text{ contains object category } c, & \text{if } R_c \geq \theta \\ X \text{ does not contain object category } c, & \text{if } R_c < \theta \end{cases}$$

4. EVALUATION PROTOCOL AND IMAGE COLLECTION

All experiments have been performed on the database *Caltech 101* [6, 5]. It is a data set containing 9,144 images spread into 101 different object categories, plus a background category. Every category contains from 40 to 800 images and most objects are centered in the foreground area and are in a stereotypical pose. These images were collected via the internet service Google image search. As illustrated in Fig. 4, Caltech 101 also defines for each image the region of interest which is a more precise location of the object within the images. We actually compared the impact of modeling only the region of interest versus modeling the whole images in our evaluations. As the definitions of region of interests are only available for 98 categories, we limited ourselves to the use of these 98 categories.

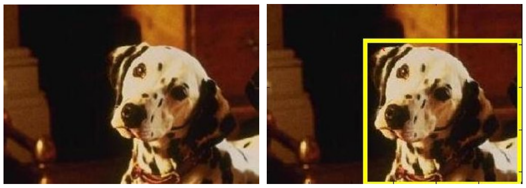

Figure 4: Sample image from Caltech 101.

All our experiments are performed using the same evaluation protocol on Caltech 101. An evaluation batch is actually composed of 10 independent evaluations using different partitions of the data set. For each partition, we choose randomly two sets of images from each object category. The first one is used as training set and the second one as set of relevant test images. A set of non-relevant images is built by randomly choosing images from the other categories. Fig. 5 illustrates schematically the constitution of training and test sets for the evaluation of category 1. An equal number of images is used for training the models of each category and, as explained below, we varied this number from 5 to 30 to see the impact of having more training data. During testing, the system can make two types of error. We measure a percentage of false rejection (FR) by counting the number of relevant images falsely rejected. A percentage of false acceptation (FA) is also measured by counting the amount of non-relevant images incorrectly accepted. These rates are actually varying as a function of the rejection threshold θ. A specific value of θ is leading to equal values of FR and FA, which is the definition of the so-called Equal Error Rate (EER) for detection tasks. For all our experiments, we used the EER as operating point. As we use 10 partitions of the data set, the EER for a given category is computed as the average of the 10 EER obtained with the partitions.

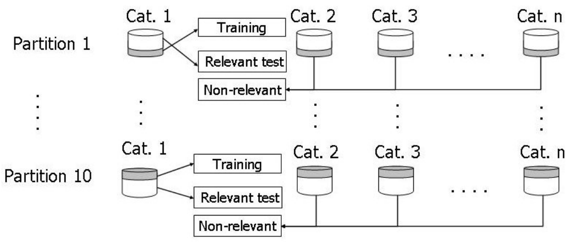

Figure 5: evaluation protocol on caltech 101

5. EXPERIMENTS

As a general strategy, we compared different approaches by varying a single parameter and explored systematically the impact of this parameter. Many different experiments were performed and we present here a summary of the most significant optimization and improvements. We started from a baseline system with 128 Gaussians trained using 10 images for each category and using only the region of interest defined in the annotation files. The world model for this system was trained using 400 images taken from the 'background' category of Caltech 101. This system gave us an overall EER of 27.4%.

Number of training images. We investigated the impact of using more or less training data to build the models. We trained with 5, 10, 15, 20, 25 and 30 images and, as expected, the more the data to train the GMMs, the best are the EER. The observed EER values were, respectively, 29.6%, 27.4%, 26.2%, 25.0%, 24.3% and 23.2%.

Parameters of the training. As suggested in some papers (for ex. [14], some pattern recognition tasks seem to benefit from adapting only the average of the Gaussians. This configuration did not bring improvement for our task. We also varied the number of EM and MAP iterations and observed that a slight improvement can be obtained increasing the number of iterations from 20 to 25. Our best previous score 23.2% moved down to 23.0%.

Modeling whole images. We wanted to evaluate the importance of limiting the modeling to the region of interest versus using all the image. Surprisingly, by modeling the whole image, the EER dropped from 23.0% to 21.7%. An interpretation to this is potentially in the fact that background of images are adding contextual information to a given object. For example, having a road in background adds to the confidence that the claimed object is a car.

Modeling center of images. Objects are usually centered in images. In this experiment, we tried to give more importance to the central part of the image, without using the definition of regions of interests available in Caltech 101. We therefore applied the feature extraction to the central part of the image, limiting blindly ourself to an area of 100 × 100 pixels. The focus on the central part increased the EER from 21.7% to 22.0% but interestingly, probably for the same reason as in the previous experiments.

Number of Gaussians. The main advantage of GMMs is in their ability to model complex forms of probability density functions by increasing the number of mixtures I. However, increasing I also means that more parameters have to be estimated, which requires larger training sets. It also means more computations. Starting from the 128 Gaussians system trained on whole images, we increased exponentially the number of Gaussians up to 1024. As illustrated on the curve A of Fig. 6, we observed a decrease of the EER down to 20.6% with 1024 Gaussians. Curve B on this Figure was obtained when modeling the central part of the image. Interestingly, going up to 1024 Gaussians seems less feasible in this case as fewer training data is available. As shown looking at the minimum of curve B, 512 Gaussians give the best results for this configuration.

Fusion of models. As illustrated with curve C in Fig. 6, a simple summation based fusion of the scores of the two systems modelling the whole image and the central part of the images lead to a significant improvement of the performances. The EER is decreased down to 17% with a system using 256 Gaussians. The explanation is to be found in the fact that both systems are modelling different kind of information. The first one model the whole image including 4 times more sub windows with many features associated to the background than the second one.

Further improvements. We could reduce down further the EER to 13% by introducing 3 modifications to our system. First, we removed sub-windows that are almost fully black or white as they do not bring much discriminant information. Second, we introduced multi-scales of the analysis window used for the feature extraction by adding 32 × 32 sizes to the 16 × 16 sizes. Third we performed a score normalization by dividing the score R_c obtained for a given image and category by the average of the scores obtained for the same image fed into all categories.

The different improvements that we have introduced allowed us to reduce by a factor of 2 the EER, going from 27.4% measured for the baseline system down to 13% for the final system. These improvements are summarized in Table 1.

6. CONCLUSIONS

We have presented a system for verifying that images are correctly associated to labels. The system is based on generic approaches that can potentially be applied to any images and labels with limited supervision. A potential application is the post-filtering of images returned by search engines to prune out or to reorder images that are less rele-

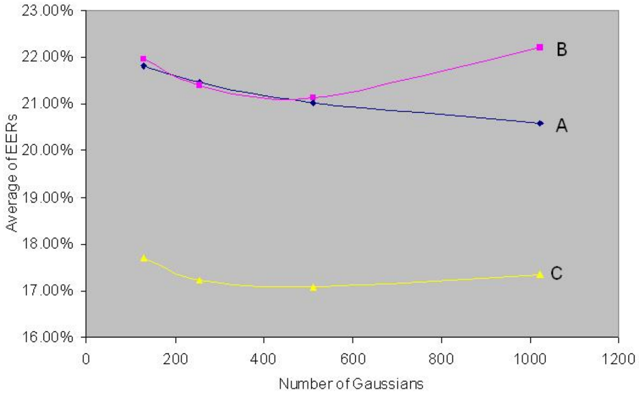

Figure 6: EER values as a function of the number of Gaussians, (A) modeling whole image, (B) modeling the central part of the image and (C) fusing A and B at the score level.

Table 1: Summary of EER performances.

Caltech 101	EER (%)
Baseline system	27.4
30 training images instead of 10	23.2
Tuning parameters of the training	23.0
Modeling whole images	21.7
Number of Gaussians up to 1024	20.6
Fusion of models	17.0
Further improvements	13.0

vant. The system is composed of a feature extraction module based on the computation of DCT coefficients and of a robust and flexible probability density function estimator based on GMMs. A tuning of the parameters of the system as well as several modifications at the feature extraction and modeling levels are introduced and justified. Evaluations of the system have been carried out on the Caltech 101 database, showing a reduction of the EER down to 13%. Considering that publicly available indexation engines are currently returning a large proportion of images that are not relevant to the search keyword (sometimes up to 50%), our approach could potentially be used to enhance their performance. Future work will be dedicated to evaluating our system on more realistic databases of images and on comparing our approach to competing systems. In this direction, we have identified the ImageCLEF evaluations and more specifically the "concept detection task" [8].

7. ADDITIONAL AUTHORS

Author 3: Andreas Humm (University of Fribourg, Department of informatics, email:`andreas.humm@unifr.ch`) and author 4: Rolf Ingold (University of Fribourg, Department of Informatics, email:`rolf.ingold@unifr.ch`).

8. REFERENCES

[1] F. Cardinaux, C. Sanderson, and S. Bengio. User authentication via adapted statistical models of face images. *Signal Processing, IEEE Transactions on*, 54(1):361–373, Jan. 2006.

[2] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, March 2000.

[3] B. V. Dasarathy. *Nearest Neighbor (NN) norms: NN pattern classification techniques*. Los Alamitos: IEEE Computer Society Press, 1990.

[4] T. Deselaers, D. Keysers, and H. Ney. Fire – flexible image retrieval engine: Imageclef 2004 evaluation. volume 3491 of *LNCS*, pages 688–698, Bath, UK, 15/09/2004 2005. Springer.

[5] L. Fei-Fei, M. Andreetto, and M. A. Ranzato. Caltech 101 - image database, http://vision.caltech.edu/, 2003.

[6] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, In Press, Corrected Proof.

[7] J. Hennebert, A. Humm, and R. Ingold. Modelling spoken signatures with gaussian mixture model adaptation. *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 2:II–229–II–232, April 2007.

[8] ImageCLEF. Concept detection task, http://www.imageclef.org/2008/vcdt, 2008.

[9] T. Lehmann, M. O. Güld, T. Deselaers, D. Keysers, H. Schubert, K. Spitzer, H. Ney, and B. B. Wein. Automatic categorization of medical images for content-based retrieval and data mining. *Computerized Medical Imaging and Graphics*, 29:143–155, 03/2005 2005.

[10] T. Lim and A. Guntoro. Car recognition using gabor filter feature extraction. *Circuits and Systems, 2002. APCCAS '02. 2002 Asia-Pacific Conference on*, 2:451–455 vol.2, 2002.

[11] R. Lippmann. Pattern classification using neural networks. *Communications Magazine, IEEE*, 27(11):47–50, 59–64, Nov 1989.

[12] T. K. Moon. The expectation-maximization algorithm. *Signal Processing Magazine, IEEE*, 13(6):47–60, Nov 1996.

[13] T. Quack, V. Ferrari, B. Leibe, and L. V. Gool. Efficient mining of frequent and distinctive feature configurations. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.

[14] D. Reynolds, T. Quatieri, and R. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10:19–41, January 2000.

[15] C. Sanderson and K. Paliwal. Polynomial features for robust face authentication. *Image Processing. 2002. Proceedings. 2002 International Conference on*, 3:997–1000, June 2002.

[16] C. Sanderson and K. K. Paliwal. Fast features for face authentication under illumination direction changes. *Pattern Recogn. Lett.*, 24(14):2409–2419, 2003.

[17] A. B. Watson. Image compression using the discrete cosine transform. *Mathematica Journal*, 4:8–1, 1994.

[18] H. Zhang, A. C. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, 2:2126–2136, 2006.