Image Characterization from Statistical Reduction of Local Patterns

Philippe Guermeur and Antoine Manzanera

ENSTA - Elec. and Comp. Sc. lab
32 Bd Victor, 75015 Paris, France
philippe.guermeur@ensta.fr
http://www.ensta.fr/~guermeur

Abstract. This paper tackles the image characterization problem from a statistical analysis of local patterns in one or several images. The induced image characteristics are not defined a priori, but depends on the content of the images to process. These characteristics are also simple image descriptors and thus considering an histogram of these elementary descriptors enables to apply "bags of words" techniques. Relevance of the approach is assessed when dealing with the image recognition problem in a robot application framework.

Keywords: Image recognition, Vector quantization, Histogram comparison.

1 Introduction

Local image description techniques usually relate to interest point detection methods. Many image processing methods define the notion of interest point from using a theoretical model of gray level variations on a local image neighbourhood [1], [2], [3]. More recent approaches combine interest points to image descriptors. A typical approach is the SIFT method which computes histograms from gradient orientations near interest points locations. [4] gives an overview of these various approaches which have often been used in "bags of words" methods for image indexing applications or robot navigation tasks such as Simultaneous Localization and Mapping of the Environment (SLAM). All of these techniques define the notion of interest point from designing an a priori model of what is a corner or some other interesting geometric element and so elaborating a model of what is the ideal gray level variation in the neighbourhood of the corresponding image points.

Others approaches aim at analysing images through statistical studies concerning the appearance of local image configurations (patterns). Thanks to a coding phase the number of patterns can be decreased and the Zipf law can be applied to model their distribution in the images [5]. This model enables to characterize texture complexity but gives no piece of spatial information as for the interest area location. This difficulty can be got round by previously partitioning the image into small regions before carrying out any statistical analysis. This is

the way [6] characterizes every region with a 58-value local binary pattern (LPB) histogram and then concatenates the various image histograms to compose a single vector which is expected to be peculiar to the image being processed. This method has been efficiently experimented in facial expression recognition. In [7] a codebook is built from quantizing gray level image neighbourhoods and used within the context of texture recognition.

Our study is concerned primarily with automatic classification of basic image patterns, without any prior knowledge regarding their configuration. These patterns are local image neighbourhoods, we propose to transform before processing, in order to give the approach the desired invariance properties. How many classes are obtained, depends on a similarity threshold given as input to the algorithm. Next, we propose to characterize the images from the statistical properties of the detected patterns, through "bags of words" techniques. A pattern codebook is built from a subset of images (learning step) and then applied to generate a feature histogram for every image (analysis step). Section 2 gives details about this extraction of characteristics. The various histograms are stored in a database for being used as feature vectors later. Image recognition tasks can be made by comparing the new histogram with the registered individual histograms (section 3) and selecting the best match as the recognition result. Section 4 evaluates the effectiveness of this method in a piece recognition framework with several comparison functions.

2   Characteristics Extraction

This section deals with the technique of codebook generation, including a K-means classification method.

2.1   Algorithm

We consider very basic image structures composed of image neighbourhoods. For each neighbourhood, we assume to be of size k (let us note $k = n \times n$), we pick up the pixels intensities to compose vectors in a k-dimensional Euclidean space in case of a gray level image, or in a mk-dimensional space in case of a m-component image (e.g. colour image).

An incremental (K-means modified) clustering method is applied to construct the codebook (learning step). Then, during the analysis step every new vector is identified to a codeword. The codebook construction is based on three steps:

```
Step 1. Begin with an initial codebook.
Step 2. For each neighbourhood, find the nearest neighbour in the
codebook.
Step 3. If the distance to this codeword is less than a given
radius denoted dmax afterwards, compute the new centroid and
replace the codeword with it. Otherwise add the new word to the
codebook.
```

2.2   Preliminary Image Transformation

The clustering algorithm can be applied directly on the original image, but in order to give the algorithm some convenient invariance properties a preliminary image transformation should be useful. The overall process is illustrated in Fig. 1 where the pre-processing step is represented with the spatio-temporal function Ψ.
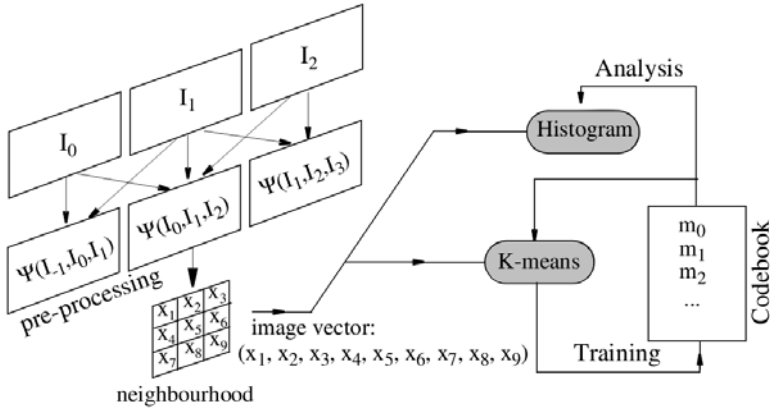

Fig. 1. Overall principle of the algorithm (with $k = 9$)

As we are more particularly concerned with invariance to lightning change, we propose an implementation in which the output of the pre-processing step represents the image gradient arguments. We suggest to code the resulting image data with 16 values, which means a $\pi/4$ quantization step. When considering colour images, only the argument of the gradient which have the maximal magnitude (computed as in [11]) is kept, if this magnitude is above a given threshold.

2.3   Distance Function

Whether being concerned with the training step or with the analysis step, the vectors extracted from an image have to be compared to those gathered in the dictionary. The metric to use depends on whether or not the image has been pre-processed.

Referring to an original image, we propose this metric to be represented by the Euclidian distance between the two vectors after they have been normalized. Let us assume that x and y are the two normalized vectors to compare, the distance between these vectors is given by:

$$E = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2} \tag{1}$$

As this metric is representative of the angle and does not consider the vector modulus, it is supposed to be invariant to illumination changes. The algorithm elaborated with this metric will be denoted algorithm 1 afterwards.

The alternative technique we propose to apply consists in pre-processing the original image in order to make each of the vector components invariant to illumination changes. So, we propose that this step comes down to replace each pixel by a representation of the argument of the gradient. In order to keep only the significant values, only the pixels whose gradient is above a given threshold are considered.

A quantization step of $\pi/4$ leads us to get a 16-value code that have to be completed with an additional value to code the non-significant data. Regardless to the original image the resulting data are reduced and a simple distance can be elaborated to compare two vectors. We propose to use the following one, based on the $\mathcal{L}_1$ norm:

$$E = \sum_{i=1}^{N} |\alpha_i - \beta_i| \tag{2}$$

where α_i and β_i are respectively the ith component of the two vectors to compare. The algorithm elaborated with this metric will be denoted algorithm 2 afterwards.

3   Histogram Comparison

The analysis step consists in searching for every image neighbourhood, the nearest vector in the codebook. The various results enable either to quantize the original image or to implement an efficient histogram algorithm. At the end of this step, any image is supposed to be characterized by a single histogram and so comparing the obtained histograms between them should indicate how much the corresponding images are similar. This section addresses this problem of histogram comparison through a brief presentation of a few possible comparison functions drawn from statistics, signal processing and geometry. Table 1 gives expressions for such convenient functions, most of them found in the literature, by dividing them into two groups according to their origin. In this table, we assume the vectors X and Y represent the two histograms to be compared, and the vector A is an average histogram as far as such a piece of information is available.

4   Application to Place Recognition

The experiments have been conducted on the INDECS database [9][10], which contains pictures of five different rooms acquired at different times of the day under different viewpoints and locations. The system is trained for all the pictures acquired under a given illumination condition and tested under the other illumination conditions for the remaining pictures. For a given picture, the aim is to recognize the room in which it has been acquired.

Table 1. Distance functions considered for histogram comparison

min/max based functions	χ^2 and Bhattacharyya [8] based functions
$\bigcap(X,Y) = \sum_i \min(X_i, Y_i)$	$\chi^2(X,Y) = \sum_i \dfrac{(X_i - Y_i)^2}{Y_i}$
$\bigcap_{av}(X,Y) = \sum_i \dfrac{1}{A_i} \min(X_i, Y_i)$	$\chi^2_m(X,Y) = \sum_i \dfrac{(X_i - Y_i)^2}{(X_i + Y_i)}$
$\bigcap_{yav}(X,Y) = \sum_i \dfrac{Y_i}{A_i} \min(X_i, Y_i)$	$\chi^2_{av}(X,Y) = \sum_i \dfrac{(X_i - Y_i)^2}{A_i(X_i + Y_i)}$
$\Psi(X,Y) = \sum_i \dfrac{\min(X_i, Y_i)}{\max(X_i, Y_i)}$	$Bha(X,Y) = \dfrac{\sum_i \sqrt{X_i Y_i}}{\sqrt{\sum_i X_i}\sqrt{\sum_i Y_i}}$
$\Psi_{av}(X,Y) = \sum_i \dfrac{\min(X_i, Y_i)}{A_i \max(X_i, Y_i)}$	$Bha_{av}(X,Y) = \dfrac{\sum_i \dfrac{\sqrt{X_i Y_i}}{A_i}}{\sqrt{\sum_i X_i}\sqrt{\sum_i Y_i}}$

The first tests have been done with our two algorithms in order to evaluate the different distance functions used for histogram comparison. They consist to evaluate the classification rate (that is the successful room recognition rate) for every image of the database, by giving each image the same weight. For better comparison with the results obtained in [9] (where the pictures containing less than 5 interest points were rejected) we choose to discard the images whose histogram contains no bin. This concerns about 1% of the total number of images. The results are plotted on figure 2 and figure 3. They indicate that the best results are obtained with Bhattacharyya metric when we consider algorithm 1 and the χ^2_m metric when we consider algorithm 2. Using this last metric, the classification rate of algorithm 2 are around 80% and the best rate is 82.42% with $dmax = 16$. The performances of algorithm 1 are weaker, but the overall performance increases when $dmax$ decreases. However, the evaluation has not been done for $dmax$ inferior to 0.35 as for this value the algorithm 1 has a great number of clusters (6 319 clusters) and consequently becomes too slow. Figures 4(a) and 4(b) illustrate the swift variation of the number of clusters when $dmax$ decreases, for both algorithms.

The classification rate is then calculated separately for each room and according to [9] a single measure of performance is computed by averaging all the results with equal weights. These results are plotted on figure 5 and figure 6, with regard to the results obtained in [9] and [10] which use a local feature detector constructed from the combination of the Harris-Laplace detector and the SIFT descriptor. These figures clearly indicate that the performance measure obtained with our algorithm is the best, regardless of which training set is chosen.
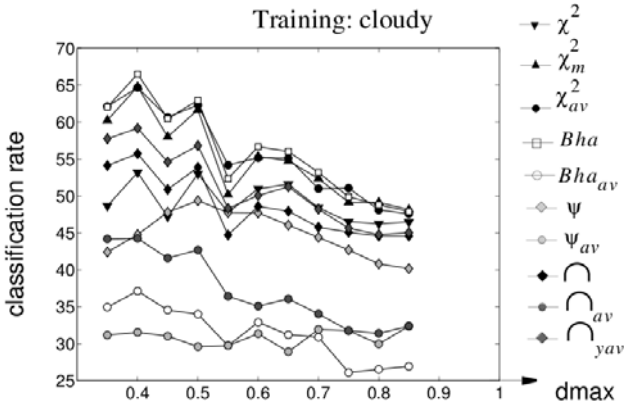
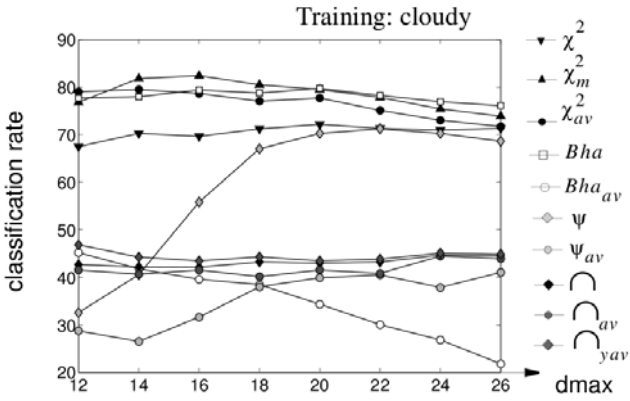**Fig. 2.** Global place classification rate using algorithm 1

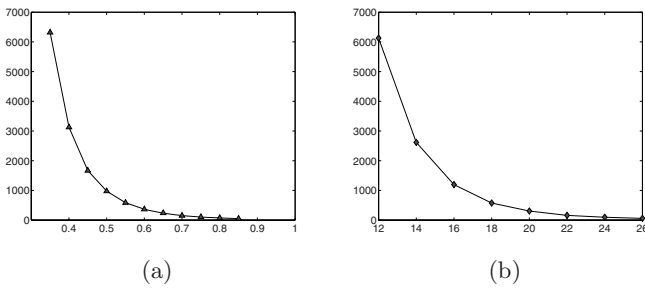**Fig. 3.** Global place classification rate using algorithm 2

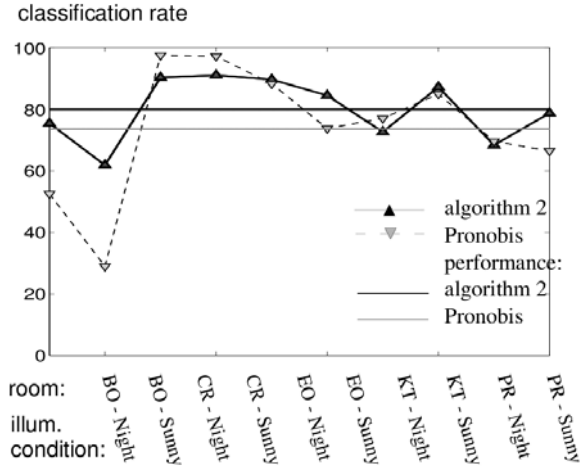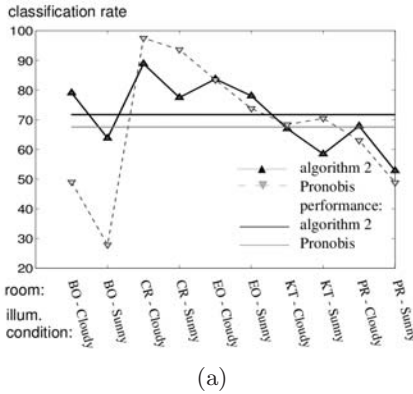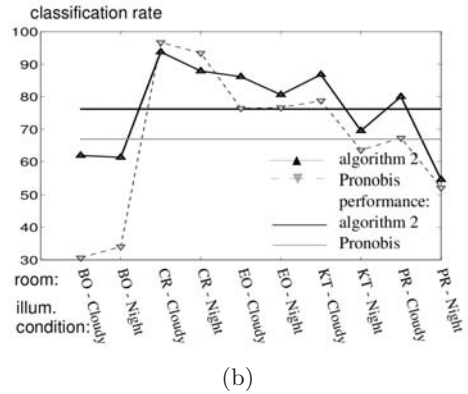**Fig. 4.** Number of clusters with regard to the radius *dmax*, using (a) Algorithm 1, or (b) Algorithm 2

Fig. 5. Classification rates and performance measure after training the algorithm with "cloudy" illumination condition

Fig. 6. Classification rates and performance measure after training the algorithm with: (a) "nighty" or (b) "sunny" illumination condition

5 Conclusion

This study has tackled the image characterization problem by proposing a new approach based on a statistical analysis. Though this approach is very simple, it has been proven to be efficient and it has been empirically validated in the robotic framework of place recognition. Studies are in progress in order to improve the classification rates (splitting of the image into small regions, higher order statistic reduction) or to lead to real-time implementations of the proposed method.

Acknowledgments

References

1. Wang, H., Brady, M.: Real-time corner detection algorithm for motion estimation. Image and Vision Computing 13(9), 695–703 (1995)
2. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceeding of the 4th Alvey Vision Conference, pp. 147–151 (1988)
3. Trajkovic, M., Hedley, M.: Fast corner detection. Image and Vision Computing 16(2), 75–87 (1998)
4. Mikolajczyk, K., Schmid, C., et al.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(10), 1615–1630 (2005)
5. Caron, Y., Charpentier, H., Makris, P., Vincent, N., et al.: Une mesure de complexité pour la détection de la zone d'intérêt d'une image. In: DGCI 2003 (2003)
6. Feng, X., Pietikäinen, M., Hadid, A., et al.: Facial Expression Recognition Based on Local Binary Pattern. Pattern Recognition and Image Analysis 17(4), 592–598 (2007)
7. Varma, M., Zisserman, A.: A Statistical Approach to Material Classification Using Image Patch Exemplars. To be published in IEEE PAMI
8. Thacker, N.A., Aherne, F.J., Rockett, P.I.: The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data, Kybernetika, Prague, June 9-11 (1997)
9. Pronobis, A.: Indoor Place Recognition Using Support Vector Machines, Master's Thesis in Computer Science (2005)
10. Pronobis, A., Caputo, B., Jensfelt, P., Christensen, H.I.: IA Discriminative Approach to Robust Place Recognition. In: ICIRS, Beijing, China (2006)
11. Zenzo, S.D.: A note on the Gradient of a multi-image. In: CVGIP, vol. 33, pp. 116–125 (1986)