Available online at www.sciencedirect.com
SCIENCE @ DIRECT®
Pattern Recognition 39 (2006) 695 – 706
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

A study of Gaussian mixture models of color and texture features for image classification and segmentation

Haim Permuter[a,*], Joseph Francos[b], Ian Jermyn[c]

[a]*Department of Electrical Engineering, Stanford University, CA 94305, USA*
[b]*Electrical and Computer Engineering Department, Ben-Gurion University, Beer Sheva 84105, Israel*
[c]*Ariana (Joint INRIA/I3S Research Group), INRIA, B.P. 93, 06902 Sophia Antipolis, France*

Received 11 February 2005; received in revised form 19 October 2005; accepted 19 October 2005

Abstract

The aims of this paper are two-fold: to define Gaussian mixture models (GMMs) of colored texture on several feature spaces and to compare the performance of these models in various classification tasks, both with each other and with other models popular in the literature. We construct GMMs over a variety of different color and texture feature spaces, with a view to the retrieval of textured color images from databases. We compare supervised classification results for different choices of color and texture features using the Vistex database, and explore the best set of features and the best GMM configuration for this task. In addition we introduce several methods for combining the 'color' and 'structure' information in order to improve the classification performances. We then apply the resulting models to the classification of texture databases and to the classification of man-made and natural areas in aerial images. We compare the GMM model with other models in the literature, and show an overall improvement in performance.

Keywords: Image classification; Image segmentation; Texture; Color; Gaussian mixture models; Expectation maximization; *k*-means; Background model; Decision fusion; Aerial images

1. Introduction

In many domains of image processing, there is a strong correspondence between entities in the scene and textures[1] in the image. This implies that the ability to recognize these textures can furnish important semantic information about the scene. Consequently, the problems of texture description and classification, and the closely related problem of segmentation, have received considerable attention, with numerous approaches being proposed (Refs. [1,2] and references therein). In particular, in the field of content-based image retrieval, the ability to answer the question: "Is there a significant amount of such-and-such texture in this image?", can be the basis for many types of query.

Two variations on the problem exist: supervised and unsupervised segmentation. In the former, models of the texture associated with different entities in the scene are assumed known, and are then applied to the image in the hope of segmenting it into regions corresponding to those entities. Clearly this requires a training stage in which human beings group texture exemplars into classes, corresponding to the entities involved, from which the corresponding model parameters are then learnt. In the unsupervised case, no models are known a priori. Instead, the aim is to discover similarities in the data that betray the existence of one or more distinct classes into which the data can be divided. This may or may not involve explicitly learning the model parameters. When the entities in the scene into which the image should be segmented are not decided upon beforehand, as they often are not, unsupervised segmentation is methodologically

* Corresponding author.
E-mail addresses: haim1@stanford.edu (H. Permuter), francos@ee.bgu.ac.il (J. Francos), ian.jermyn@inria.fr (I. Jermyn).

[1] By the word 'texture', we denote both what we will later call 'structure' information, and color information.

0031-3203/$30.00 © 2005 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2005.10.028

ill-defined, since no specification of the ideal result is given. In supervised segmentation on the other hand, texture classes necessarily correspond to distinct entities in the scene, and the success or failure of the segmentation can be decided on this basis. In this paper, we consider the supervised texture segmentation problem.

1.1. Literature

Many kinds of statistical models have been applied to texture classification. These include Bayes classifiers assuming multivariate Gaussian distributions for the features [3–6]; Fisher transformation [7,8]; nonparametric nearest-neighbor classification [9–12]; classification trees [8]; learning vector quantization [13,14]; feed-forward neural networks [15]; and recently support vector machine [16,17] and multiple histogram combined with self-organized map [18] . In some earlier cases, the statistical modelling after the feature extraction is just thresholding [19–22]; or simple extremum picking [23–25]. Markov random fields, and especially Gaussian Markov random fields have been extensively used for texture modelling and segmentation since the early work in Ref. [26]. For a good review, see the paper by Geman and Graffigne [27]. Li and Gray [28] proposed a 2D hidden Markov model (HMM) for image classification, while a somewhat different model is the noncausal HMM described in Ref. [29].

Another recent class of models uses hidden Markov trees (HMTs) to model the joint statistics of wavelet coefficients. Tree models sacrifice some descriptive power (usually only inter- rather than intra-scale dependencies) to ease of implementation (many algorithms that work in the case of linear graphs, also work on trees, but not on more complicated models). HMT models were first introduced in Ref. [30], and were applied to texture analysis in Refs. [31,32]. They are typically used, even in texture applications, with binary-valued hidden state variables that switch between high and low variance Gaussian distributions for the wavelet coefficients. This behavior is intended to capture the difference between edges and noisy but otherwise smooth regions in images, an important distinction for 'edge-preserving denoising'. Indeed, for denoising, HMTs result in state of the art algorithms. It is not clear however, that they remain appropriate for single textures, whose statistics may differ markedly from those for natural images considered as a whole. In particular, the division into 'edges' and 'noise' seems strange in this context. HMTs are used in Ref. [33], where texture and color are combined in an HMT model. Texture is described using HMTs of grayscale wavelet coefficient magnitudes, while color is described using independent Gaussian distributions at each scale for the colored scaling coefficients.

In recent work [34], we proposed the use of Gaussian mixture models (GMMs) for texture classification, demonstrating improved performance over other, computationally more expensive methods. This paper is an extension of the work presented there. In related but differently directed work, Gray et al. [35] also used GMMs for image classification.

2. Classification, GMMs, and feature spaces

In this section, we describe in top-down fashion the models we will use. We begin with our general approach to the classification problem, and continue by describing the place of GMMs within that framework. Finally, we describe the various feature spaces on which the GMMs are defined. We assume throughout that we are dealing with N classes, labelled by $n \in N$.[2]

2.1. Classification

Any classification model is defined on the space $\mathcal{N}$ of maps from the image domain to the set N of classes (each class n corresponds to an entity of interest in the scene), the possible 'classifications'. Thus each classification $v \in \mathcal{N}$ assigns a class $n = v(p) \in N$ to each pixel p giving the class of that pixel. By defining a posterior probability distribution on $\mathcal{N}$, and using a suitable loss function, an optimal classification can be chosen. The loss function is more often than not taken to be the negative of a delta function, the resulting estimate then being a maximum a posteriori (MAP) estimate. The posterior distribution is expressed as the (normalized) product of a likelihood, such as the GMM models that we will discuss in this paper, which gives the distribution of images corresponding to a given class, and a prior probability distribution on the classifications.

Prior models for the classification v usually have a minimum sub-image that can be analyzed. Typically such a model assumes that the regions corresponding to classes are larger than the minimum sub-image. A similar but different restriction is that the neighboring pixels of a given pixel will be with a higher probability from the same class than from an other class. A standard choice for the prior is thus the Potts model, which penalizes a classification by the total length of class boundary it contains. Unfortunately, the use of such a model renders the MAP estimation problem hard to solve, at least rapidly. In order to avoid this problem of computational complexity while producing a similar effect, we use two heuristics: we assume that v is constant on $S \times S$ subimages, called 'blocks', but that its values on different blocks are independent and equiprobable; and we use a loss function/classification rule that incorporates a local 'averaging' of the class over block neighborhoods called 'patches'. The set of blocks in an image will be denoted B, and individual blocks by b. The neighborhood patch $P(b)$ of a block b is the set of blocks in a larger $T \times T$ subimage with b at its center.

[2] Throughout we use an integer N to represent both the number itself and the set $\{1, \ldots, N\}$.