Presented at ICCV'99, Kerkyra, 1999.

Independent Component Analysis of Textures

Roberto Manduchi

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
manduchi@jpl.nasa.gov

Javier Portilla

Instituto de Optica
Consejo Super. de Investig. Cientificas
Serrano 121, Madrid, Spain
iodpm79@pinar2.csic.es

Abstract

A common method for texture representation is to use the marginal probability densities over the outputs of a set of multi-orientation, multi-scale filters as a description of the texture. We propose a technique, based on Independent Components Analysis, for choosing the set of filters that yield the most informative marginals, meaning that the product over the marginals most closely approximates the joint probability density function of the filter outputs. The algorithm is implemented using a steerable filter space. Experiments involving both texture classification and synthesis show that compared to Principal Components Analysis, ICA provides superior performance for modeling of natural and synthetic textures.

1 Introduction

One of the main goals of computer vision is the compact representation of visual entities. Textures, the visual objects considered in this paper, are best represented by their statistical properties. We introduce an information-theoretic framework that provides a better understanding of filter-based approaches to texture analysis and leads to a technique for improving the quality of those texture representations that are based on marginal statistics over filter outputs.

Loosely speaking, textures are characterized by two basic properties: *homogeneity* and *locality of representation*. In other words, the "visual flavour" of a texture can be captured by its statistical behavior within a limited size window. Suitable statistical models for textures are stationary Markov Random Fields (MRF), which are completely characterized by their joint probability density function (pdf) within a suitable neighborhood. Estimating and representing even low-dimensional joint pdf's, however, can be an overwhelming task. The size of the representation grows exponentially with the model dimension, as does the minimum number of samples required to achieve a given estimation accuracy [10]. In fact, techniques that attempt to completely characterize joint distributions either consider very small neighborhoods [27], measure only pairwise dependency [17], or use specific MRF models [12],[8].

A different approach is based on the analysis of feature vectors formed by the output of a filter bank [20],[9],[21],[23]. With a suitable choice of the analysis filters, we can assume that the feature vectors capture the local visually significant texture character. In other words, the filters map the image values in each neighborhood onto a "perceptually relevant" subspace, thereby reducing the representation size while preserving structural information.

Barring a few exceptions [27],[13],[30], typical filter-based algorithms do not estimate the joint statistical description of the texture features. Instead, they build models based on the *marginal statistics* of the feature components, usually represented by the channel variances or their histograms. For example, by analyzing the directional characteristics of a texture along a dense (ideally continuous) set of orientations and scales, one obtains its "scale/orientation signature", which can be used for classification [23],[29].

Texture representation by marginal statistics is a simple and attractive approach. To make efficient use of it, though, we should understand how well a given set of marginals represents the joint statistical description of a texture feature. Zhu et al. [32] pointed out that, in general, one needs the marginal pdf's along every possible projection direction to completely characterize the joint feature vector pdf. Thus, one approach to improving the quality of representation is to use a large number of filters, perhaps exploiting steerable schemes [15],[26] for computational efficiency.

In this work we follow a different route, and devise a technique for finding the basis of a given filter space which generates the most informative marginals for a given texture, meaning that the product over the marginal densities most closely approximates the joint pdf. We show by experimental tests of classification and synthesis that by selecting such "optimal" filter basis the quality of the representation increases significantly.

Our basis selection algorithm is based on Independent Component Analysis (ICA) theory. ICA has been an area of intense research during the last decade, stimulated by problems of blind source separation and deconvolution. ICA can be regarded as a technique for statistical modeling that proves superior to Principal Component Analysis in the case of non-gaussian random vectors. It is an established notion that natural images do not behave like gaussian processes [22],[4]; our experiments show that ICA provides better models for natural and synthetic textures.

Other filter selection techniques have been proposed for texture analysis, usually based on energy [7] or class separability [14]. Perhaps closest to our approach is the entropy–based algorithm of Zhu et al. [32], which uses a greedy strategy to sequentially introduce one filter at a time.

This paper is organized as follows. Section 2 describes the filter spaces that we use for texture analysis, and introduces the filter basis selection problem. Section 3 provides some necessary background of ICA, and shows its application to texture modeling. Section 4 describes texture classification and synthesis experiments from which we were able to assess the performance of ICA modeling. Section 5 has the conclusions.

2 Steerable spaces for texture analysis

Given an image $l(x)$ and a set of N filters $\{h_i(x)\}$, we will say that the vector $f(x) = [f_1(x) = l * h_1(x), \ldots, f_N(x) = l * h_N(x)]$ is the *feature representation* of the image at point x. We will assume that a texture is completely represented by the joint pdf of its feature vector $f(x)$ (which, by stationariety, is independent of x.) By this we mean that two textures with the same joint distribution of their feature vectors are "perceptually" indistinguishable.

Which filters produce meaningful feature representations of textures? Several design criteria have been proposed. Basically, they all share the following characteristics: 1) zero-DC response, to enforce invariance to slow-varying illumination bias; 2) good spatial/spectral concentration, to ensure energy separation while preserving locality of description. Another useful property of analysis filter banks is *steerability* [15],[16] defined as follows. Let H be the linear space spanned by the N kernels $\{h_i(x)\}$, and let $\mathcal{G}$ be a suitable group of domain transformations (e.g., rotations or isotropic scaling.) We will say that H is steerable over $\mathcal{G}$ if for every kernel $h(x)$ in H, and for every transformation $A(\cdot)$ in $\mathcal{G}$, the "transformed" kernel $h_A(x) = h(A(x))$ belongs to H. For example, assume the filter space is steerable over the group of rotations. Let $f(x)$ and $f^R(x)$ be the feature vector description of a texture and of the rotated version of the texture, respectively (R is the rotation matrix). Then, $f(x)$ and $f^R(R^{-1}x)$ are related by a linear transformation (a matrix), which is a function of the rotation angle only. Steerability, thus, ensures uniformity of representation for transformed versions of the same texture. Methods exist to design exact and approximate finite dimensional steerable filter spaces over rotations and isotropic scaling over a finite scale range of interest [26].

Joint spatial/spectral concentration is achieved, for example, using Gabor kernels or directional derivatives of gaussian kernels [21],[23]. Such filters effectively capture directional texture attributes. If a N-dimensional steerable analysis space is used, the texture signatures are completely characterized by the joint statistics of the output of N basis filters. Note that one can always find a set of basis filters which are scaled/rotated versions of a prototype kernel in the steerable space.

In this paper we study reduced representations formed by the marginal statistics of the feature vectors. Let us recall that the marginal density of the i-th component of a random vector z, $p_i(z_i)$, is the projection of the joint pdf $p(z)$ onto the i-th axis. The set of marginal statistics is represented by the outer product of the marginal pdf's: $\bar{p}(z) = \prod_i p_i(z_i)$, which is equal to $p(z)$ if and only if the components of the feature vector are statistically independent. In particular, we are concerned with the selection of the "optimal" basis in a steerable filter space. Note that the choice of the basis is irrelevant if the joint pdf of the feature vector is considered. Indeed, if f_1 and f_2 are the feature vectors at point x, computed using two different filter bases, then $f_2 = Af_1$, where A is a full-rank matrix. Thus, the joint pdf's of the feature vectors, $p_1(z)$ and $p_2(z)$, are related to one another as $p_2(z) = p_1(A^{-1}z)/|\det(A)|$ [25]. This one-to-one relation, however, holds for joint pdf's only. In general, the marginal pdf's of f_1 *cannot* be computed from the marginal pdf's of f_2. This somewhat paradoxical observation stems from the fact that marginal pdf's are projections of a joint pdf along different directions. A finite set of marginals does not carry enough information in general to allow reconstructing all the other projections.

Assume, however, that a particular choice of basis filters gives statistically independent feature components. Then, as noted above, the marginals completely characterize the joint pdf of the feature vector: from this set of marginals, all the other marginals can be reconstructed. The converse is not true: other filter bases will produce, in general, features with "less informative" marginal statistics.

Unfortunately, only very few textures can be represented by independent component features. Still, it is almost always the case that some filter bases produce "more informative" marginals than others. This intuitive notion can be made more rigorous using the theory of Independent Components Analysis (ICA); ICA also leads to an algorithm for selecting an "optimal" filter basis.

3 Independent Component Analysis

In this section we report some results of Independent Component Analysis theory that are instrumental to the development of the texture classification and synthesis algorithms discussed in later sections. We refer the reader to the excellent tutorial of Cardoso [5] for a general overview of the theory.

We first introduce the ICA problem as a statistical approximation tool, and then briefly outline Comon's ICA algorithm, which we have adopted in our experiments.

3.1 Problem formulation

The theory of Independent Component Analysis is traditionally associated with the Blind Source Separation (BSS) problem. In its simplest formulation, the BSS problem assumes that N independent causes (random variables) have been linearly combined by a full rank matrix A to produce N observed variables. The goal is to identify the mixing matrix from the observations, possibly using prior information about the statistics of the causes if available.

ICA, however, may also be regarded as a general estimation method [11], without explicit reference to the BSS model. If z is a random vector with joint pdf $p(z)$ and full-rank covariance matrix[1], ICA seeks a full rank matrix A such that the pdf of the vector $y = Az$ is "best represented" by the outer product of its marginal pdf's. The quality of the separable approximation is measured by the *contrast*, which is a deterministic function of a joint pdf. Let $e(z)$ be the "model error", defined as the difference between the logarithm of the density $p(z)$ and the log-likelihood of the separable model:

$$e(z) = \log p(z) - \log \prod_i p_i(z_i) \qquad (1)$$

where $p_i(z_i)$ are the marginal pdf's of the components of z. The contrast [11] of $p(z)$ is defined as the expectation of the model error $e(z)$, which is equal to the Kullback-Leibler (KL) divergence between $p(z)$ and the separable model pdf:

$$\begin{aligned} E\left[e(z)\right] &= \int_{-\infty}^{\infty} p(z) \log \frac{p(z)}{\prod_i p_i(z_i)}\,\mathrm{d}z \qquad (2) \\ &= \sum_i H_i(z_i) - H(z) \end{aligned}$$

where $H(\cdot)$ represents the differential entropy of its argument: $H(z) = -\int_{-\infty}^{\infty} p(z)\log p(z)$ (terms $H_i(z_i)$ are called *marginal entropies* of z.) This contrast takes also the name of *mutual information* of z, and being a KL divergence it is always positive (it vanishes if and only if z has independent components). Intuitively, the mutual information tells us how much we lose in terms of average information if we neglect to consider the statistical dependence among the components of z.

Other contrast functions have been proposed for ICA. For example, if the distributions $g_i(z_i) = \int_{-\infty}^{z_i} p_i(\hat{z})\mathrm{d}\hat{z}$ of the original independent components in a BSS model are known, we may define the "infomax" contrast [1] of the vector z as $-H(g(z))$, where $g(z) = [g_1(z_1), \ldots, g_N(z_N)]$. However, in the case of texture descriptors we don't know the distributions $g_i(z_i)$, and mutual information is a more appropriate criterion. Thus, in this paper we will use the following definition of the ICA problem:

> **ICA problem:** find a full-rank matrix A such that $y = Az$ has minimal mutual information.

Note that the solution to the ICA problem is not unique: if A minimizes the mutual information of Az, so do all matrices obtained by permuting the rows of A or by pre-multiplying A by a diagonal matrix.

If z is jointly gaussian, its mutual information is null if and only if its covariance matrix is diagonal. Principal Component Analysis (PCA, also known as Karhunen-Loève transform) is a technique to diagonalize a vector covariance by an orthonormal matrix, and therefore solves the ICA problem for gaussian vectors. In the general non-gaussian case, however, diagonal covariance is not sufficient to ensure the minimal mutual information condition; hence, more work is needed.

[1] If the covariance matrix V is not full rank, we may consider the projection of z onto the range space of V [11].

It can be shown [11] that, if z has been "pre-whitened" to unit covariance (by PCA followed by axis rescaling), the ICA problem is solved by an orthonormal linear transformation that minimizes the sum of the marginal entropies of the transformed vector. This fact has a nice counterpart in the context of BSS. Since the gaussian distribution has the highest entropy for a given variance, we may argue that the demixing matrix is the one that produces marginals as "far from gaussian" as possible. This observation agrees with intuition: indeed, as suggested by the central limit theorem, linear mixing of independent causes produces gaussian-like distributions.

3.2 A high–order contrast algorithm

ICA algorithms try to minimize the contrast of a vector without explicitly computing its joint pdf. Comon [11] proposed an efficient technique based on high-order statistics[2]. In fact, Comon's algorithm minimizes a different contrast than mutual information; the relation between the two contrast functions can be highlighted using Edgeworth functional expansions [19]. An intuitive justification of high-order methods is based on the fact that all the cross-cumulants of an independent component vector are null [19]. Thus, one may try to minimize mutual information by actually minimizing the n-th order dependence among the vector components. For example, Comon's technique finds an orthonormal matrix A that minimizes the sum of the squared fourth-order cross-cumulants of $y = Az$, provided that z has been pre-whitened. It can be shown that this is equivalent to maximizing the sum of the squared marginal kurtosis (fourth-order cumulants) of y. It is a well known fact that gaussian distributions have zero kurtosis [19]: cumulant–based algorithms do push the vector components away from gaussianity. Leptokurtic, heavy tailed exponential-like distributions are characteristic of "sparse codes" [24], and are often used to model the output of wavelet filter banks for image coding [22]. Platykurtic distributions are also often observed in texture analysis.

Comon's design algorithm is based on Jacobi iterations. Its complexity is $\mathcal{O}(N^4)$ (including pre-whitening,) and it always converges to a (possibly local) minimum of the contrast. Due to these desirable properties, we have selected such a technique for our experiments. It should be noted that several other fast algorithms for ICA exist [5]; we plan to experiment with some of those in future work.

[2]High–order statistics methods have often been neglected by neural network researchers involved in ICA. A recent paper by Cardoso [6] compares cumulant–based algorithms with gradient–descent techniques, and shows the practical effectiveness of the former from an algorithmic point of view, as well as on the ground of experimental results.

4 Experimental tests

4.1 Supervised classification

In this section we use ICA pdf modeling in a test of supervised texture discrimination. For the sake of simplicity, we perform training and classification on the same data set, formed by the $K = 8$ textures of Figure 1 (chosen from the MIT VisTex database). Each texture is modeled by the marginal pdf's over the selected channels in a fixed steerable space, together with the corresponding ICA matrix. The marginal densities are estimated and represented by 50-bins histograms.

An image formed by the mosaic of all textures is classified pixel-wise using a Bayesian technique [3]. Spatial coherence is enforced by using a "soft" version of Besag's Iterated Conditional Modes (ICM) algorithm [2], which uses MRF modeling to estimate the posterior class probabilities for each pixel from the conditional likelihoods (as produced by ICA modeling.) The experiment proceeds as follows:

- **Training:** for each texture model k,
  1. filter the training image with a fixed filter bank;
  2. compute the ICA matrix A_k from the output of the filters (we will assume, without loss of generality, that $|\det(A_k)| = 1$;)
  3. multiply the output vectors by the ICA matrix A_k and compute the channel histograms.
- **Classification:** apply the fixed filter bank to the test image. For each texture model k,
  1. multiply the filter output vectors by the model ICA matrix A_k, and use the corresponding channel histograms to compute the marginal conditional likelihoods $p_i(z_i|k)$;
  2. the conditional likelihood $p(z|k)$ of any pixel value z is equal to the product of the corresponding marginal conditional likelihoods:

$$p(z|k) = \prod_i p_i(z_i|k) \tag{3}$$

  From the conditional likelihoods, compute the posterior class probabilities using the ICM algorithm and perform pixel-wise Bayesian classification.

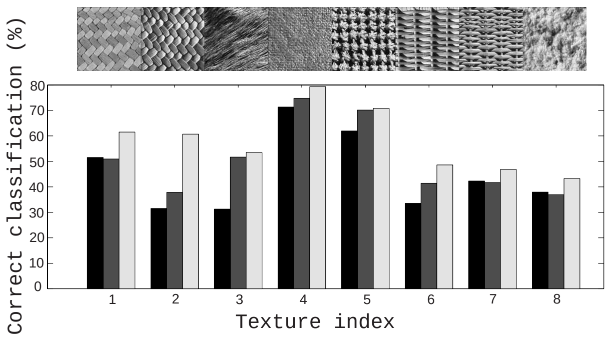

Figure 1: Supervised classification test (see text). The system is trained and tested on the eight textures shown above. The plots represent the percentage of correct classification for each texture. White bars: using ICA transformation. Gray bars: using PCA transformation. Black bars: without transformation.

For our experiment we have chosen a steerable pyramid filter bank [31] with just two orientations and three scales ($N = 6$ filters overall). For each training texture (size 256×256 pixels), Comon's Matlab implementation of ICA required 27 seconds of computation time on a Power Macintosh G3 266 MHz. The results, in terms of percentage of correctly classified pixels for each texture, are shown by the white bar plot of figure 1. If PCA is used instead of ICA, we obtain the correct classification rates shown by grey bars; the classification performances of the system without transformation (i.e., using the original filters) are shown by the black bars. ICA pdf modeling consistently yields the highest correct classification rates.

We have observed that the performance gain provided by PCA and ICA modeling relative to the nominal feature vector representation decreases if the number of orientations in the steerable filter bank is increased. This should not come as a surprise: many marginal statistics can provide sufficient information even if they are not statistically independent; however, the benefit comes at the price of increased representation size.

4.2 Texture synthesis

Synthesis-by-analysis algorithms [18],[28],[32],[30] create new textures which "look like" a given prototype. This is equivalent to sampling a random process whose statistical description has been estimated from a given sample.

A simple and effective synthesis technique has been proposed by Heeger and Bergen [18]. First, the prototype texture is analyzed by a filter bank which includes the "identity" filter, and the marginal channel histograms are recorded. Then, a random image is generated, and its channel histograms (computed by the same filter bank) are iteratively adjusted to match those of the prototype texture. The algorithm usually converges to a texture that has the same channel histograms as the prototype: the two textures are therefore identical on the grounds of marginal statistical description.

We have adopted such an algorithm as a testbed for our experiments because it represents the ideal "Turing test" to validate texture models based on marginal channel statistics. Our only addition to Heeger and Bergen's model is the multiplication of the channel vectors at the output of the filter bank by the ICA matrix computed for the prototype texture, as in the scheme of section 4.1. After the channel histograms have been adjusted to match those of the prototype texture, the channel vectors are multiplied by the inverse of the ICA matrix. Note that the presence of the ICA matrix requires that all the channels be kept to the same sampling rate.

Heeger and Bergen use steerable pyramid analysis filter banks [31], and synthesize samples that successfully reproduce some unstructured characteristics of the prototype texture. Such simple marginal statistics modeling, though, fails to capture more complex spatial structures, such as elongated patterns. A simple way to boost the performance of Heeger and Bergen's scheme is to enrich the filter space with the *shifted* versions of the filters. In other words, for each filter $h(x)$ we add the filters $h^n(x) = h(x+n)$ with n belonging to a suitable neighborhood of the origin. The feature representation thus "looks around" over a larger neighborhood of each point, and allows us to synthesize patterns with highly structured spatial dependency. This new information comes at no cost: the output of filter $h^n(x)$ at point x_0 is equal to the output of filter $h(n)$ at point $x_0 + n$. In particular, the shifting grid in our implementation is a function of the filter scale. For example, if for the smallest scale filters $h(x)$ (with scale $\sigma = \sigma_0$) we use the shifted kernels $h^n(x)$ with $n \in \{(0,0),(-1,0),(0,-1),(1,0),(0,1)\}$, for wider filters with scale $\sigma = \sigma_0 * k$ we will use $n \in \{(0,0),(-k,0),(0,-k),(k,0),(0,k)\}$.

Shifted filters, though, are useless if the marginals statistical description is built directly from their output, like in Heeger and Bergen's original algorithm: by stationariety, the marginal pdf of a filter output does not change if the filter is shifted. The contribution of the shifted filters becomes relevant if a different basis of the steerable space is chosen, that is, if the output

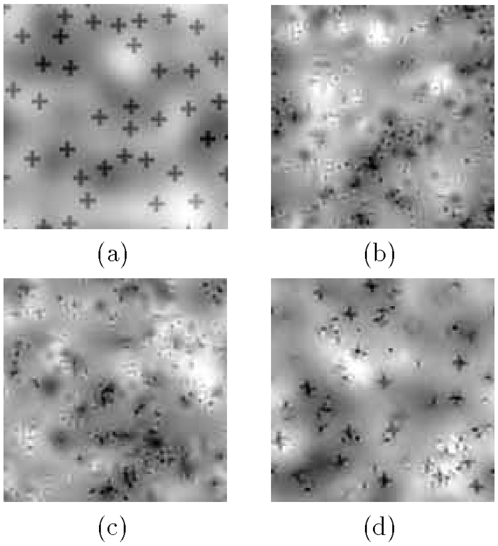

Figure 2: Examples of texture synthesis. (a) The prototype texture "Crosses on clouds" is synthesized using Heeger and Bergen's technique (b) with no channel transformation , (c) with PCA transformation and (d) with ICA transformation.

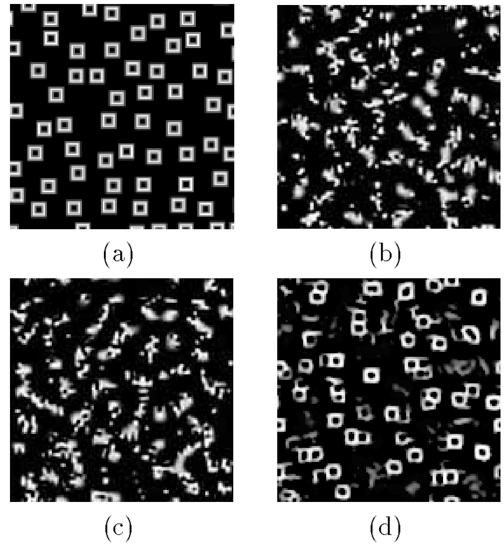

Figure 3: Examples of texture synthesis for the prototype texture "Squares" (see caption of figure (2)).

of the filters is multiplied by the PCA or ICA matrix.

We show in figures 2 and 3 examples of synthesis from the prototype textures "Crosses on clouds" and "Squares". For the "Crosses on clouds" texture we have used a steerable pyramid filter bank which comprised both odd-symmetric and even-symmetric kernels at two orientations and four scales (overall $N = 16$ filters). For the "Squares" texture we have used odd-symmetric kernels at two orientations and three scales, shifted over five different positions (overall $N = 30$ filters). These prototype textures are highly non-gaussian, which makes PCA modeling inadequate (see figure 2(c) and 3(c)). ICA modeling proves superior in both cases. In particular, for the "Crosses on clouds" texture, ICA does a good job at separating the low-pass, isotropic components (the "clouds") from the line-shaped, oriented ones (the "crosses".) In the case of the "Squares" texture, ICA is able to extract the local structure by correctly combining the shifted filters outputs.

5 Conclusions

We have presented an algorithm that chooses the basis filters in a steerable space in such a way as to yield the most informative marginals for texture representation. The method is based on the minimization of the mutual channel information via Independent Component Analysis. The experimental results show the superiority of ICA modeling with respect to PCA for natural and synthetic textures and small dimensional filter spaces.

An interesting open problem, which we are currently investigating, is subspace selection based on information–theoretic criteria. We expect that by choosing a small number of highly informative channels, one should be able to obtain reduced size representations that maintain good discrimination properties.

Acknowledgments

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. RM was supported in part by the NASA Telerobotics program and by the NASA Remote Exploration and Experimentation task. JP was supported by the Consejo Superior de Investigaciones Cientificas (CSIC, Spain), and the Comision Interministerial de Ciencia y Tecnologia (CICYT, Spain), under grant TIC97-325. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

The authors would like to thank Dr. Michael Burl for his proofreading of the paper, which unquestionably improved the clarity of the presentation.

References

[1] A.J. Bell and T.J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129-1159, 1995.

[2] J. Besag. On the statistical analysis of dirty pictures. *J. R. Statist. Soc. B*, 48(3):259–302, 1986.

[3] C.M. Bishop. *Neural networks for pattern recognition.* Clarendon Press, Oxford, U.K., 1995.

[4] R.W. Buccigrossi and E. Simoncelli. Progressive wavelet image coding based on a conditional probability model. *Proc. IEEE ICASSP*, 4:2957–2960, Munich, 1997.

[5] J.-F. Cardoso. Blind signal separation: Statistical principles. *Proceedings of the IEEE*, 86(10):2009–2025, October 1998.

[6] J.-F. Cardoso. High–order contrasts for Independent Component Analysis. *Neural Computation*, 11:157–192, 1999.

[7] T. Chang and C.-C. J. Kuo. Texture analysis and classification with tree–structured wavelet transform. *IEEE Trans. Image Proc.* , 2(4):429–441, October 1993.

[8] R. Chellappa and S. Chatterjee. Classification of textures using Gaussian Markov Random Fields models. *IEEE Trans. Acoust., Speech, Signal Proc.*, 33(4):959–963, August 1985.

[9] J.M. Coggins and A.K. Jain. A spatial filtering approach to texture analysis. *Pattern Recognition Letters*, 3:195–203, May 1985.

[10] P. Comon. Supervised classification: A probabilistic approach. *Proc. ESANN*, 111–128, Verleysen, April 1995.

[11] P. Comon. Independent Component Analysis: A new concept? *Signal Processing*, 36(3):287–314, April 1994.

[12] G.C. Cross and A.K. Jain. Markov Random Field texture models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 5:25–39,1983.

[13] J.S. De Bonet and P. Viola. Texture recognition using a non-parametric multi-scale statistical model. *Proc. IEEE CVPR*, 641–647, Santa Barbara, June 1998.

[14] K. Etemad and R. Chellappa. Separability–based multiscale basis selection and feature extraction for signal and image classification. *IEEE Trans. Image Proc.*, 7(10):1453–1465, October 1998.

[15] W. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13:891-906, 1991.

[16] H. Greenspan, S. Belongie, R. Goodman, P. Perona. Rotation invariant texture recognition using a steerable pyramid. *Proc. 12th IAPR*, 2:162-167, Jerusalem, Israel, 1994.

[17] R.M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804. May 1979.

[18] D. Heeger and J. Bergen. Pyramid-based texture analysis/synthesis. *Proc. ACM SIGGRAPH*, August 1995.

[19] M. Kendall and A. Stuart. *The advanced theory of statistics.* MacMillan Publishing, New York, 1977.

[20] K.I. Laws. Texture energy measures. *Proc. Image Underst. Workshop*, 47–51, November 1979.

[21] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *Journ. Optical Society of America*, 7(5):923–932, May 1990.

[22] S.G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11:674–693, July 1989.

[23] B.S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(8):837–842, August 1996.

[24] B.A. Olshausen and D.J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, June 1996.

[25] A. Papoulis. *Probability, random variables and stochastic processes.* McGraw-Hill, New York, 1965.

[26] P. Perona. Deformable kernels for early vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(5):488-499, May 1995.

[27] K. Popat and R. Picard. Cluster-based probability model and its application to image and texture processing. *IEEE Trans. Image Proc.*, 6(2):268–284, February 1997.

[28] J. Portilla, R. Navarro, O. Nestares and A. Tabernero. Texture synthesis-by-analysis based on a multiscale early-vision model. *Optical Engineering*, 35(8):2403-2417, 1996.

[29] Y. Rubner and C. Tomasi. A metric for distributions with applications to image databases. *Proc. 6th ICCV*, 59–66, Bombay, 1998.

[30] E.P. Simoncelli and J. Portilla. Texture characterization via second-order statistics of wavelet coefficient amplitudes. *Proc. IEEE ICIP'98*, 1998.

[31] E. Simoncelli and W.T. Freeeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. *Proc, of 2nd IEEE ICIP*, 444-447, Washington DC, October 1995.

[32] S.C. Zhu, Y. Wu and D. Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8), November 1997.