

# Texture Segmentation Using Independent Component Analysis of Gabor Features

Yang Chen and Runsheng Wang

ATR Lab, National University of Defense Technology, Changsha, 410073, Hunan, China  
seemeinwinter@sohu.com, rswang@nudt.edu.cn

## Abstract

*This paper proposes a novel method for texture segmentation using independent component analysis (ICA) of Gabor features (called ICAG). It has three distinguished aspects. (1) Gabor wavelets transformation first produces distinct textural features characterized by spatial locality, scale and orientation selectivity. (2) Principal component analysis (PCA) then reduces the dimensionality of these features and ICA finally derives independent features for texture segmentation. (3) Two different frameworks for ICA are discussed. Framework I regards pixels as random variables and represents them as a column vector by re-shaping all the transformed images row-by-row, while framework II treats the statistical features, viz. the mean and standard deviation of image, as random variables. The statistical features of all the transformed images construct a column vector. Comparative experiment results among ICAG, Gabor wavelets and ICA indicate that ICAG provides the best performance and framework II is more efficient and applicable for texture segmentation.*

## 1. Introduction

Texture segmentation is a very fundamental area of study in computer vision and image processing. It is a key problem in many applications such as object recognition, quality inspection, remote sensing and so on. The major difficulty of texture segmentation is feature extraction. Many techniques on this topic have been developed and broadly divided into four categories: statistical methods, structural methods, model based methods and signal processing methods (essentially filtering methods) [1].

Recently, the multi-scale filtering approach using Gabor wavelets has shown promising potential for texture analysis, which is inspired by psychophysical studies first proposed by Campbell and Robson [2]. The transformed images through Gabor wavelets can exhibit distinct textural features characterized by

spatial locality, scale and orientation selectivity [3]. However owing to the non-orthogonality of Gabor wavelets, there is redundant information in the transformed images. A popular solution is employing the well-known linear projection technique, namely principal component analysis (PCA) to remove it. But PCA can only decorrelate the second order dependencies and provide partial information on the statistics of textures. So that it might become necessary to incorporate higher order statistics as well. Independent component analysis (ICA) is a relatively recent technique that exploits higher order statistical structure in data. It can linearly transform the observed variables into such components that are not only uncorrelated but also as statistically independent from each other as possible. ICA has been proposed as a generic statistical model [4] for images and directly applied to texture analysis[5, 6].

This paper proposes a supervised method for texture segmentation using independent component analysis (ICA) of Gabor features (called ICAG). Firstly, training samples are randomly selected from the texture images, and then exposed to the Gabor wavelets transformation. Secondly, a feature vector corresponding to each sample can be constructed from the transformed images in two different frameworks. Framework I regards pixels as random variables and represents them as a column vector by re-shaping all the transformed images row-by-row, while framework II treats the statistical features, viz. the mean and standard deviation of image, as random variables. The statistical features of all the transformed images construct a column vector. Thirdly, the dimensionality and redundancy of the feature vectors for all the training samples are reduced by PCA and ICA. The new resulting vectors with statistically independent components are taken as texture features. Finally, a nearest neighbor classifier with Mahalanobis distance is chosen to segment the test images based on these features.

This paper is organized as follows: section 2 briefly reviews the formulas and design of Gabor wavelets. Section 3 elaborates the independent analysis of Gabor

features with two different frameworks. Before concluding in section 5, section 4 performs a series of comparative experiments.

## 2. Gabor Wavelets and Design

A 2D Gabor Function essentially is a 2D Gaussian modulated with a complex exponential defined as follow:

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\left(\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}\right) + j2\pi(\mu_0x + \nu_0y)\right) \quad (1)$$

and its Fourier transform:

$$G(\mu, \nu) = \exp\left(-2\pi^2\left(\sigma_x^2(\mu - \mu_0)^2 + \sigma_y^2(\nu - \nu_0)^2\right)\right) \quad (2)$$

where the radial frequency  $f_0 = \sqrt{\mu_0^2 + \nu_0^2}$ , the orientation  $\Phi = \tan^{-1}(\nu_0/\mu_0)$  from the coordinate axis  $\mu$ . Let  $g(x, y)$  be the mother wavelet, then the self-similar filter bank can be obtained by appropriate dilations and rotations of  $g(x, y)$  through the generating function:

$$g_{mn}(x, y) = a^{-m}g(x'_m, y'_m) \quad (3)$$

$$a > 1, m = 0, \dots, S-1, n = 0, \dots, K-1$$

and

$$x'_m = a^{-m}(x\cos\theta_n + y\sin\theta_n), y'_m = a^{-m}(-x\sin\theta_n + y\cos\theta_n) \quad (4)$$

where  $\theta_n = n\pi/K$ ,  $a$  is scale,  $S$  and  $K$  is the total number of scales and orientations respectively.

Assuming a training sample is  $N \times N$  in size, the following values of the radial frequency  $f_0$  are used:

$$\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}, \dots, (N/4)\sqrt{2} \text{ cycles / image-width}$$

The above choice of the radial frequency guarantees that the pass-band of the filter with the highest radial frequency, viz.  $(N/4)\sqrt{2}$  cycles / image-width, falls inside the  $N \times N$  window [7]. Accordingly, the different numbers of orientation ( $K = 4, 6, 8$ ) are used along with the values of  $N$ . This design yields the following equations for the necessary parameters:

$$a = \left(\frac{U_h}{U_l}\right)^{\frac{1}{S-1}} \quad (5)$$

$$\sigma_x = \left(\frac{1}{2\pi}\right) \left(\frac{a+1}{a-1}\right) \left(\frac{\sqrt{2\ln 2}}{U_h}\right) \quad (6)$$

$$\sigma_y = \left(\frac{1}{2\pi}\right) \left[\tan\left(\frac{\pi}{2K}\right) \sqrt{\frac{U_h^2}{2\ln 2} - \frac{1}{(2\pi\sigma_x)^2}}\right] \quad (7)$$

where  $U_l$  and  $U_h$  denote the lower and upper center frequencies of interest. Here they are  $\sqrt{2}$  and  $(N/4)\sqrt{2}$  cycles / image-width respectively.

## 3. Independent Component Analysis of Gabor Features

Assume a set of training samples from the given texture:  $t(x, y) = \{t_1(x, y), t_2(x, y), \dots, t_l(x, y)\}$ , where each element  $t_i(x, y)$  represents a sample in size of  $N \times N$  and the total number of samples is  $l$ . Apply the family of Gabor wavelets with  $S$  scales and  $K$  orientations designed in section 2 to the above training samples,  $SK$  transformed images for each sample are then obtained as follows:

$$O_{mn}^{(i)}(x, y) = \text{abs}(t_i(x, y) * g_{mn}(x, y)) \quad (8)$$

$$m = 0, 1, \dots, S-1; n = 0, 1, \dots, K-1; i = 1, 2, \dots, l.$$

From these transformed images, we construct Gabor feature vectors in two frameworks for ICA. Framework I (called ICAG I) regards pixels as random variables, while framework II (called ICAG II) treats the statistical features of image as random variables.

### 3.1 Framework I: Regard Pixels as Random Variables.

Framework I regards pixels as random variables and represents them as a column vector  $\mathbf{o}_i$  by re-shaping all the  $SK$  transformed images of the training sample  $t_i(x, y)$  row-by-row, that is, the vector  $\mathbf{o}_i$  is written as:

$$\mathbf{o}_i = \left\{ o_{00}^{(i)T}, o_{01}^{(i)T}, \dots, o_{S-1K-1}^{(i)T} \right\}^T \quad (9)$$

where  $T$  is the transpose operator, and  $o_{mn}^{(i)}$  is also a column vector constructed by concatenating the rows of the output  $O_{mn}^{(i)}(x, y)$ . Thus the dimensionality of the vector  $\mathbf{o}_i$  is very high:  $N^2SK$ . So before the independent component analysis, we first down-sample the vector  $\mathbf{o}_i$  by a factor  $d$  to reduce the dimensionality. Different values of  $d$  are used according to the values of  $N$ . The  $l$  training samples form the matrix  $\mathbf{O}^{(d)} = \left[ \mathbf{o}_1^{(d)}, \mathbf{o}_2^{(d)}, \dots, \mathbf{o}_l^{(d)} \right]$  down-sampled by the factor  $d$ , which would be taken as the observed vectors for ICA.

### 3.2 Framework II: Regard Statistical Features as Random Variables.

In framework II, due to the spatially homogeneous property of local texture regions, the mean  $\mu_{mn}^{(i)}$  and standard deviation  $\sigma_{mn}^{(i)}$  of the output  $O_{mn}^{(i)}(x, y)$  can be treated as the texture features of sample  $t_i(x, y)$  at the particular scale  $m$  and orientation  $n$ . The statistical features of all the transformed images at  $S$  scales and  $K$  orientations construct the column vector  $\mathbf{f}_i$  as follows:

$$\mathbf{f}_i = \left[ \mu_{00}^{(i)}, \sigma_{00}^{(i)}, \mu_{01}^{(i)}, \sigma_{01}^{(i)}, \dots, \mu_{S-1K-1}^{(i)}, \sigma_{S-1K-1}^{(i)} \right]^T \quad (10)$$

$$\mu_{mn}^{(i)} = \frac{1}{N^2} \sum_{(a,b) \in N \times N} |O_{mn}^{(i)}(a, b)| \quad (11)$$

$$\sigma_{mn}^{(i)} = \left( \frac{1}{(N^2-1)} \sum_{(a,b) \in N \times N} \left( |O_{mn}^{(i)}(a, b)| - \mu_{mn}^{(i)} \right)^2 \right)^{\frac{1}{2}} \quad (12)$$

So the observed vectors for ICA can be written as  $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_I]$  and the vector  $\mathbf{f}_i$  has a fairly lower dimensionality:  $2SK$ , compared with that of framework I.

### 3.3 ICA

ICA theory is traditionally associated with the Blind Source Separation (BBS) problem. In brief, the BBS problem assumes that  $N$  independent causes (random variables) have been linearly combined by a full bank matrix to produce  $M$  observed variables. The goal of ICA is to identify this mixing matrix from the observations, possibly using prior information about the statistics of the causes if available. When ICA is applied in texture analysis, a given texture image is considered as the mixture of an unknown set of statistically independent variables by an unknown mixing matrix. A separating matrix is learnt by ICA to recover this set of independent variables, which can be regarded as the features to represent that kind of texture image.

In this paper, before the independent component analysis of the above vectors ( $\mathbf{O}^{(d)}$  or  $\mathbf{F}$ ), principal component analysis is performed to reduce the dimensionality. It is very helpful for the computation of ICA:

$$\mathbf{X} = \mathbf{P}\mathbf{O} \text{ or } \mathbf{X} = \mathbf{P}\mathbf{F} \quad (13)$$

where  $\mathbf{P} \in \mathbf{R}^{p \times q}$ ,  $p < q$  and  $q = N^2SK$  in ICAG I or  $q = 2SK$  in ICAG II.

Let the output of PCA  $\mathbf{X}$  be the observed vectors and  $\mathbf{S}$  be the unknown independent sources. If  $\mathbf{A}$  is the unknown mixing matrix, then the general model of ICA can be written as follows:

$$\mathbf{X} = \mathbf{A}\mathbf{S} \quad (14)$$

ICA is to find a separating matrix  $\mathbf{W}$ , thus

$$\mathbf{Y} = \mathbf{W}\mathbf{X} \quad (15)$$

However, there is no restricted form expression to find  $\mathbf{W}$  and many iterative algorithms are developed to approximate  $\mathbf{W}$  for optimizing the independence of  $\mathbf{Y}$ . The vector  $\mathbf{Y}$  is actually an estimate of the true  $\mathbf{S}$  and possibly permuted and rescaled. This paper utilizes the fixed-point algorithm proposed by Hyvärinen [8], which is described in formulation as:

$$\mathbf{W}^+ = \mathbf{W} - \frac{\mu[\mathbf{C}^{-1} \mathbf{E}\{\mathbf{X}g(\mathbf{W}^T \mathbf{X})\} - \beta \mathbf{W}]}{[\mathbf{E}\{g'(\mathbf{W}^T \mathbf{X})\} - \beta]} \quad (16)$$

$$\mathbf{W}^* = \frac{\mathbf{W}^+}{\sqrt{(\mathbf{W}^+)^T \mathbf{C} \mathbf{W}^+}} \quad (17)$$

where  $\beta = \mathbf{E}\{\mathbf{W}^T \mathbf{X}g(\mathbf{W}^T \mathbf{X})\}$ ,  $g(u) = u^3$  and  $\mathbf{E} = \{\mathbf{X}\mathbf{X}^T\}$ . For further details on this algorithm we refer to [8].

## 4. Comparative Experiments

To evaluate the performance and applicability of the ICAG method, we not only use the composite image T1 (the left in figure 1) comprising five narrowband textures (D106, D21, D16, D77, D93), but also challenge the composite image T2 (the right in figure 1), which consists of five natural textures (D28, D12, D92, D112, D15) with broader bands in frequency. Furthermore, a series of comparison experiments on the two test images are performed, particularly ICAG I versus ICA, ICAG II versus Gabor wavelets, and ICAG I versus II. Figure 2 contrastively illustrates the procedures of the four methods.

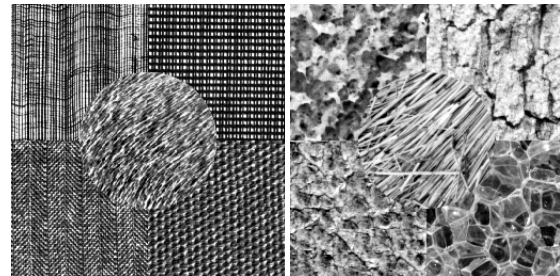
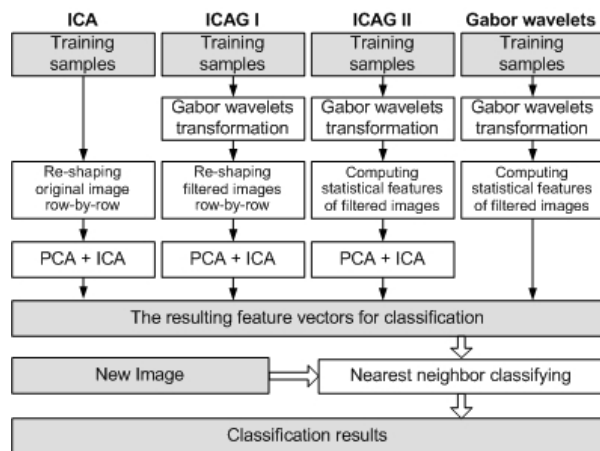
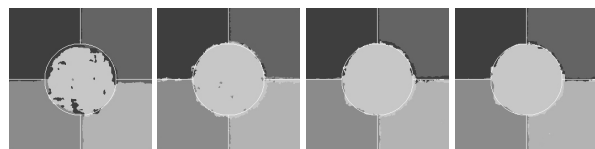


Figure 1. Brodatz textures used for segmentation.



**Figure 2. The procedures of ICA, ICAG I, ICAG II and Gabor wavelets.**

As illustrated in figure 3, the Gabor wavelets, ICA, ICAG I and II applied into T1 provide nearly perfect results with the window size  $N = 17$ . Table 1 also gives the misclassification rates of the four methods on T1 using various values of  $N$ :  $9 \times 9$ ,  $13 \times 13$ ,  $17 \times 17$  and  $21 \times 21$ , which shows that both ICAG I and II achieve less error rate compared with the Gabor wavelets and ICA. More significantly, the segmentation results of T2 (see figure 4 and table 2) strongly demonstrate the excellent performance of ICAG I and II. Oppositely, the ICA and Gabor wavelets are not suitable for segmenting these natural textures. In addition, although ICAG I provides a little better results relative to ICAG II, its dimensionality is so high that causes the “curse of dimensionality” in statistical estimations and a lot of inconvenience in computation. Therefore, ICAG II is a more efficient and applicable approach for texture segmentation.

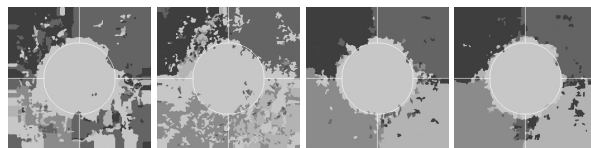


**Figure 3. The segmentation results of T1 by Gabor wavelets, ICA ICAG I and II from left to right.**

**Table 1. Mosaic1 misclassification rate (%) of Gabor, ICA and ICAG**

$N$	$9 \times 9$	$13 \times 13$	$17 \times 17$	$21 \times 21$
ICA	4.96	4.32	3.67	5.18
ICAG I*	3.12(4)	3.08(6)	2.16(8)	2.99(10)
ICAG II	4.22	3.13	2.57	3.27
Gabor	19.72	14.80	5.32	5.94

\* The values in bracket are the down-sampling factor  $d$ .



**Figure 4. The segmentation results of T2 by Gabor wavelets, ICA ICAG I and II from left to right.**

**Table 2. Mosaic2 misclassification rate (%) of Gabor, ICA and ICAG**

$N$	$21 \times 21$	$25 \times 25$	$29 \times 29$	$33 \times 33$
ICA	39.72	35.57	31.23	41.68
ICAG I*	18.56 (11)	15.69 (12)	11.28 (14)	13.79 (16)
ICAG II	21.32	16.15	13.31	15.36
Gabor	50.73	49.16	40.18	48.64

\* The values in bracket are the down-sampling factor  $d$ .

## 5. Conclusions

In this paper, we propose a new algorithm, ICAG, for texture segmentation and discuss its two different frameworks. Experimental results suggest ICAG II is a more efficient method to extract texture features, especially for some natural textures.

## Reference

- [1] M. Tuceryan and A.K. Jain, "Texture Analysis", in The Handbook of Pattern Recognition and Computer Vision (2nd Edition), by C.H. Chen, L.F. Pau, and P.S.P.W. (eds.), pp. 207-248, World Scientific Publishing Co., 1998.
- [2] F.W. Campbell and J.G. Robson, "Application of Fourier Analysis to the Visibility of Gratings". Journal of Physiology, vol.197, pp. 551-566, 1968.
- [3] A.C. Bovik, M. Clark, and W.S. Geisler, "Multichannel Texture Analysis Using Localized Spatial Filters". IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.12, no.1, pp. 55-73, 1990.
- [4] A.J. Bell and T.J. Sejnowski, "The " independent coponents" of Natural Scenes are Edge Filters". Vision Research, vol.37, pp. 3327-3338, 1997.
- [5] R. Manduchi and J. Portilla, "Independent Component Analysis of Textures [A]", in Proceedings of International Conference on Computer Vision, Corfu, Greece, 1999.
- [6] R. Jenssen and T. Eltoft, "ICA Filter Bank for Segmentation of Textured Images", in The 4th International Symposium on Independent Component Analysis and Blind Signal Separation, Nara, Japan, 2003.
- [7] A.K. Jain and F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters". Pattern Recognition, vol.24, no.12, pp. 1167-1186, 1991.
- [8] A. Hyvärinen, "Fast and Robust Fixed-Point Algorithms for Independent Component Analysis". IEEE Transactions on Neural Networks, vol.10, no.3, pp. 626-634, 1999.