Logical Operators for Texture Image Analysis

Vidya Manian and Ramón Vásquez

Electrical and Computer Engineering Department
University of Puerto Rico, Mayagüez, PR 00681-5000
manian@ece.uprm.edu, reve@ece.uprm.edu

Abstract

In this paper, logical operators are used for analyzing texture properties and an algorithm is presented for texture image classification. Operators constructed from logical building blocks are convolved with texture images. An optimal set of six operators are selected based on their texture discrimination ability. The responses are then converted to standard deviation matrices computed over a sliding window. Zonal sampling features are computed from these matrices. A feature selection process is applied and the new set of features are used for texture classification. Classification of texture images and a segmentation experiment are presented. The Euclidean distance classifier is found to perform best with this algorithm. Results show that this algorithm performs efficiently in classifying texture images.

1. Introduction

Texture classification is an image processing technique by which different regions of an image are identified based on texture properties. This process plays an important role in many industrial, biomedical and remote sensing applications. Early work utilized statistical and structural methods for texture feature extraction [1]. Gaussian Markov random field (GMRF) and Gibbs distribution texture models were developed and used for texture recognition [2]. Power spectral methods using the Fourier spectrum have also been used. DCT, Walsh-Hadamard and DHT have been used for recognition of two dimensional binary patterns [3]. One of the major developments recently in texture segmentation has been the use of multiresolution and multichannel descriptions [4] of the texture images. Logical operators have been used for Boolean analysis, minimization, spectral layered network decomposition, spectral translation synthesis, image coding, cryptography and communication. Logical systems considered in this work are Logical Hadamard transform, Adding and Arithmetic transforms and logical operators such as Equivalence, Negation and Conjunction. This work is a unique attempt in the following respects

(a) construction of a texture feature space using logical operators,
(b) the algorithm for image classification is computationally attractive with excellent performance over a wide variety of images.

This paper is organized as follows. Logical operators are described in Section 2. In Section 3, texture analysis using the operators is explained and the algorithm for texture classification is presented. In Section 4 experimental results of classifying different types of images. Finally, Section 5 gives the conclusions and a few pointers on future directions.

2. Logical Operators

The logical operators considered here are order-2 elementary matrices. The building blocks for defining these matrices are 0, 1, -1, matrices of order 1×1. These matrices can be formed by row-wise join or column wise join operations described in [5]. Some of the matrices are shown in Fig. 1.

$$\begin{bmatrix}1&1\\1&1\end{bmatrix}\ \begin{bmatrix}1&1\\1&-1\end{bmatrix}\ \begin{bmatrix}1&-1\\1&1\end{bmatrix}\begin{bmatrix}1&-1\\1&-1\end{bmatrix}$$

(a) Hadamard

$$\begin{bmatrix}1&0\\1&1\end{bmatrix}\ \begin{bmatrix}1&1\\1&0\end{bmatrix}\begin{bmatrix}1&0\\1&-1\end{bmatrix}\begin{bmatrix}1&-1\\1&0\end{bmatrix}\begin{bmatrix}1&0\\1&0\end{bmatrix}$$

(b) adding

$$\begin{bmatrix}1&1\\1&1\end{bmatrix}\begin{bmatrix}1&1\\1&1\end{bmatrix}\begin{bmatrix}1&-1\\1&1\end{bmatrix}\begin{bmatrix}1&-1\\1&1\end{bmatrix}\begin{bmatrix}1&0\\1&1\end{bmatrix}\begin{bmatrix}1&0\\1&0\end{bmatrix}\begin{bmatrix}1&0\\1&1\end{bmatrix}\begin{bmatrix}1&-1\\1&0\end{bmatrix}\begin{bmatrix}1&0\\1&0\end{bmatrix}$$

(c) arithmetic

$$\begin{bmatrix}1&1\\0&1\end{bmatrix}\ \begin{bmatrix}1&1\\0&1\end{bmatrix}\ \begin{bmatrix}1&0\\0&1\end{bmatrix}$$

(d) equivalence

$$\begin{bmatrix}1&1\\0&0\end{bmatrix}\ \begin{bmatrix}0&0\\0&1\end{bmatrix}$$

(e) conjunction

$$\begin{bmatrix}1&0\\0&1\end{bmatrix}\ \begin{bmatrix}0&1\\1&1\end{bmatrix}$$

(f) disjunction

Fig. 1. Examples of logical operators

Partially sponsored by: NSF EIA 9977071 and NASA NCCW-0088

3. Texture Analysis and Classification with Logical Operators

The operators described in Section II can be exploited for their characteristic to relate texture elements or primitives in a logical context. Their ability to extract texture features and the algorithm for texture classification are presented below.

3.1 Texture analysis

The operator masks are first convolved with texture regions,

$$G(u,v) = F(u,v) * O(u,v) \tag{1}$$

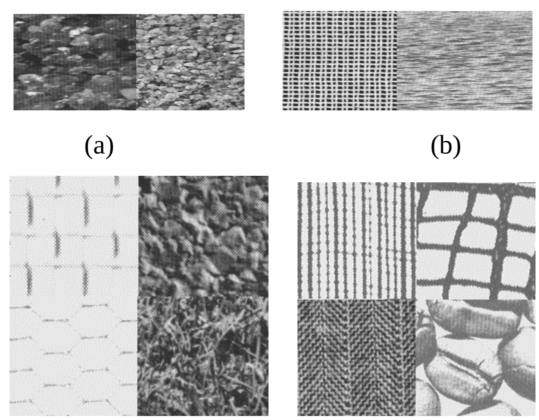
where *F* is the image function and O is one of the set of logical operators. The response of the texture images to the 6 operators given in Eq. (1) is used to compute a standard deviation matrix using a sliding window.

$$SD(u,v) = \frac{1}{W^2}\left\{\sum_{m=-w}^{w}\sum_{n=-w}^{w}\left[G(u+m,v+n) - M(u+m,v+n)\right]^2\right\}^{1/2} \tag{2}$$

The size of the scanning window is 5 x 5 and it slides pixel by pixel. The center pixel is assigned the standard deviation value. In order to avoid loosing boundary information the images are padded with zeros on all sides. Twelve images from the Brodatz album [6] with typical texture characteristics are used (see Fig. 2). The operators in Fig. 1 are convolved with the texture samples of size 64 x 64, as per Eq. (1) which constitutes a filtering operation. The standard deviation matrix for each operator response is computed as in Eq. (2), which can be seen as a smoothing operation. The average l_1-norm of Eq. (2) is computed as

$$e = \frac{1}{N_1 N_2}\sum_{u=1}^{N_1}\sum_{v=1}^{N_2}\left|SD(u,v)\right| \tag{3}$$

where N_1 and N_2 are the size of the standard deviation matrix. From these values the properties of the operators can be examined. It is known that the Hadamard operators have good energy compaction properties. It can also be verified that they yield maximum discrimination for coarse and fine textures as can be seen from the values of Eq. (3) (see Fig. 3) for the 4 Hadamard operators (H1-H4) for textures d28 and d29 (shown on left and right of Fig. 2 (a)). The values have been normalized, 1 represents maximum coarseness. The adding and arithmetic operators (AD1-AD5 and AR1 to AR9) extract contrast properties from the textures with a value close to 1 for high contrast textures. High and low contrast textures d21 and d38 are shown in the left and right of Fig. 2(b), respectively. The corresponding values are plotted in Fig. 3 which shows that highest separation is obtained with operators AD1 and AR5. The conjunction and disjunction operators (CN1- CN2 and DN1-DN2) extract randomness information from the texture and have higher values close to 1 for irregular or random textures (d4 and d9 shown in right top and bottom of Fig. 2(c)), and lesser values for regular and periodic textures (d6 and d34 shown in left top and bottom of Fig. 2(c)). As seen from the plot in Fig. 3, CN1 and DN1 operators yield maximum separability. The equivalence operators (EQ1-EQ3) perform an averaging operation over the textured image, and give a cue of the size of the texture element yielding higher values for smaller elements (textures d105 and d16 shown in left top and bottom of Fig. 2(d)) and lesser values for larger texture structures (textures d103 and d74 shown in right top and bottom of Fig. 2(d)). The graph in Fig. 3 shows maximum discrimination with the equivalence operator EQ3. From this analysis, one operator from each class of logical operator that gives the maximum separability among textures is chosen as the most powerful among the rest. The final set of six operators (H2, AD1, AR5, EQ3, CN1 and DN1) is shown in Fig. 4.



(a) (b) (c) (d)

Fig. 2. Texture samples (a) coarseness, (b) contrast, (c) randomness, (d) texture element size

3.2 Algorithm for texture classification

The texture samples are convolved with the operators as in Eq. (1). The standard deviation matrix for each response is computed as in Eq. (2). Features are extracted by zonal-filtering using zonal masks which are applied to the standard deviation matrix. The zonal mask, also called zonal filter, is a simple slit/mask or an aperture. A combination of an angular slit with a bandlimited low-pass, band-pass or high-pass filter can be used for yielding good discriminating features for periodic or quasiperiodic textures. SD(u,v) is the standard deviation matrix, where

$1 \le u \le N_1$ and $1 \le v \le N_2$, N_1 and N_2 are the number of rows and columns in the matrix. Masks are sets of integers that are used to extract features from the standard deviation matrix.

Horizontal slit feature $Y_1 = \sum_{(u,v)\in H_m} SD(u,v)$ (4)

where the horizontal slit mask $H_m = \{(u,v) : u, v \text{ integer}, U_1 \le u \le U_2; 1 \le v \le N_2\}$

Vertical slit feature: $Y_2 = \sum_{(u,v)\in V_m} SD(u,v)$ (5)

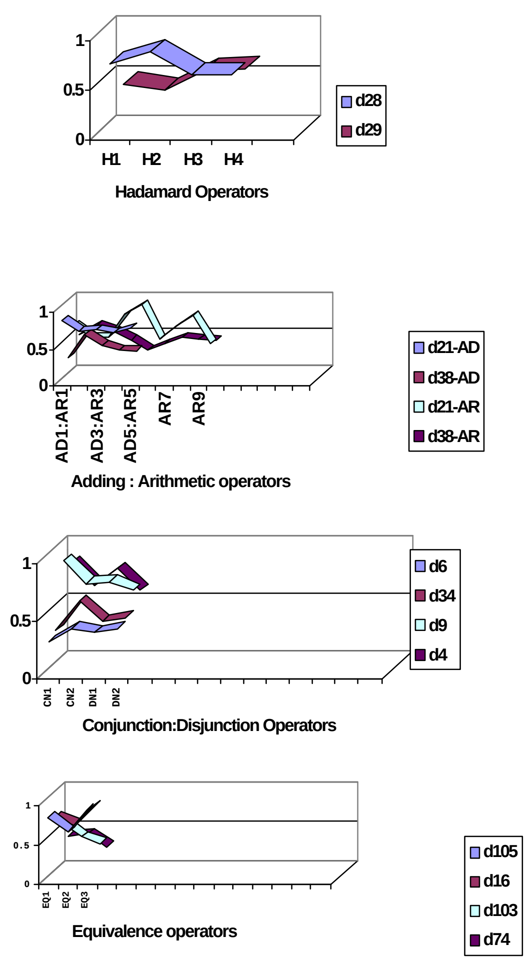


Fig. 3. Operator response plots

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

Fig. 4. Selected logical operators

where the vertical slit mask $V_m = \{(u,v) : u, v \text{ integer}, 1 \le u \le N_1; V_1 \le v \le V_2\}$ If x and y are defined as $x = c + dv;\ y = a + bu$, where $c = -\frac{N_2+1}{N_2-1}$, $d = \frac{2}{N_2-1}$, $a = \frac{N_1+1}{N_1-1}$ and $b = -\frac{2}{N_1-1}$.

Then, ρ and θ are in polar coordinates and are defined as $r(u,v) = \sqrt{(a+bv)^2 + (c+du)^2}$ and $\tan q(u,v) = \frac{a+bu}{c+dv}$.

Ring feature: $Y_3 = \sum_{(u,v)\in R_m} SD(u,v)$, (6)

where the ring mask $R_m = \{(u,v) : u, v \text{ integer}, r_1 \le r(u,v) \le r_2\}$

Circular feature: $Y_4 = \sum_{(u,v)\in C_m} SD(u,v)$, (7)

where circular mask $R_m = \{(u,v) : u, v \text{ integer}, r_1 \le r(u,v) \le r_2\}$

Sector feature: $Y_5 = \sum_{(u,v)\in S_m} SD(u,v)$, (8)

where $S_m = \{(u,v) : u, v \text{ integer}, q_1 \le q \le q_2\}$

These features can be computed with different sizes of masks and they form a feature vector Y for samples from each texture class. Let the length of the feature vector Y be L, $\hat{Y}$ is the normalized resulting vector, i is the index for Y, such that i ≤ L. A combination of two criteria, the distance between the means of each feature and the measure of standard deviation are used to quantify the separation between classes. If M is the mean and σ is the standard deviation of the features in the training matrix, the sum of the distances of each feature from M and σ are computed as

$$D_L = \sum_j \left| \hat{Y}_{i,j} - M_i \right| \quad (9)$$

where i is the feature index, L is the number of features, j is the class index and J is the total number of classes. The standard deviation value for each feature is computed as

$$s_L = \sqrt{\frac{1}{J} \sum_{j=1}^{J} \left(Y_{i,j} - M_i\right)^2} \quad (10)$$

Values computed from Eqs. (9-10) are sorted in ascending order and the features with the first half indices are selected as the best overall set of features. Both the parametric and non-parametric classifiers are used in the experiments.

4. Experimental Results

Nine experiments with 6 textures [6], each were conducted with the minimum Euclidean distance classifier: $d_j(x) = \sum_{q=1}^{Q} \left(x_q - m_{i,q}\right)^2$ (11)

And K-nearest neighbor classifier:

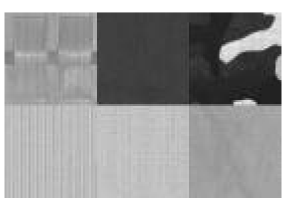
$P(w_j \mid x)$ if $k-nearest\ neighbors$ of x are labeled w_j (12)

The PCCs for this experiment are shown in Table 1. The average number of features selected in 7. Both classifiers perform closely. The Euclidean distance measure is found to be the most suitable one for this algorithm due to the nature of features used, the classifier simplicity and speed.

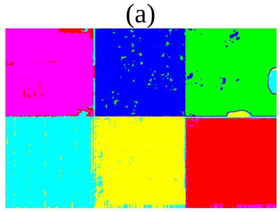
Segmentation of 6 textured images is done in this experiment. The original mosaic of six textile textures (basket, naugahyde leather, a fabric texture, corduroy, cotton and tanned leather) is shown in Fig. 5(a). Each texture is of size 256 x 256 and the mosaic is of size 512 x 768. The algorithm is applied to segment this image using a moving window of size 8 x 8 with an overlap of 7 pixels. The segmentation result is shown in Fig. 5(b) which shows excellent classification with minimal errors in the boundary.

Table 1. Classification results

Textured Image	% Correct Classification (PCC) results	
Mosaic of six Brodatz Textures with Texture ID's	k-nearest neighbor classifier	Euclidean Distance Classifier
D-94/101/36/84/103/56	93	93
D-22/28/9/38/4/57	99	100
D-28/20/9/38/50/57	96	99
D-6/105/24/19/68/16	97	98
D-77/4/16/15/24/9	94	97
D-90/74/93/34/65/53	88	93
D-105/79/82/52/19/78	89	91
D-28/9/57/24/4/38	99	99
D-103/105/12/78/79/82	90	95
Average PCC	94	96



(a)



(b)

Fig. 5. (a) Original textile composite image, (b) Segmented image

In the experiments with Brodatz textures, texture samples of 64x64 size is optimal due to the wide range of textures involved with small to large structures. In general, the sample size should be large enough to yield reliable features and small enough to produce accurate boundaries in segmentation problems. Hence, 8x8 window size has been used in segmentation of Fig. 5(a), where texture structure to be characterized are small. To avoid the curse of dimensionality, a feature selection process has been applied to select the optimal set of features.

5. Conclusions

Logical operators has been used to analyze texture images and shown to extract different properties from them useful for classification. An algorithm for using the same for feature extraction and classification of images has been presented. Results with textured images and segmentation of a mosaic image gives good results which proves the efficiency of the algorithm. The algorithm can be employed for classifying other types of images such as medical images and for object recognition.

References

[1] A. K. Jain, "Statistical pattern recognition: a review," *IEEE Trans. PAMI*, , Vol. 22, No. 1, pp. 4-37, Jan. 2000.

[2] S. Krishnamachari and R. Chellappa, "Multiresolution Gauss-Markov random field models for texture segmentation." *IEEE. Trans. Image Processing*, vol. 6, pp. 251-267. Feb 1997.

[3] J. Wu and W. Duh, "Feature extraction capability of some discrete transforms," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-36, pp. 1687, Oct. 1991.

[4] A. C. Bovik, M. Clark and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. and Machine Intell*., vol. 12, pp. 55-73, Jan. 1990.

[5] B. J. Falkowski, and M. A. Perkowski, "A family of all essential Radix addition/subtraction multipolarity transforms: Algorithms and interpretations in Boolean domain," *Proc. 23rd IEEE Int. Symp. On Circuits & Systems*, New Orleans, LA, pp.1596-1599, May 1990.

[6] P. Brodatz, *Textures-A photographic Album for Artists and Designers*, Dover, New York, 1966.