Texture information in run-length matrices

Article *in* IEEE Transactions on Image Processing · December 1998

DOI: 10.1109/83.725367 · Source: IEEE Xplore

Xiaoou Tang
The Chinese University of Hong Kong

References

[1] M. F. Barnsley, *Fractals Everywhere.* New York: Academic, 1988.

[2] A. E. Jacquin, "Fractal image coding: A review," *Proc. IEEE*, vol. 81, pp. 1451–1465, Oct. 1993.

[3] G. E. Øien, S. Lepsøy, and T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," in *Proc. ICASSP*, 1991, pp. 2773–2776.

[4] D. M. Monro, "A hybrid fractal transform," in *Proc. ICASSP*, 1993, vol. 5, pp. 169–172.

[5] S. Lepsøy, G. E. Øien, and T. A. Ramstad, "Attractor image compression with a fast noniterative algorithm," in *Proc. ICASSP*, 1993, vol. 5, pp. 337–340.

[6] M. Gharavi-Alkhansari and T. S. Huang, "A fractal-based image block-coding algorithm," in *Proc. ICASSP*, 1993, vol. 5, pp. 345–348.

[7] G. Vines, "Signal modeling with iterated function systems," Ph.D. dissertation, Georgia Inst. Technol., Atlanta, 1993.

[8] Y. Fisher, Ed. *Fractal Image Compression-Theory and Application*. Berlin, Germany: Springer-Verlag, 1995.

[9] L. Thomas and F. Deravi, "Region-based fractal image compression using heuristic search," *IEEE Trans. Image Processing*, vol. 4, pp. 832–838, June 1995.

[10] C. J. Wein and I. F. Blake, "On the performance of fractal image compression with clustering," *IEEE Trans. Image Processing*, vol. 5, pp. 522–526, Mar. 1996.

[11] I. K. Kim and R.-H. Park, "Still image coding based on vector quantization and fractal approximation," *IEEE Trans. Image Processing*, vol. 5, pp. 587–597, Apr. 1996.

[12] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.*Boston, MA: Kluwer, 1992.

[13] M. B. Brahmanandam, S. Panchanthan, and M. Goldberg, "An affine transform based image vector quantizer," in *Proc. SPIE: VCIP*, 1993, pp. 1639–1647.

Texture Information in Run-Length Matrices

Xiaoou Tang

Abstract—**We use a multilevel dominant eigenvector estimation algorithm to develop a new run-length texture feature extraction algorithm that preserves much of the texture information in run-length matrices and significantly improves image classification accuracy over traditional run-length techniques. The advantage of this approach is demonstrated experimentally by the classification of two texture data sets. Comparisons with other methods demonstrate that the run-length matrices contain great discriminatory information and that a good method of extracting such information is of paramount importance to successful classification.**

***Index Terms*—Pattern classification, run-length matrix, texture analysis.**

I. Introduction

Texture is the term used to characterize the surface of a given object or region and it is one of the main features utilized in image processing and pattern recognition. Many texture analysis methods have been developed over the past few decades [2], [7]–[11], [14]. One such method characterizes texture images based on run-lengths of image gray levels. First introduced by Galloway [7], the *run-length method* has not been widely accepted as an effective texture analysis approach. Several comparison studies conducted by Weszka *et al.* [15] and Conners and Harlow [4] have shown that the run-length features are the least efficient texture features among a group of traditional texture features, such as the co-occurrence features, the gray level difference features, and the power spectrum features. Applications of the run-length method have been very limited compared to other methods.

In this correspondence, we investigate the run-length method with a new approach. By using a multilevel dominant eigenvector estimation algorithm and the Bhattacharyya distance measure for feature extraction, we demonstrate that texture features extracted from the run-length matrix can produce great classification results. Experimental comparison with the widely used co-occurrence method and the recently proposed wavelet method show that the run-length matrices contain sufficient discriminatory information and that a good method of extracting such information is crucial to a successful classification.

This work is organized into four sections. Section II introduces the original definition of the run-length matrix and several of its variations, then reviews the traditional run-length features and describes the new run-length feature extraction algorithm. Section III presents the texture classification experimental results. The conclusions are summarized in Section IV.

II. Methodology

A. Definition of the Run-Length Matrices

With the observation that, in a coarse texture, relatively long gray-level runs would occur more often and that a fine texture should contain primarily short runs, Galloway proposed the use of a run-length matrix for texture feature extraction [7]. For a given image, a run-length matrix $p(i, j)$ is defined as the number of runs with pixels of gray level i and run length j. Various texture features can then be derived from this run-length matrix.

Here, we design several new run-length matrices, which are slight but unique variations of the traditional run-length matrix. For a run-length matrix $p(i, j)$, let M be the number of gray levels and N be the maximum run length. The four new matrices are defined as follows.

1) Gray Level Run-Length Pixel Number Matrix:

$$p_p(i, j) = p(i, j) \cdot j. \tag{1}$$

Each element of the matrix represents the number of pixels of run-length j and gray-level i. Compared to the original matrix, the new matrix gives equal emphasis to all length of runs in an image.

2) Gray-Level Run-Number Vector:

$$p_g(i) = \sum_{j=1}^{N} p(i, j). \tag{2}$$

This vector represents the sum distribution of the number of runs with gray level i.

3) Run-Length Run-Number Vector:

$$p_r(j) = \sum_{i=1}^{M} p(i, j). \tag{3}$$

This vector represents the sum distribution of the number of runs with run length j.

Manuscript received April 22, 1996; revised October 13, 1997. This work was supported in part by the Chinese University of Hong Kong through Direct Grant C001–2050183, and by the Office of Naval Research under Grant N00014-93-1-0602. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Maria Petrou.

The author is with the Department of Information Engineering, the Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: xtang@alum.mit.edu).

Publisher Item Identifier S 1057-7149(98)06813-4.

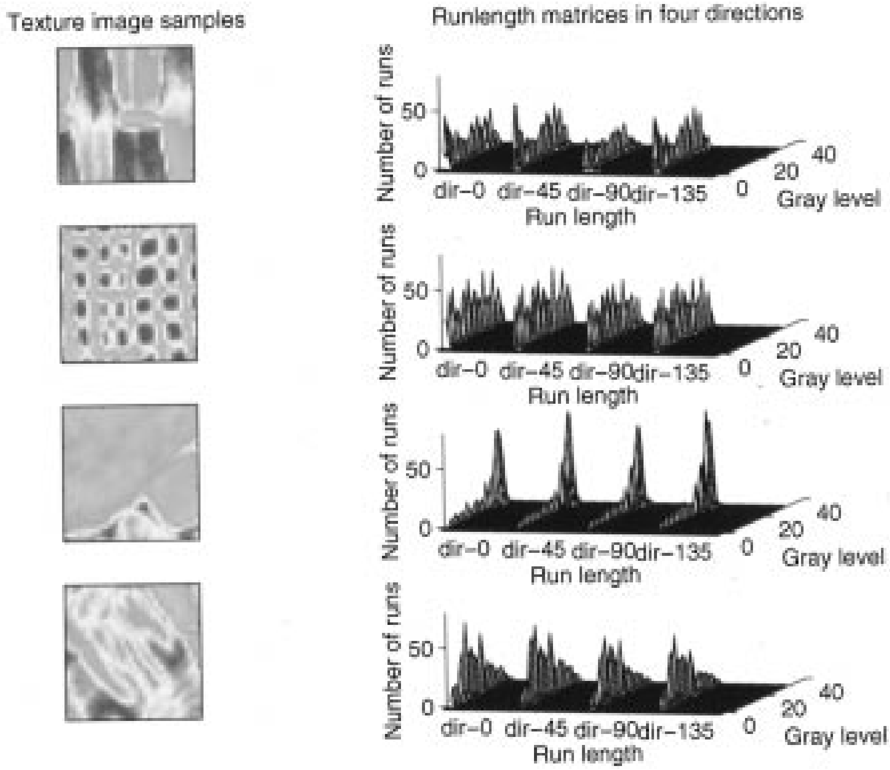

Fig. 1. Four directional run-length matrices of several Brodatz texture samples. Each image sample is of size 32 × 32 with 32 gray levels. The four directional (0°, 45°, 90°, and 135° directions) run-length matrices are combined into a single matrix. The left-most column of each directional matrix is the run-length-one vector, which has much larger values than the other columns.

4) Gray-Level Run-Length-One Vector:

$$p_o(i) = p(i,\, 1). \tag{4}$$

Fig. 1 shows the four-directional run-length matrices of several natural texture samples. Notice that the first column of each of the four directional run-length matrices is overwhelmingly larger than the other columns. This may mean that most texture information is contained in the run-length-one vector. The advantages of using this vector are that it offers significant feature length reduction and that a fast parallel run-length matrix computation can replace the conventional serial searching algorithm. For example, the positions of pixels with run-length one in the horizontal direction can be found by a logical "and" operation on the outputs of the forward and backward derivative of the original image:

$$f(i,\, j) = x(i,\, j) - x(i,\, j-1) \tag{5}$$

$$b(i,\, j) = x(i,\, j-1) - x(i,\, j) \tag{6}$$

$$o(i,\, j) = f(i,\, j) \cap b(i,\, j) \tag{7}$$

where $x(i,\, j)$ is the texture image whose pixels outside the image boundary are set to zero, and $\cap$ represents the logical "and" operation. Then $p_o(i)$ can be obtained by computing the histogram of $x(i,\, j)_{o(i,\, j)=1}$. To find the starting pixel position for runs with length two, a similar scheme can be employed as follows:

$$f_2(i,\, j) = (f(i,\, j) \neq 0) - o(i,\, j) \tag{8}$$

$$b_2(i,\, j) = (b(i,\, j) \neq 0) - o(i,\, j) \tag{9}$$

$$o_2(i,\, j) = f_2(i,\, j) \cap b_2(i,\, j+1). \tag{10}$$

In fact, the gray level run number vector $p_g(i)$ can also be obtained with the above approach by computing the histogram of $x(i,\, j)_{f(i,\, j)\neq 0}$.

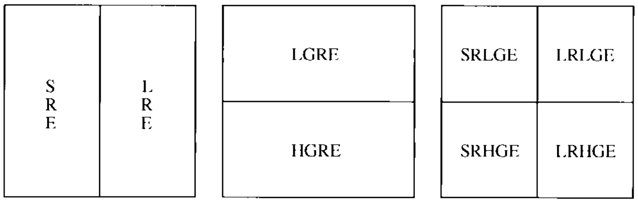

Fig. 2. Run-emphasis regions of several traditional run-length texture features.

The matrix and vectors defined above are not designed for the extraction of traditional features. Along with the original run-length matrix, they are used in the new feature extraction approach in Section II-C. The next section gives a review of the traditional feature extraction.

B. Traditional Run-Length Features

From the original run-length matrix $p(i,\, j)$, many numerical texture measures can be computed. The five original features of run-length statistics derived by Galloway [7] are as follows.

1) *Short Run Emphasis (SRE)*:

$$\text{SRE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{p(i,j)}{j^2} = \frac{1}{n_r} \sum_{j=1}^{N} \frac{p_r(j)}{j^2}. \tag{11}$$

2) *Long Run Emphasis (LRE)*:

$$\text{LRE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} p(i,\, j) \cdot j^2 = \frac{1}{n_r} \sum_{j=1}^{N} p_r(j) \cdot j^2. \tag{12}$$

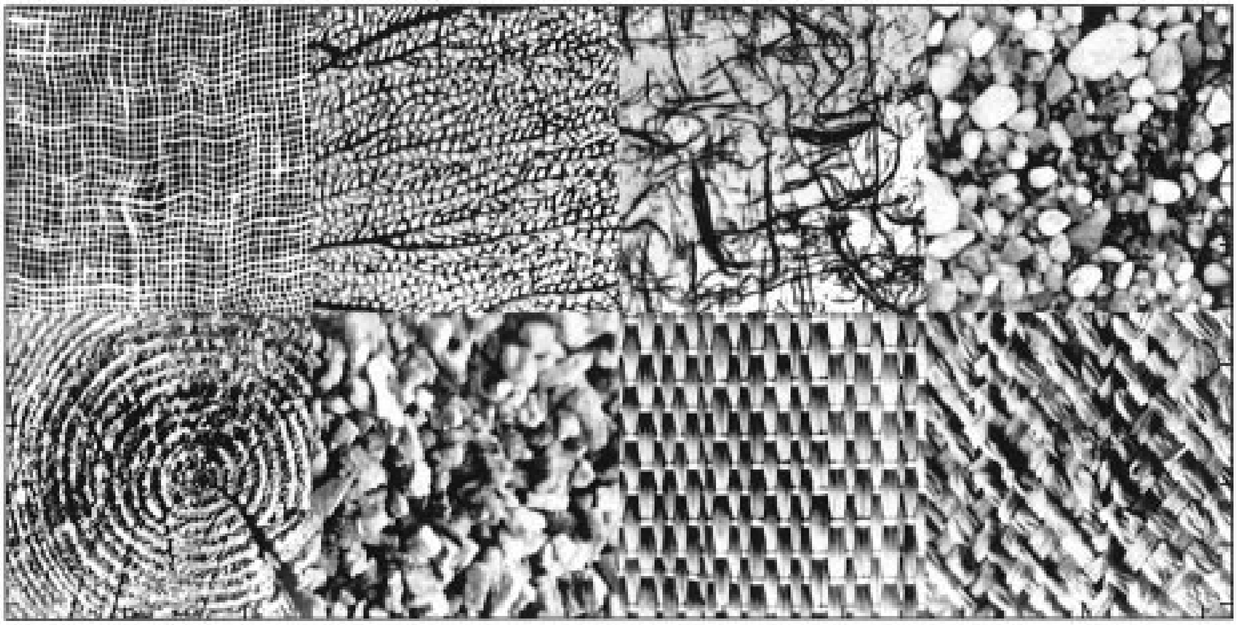

Fig. 3. Eight Brodatz textures. Row 1: burlap, seafan, rice paper, pebbles23. Row 2: tree, mica, straw, raffia.

3) *Gray-Level Nonuniformity (GLN)*:

$$\mathrm{GLN} = \frac{1}{n_r} \sum_{i=1}^{M} \left(\sum_{j=1}^{N} p(i, j) \right)^2 = \frac{1}{n_r} \sum_{i=1}^{M} p_g(i)^2. \tag{13}$$

4) *Run Length Nonuniformity (RLN)*:

$$\mathrm{RLN} = \frac{1}{n_r} \sum_{j=1}^{N} \left(\sum_{i=1}^{M} p(i, j) \right)^2 = \frac{1}{n_r} \sum_{j=1}^{N} p_r(i)^2. \tag{14}$$

5) *Run Percentage (RP)*:

$$\mathrm{RP} = \frac{n_r}{n_p}. \tag{15}$$

In the above, n_r is the total number of runs and n_p is the number of pixels in the image. Based on the observation that most features are only functions of $p_r(j)$, without considering the gray level information contained in $p_g(i)$, Chu *et al.* [3] proposed two new features, as follows, to extract gray level information in the matrix.

1) *Low Gray-Level Run Emphasis (LGRE)*:

$$\mathrm{LGRE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{p(i,j)}{i^2} = \frac{1}{n_r} \sum_{i=1}^{M} \frac{p_g(i)}{i^2}. \tag{16}$$

2) *High Gray-Level Run Emphasis (HGRE)*:

$$\mathrm{HGRE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} p(i, j) \cdot i^2 = \frac{1}{n_r} \sum_{i=1}^{M} p_g(i) \cdot i^2. \tag{17}$$

In a more recent study, Dasarathy and Holder [5] described another four feature extraction functions following the idea of joint statistical measure of gray level and run length, as follows.

1) *Short Run Low Gray-Level Emphasis (SRLGE)*:

$$\mathrm{SRLGE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{p(i,j)}{i^2 \cdot j^2}. \tag{18}$$

2) *Short Run High Gray-Level Emphasis (SRHGE)*:

$$\mathrm{SRHGE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{p(i,j) \cdot i^2}{j^2}. \tag{19}$$

3) *Long Run Low Gray-Level Emphasis (LRLGE)*:

$$\mathrm{LRLGE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{p(i,j) \cdot j^2}{i^2}. \tag{20}$$

4) *Long Run High Gray-Level Emphasis (LRHGE)*:

$$\mathrm{LRLGE} = \frac{1}{n_r} \sum_{i=1}^{M} \sum_{j=1}^{N} p(i, j) \cdot i^2 \cdot j^2. \tag{21}$$

Dasarathy and Holder [5] tested all eleven features on the classification of a set of cell images and showed that the last four features gave better performance.

These features are all based on intuitive reasoning, in an attempt to capture some apparent properties of run-length distribution. For example, the eight features illustrated in Fig. 2 are weighted-sum measures of the run-length concentration in the eight directions, i.e., the positive and negative 0°, 45°, 90°, and 135° directions. Two drawbacks of this approach are: there is no theoretical proof that, given a certain number of features, maximum texture information can be extracted from the run-length matrix, and many of these features are highly correlated with each other.

C. Dominant Run-Length Method (DRM)

Instead of developing new functions to extract texture information, we use the run-length matrix as the texture feature vector directly to preserve all information in the matrix. However, this again introduces two problems: the large dimensionality of the feature vector and the high-degree correlation of the neighborhood features.

To alleviate the first problem, observe the run-length matrix in Fig. 1 more closely. We see that most nonzero values concentrate

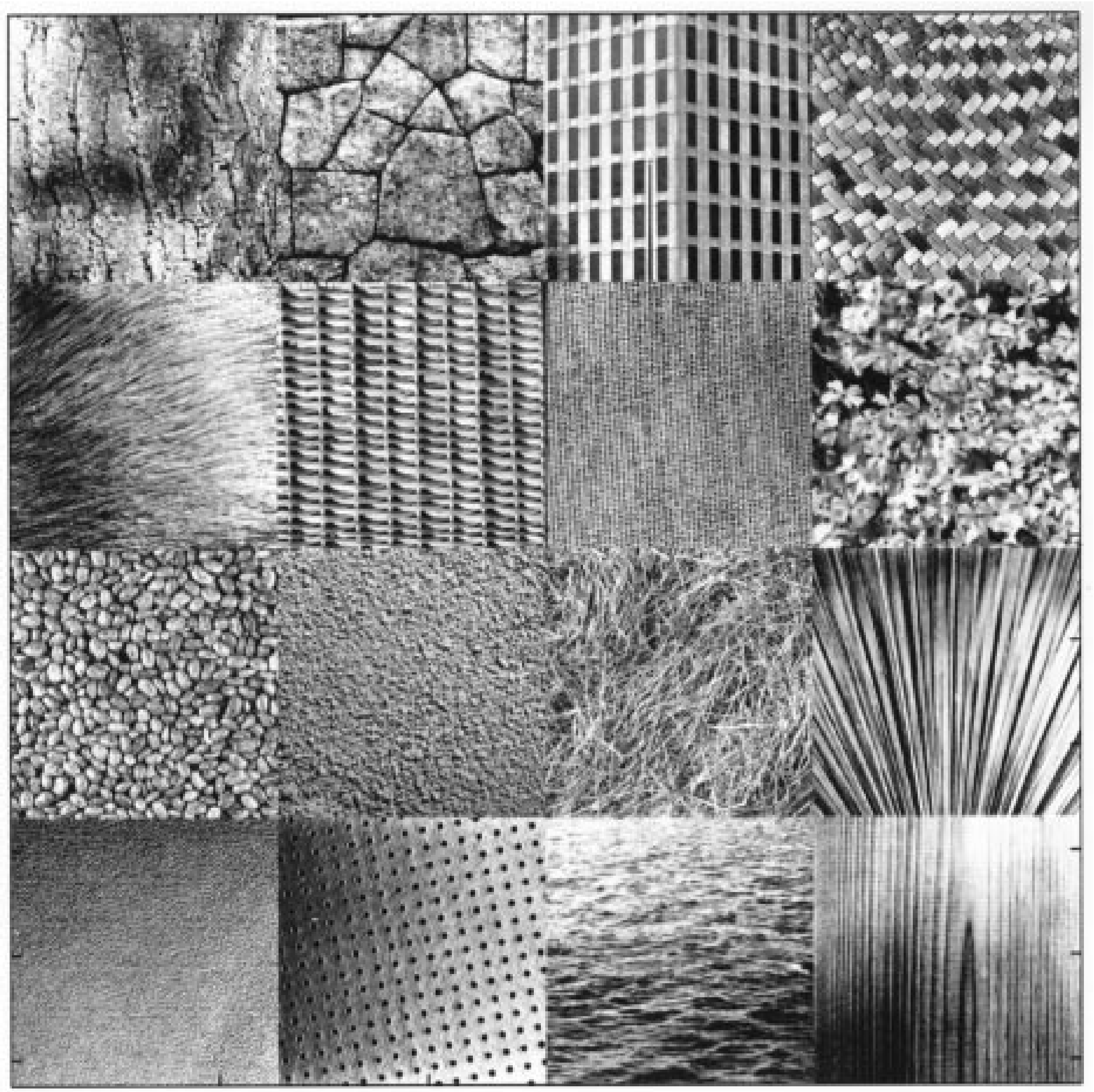

Fig. 4. Sixteen Vistex textures.

TABLE I
BRODATZ TEXTURE CLASSIFICATION RESULTS USING THE NEW FEATURE SELECTION METHOD ON THE TRADITIONAL RUN-LENGTH FEATURES

Feature name	Original feature length	Number of selected features	Correct classification rate		
			Training data	Testing data	All data
G5	20	12	88.5	74.9	78.6
C2	8	8	61.2	41.8	47.0
D4	16	16	84.4	59.1	65.8
ALL	44	24	99.4	83.7	87.9

in the first few columns of the matrix. Moreover, because of the correlation between the short-run section and the long-run section, using only the first few columns as the feature vector will also preserve most of the information in the long-run section. Another advantage of using only the first few columns is that the fast parallel run-length matrix computation algorithm described in Section II-A can be used.

To further reduce the feature vector dimension and to decorrelate neighboring element values in the matrices, we use the principal component analysis method, also called Karhunen–Loeve transform

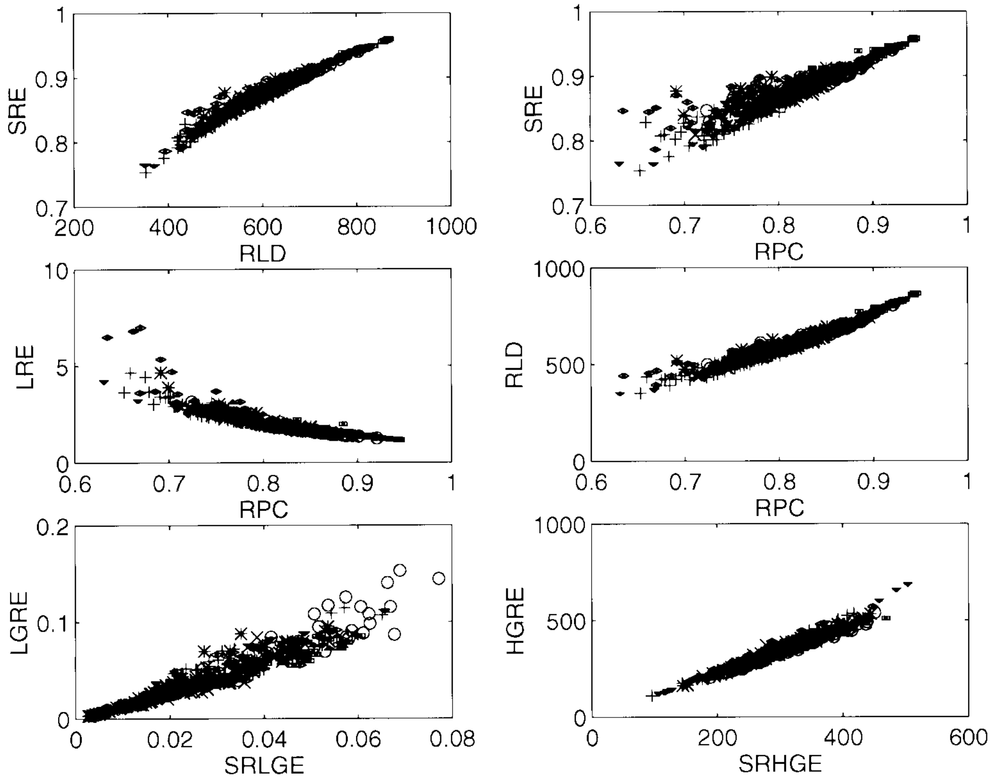

Fig. 5. Scatter plots of several highly correlated traditional run-length texture features of the eight Brodatz textures. Due to overlap, not all eight class symbols can be discerned.

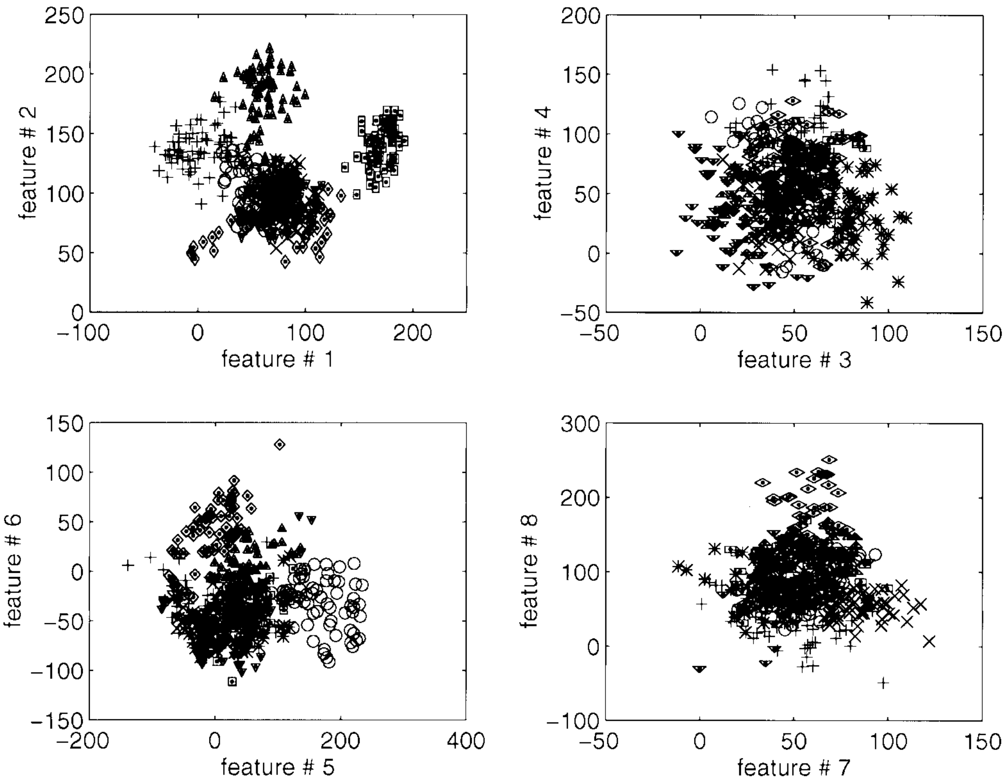

Fig. 6. Scatter plots of the top eight features extracted by applying an MDEE transform on the original run-length matrices of the Brodatz textures. Linearly separable clustering is observed for most of the eight texture classes.

TABLE II
BRODATZ TEXTURE CLASSIFICATION RESULTS USING THE NEW DOMINANT RUN-LENGTH MATRIX FEATURES

Feature name	Original feature length	Number of selected features	Correct classification rate		
			Training data	Testing data	All data
p: columns 1:4	512	11	100.0	100.0	100.0
p: columns 5:32	3584	8	53.3	41.3	44.5
p: whole matrix	4096	11	100.0	100.0	100.0
p_p: columns 1:4	512	7	100.0	100.0	100.0
p_p: columns 5:32	3584	17	69.6	41.4	48.9
p_p: whole matrix	4096	10	100.0	100.0	100.0
p_g: GLRNV	128	8	100.0	100.0	100.0
p_r: RLRNV	128	20	95.2	63.9	72.3
p_o: GLRLOV	128	11	100.0	100.0	100.0

TABLE III
VISTEX TEXTURE CLASSIFICATION RESULTS USING THE NEW DOMINANT RUN-LENGTH MATRIX FEATURES

Feature name	Original feature length	Number of selected features	Correct classification rate		
			Training data	Testing data	All data
p: columns 1:4	512	17	99.9	96.8	97.6
p: whole matrix	4096	18	99.9	98.0	98.5
p_p: columns 1:4	512	19	100.0	96.8	97.6
p_p: whole matrix	4096	24	100.0	97.5	98.1
p_g: GLRNV	128	23	100.0	93.9	95.6
p_o: GLRLOV	128	18	99.8	97.0	97.8

(KLT), and then use the Bhattacharyya distance measure to rank the eigenfeatures according to their discriminatory power.

1) Multilevel Dominant Eigenvector Estimation (MDEE) Method: To compute the KLT, let x_i be a feature vector sample. We form an n by m matrix

$$A = \begin{bmatrix} x_1(1)x_2(1)\cdots x_m(1) \\ x_1(2)x_2(2)\cdots x_m(2) \\ \vdots\vdots\vdots\vdots \\ x_1(n)x_2(n)\cdots x_m(n) \end{bmatrix} \tag{22}$$

where n is the feature vector length and m is the number of training samples. The eigenvalues of the sample covariance matrix are computed in two ways, depending on the size of the feature vector. If the feature vector length n is a small number, eigenvalues are computed by a standard procedure. The sample covariance matrix is estimated by

$$W = \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu)(x_i - \mu)^T = \frac{1}{m}AA^T \tag{23}$$

where μ is the mean vector. The eigenvalues and eigenvectors are computed directly from W. However, for the feature vector formed by the four directional run-length matrices, n is a large number. For a neighborhood of 32 × 32 with 32 gray levels, n can reach a maximum of 4096, producing a covariance matrix of size 4096 × 4096. Direct computation of the eigenvalues and eigenvectors becomes impractical.

To avoid this problem, we use a new multilevel dominant eigenvector estimation method developed in [12]. By breaking the long feature vector into $g = n/k$ groups of small feature vectors of length k,

$$A = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_g \end{bmatrix} \tag{24}$$

where

$$B_g = \begin{bmatrix} x_1((g-1)k+1)\cdots x_m((g-1)k+1) \\ \vdots\vdots\vdots \\ x_1(gk)\cdots x_m(gk) \end{bmatrix}$$

we can perform the KLT on each of the g group short feature vector set B_i. Then a new feature vector is formed by the first few selected dominant eigenfeatures of each group. The final eigenvectors are computed by applying the KLT to this new feature vector. For proof that the eigenvalues computed by MDEE are a close approximation of the standard KLT, refer to [12].

Significant reduction of computational time can be achieved by the MDEE over the standard KLT. For example, if a feature vector of length $n = 1000$ is broken into ten vector groups of length 100, and 10% of the eigenfeatures in each group are saved for the second-level eigenvalue computation, the computational complexity for the MDEE is $11(n/10)^3$, which is nearly two orders of magnitude faster than the KLT's 1000^3.

2) Bhattacharyya Distance Measure: However, it is well known that the KLT features are optimal for data representation but not necessarily the best for discrimination. To measure the class separability of each feature, some other criterion must be employed. We choose the Bhattacharyya distance measure. An analytical form of this measure is [6]

$$\beta = \frac{1}{8}(\mu_1 - \mu_2)^T\left(\frac{\sigma_1 + \sigma_2}{2}\right)^{-1}(\mu_1 - \mu_2) + \frac{1}{2}\ln\frac{\left|\frac{1}{2}(\sigma_1 + \sigma_2)\right|}{|\sigma_1|^{1/2}|\sigma_2|^{1/2}} \tag{25}$$

where σ_i and μ_i are the class variance and mean. From (25) we see that the Bhattacharyya distance β is proportional to both the distance of class means and the difference of class variances. The feature selection criterion is to only retain those decorrelated features with large β value.

Throughout the experiments, we select the first 30 features with largest eigenvalues, rank these KLT-decorrelated features by their β values, and use the first n features with the largest β for classification. We run the feature length n from one to 30 to select the one that gives the best performance as the final feature vector length.

D. Classification Algorithm

Since the main focus of this work is the feature extraction algorithm, we use a simple Gaussian classifier for the experiments. Let the class mean and covariance matrix of the feature vector x be μ_i and W_i, respectively, a distance measure is defined as [13]

$$D_i = (x - \mu_i)^T W_i^{-1}(x - \mu_i) + \ln|W_i| \tag{26}$$

where the first term on the right of the equation is actually the Mahalanobis distance. The decision rule is

$$x \in C_L \quad \text{when} \quad D_L = \min\{D_i\}. \tag{27}$$

TABLE IV
VISTEX TEXTURE CLASSIFICATION RESULTS USING THE CO-OCCURRENCE, WAVELET, AND NEW RUN-LENGTH FEATURES

Feature name		Original feature length	Number of selected features	Correct classification rate		
				Training data	Testing data	All data
Co-occurrence		52	29	100.0	97.4	98.1
Wavelet	Level 2	16	13	98.2	90.6	92.7
	Level 3	64	20	98.6	90.1	92.4
	All Levels	84	15	97.9	90.6	92.5
Run-length		4096	18	99.9	98.0	98.5

TABLE V
VISTEX TEXTURE CLASSIFICATION RESULTS USING THE CO-OCCURRENCE, WAVELET, AND RUN-LENGTH FEATURES, WITH IMAGE SAMPLE SIZE 64 × 64

Feature name		Original feature length	Number of selected features	Correct classification rate		
				Training data	Testing data	All data
Co-occurrence		52	24	100.0	99.8	99.8
Wavelet	Level 2	16	10	100.0	99.5	99.6
	Level 3	64	12	100.0	99.3	99.5
	Level 4	256	22	100.0	98.2	98.6
	All Levels	340	24	100.0	98.1	98.5
Run-length		8192	13	100.0	100.0	100.0

III. EXPERIMENTS AND DISCUSSION

In this section, two separate data sets are used for the texture classification experiment. We first make detail comparisons between various DRM features and the traditional run-length features on the classification of eight Brodatz images. We then compare the best DRM features with the co-occurrence and wavelet features on the classification of a larger data set—16 Vistex images.

A. Data Description

The eight Brodatz images [1] are shown in Fig. 3. Each image is of size 256 × 256 with 256 gray levels. The images are first quantized into 32 gray levels using equal-probability quantization. Each class is divided into 225 sample images of dimension 32 × 32 with 50% overlapping. Sixty samples of each class are used as training data, so the training data size is 480 samples and the testing data size is 1320.

To further compare the performance consistency of the DRM features, we conducted a second experiment on a larger data set—16 images from Vistex texture image data base of the MIT Media Laboratory. Unlike Brodatz images, which are mostly obtained in well controlled studio conditions, the Vistex images are taken under natural lighting conditions. The Vistex images shown in Fig. 4 are: Bark.08, Brick.04, Buildings.09, Fabric.01, Fabric.05, Fabric.13, Fabric.17, Flowers.07, Food.01, Food.05, Grass.02, Leaves.02, Metal.01, Tile.07, Water.06, Wood.02. The same 32 gray level quantization is applied to each image. Each class is again divided into 225 samples of dimension 32 × 32 with 50% overlapping. Sixty samples of each class are used as training data. So the training data has 960 samples and the testing data has 2640 samples.

B. Comparison of the Traditional Run-Length Features and the New DRM Features

We first use the feature selection algorithms on the traditional run-length features for the classification of the Brodatz images. Similar to [5], the feature groups tested are the original five features of Galloway [7], the two features of Chu *et al.* [3], and the four new features of Dasarathy and Holder [5]. All four-direction features are used. Results are shown in Table I. The feature vector containing all three group features gives the best result of 88% accuracy. Fig. 5 shows the scatter plots of several run-length features. The strong correlations among different features shown in the figure indicate that these features contain similar information.

Fig. 6 shows the scatter plots of the top eight features obtained by applying the MDEE transform on the run-length matrix. Almost perfectly separable clustering can be seen for most of the eight image classes, in sharp contrast to the overlapping clusters of the traditional run-length features in Fig. 5. This is reflected by the good classification results using the DRM features in Table II. With only a small number of features, perfect classification is achieved with the original matrix and with most of the new matrices and vectors. The only exceptions are the run-length run-number vector and the long-run region of the run-length matrix. The poor performance of the long-run matrix and the good performance of the short-run matrix indicate that most texture information is indeed concentrated in the short-run region.

C. Comparison with Other Methods

We now compare the new run-length method with the co-occurrence method and the wavelet method on the Vistex data set. For the co-occurrence method, 13 co-occurrence features are computed for each of the four directions as described in [8]; for the wavelet method, the texture feature used for each wavelet decomposition channel is the energy feature. The same feature selection method in Section II-C is applied to the co-occurrence and wavelet feature vectors.

The classification results on the sixteen Vistex images using various DRM features are first shown in Table III. About 97% classification accuracy is achieved by most feature vectors. An especially interesting result is that the run-length-one vector gives excellent performance, similar to that of the original full matrix.

Classification results using co-occurrence and wavelet features on the sixteen Vistex images are shown in Table IV. From the results, we can see that the run-length features perform comparably well with the co-occurrence features and better than the wavelet features. This demonstrates that there is rich texture information contained in the run-length matrices and that a good method of extracting such information is important to successful classification.

The poor results of the wavelet features are inconsistent with several previous studies [2], [10], where wavelet features generate near perfect classifications. This is mainly because that we use a much smaller texture sample size, 32 × 32, than the ones used in most previous studies, 64 × 64 or 128 × 128 [2], [10]. Such a small image size may not be enough to estimate a stable frequency energy distribution. To confirm this sample size effect, we divide each Vistex image class into 169 sample images of dimension 64 × 64 with 75% overlapping between neighborhood samples. Only 39 samples in each class are used as training data, so the training data size is 624 samples and the testing data size is 2080 samples. Table V shows the classification results. Near perfect classifications are achieved by all three methods, similar to results in [2] and [10].

IV. CONCLUSION

We extract a new set of run-length texture features that significantly improve image classification accuracy over traditional run-length features. By directly using part or all of the run-length matrix as a feature vector, much of the texture information is preserved. This approach is made possible by the utilization of the multilevel dominant eigenvector estimation method, which reduces the computation complexity of KLT by several orders of magnitude. Combined with the Bhattacharyya measure, they form an efficient feature selection algorithm.

The advantage of this approach is demonstrated experimentally by the classification of two independent texture data sets. Experimentally, we also observe that most texture information is stored in the first few columns of the run-length matrix, especially in the first column. This observation justifies development of a new, fast, parallel run-length matrix computation scheme.

Comparisons of this new approach with the co-occurrence and wavelet features demonstrate that the run-length matrices possess as much discriminatory information as these successful conventional texture features and that a good method of extracting such information is key to the success of the classification. We are currently investigating the application of the new feature extraction approach on other texture matrices. We hope our work here will renew interest in run-length texture features and promote future applications.

References

[1] P. Brodatz, *Textures: A Photographic Album for Artists and Designers.* New York: Dover, 1966.

[2] T. Chang and C.-C. J. Kuo, “Texture analysis and classification with tree-structured wavelet transform,” *IEEE Trans. Image Processing,* vol. 2, pp. 429–441, Oct. 1993.

[3] A. Chu, C. M. Sehgal, and J. F. Greenleaf, “Use of gray value distribution of run lengths for texture analysis,” *Pattern Recognit. Lett.,* vol. 11, pp. 415–420. June 1990.

[4] R. W. Conners and C. A. Harlow, “A theoretical comparison of texture algorithms,” *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 2, pp. 204–222, May 1980.

[5] B. R. Dasarathy and E. B. Holder, “Image characterizations based on joint gray-level run-length distributions,” *Pattern Recognit. Lett.,* vol. 12, pp. 497–502, 1991.

[6] K. Fukunaga, *Introduction to Statistical Pattern Recognition.* New York: Academic, 1972.

[7] M. M. Galloway, “Texture analysis using gray level run lengths,” *Comput. Graphics Image Process.,* vol. 4, pp. 172–179, June 1975.

[8] R. M. Haralick, K. S. Shanmugam, and I. Dinstein, “Textural features for image classification,” *IEEE Trans. Syst., Man, Cybern.,* vol. SMC-3, pp. 610–621, 1973.

[9] M. E. Jernigan and F. D’Astous, “Entropy-based texture analysis in the spatial frequency domain,” *IEEE Trans. Pattern Anal. Machine Intell.,* vol. PAMI-6, pp. 237–243, Mar. 1984.

[10] A. Laine and J. Fan, “Texture classification by wavelet packet signatures,” *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 15, pp. 1186–1191, Nov. 1993.

[11] S. Liu and M. E. Jernigan, “Texture analysis and discrimination in additive noise,” *Comput. Vis., Graph., Image Process.,* vol. 49, pp. 52–67, 1990.

[12] X. Tang, “Dominant run-length method for image classification,” Woods Hole Oceanog. Inst., Tech. Rep. WHOI-97–07, June 1997.

[13] C. W. Therrien, *Decision Estimation and Classification, An Introduction to Pattern Recognition and Related Topics.* New York: Wiley, 1989.

[14] M. Unser and M. Eden, “Multiresolution feature extraction and selection for texture segmentation,” *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 11, pp. 717–728, July 1989.

[15] J. S. Weszka, C. R. Dyer, and A. Rosenfeld, “A comparative study of texture measures for terrain classification,” *IEEE Trans. Syst., Man, Cybern.,* vol. SMC-6, pp. 269–285, 1976.

Real-Time Computation of Two-Dimensional Moments on Binary Images Using Image Block Representation

Iraklis M. Spiliotis and Basil G. Mertzios

***Abstract*—This work presents a new approach and an algorithm for binary image representation, which is applied for the fast and efficient computation of moments on binary images. This binary image representation scheme is called *image block representation*, since it represents the image as a set of nonoverlapping rectangular areas. The main purpose of the image block representation process is to provide an efficient binary image representation rather than the compression of the image. The block represented binary image is well suited for fast implementation of various processing and analysis algorithms in a digital computing machine. The two-dimensional (2-D) statistical moments of the image may be used for image processing and analysis applications. A number of powerful shape analysis methods based on statistical moments have been presented, but they suffer from the drawback of high computational cost. The real-time computation of moments in block represented images is achieved by exploiting the rectangular structure of the blocks.**

***Index Terms*—Binary image, image analysis, image block representation, moments.**

I. Introduction

The most common image representation format is a two-dimensional (2-D) array, each element of which has the brightness value of the corresponding pixel. For a binary image these values are zero or one. In a serial machine, only one pixel is to be processed at a time, by using the 2-D array representation. Many research efforts have considered the problem of selecting an image representation suitable for concurrent processing in a serial machine. The need for such approaches arises from the fact that an image contains a great amount of information, thus rendering the processing a difficult and slow task. Existing approaches to image representation aim to provide machine perception of images in pieces larger than a pixel and are separated in two categories: 1) boundary-based methods and 2) region-based methods. Such approaches include quadtree representations [1], chain code representations [2], contour control point models [3], autoregressive models [4], the interval coding representation [5], and block implementation techniques [6]–[8]. One common objective of the above methods is the representation of an image in a more suitable form for a specific operation.

This correspondence presents a new advantageous representation for binary images called *image block representation* (IBR) and constitutes an efficient tool for image processing and analysis techniques [9], [10]. Using the block represented binary images, real-time computation of 2-D statistical moments is achieved through analytical formulae. The computational complexity of the proposed technique is $O(L^2)$, where $(L-1, L-1)$ is the order of the 2-D moments to be computed.

Various sets of 2-D statistical moments constitute a well-known image analysis and pattern recognition tool [11]–[20]. In pattern recognition applications, a small set of the lower order moments is

Manuscript received June 4, 1994; revised September 18, 1995. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. John Juyang Weng.

The authors are with the Automatic Control Systems Laboratory, Department of Electrical and Computer Engineering, Democritus University of Thrace, 67100 Xanthi, Greece (e-mail: spiliot@demokritos.cc.duth.gr; mertzios@demokritos.cc.duth.gr).

Publisher Item Identifier S 1057-7149(98)07752-5.