

Texture Analysis Using Multidimensional Histogram

Payel Saha
TCET, Mumbai
Shyamnarayan Thakur Marg
Thakur Village, Kandivali(E)-101

Sudhir Sawarkar
D.M.C.E.
Airoli-708

ABSTRACT

Texture features have long been used in remote sensing applications for representing and retrieving regions similar to a query region. Various representations of texture have been proposed based on the power spectrum, grey-level co-occurrence matrices, wavelet features, Gabor features, etc. Analysis of several co-occurring pixel values may benefit texture description but is impeded by the exponential growth of histogram size. Multidimensional histograms can be reduced by using methods like linear compression, dimension optimization and vector quantization. Experiments with natural textures showed that multidimensional histograms provided higher classification accuracies than the channel histograms and the wavelet packet signatures.

Categories and Subject Descriptors

I.4 Image Processing and Computer Vision

General Terms

Algorithm

Keywords

Texture classification, multidimensional histograms, vector quantization, self-organizing map, feature selection

1. INTRODUCTION

Texture is an important characteristic for the analysis of many types of images. It can be seen in all images from multispectral scanner images obtained from aircraft or satellite platforms to microscopic images of cell cultures or tissue samples (which the biomedical community analyzes). Despite its importance and ubiquity in image data, a formal approach or precise definition of texture does not exist. The texture discrimination techniques are, for the most part, *ad hoc*. In this paper, some of the extraction techniques and models which investigators have been using to measure textural properties are reviewed and analyzed. Texture is considered as an organized area phenomenon. When it is decomposable, it has two basic dimensions. The first dimension is for describing the primitives out of which the image texture is composed, and the second dimension is for the description of the spatial dependence or interaction between the primitives of an image texture. The first dimension is concerned with tonal primitives or local properties, and the second dimension is concerned with the spatial organization of the tonal primitives.

Tonal primitives are regions with tonal properties. The tonal primitives can be described in terms such as the average tone, or maximum and minimum tone of its region. The region is a maximally connected set of pixels having a given tonal property. The tonal region can be evaluated in terms of its area and shape. The tonal primitive includes both its gray tone and tonal region properties.

An image texture is described by the number and types of its primitives and the spatial organization or layout of its primitives. The spatial organization may be random, may have a pair wise dependence of one primitive on a neighboring primitive, or may have a dependence of n primitives at a time. The dependence may be structural, probabilistic, or functional (like a linear dependence). Image texture can be qualitatively evaluated as having one or more of the properties of fineness, coarseness, smoothness, granulation, randomness, lineation, or being mottled, irregular, or hummocky. Each of these adjectives translates into some property of the tonal primitives and the spatial interaction between the tonal primitives. Unfortunately, few experiments have been done attempting to map semantic meaning into precise properties of tonal primitives and their spatial distributional properties. To objectively use the tone and textural pattern elements, the concepts of tonal and textural feature must be explicitly defined. With an explicit definition, tone and texture are not independent concepts. They bear an inextricable relationship to one another very much like the relation between a particle and a wave. There really is nothing that is solely particle or solely wave. Whatever exists has both particle and wave properties and depending on the situation, the particle or wave properties may predominate. Similarly, in the image context, tone and texture are always there, although at times one property can dominate the other and we tend to speak of only tone or only texture. Hence, when defining tone and texture, instead of using two concepts a single one tone-texture concept is used. The basic interrelationships in the tone-texture concept are the following when a small-area patch of an image has little variation of tonal primitives; the dominant property of that area is tone. When a small-area patch has wide variation of tonal primitives, the dominant property of that area is texture. Crucial in this distinction are the size of the small-area patch, the relative sizes and types of tonal primitives, and the number and placement or arrangement of the distinguishable primitives. As the number of distinguishable tonal primitives decreases, the tonal properties will predominate. In fact, when the small-area patch is only the size of one resolution cell, so that there is only one discrete feature, the only property present is simple gray tone. As the number of distinguishable tonal primitives increases within the small-area patch, the texture property will dominate. When the spatial pattern in the tonal primitives is random and the gray tone variation between primitives is wide, a fine texture results. As the spatial pattern becomes more definite and the tonal regions involve more and more resolution cells, a coarser texture results. In summary, to characterize texture, the tonal primitive properties as well as the spatial interrelationships between them can be used. This implies that texture-tone is really a two-layered structure, the first layer having to do with specifying the local properties which manifest themselves in tonal primitives and the second layer having to do with specifying the organization among the tonal primitives.

2. REDUCED MULTIDIMENSIONAL HISTOGRAMS

Monochrome textures are conventionally described by one dimensional difference histograms or two-dimensional co-occurrence histograms of gray levels. Histograms may be used as such for texture description [2] but, typically, the description is based on various statistics computed from the histogram [3]. Histograms with more than two dimensions have only occasionally been applied to monochrome texture description [4], [5]. Kimmo Valkealahti and Erkki Oja [1] suggests that an increase in the co-occurrence dimensionality, which improves the description of spatial relationships, benefits both monochrome and color texture classification [6]. In the present study, ways are explored to speed up the collection of multidimensional histograms. The method of [1] is compared with the channel histogram method of Unser [7] which is a well-justified improvement of the co-occurrence matrix method, and with the wavelet packet signature method of Laine and Fan [8] exemplifying recent developments in texture analysis without histogram basis. Moreover, a comparison is made with multidimensional channel histograms which are somewhat related to the method of He and Wang [4]. All comparisons are performed using Brodatz textures, and to increase the generalizability of the results, the textures are also rotated and scaled.

2.1 Texture Preprocessing

The 32 Brodatz textures [14] used in the study (Fig. 1) were equalized to 256 X 256 pixels and 256 gray levels. The images were selected according to visual judgement so that 64 X 64 subimages captured the essential substructures. Otherwise, the selection was independent of the classifiers used. Each image was divided into 16 disjoint 64 X 64 blocks, and each block was independently histogram-equalized to abolish luminance differences among textures. Each texture block was transformed into three additional blocks: a block rotated by 90 degrees, a 64 X 64 scaled block obtained from 45 X 45 pixels in the middle, and a block which was both rotated and scaled. The 32 texture categories included 2,048 blocks altogether. Construction and testing of the classifiers were carried out with disjoint sets of blocks: Eight blocks in each texture image, together with the corresponding 24 transformed blocks, were randomly selected into a design set and the other 8 + 24 blocks were used for the evaluation of classifier performance. The classifier performance was evaluated statistically with 10 different randomly selected design and test sets.

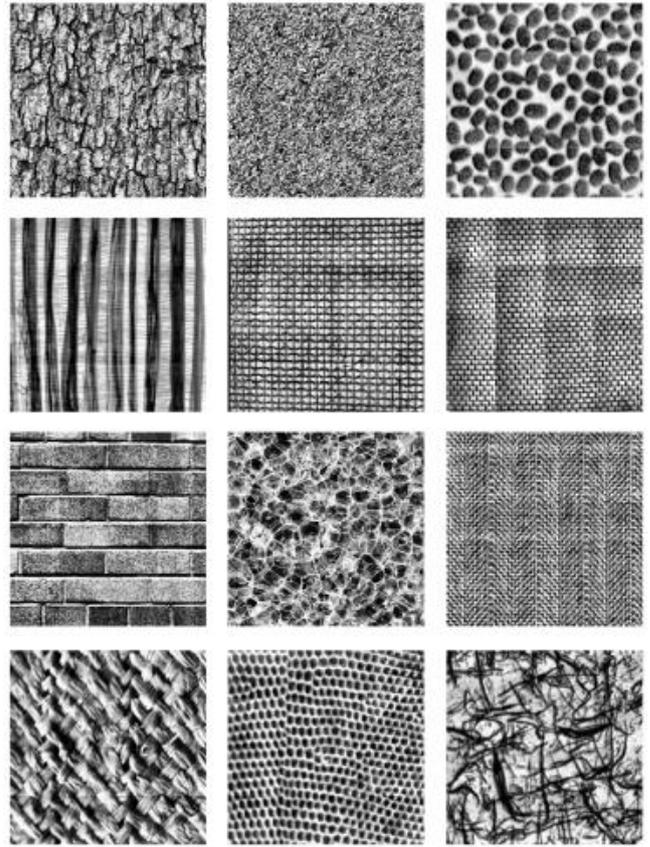


Fig 1. Twelve of the thirty-two histogram-equalized Brodatz textures used in the study.

2.2 Co-Occurrence Vectors

Two kinds of co-occurrences were analyzed: mean-subtracted gray levels and variance-equalized discrete cosine transform coefficients. Let matrix \mathbf{G} represent the gray levels in a 4 X 4 neighborhood at some location in a texture block. A 16-dimensional gray level vector \mathbf{g} is formed by collecting entries $G_{i,j}$ of matrix \mathbf{G} into vector $\mathbf{g} = (g_1, g_2, \dots, g_{16}) = (G_{1,1}, G_{1,2}, \dots, G_{4,4})$. The components of the mean-subtracted gray-level co-occurrence vector $\mathbf{s} = (s_1, s_2, \dots, s_{16})$ are defined as

$$s_i = b_i \left(g_i - \frac{\sum_{j=1}^{16} b_j g_j}{\sum_{j=1}^{16} b_j} \right)$$

in which binary coefficients b_i determine the active components of \mathbf{s} . The co-occurring gray levels are selected from adjacent positions because then their mutual dependence is strongest. During feature selection as mentioned in section 2.6, the coefficients are adjusted to find the subset of vector components which minimize the classification error rate. Setting one or more coefficients b_i to zero annuls the corresponding elements in the co-occurrence vectors. Therefore, the effective dimensionality of the co-occurrence vectors is

$$K = \sum_{i=1}^{16} b_i$$

As an alternative representation, the discrete cosine transform (DCT) was used to decorrelate the gray-level co-occurrences [7]. Let matrix \mathbf{C} be the DCT matrix with un-normalized column vectors,

$$C = \begin{bmatrix} 1 & \cos \frac{\pi}{8} & \cos \frac{2\pi}{8} & \cos \frac{3\pi}{8} \\ 1 & \cos \frac{3\pi}{8} & \cos \frac{6\pi}{8} & \cos \frac{9\pi}{8} \\ 1 & \cos \frac{5\pi}{8} & \cos \frac{10\pi}{8} & \cos \frac{15\pi}{8} \\ 1 & \cos \frac{7\pi}{8} & \cos \frac{14\pi}{8} & \cos \frac{21\pi}{8} \end{bmatrix}$$

The DCT-coefficient vector is collected from entries $F_{i,j}$ of matrix

$F = C^T G C$ into vector $f = (f_1, f_2, \dots, f_{16}) = (F_{1,1}, F_{1,2}, \dots, F_{4,4})$. Vector f contains the same information as vector g , but the components of f are almost uncorrelated. The components of the variance equalized DCT co-occurrence vector s are defined by

$$s_i = b_i \frac{f_i}{\sqrt{\text{Var}\{f_i\}}}$$

in which the denominator is the estimated standard deviation of the numerator. The equalization of component variances means that the DCT coefficients with small variance are assumed to possess similar discriminatory power as the coefficients with large variance.

2.3 Histogram Classification

Using the vector quantizer, sample histograms were computed for each texture block separately. The histograms were represented as sequences $\{h_{ijq}\} = \{h_{ij1}, \dots, h_{ijN}\}$, in which $i = 1, \dots, 32$ indexes the different texture categories, $j = 1, \dots, 32$ indexes the 64 X 64 blocks in the design set of the i th texture, and N is the histogram size equaling the number of code words. A texture model histogram

$$\{m_q\} = \left\{ \sum_{j=1}^{32} h_{ijq} \right\}$$

was obtained by summing up the histograms computed from each block of the modeled texture. A new histogram $\{h_q\}$ was classified according to the model histogram that maximized the log-likelihood function

$$L(\{h_q\}, \{m_q\}) = \sum_{q=1}^N h_q \ln \frac{m_{iq}}{M_i}$$

2.4 Quadtree Vector Quantizer

In a quadtree vector quantizer, each non-terminal node branches to four nodes at each succeeding level. In the present case, the tree structured codebook had six levels $l = 1, 2, \dots, 6$, each with 4^l vectors. Each codebook with $N = 4,096$ code words was trained with 41,000 vectors s which were randomly sampled from a design set of all 32 textures. At the beginning of training, the first level is initialized with four distinct vectors close to the mean of all training vectors. The code vectors are trained by repeating the Lloyd iteration [12] until the decrease of quantization error in succeeding iterations is less than 5 percent. During Lloyd iteration, each code vector is replaced with the mean of the sample vectors closer to it than to any other code vector. After one level has been trained, vector values are fixed and each value is copied to the descendant code vectors, which are then made unequal by changing their values slightly. The search for the best-matching code vector always starts from the first level and proceeds to the subtree descending from the code vector which is closest to the sample vector. Decisions among branches are made until the level under training is reached. During vector quantization with a trained quadtree, only $6 \times 4 = 24$ code vectors, instead of 4,096, are matched to a sample.

2.5 Tree-Structured Self-Organizing Map

The quadtree-structured self-organized map [11], [6] consisted of six levels in which the nodes were arranged in squares. The levels are trained using repeated Lloyd iterations as the quadtree vector quantizers. After training of one level, each code vector value is copied to the four descendant code vectors and the succeeding level is initialized by giving each code vector the mean value of its four-connected neighbor nodes. The search for the best-matching code vector always starts from the first level, and when proceeding to the succeeding levels it is limited to the descendant nodes of the current best-matching code vector and its four-connected neighbors. The similarity of neighboring code vectors emerges during the training from the initialization of tree levels and limited lateral searches. In a trained map, about 90 code vectors are matched to a sample during the quantization.

2.6 Selection of Co-Occurrence Components

The classifier was adjusted to the current task by selecting the co-occurrence vector components which minimized the classification error rate of a design set. The error rate was determined with the leave-one-out method: When a sample histogram for a texture block was compared with the model of the same texture, shares of the histogram itself and the histograms collected from all other rotated and scaled variants of the same texture block were subtracted from the model histogram (changing M_i accordingly) prior to the likelihood computation of [5]. A genetic algorithm was used to find the values of coefficients b_i in [2] and [4]. Vector $b = (b_1, b_2, \dots, b_{16})$ which minimized the error rate determined the component selection. The genetic algorithm was implemented according to the instructions given by Davis [15]. The optimization was carried out using a population of 50 vectors with random initial values. A new unique vector was reproduced from two parent vectors which were selected with the roulette wheel method. Each component was randomly assigned the value of either of the parents. The value was inverted with probability 0.04. If the classification error rate of the reproduced vector was lower than that of the worst population member then a replacement was made. The number of evaluated vectors was 500.

3 CHANNEL HISTOGRAMS

Channel histograms estimate one-dimensional marginal densities of a decorrelated feature distribution [7]. Thus, a K dimensional co-occurrence distribution is represented by K one dimensional histograms. The components of co-occurrence vectors s in (4) are approximately decorrelated by the discrete cosine transform. Therefore, the i th channel histogram is collected from the values of component s_i . The component values were scalar-quantized to $N_1 = 2, 4, 8, \dots, 256$ levels so that each bin of the i th histogram had about the same share of the distribution of s_i . This means that the channel histograms over the whole texture data are flat. Each

$$K = \sum_{i=1}^{16} b_i$$

texture is modeled with a combination of channel histograms. Multidimensional channel histograms were also used as texture models. The component values were scalar-quantized to a small number of levels, $N_2 = 2, 3, \dots, 8$, and were collected into K dimensional histograms. The optimization of channel histogram classifiers included selection of the number of quantization levels N_1 or N_2 . For the selection, component b_{17} is added to vector b . Components b_1, b_2, \dots, b_{17} were constrained so that histogram size $N = KN_1$ or $N = KN_2$ ensuing from their values did not exceed 4,096 bins. During the selection, component b_{17} was given a random value within its range with probability

0.04, and it was increased or decreased by one step with probability 0.3. The number of evaluated vectors was 2,000.

4 RESULTS AND DISCUSSION

4.1 Reduced Multidimensional Histograms

The reduced multidimensional histograms provided significantly higher classification accuracies than the channel histograms or wavelet packets. The results of different methods are shown in Table 1, where they are in descending order of classification accuracy; each accuracy is an average of ten experiments with different design and test sets. Each P value in the table shows the significance of the difference between two consecutive test set classification accuracies (analysis of variance with Tukey test). As shown in Table 1, the two vector quantizers provided similar classification accuracies. There was a significant difference ($p < 0.002$; paired sign test) between the quantization error per dimension of the tree-structured self-organizing map (median 0.362) and the quadtree vector quantizer (median 0.418). This shows that the quantization accuracy was not critical for the classification performance. Table 1 also shows that the DCT-coefficient and gray-level vectors performed equally well. During the optimization, the reduction of the mean-removed gray-level components was modest: The median number of selected components was 13.5 (lower quartile 12, upper 15). The high number of selected components demonstrates that our method can represent quite high dimensional co-occurrences. The reduction of DCT coefficients was more pronounced: The median number of selected components was 8 [8, 9] with both vector quantizers. The DCT coefficients with the highest variance preceding the normalization in [4] were most frequently selected and those with the lowest variance were most frequently rejected. The optimization of DCT coefficients thus resembled principal-component-type feature reduction:

The “principal” DCT coefficients captured most of the discriminatory information, whereas the “minor” DCT coefficients

could be regarded as noise. As a separable two-dimensional transform, the DCT is faster to compute than the principal component transform. Suppose that it takes 100 time units to compute a multidimensional histogram of the optimized mean-removed gray-level vectors with the tree-structured self-organizing map. The time was decreased to 72 units by the use of the optimized DCT coefficients. Replacing the tree-structured self-organizing map with the quadtree vector quantizer decreased the time further to 32 units.

Table 1: Average classification results in 10 experiments

Method	Accuracy % (SD)		P value
	Design set	Test set	
Reduced multidimensional histograms			
of DCT coefficients with TSOM	93.3 (0.8)	93.9 (0.7)	-
of DCT coefficients with QVQ	93.3 (0.8)	93.4 (0.8)	n.s.
of mean-removed gray levels with TSOM	92.5 (0.8)	92.8 (0.9)	n.s.
Multidimensional channel histograms	89.9 (0.8)	90.4 (0.8)	$p < 0.001$
Wavelet packets	86.4 (0.9)	85.1 (1.1)	$p < 0.001$
One-dimensional channel histograms	78.8 (0.8)	78.2 (1.6)	$p < 0.001$

Where, *SD*: sample standard deviation.

DCT: discrete cosine transform.

TSOM: tree-structured self-organizing map.

QVQ: quadtree vector quantizer.

4.2 Minimization of Leave-One-Out Error

Table 1 shows no significant differences between the design and test set accuracies for any method. Thus, the minimization of the leave-one-out error during the optimization did not result in significant over fitting of the texture models. Owing to the leave-one out method, a lower number of texture blocks were used to compute the models of the design than the test sets. This may have

somewhat favored the test set accuracies. The suitability of genetic algorithms for complex tasks, such as the optimization of classifiers, is well documented [17] but it is not excluded that simpler search methods [18] might perform equally well in the present case.

5 CONCLUSION

In comparison with the other methods, the reduced multidimensional histograms provided the highest classification accuracies. In our previous study [6], the tree-structured self-organizing map was chosen instead of the traditional tree-search vector quantizer because it was suggested to provide lower quantization error. The present results verified this suggestion, but they also showed that the significant difference in the quantization

error was not reflected in the classification performance. In comparison to the previous study, the use of traditional tree-search vector quantizer and optimized linear compression significantly speeded up the classification.

6 REFERENCES

- [1] “Reduced Multidimensional Co-Occurrence Histograms in Texture Classification”, Kimmo Valkealahti and Erkki Oja, *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 1, January 1998.
- [2] M. Unser, “Sum and Difference Histograms for Texture Classification,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 118–125, Jan. 1986.
- [3] R.M. Haralick, “Statistical and Structural Approaches to Texture,” *Proc. IEEE*, vol. 67, no. 5, pp. 786–804, May 1979.
- [4] D.-C. He and L. Wang, “Unsupervised Textural Classification of Images Using the Texture Spectrum,” *Pattern Recognition*, vol. 25, no. 3, pp. 247–255, 1992.
- [5] V. Kovalev and M. Petrou, “Multidimensional Co-Occurrence Matrices for Object Recognition and Matching,” *Graphical Models and Image Processing*, vol. 58, no. 3, pp. 187–197, May 1996.
- [6] E. Oja and K. Valkealahti, “Co-Occurrence Map: Quantizing Multidimensional Texture Histograms,” *Pattern Recognition Letters*, vol. 17, no. 7, pp. 723–730, June 1996.
- [7] M. Unser, “Local Linear Transforms for Texture Measurements,” *Signal Processing*, vol. 11, no. 1, pp. 61–79, July 1986.
- [8] A. Laine and J. Fan, “Texture Classification by Wavelet Packet Signatures,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1,186–1,191, Nov. 1993.
- [9] G.F. McLean, “Vector Quantization for Texture Classification,” *IEEE Trans. Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 637–649, May/June 1993.
- [10] T. Kohonen, *Self-Organizing Maps*. Berlin: Springer-Verlag, 1995.

- [11] P. Koikkalainen, "Fast Deterministic Self-Organizing Maps," *Proc. Int'l Conf. Artificial Neural Networks*, vol. 2, pp. 63–68, Paris, 9–13 Oct. 1995.
- [12] R.M. Gray, "Vector Quantization," *IEEE ASSP Magazine*, vol. 1, no. 2, pp. 4–29, Apr. 1984.
- [13] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*. Boston: Kluwer Academic Publishers, 1992.
- [14] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. New York: Dover Publications, 1966.
- [15] L. Davis, ed., *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.
- [16] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Englewood Cliffs, N.J.: Prentice Hall, 1995.
- [17] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, Mass.: Addison-Wesley Publishing Company, 1989.
- [18] P.A. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*. London: Prentice Hall International, 1982.
- [19] Robert m. Hawlick, "Statistical and Structural Approaches to Texture" *IEEE Proceedings of* Vol. 67, No.5, May 1979