Journal of Visual Communication and Image Representation **11**, 327–342 (2000)
doi:10.1006/jvci.2000.0448, available online at http://www.idealibrary.com on IDEAL®

Texture Recognition and Image Retrieval Using Gradient Indexing

Bo Tao

Sony US Research SJ2C5, 3300 Zanker Road, San Jose, California 95134
E-mail: Bo_Tao@mail.sel.sony.com

and

Bradley W. Dickinson

Electrical Engineering Department, Princeton University, Princeton, New Jersey 08544
E-mail: bradley@ee.princeton.edu

Received April 21, 1999; accepted September 7, 1999

Our starting point is gradient indexing, the characterization of texture by a feature vector that comprises a histogram derived from the image gradient field. We investigate the use of gradient indexing for texture recognition and image retrieval. We find that gradient indexing is a robust measure with respect to the number of bins and to the choice of the gradient operator. We also find that the gradient direction and magnitude are equally effective in recognizing different textures. Furthermore, a variant of gradient indexing called *local activity spectrum* is proposed and shown to have improved performance. Local activity spectrum is employed in an image retrieval system as the texture statistic. The retrieval system is based on a segmentation technique employing a distance measure called Sum of Minimum Distance. This system enables content-based retrieval of database images from templates of arbitrary size.

1. INTRODUCTION

Texture, which refers to a characterization of the surface of an object or a collection of objects, has traditionally played an important role in the analysis of aerial, satellite, and medical images, among other image types [1]. In recent years, it has found applications in content-based image retrieval systems as well [2, 3].

Texture recognition has been studied for many years; some of the typical methods include, but are not limited to, parametric statistical model-based techniques [4, 5], structural techniques [6, 7], techniques based on empirical second-order statistics [8, 9] and various transform-based techniques [10–12]. Another major class of approaches for texture classification is based on the analysis of local properties of a texture sample by using either linear

1047-3203/00 $35.00

or nonlinear local operators [13–16]. We adopt the view that neighborhood properties are the dominant reasons for the appearance of a texture, and so the empirical distribution of a local operator describes a texture primitive. Hence, we expect a texture recognition scheme based on local properties to give good results.

Many local neighborhood properties have been studied for texture representation purposes. The first kind is based on linear filtering [14, 16]. The filters are often designed to detect various spatial structures, such as "spots" and "edges." Another class is based on two-dimensional local transforms. Statistical properties of each transform coefficient are used to classify a texture [13]. Nonlinear methods have also been introduced to characterize neighborhood properties; one such approach was proposed in [15] by comparing one pixel with its neighbors. Local operator-based approaches often enjoy the advantage of fast computation, and thus are suitable for real-time applications.

The purpose of this paper is to study the use of gradient operators for texture analysis. Three gradient operators, namely Sobel, Prewitt, and Roberts [17], will be compared. A gradient histogram is used as the basis for determining a distance between different textures. It is found that such a gradient histogram method, which we call *gradient indexing*, is insensitive to the number of bins and to the choice of the gradient operator. Furthermore, it is found that the gradient direction is as effective a measure as the gradient magnitude. By treating variations along horizontal, vertical, diagonal, and anti-diagonal directions separately, we introduce a modified version of gradient indexing, called *local activity spectrum* (LAS), which has an improved performance over histograms of the three operators above. The local activity spectrum technique only requires addition, absolute value, and counting operations, and thus is highly computationally efficient.

Besides its conventional applications in such areas as terrain type recognition, medical image analysis, and industrial inspection, texture is among the most frequently used features in content-based image retrieval systems. We will apply the local activity spectrum to an image retrieval system. Our image retrieval system permits users to search a database using a target image (called a *template*) of arbitrary size to retrieve images containing similar texture patterns. The system is based on image segmentation and a distance measure called *sum of minimum distance* (SMD).

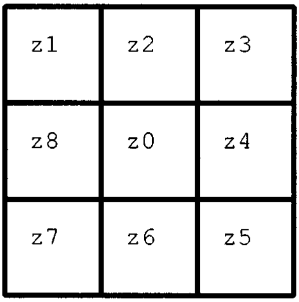
In what follows, the gradient indexing technique and local activity spectrum will be introduced in Section 2. The retrieval system is described in Section 3. Section 4 presents experimental results on both texture classification and texture image retrieval. The paper concludes with discussions in Section 5.

2. GRADIENT INDEXING

It has long been recognized that edges can be effective features for describing a texture. In [18], an edgeness measure is defined as a function of distance, r,

$$M(r) = \sum_{i,j} |f(i,j) - f(i+r,j)| + |f(i,j) - f(i-r,j)| \\ + |f(i,j) - f(i,j+r)| + |f(i,j) - f(i,j-r)|,$$

where $f(i,j)$ is the image function value at pixel (i,j). Based on the values of $M(r)$ as r varies, the authors were able to detect the occurrence of certain diseases. In another



z1	z2	z3
z8	z0	z4
z7	z6	z5

FIG. 1. A 3 × 3 window used for computing the gradient at the center pixel z0.

paper where texture edgeness is used for segmentation [19], an edge detection algorithm is proposed and applied to texture segmentation; the Roberts gradient is employed to transform the gray-level image to a gradient field on which edge detection is carried out. In [20], a joint representation of color, texture, and shape is proposed in the spatio-chromatic domain. Laplacian of Gaussian filtering is first applied to the image; then the features are computed at zero-crossing positions. In work on adaptive image processing, it has been concluded that the best indicator functions for local structure are those "based on some form of gradient estimation" [21].

We view a gradient operator simply as a local operator suitable for measuring the local spatial variation or "activity." Thus the distribution of the gradient, which can be characterized using a histogram, can be used for texture recognition. We call such a gradient histogram approach *gradient indexing*. (Histogram-based approaches have been proven to be robust in another field of image analysis, color-based object recognition [22].) The properties of gradient indexing include translation invariance, which is highly desirable in real-world applications.

The first-order derivative (gradient) of an image function $f(x, y)$ is defined as $\nabla f = [g_x \; g_y]^{\mathrm{T}} = [\frac{\partial f}{\partial x} \; \frac{\partial f}{\partial y}]^{\mathrm{T}}$ where the superscript T denotes transposition. The magnitude of a gradient, defined as $\mathrm{mag}(\nabla f) = |\frac{\partial f}{\partial x}| + |\frac{\partial f}{\partial y}|$, will be used throughout this work for computational efficiency. The direction of a gradient is defined as $\theta(\nabla f) = \arctan(g_x/g_y)$. When applied to a digital image, the gradient operation can be implemented in several ways. The three popular gradient operators used in our work are Sobel, Prewitt, and Roberts[1] operators; their expressions are given as follows, respectively (see Fig. 1):

$$\textit{Sobel}: \begin{cases} g_x = f(z7) + 2f(z6) + f(z5) - f(z1) - 2f(z2) - f(z3) \\ g_y = f(z3) + 2f(z4) + f(z5) - f(z1) - 2f(z8) - f(z7) \end{cases} \quad (1)$$

$$\textit{Prewitt}: \begin{cases} g_x = f(z7) + f(z6) + f(z5) - f(z1) - f(z2) - f(z3) \\ g_y = f(z3) + f(z4) + f(z5) - f(z1) - f(z8) - f(z7) \end{cases} \quad (2)$$

$$\textit{Roberts}: \begin{cases} g_x = f(z0) - f(z5) \\ g_y = f(z4) - f(z6). \end{cases} \quad (3)$$

[1] Note that the Roberts operator computes the derivatives along the diagonal and anti-diagonal directions, which are mathematically equivalent to $[g_x \; g_y]^{\mathrm{T}}$. In fact, such an operator is often called the *Roberts cross-gradient operator*.

The gradient histogram[2] is computed by counting the number of pixels whose gradient values fall into various bins. We assume each bin has the same volume and is nonoverlapping with other bins. Note that the choice of uniform bins is not optimal in that gradients are not uniformly distributed in most situations. Neither is the choice of nonoverlapping bins ideal for classification, since two gradients close to each other can represent similar activity levels, but they may fall into two different bins due to lying on different sides of a threshold. A bell-shaped overlapping bin selection, or equivalently, a fuzzy membership function, can help alleviate this problem. However, we will not address such problems in this work; nonoverlapping uniform bins are used to show the capabilities of the gradient indexing technique.

For measuring the distance between histograms, several distance measures can be employed, including the Kullback divergence, the intersection measure [22], and the city_block distance. The city_block distance dominates in both recognition accuracy and computational efficiency in our experiments when compared with the Kullback divergence measure. It also achieves higher accuracy than the intersection measure. Thus the city-block distance will be used here. Let $\{P(i)\}_{i=1}^{N}$ and $\{Q(i)\}_{i=1}^{N}$ represent the gradient histograms of two images, where N is the number of bins. The city_block metric is defined as

$$d(P, Q) = \sum_{i=1}^{N} |P(i) - Q(i)|. \tag{4}$$

Mathematically, the gradient measures the rate of change at each pixel, that is, the local activity, but when the gradient operation is applied to a discrete field such as a digital image, and approximated with a gradient operator, the output of the operator only reflects part of the local activity. In order to capture the spatial activity of the texture along horizontal, vertical, diagonal, and anti-diagonal directions separately, we introduce a modified gradient indexing technique called local activity spectrum. It is defined in the following way. Let

$$\begin{aligned}
g_1 &= |f(i-1, j-1) - f(i, j)| + |f(i, j) - f(i+1, j+1)| \\
g_2 &= |f(i-1, j) - f(i, j)| + |f(i, j) - f(i+1, j)| \\
g_3 &= |f(i-1, j+1) - f(i, j)| + |f(i, j) - f(i+1, j-1)| \\
g_4 &= |f(i, j+1) - f(i, j)| + |f(i, j) - f(i, j-1)|
\end{aligned}$$

be the four activity measures at pixel (i, j). The distribution of $[g_1\ g_2\ g_3\ g_4]^{\mathrm{T}}$ can be represented using a histogram, which is called local activity spectrum. By using such a measure, we can effectively differentiate textures according to their activities along the four directions. If we think of the histogram formation as quantization, then in our work, a product-form quantization scheme is used; i.e., each g_i is quantized independently. This is certainly not optimal, but we choose such a histogram-based method for its simplicity. The city-block metric is again used as the distance measure for local activity spectrum.

Shown in Fig. 2 are two Brodatz textures [23]: herringbone weave (D17) and wood grain (D68). Their gradient magnitude histograms computed using the Sobel operator are shown in Fig. 3. The bins at the high end are mostly empty and hence only a part (25 out of 64)

[2] In this paper, a gradient histogram refers to the histogram of the gradient, its magnitude or its direction. The actual meaning should be clear from the context.

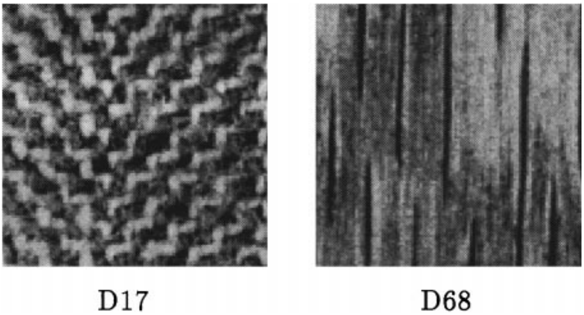


FIG. 2. Two Brodatz texture samples used for experiments.

of the histograms are shown. The local activity spectrums of the same two textures are given in Fig. 4. A total of 256 bins are used by quantizing each g_i with 4 bins. To show this four-variable histogram, we have changed it to a single-variable histogram by ordering the quantized values into a four-tuple $Q(g_1)Q(g_2)Q(g_3)Q(g_4)$ and interpreting it as a four-digit hex number. The first 100 bins are shown in Fig. 4. The local activity spectrum shows multiple modes and hence increases the discrimination capability.

In Section 4 we will compare gradient indexing with two other texture measures on texture classification and present results on using local activity spectrum for image retrieval purposes too, but first, we will describe the image retrieval system in the next section.

3. IMAGE RETRIEVAL WITH TEMPLATES OF ARBITRARY SIZE

Content-based image retrieval is becoming a major tool for modern digital image database applications. By indexing through image content instead of using keywords, semi-automatic or even automatic indexing can be realized saving a huge amount of labor.

Many retrieval methods rely on global features computed over the whole image [2, 24]. The assumption behind the use of global features is that images judged to be similar

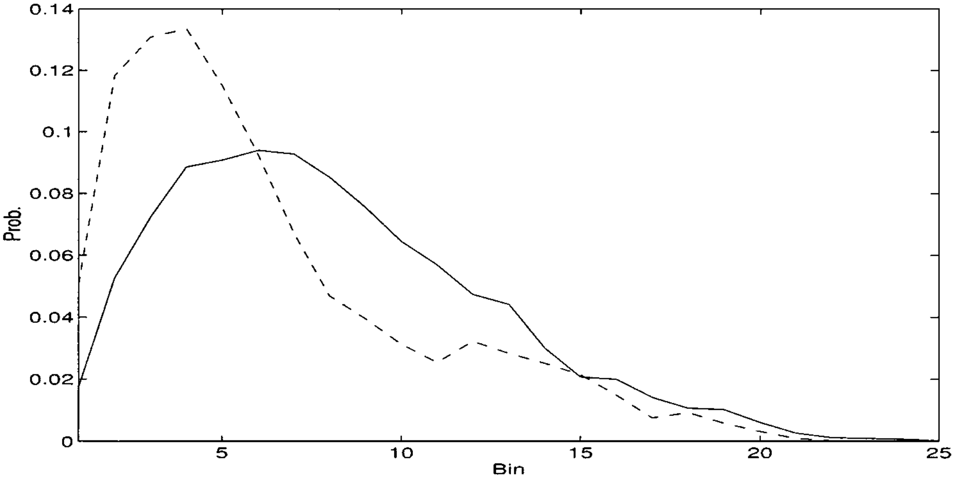


FIG. 3. Gradient magnitude histograms of two Brodatz textures: herringbone weave (solid line) and wood grain (dashed line).

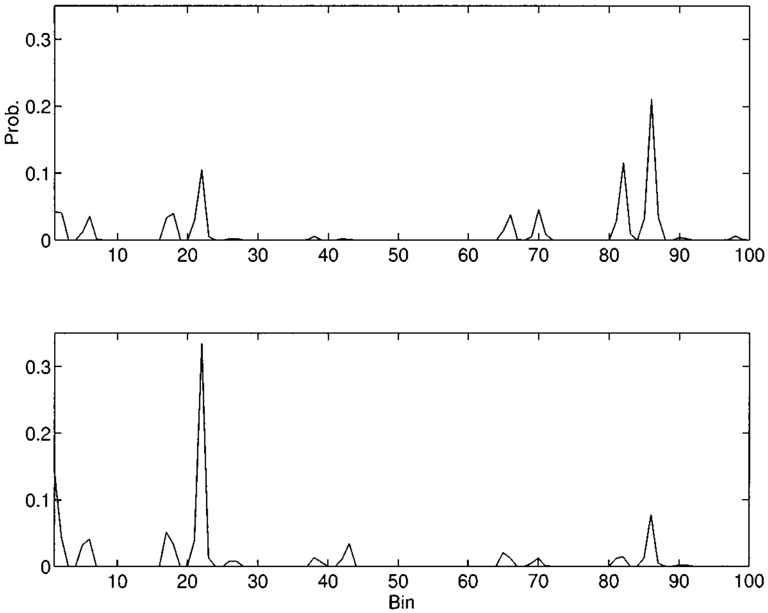


FIG. 4. Local activity spectrums of two Brodatz textures: herringbone weave (top) and wood grain (bottom).

by human viewers are close to each other in some global feature space. This is not always the case. For example, two very different images can have similar color histograms. Moreover, when one is interested in retrieving images containing a particular pattern, e.g., a political figure or a terrain type, the global feature-based approaches may be almost powerless. However, if local features, e.g., the shape of each object in the image, are employed in the retrieval process, the confidence in retrieval results can be strongly boosted.

The goal of our retrieval system is to allow a user to search the database for images containing patterns similar to the template given by the user as the query sample. The template can be of arbitrary size. For example, it can be a small region cropped from a large image. The user can retrieve all images containing portions similar to this small sample region. Template matching-based techniques have been studied for such applications [25]. Typically these require a huge amount of computation that is often considered too expensive. As an alternative, we will propose an approach based on image segmentation.

3.1. *Image Segmentation Using Local Activity Spectrum*

The segmentation technique is summarized as follows. We first compute the local activity measure $[g_1(s)\ g_2(s)\ g_3(s)\ g_4(s)]^T$ for each pixel s. Then the histogram bin index is obtained for each pixel using the product-form quantizer as discussed in the previous section. Next, suppose that the number of bins used is B. The image is divided into square blocks such that each block contains S pixels, where S should be large enough to generate accurate distribution estimation. In our implementation, B is chosen to be 256 (4 bins for each direction) and S is 16 times as large; in other words, the size of each block is 64×64. The block size should be selected so that it is at least as large as the size of the primitive element of each texture type. It appears that 64×64 is sufficient based on our experiments with Brodatz textures. For textures at different scales, a different block size may have to be used. Next, the local activity spectrum is computed for each block. Note that a fairly large S is required to ensure a stable LAS measure.

In the next step of segmentation, k-means clustering [26] is performed using the local activity spectrum of all blocks. This is an iterative procedure. Basically, during each iteration, the center of each class is calculated and then each feature vector is assigned a class lable using the newly computed centers. This process produces ragged segments with each being composed of a number of square blocks. Then an iterative process is used to refine the segment boundaries as discussed in the following.

For each boundary pixel s, suppose there are $n(s)$ segments bordering this pixel, with local activity spectrum P_i, $i = 1, \dots, n(s)$. Pixel s is assigned to one of the bordering segments so as to maximize the measure

$$\sum_{i=1}^{n(s)} \sum_{j \neq i} \| P_i - P_j \|_1,$$

where $\| P_i - P_j \|_1$ is the city_block distance between P_i and P_j. There are only at most 8 segments bordering one pixel, and for most boundary pixels, the number of bordering segments is 2. Furthermore, for the above computation, only the bin having the same index as the local activity index of the pixel is actually involved. Therefore, the computation is very efficient. This refining process is repeated until the segmentation result is stabilized or a maximum number of steps has been reached.

3.2. Distance Measure between Two Segmented Images

During the database population process, each one of a total of N images is segmented using the algorithm introduced previously. For image I_k, $k = 1, \dots, N$, suppose the segmentation is $\{\Omega_1^k, \dots, \Omega_{n_k}^k\}$ where n_k is the total number of segments inside I_k. For each segment Ω_j^k, the local activity spectrum P_j^k is computed and all the n_k feature vectors $\{P_1^k, \dots, P_{n_k}^k\}$ will be stored along with image I_k. Additionally, the neighborhod structure of each segment will be extracted. For segment Ω_j^k, we let y_j^k represent such a structure, i.e., $y_j^k = [b_{j,1}^k \dots b_{j,n_k}^k]$, where

$$b_{j,i}^k = \begin{cases} 1 & \Omega_i^k \text{ is a neighbor of } \Omega_j^k \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

and $b_{j,j}^k \equiv 1$. Then $\{y_1^k, \dots, y_{n_k}^k\}$ will also be stored with image I_k; alternatively, one can think of $\{y_1^k \dots y_{n_k}^k\}$ as the symmetric adjacency matrix of the segments of image I_k.

During the retrieval process, when a query request is generated with a template (target image) T, the search engine will first segment the template using the same segmentation algorithm as used in the construction of the database. Suppose a total of m segments, $\{\omega_1, \dots, \omega_m\}$, are found in the template. For each segment ω_j, the local activity spectrum Q_j is calculated. The spatial relationship is represented in the same way as for images in the database. Let $\{s_1, \dots, s_m\}$ represent such a relationship. Then we define the distance between the template and image I_k in the database as

$$\text{dist}(T, I_k) = D_1(f_T, f_{I_k}) + D_2(g_T, g_{I_k}), \tag{6}$$

where $f_T = \{Q_1, \dots, Q_m\}$, $g_T = \{s_1, \dots, s_m\}$, $f_{I_k} = \{P_1^k, \dots, P_{n_k}^k\}$, and $g_{I_k} = \{y_1^k, \dots, y_{n_k}^k\}$.

Then all the N distances $\{\text{dist}(T, I_k)\}_{k=1}^{N}$ will be sorted to rank the images. The distance measure D_2 is known as the relational distance measure and has been applied to various object recognition tasks. Details about relational distance are out of the scope of this work. Interested readers can refer to [27] and the references cited therein.

For D_1 we propose a distance measure called sum of minimum distance to measure the distance between f_T and f_{I_k}. We define SMD as

$$D_1\big(f_T, f_{I_k}\big) = \sum_{i=1}^{m} w_i \times \min_{1 \leq j \leq n_k} \big\| Q_i, P_j^k \big\|_1, \tag{7}$$

where $\| \cdot \|_1$ is the city_block distance. The weights $\{w_i\}$ are a function of the size of each segment (placing more weight on a larger segment). Two functions have been tested for such purposes, with one being $\omega_i = |Q_i|$ and the other being $\omega_i = 1/(1 + e^{-\lambda |Q_i|})$. It is found that for images in our database, $\lambda = 0.0004$ performs well and is used in the experiments presented in the next section. We note that a distance measure such as SMD is intuitively appealing, based on the assumption that two images are similar because their individual components are pairwise similar. The measure in Eq. (7) allows the database to be searched with a template of arbitrary size, with the comparisons being between individual objects (segments).

4. EXPERIMENTAL RESULTS

In this section, we first evaluate the capability of the gradient indexing technique for texture recognition by comparing it to two other texture features. Then, we report retrieval results using the local activity spectrum.

4.1. Texture Recognition Results

A total of 20 Brodatz textures and 10 satellite textures are used in the recognition experiments. Both micro- and macrotextures are present. Besides those shown in Fig. 2, several more texture samples are shown in Fig. 5. For each texture class, 64 samples of size 64 × 64 are used. They are cut from original 512 × 512 images and are nonoverlapping with each other. Half of the samples (i.e., 32 samples) are used for training, while the other half are used for testing purposes. During the experiments with gradient indexing techniques, the nearest neighbor classification rule is used. As mentioned previously, the city_block distance is used as the distance measure for gradient indexing including the local activity spectrum.

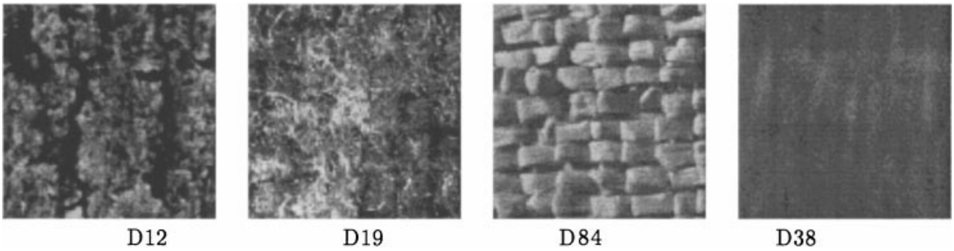


FIG. 5. Some Brodatz texture samples used for experiments.

TABLE I
Texture Classification Results (Percentages of Correct Classification) Using Several Techniques

	LAS	Sobel	Prewitt	Roberts	GMRF	Local_DCT
64 × 64	98	96	91	93	93	84
32 × 32	92	77	75	68	84	75

In the first experiment, the gradient magnitude calculated using Sobel, Prewitt, and Roberts operators is used to compute the gradient histogram. The number of bins is chosen to be 256. For the local activity spectrum measure, 4 bins are used in each of the four directions to bring the total number of bins to 256. In order to show the effectiveness of the gradient indexing techniques, classification techniques based on the Gaussian Markov random field model (GMRF) [28] and on a local linear transform [13] are used as well. These techniques have been known to have very good performance among single-scale techniques.[3]

Consider the random field model below

$$y(s) = \sum_{r \in N} \theta_r y(s \oplus r) + e(s),$$

where N is the neighborhood set with size $M \times M$, and $\{\theta_r\}$ are the random field parameters. Suppose that $\{e(s)\}$ is a zero mean Gaussian process satisfying certain neighborhood properties; then one can show that $\xi = \{C(0), C(r) \mid r \in N_s\}$ is a lossless feature vector [28], where $C(r) = M^{-2} \sum_s y(s)y(s+r)$, $N = N_s \cup -N_s$. By assuming ξ is multivariate Gaussian distributed, a Bayes classifier is designed. We use a second-order GMRF model in our experiments; that is, $N_s = (1, 0) \cup (1, 1) \cup (0, 1) \cup (-1, 1)$. So, loosely speaking, this approach also employs local properties with the same neighborhood as the gradient operators.

Local linear transforms for texture analysis are studied in [13]. Statistics, such as the second and higher moments, computed from each transform coefficient can be employed for texture classification. Similar performance is obtained using various transforms when the same window size is used. In our experiments, we chose the 3 × 3 DCT. The variance of each coefficient was used, together with a minimum distance rule with the Euclidean distance. This technique will be called Local_DCT in the following.

The texture classification results are given in Table I, where each entry in the middle is the probability (in percentage) that a texture sample is classified correctly. Also used are 32 × 32 subsamples cropped from the larger samples. We first note that in both cases, local activity spectrum scores the best. When the image size is small (32 × 32), the improvement over other gradient indexing techniques is especially significant. On the other hand, as a whole, the gradient indexing techniques perform well compared to the GMRF and Local_DCT methods. Considering the computational efficiency possessed by gradient indexing, it can be a good candidate for many applications subject to real-time constraints.

[3] We have chosen not to compare the proposed gradient indexing technique with recently proposed wavelet techniques in this work. A major reason is that gradient indexing can be generalized to the multiscale situation as well. For example, an image pyramid can be built through consecutive low-pass filtering and gradient distributions can be calculated for each scale. We are continuing research in this direction and will compare this multiscale generalization with current wavelet-based techniques in the future.

TABLE II
Percentages of Correct Classification Show the Insensitivity of Gradient Indexing to the Number of Bins

	256	128	64	32
Sobel	96	95	95	90
Prewitt	91	93	90	88
Roberts	93	92	89	82

Note. Images are of size 64 × 64.

To test the sensitivity of gradient indexing to the number of bins, three more different bin numbers are used in the second experiment: 128, 64, and 32 with results given in Table II (together with the results using 256 bins). The sample size 64 × 64 is used. The numbers in the first row are the numbers of bins. Each entry in the middle is the overall average probability of correct recognition. It is easily seen that gradient indexing is quite insensitive to the number of bins used.

In the third experiment, we test the effectiveness of gradient direction for texture recognition. The gradient direction $\theta(\nabla f)$ histogram is used. Results using gradient direction histograms are given in Table III with the number of bins given in the first column. The three methods are denoted as Sobel_θ, Prewitt_θ, and Roberts_θ. Compared with the results obtained using the gradient magnitude histogram given in Table I, we can see that the gradient direction is equally effective. As yet another example, the derivative g_x and g_y are used to form the gradient histogram. Note that $[g_x \;\; g_y]$ is equivalent to $[\mathrm{mag}(\nabla f)\,\theta(\nabla f)]$ since one can derive one from the other and thus both provide the same description of local activity. Product form histograms are used with uniform nonoverlapping bin selection in each direction. Two bin selections are used: 16 and 8 bins in each direction, to bring the total number of bins to 256 and 64, respectively. The classification accuracy is given in Table IV. For convenience, the three methods are denoted as Sobel_xy, Prewitt_xy, and Roberts_xy. Comparing these results with those given in Tables I, II, and III, it is evident that so long as the total number of bins is the same, it is equally effective to use a histogram of the gradient, the gradient magnitude, or the gradient direction.

If one looks at the performance of each of the three gradient operators in all the above experiments, it is seen that gradient indexing is insensitive to the choice of the gradient operator, which means that such a technique captures some essential local property of texture and is robust.

TABLE III
Texture Classification Accuracy (in Percent) Using the Gradient Direction Histogram

	Sobel_θ	Prewitt_θ	Roberts_θ
	(a) 64 × 64 samples		
256	88	92	91
64	86	84	86
	(b) 32 × 32 samples		
256	65	67	74
64	87	72	73

TABLE IV
Texture Classification Accuracy by Using the Gradient Itself

	64×64	32×32
(a) 16 bins each for g_x and g_y		
Sobel_xy	94	71
Prewitt_xy	93	74
Roberts_xy	94	70
(b) 8 bins each for g_x and g_y		
Sobel_xy	84	65
Prewitt_xy	84	60
Roberts_xy	87	62

To summarize this section, we have shown three experimental results demonstrating the good performance of gradient indexing with applications to texture classification, and the improvement in performance achieved by a modified gradient indexing technique, local activity spectrum. It is also shown that gradient indexing is insensitive to the number of bins (so long as this number falls into a reasonable range) and to the choice of the gradient operator, and that the gradient direction is of equal effectiveness in texture recognition as the gradient magnitude. In the next section, we present image retrieval results using the retrieval system described in the previous section and the local activity spectrum measure.

4.2. Image Retrieval Results

One-hundred images are constructed by randomly choosing several texture patches and compositing them. Two image sizes are used, 128×128 and 256×256. The size is randomly decided. The size of each patch in an image is also random. The same 20 Brodatz textures and 10 real satellite textures as used in the previous section are used to provide patches. Each patch is randomly cropped from the original texture images. We have used templates of various sizes and containing various numbers of textures to search the database. As described in Section 3, we use the local activity spectrum as the texture feature.

Several examples are given in Figs. 6, 7, and 9. In Fig. 6, the first four images contain a common texture pattern, Brodatz D38 (water). All the other texture patterns in these images are quite different. When template (e), which is randomly cut from the original D38 image, is given to generate a query process, the search engine gives the top four rankings to (a), (b), (c), and (d) in descending order. In Fig. 7, each of the images (a), (b), and (c) has two common texture patterns, Brodatz D17 (herringbone weave) and D28 (beach sand). The other two patterns are different. A template (e) containing two texture types is cropped from (a) and used for query. The search engine ranks (a), (b), and (c) as the best three matches in that order. As yet another example, (d) is cropped from (a) and consists of all four patterns. When (d) is used to search the database, (a) is ranked as the best match.

Some of the limitations of the proposed algorithm can be analyzed by looking at the image shown in Fig. 7b. It is seen that in the database image Fig. 7b, the spatial relationship of the two texture types used for retrieval purposes is different from that in the template image as shown in Fig. 7e. However, our algorithm does not use this spatial orientation as a search criterion. Rather, it merely takes into account whether two regions border each other. Therefore, it gives a high rank to the image in Fig. 7b. The segmentation result of Fig. 7b is

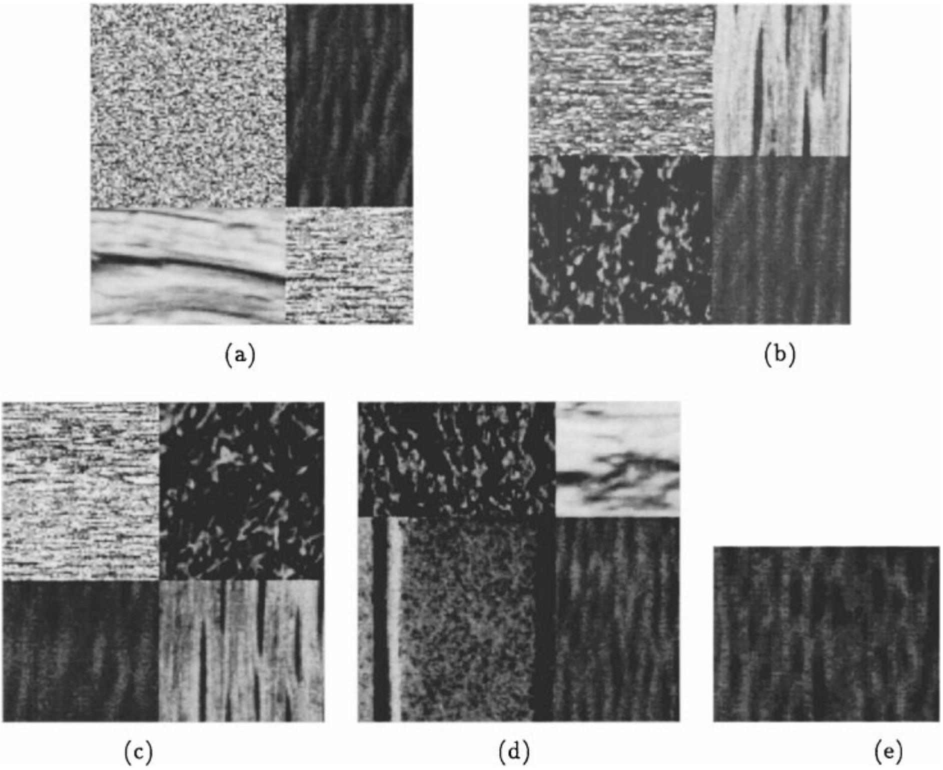


FIG. 6. (a), (b), (c), and (d) are images stored in the database. They have one common pattern, Brodatz D38. (e) is randomly cut from the original Brodatz image D38. When (e) is used as a template (search target), the top four matches are (a), (b), (c), and (d) in descending order.

given in Fig. 8, and the segmentation algorithm is such that the two target texture types are simply recognized as neighbors. Spatial orientation may be of interest for future research.

Finally, we performed tests on some satellite images. One example is given in Fig. 9 where when (b) is given as the template, (a) is output as the best match from a set of the 100 images mentioned above and 8 satellite images. However, more satellite images are to be obtained to perform a more comprehensive evaluation.

We note that all the retrieval results above cannot be obtained with an indexing/retrieval technique where only global features are computed, because the texture property varies from region to region inside each image, and from image to image. It is mandatory to have knowledge of local properties. On the other hand, our distance measure between two segmented images enables one to compare a template of arbitrary size and with an arbitrary number of patterns with an image in the database. By doing so, one is allowed a good amount of freedom in picking up any query sample that one feels appropriate for searching purposes.

5. DISCUSSION

In this work, we have investigated the use of gradient indexing for texture recognition and image retrieval. It is found that such a technique is robust, being insensitive to the choice of the operator and to the number of histogram bins. We also find that the gradient

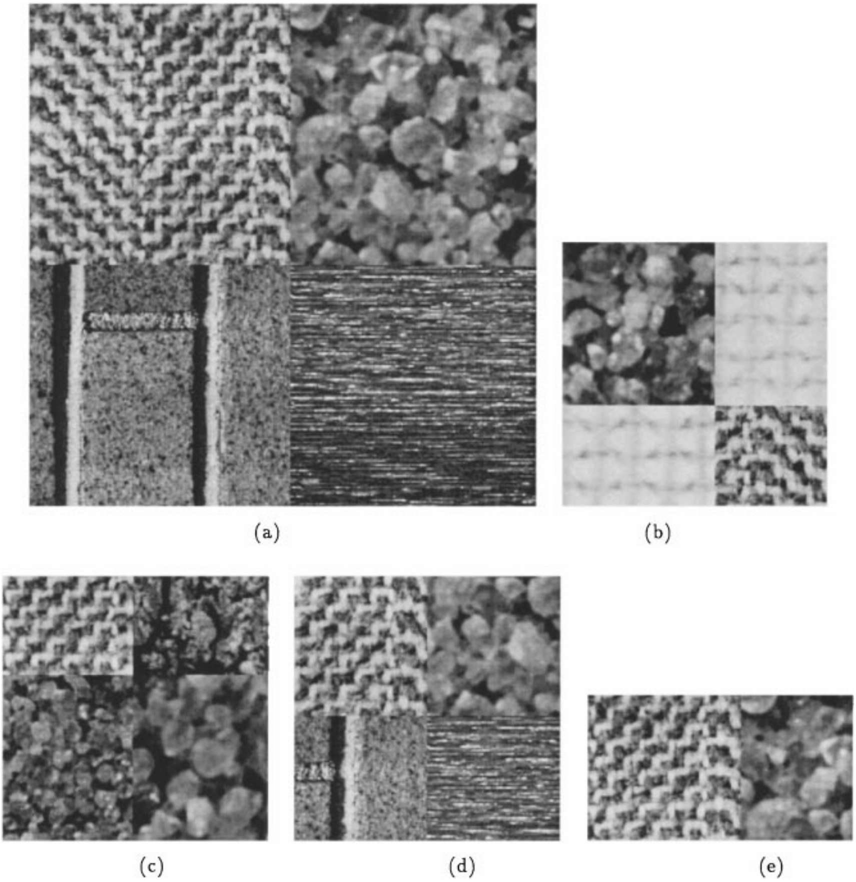


FIG. 7. (a), (b), and (c) are images stored in the database. They have two common patterns, Brodatz D17 and D28, while the other two patterns in each image are different. (d) and (e) are cropped from (a). When (d) is used as a template, the best match is (a); when (e) is used, the best three matches are (a), (b), and (c) in that order.

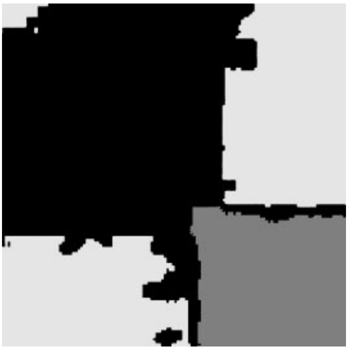


FIG. 8. The segmentation result for the image shown in Fig. 7b.

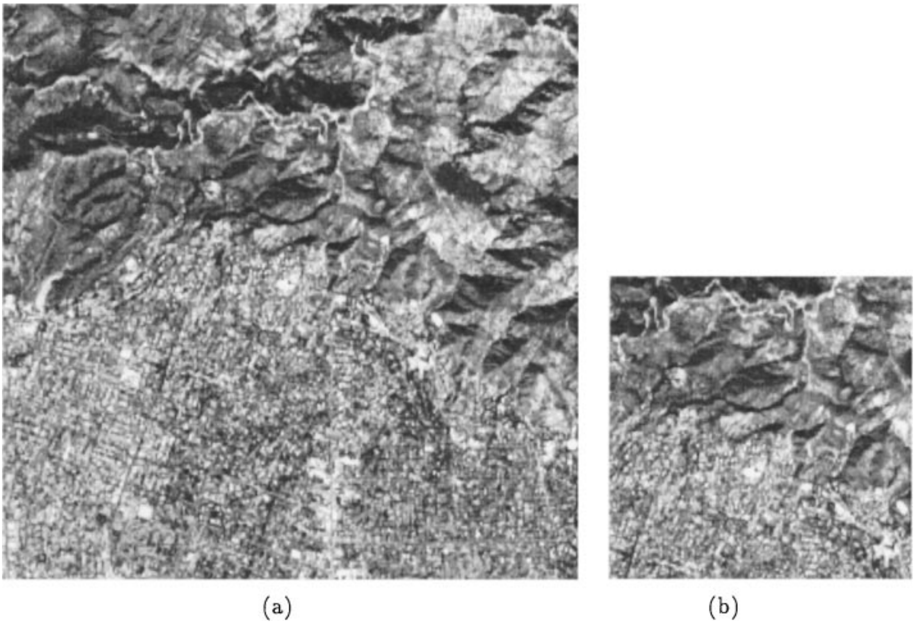


FIG. 9. The retrieval system ranks (a) as the best match when (b) is given as a template.

direction is of similar importance for texture recognition as the gradient magnitude. In other words, the gradient magnitude and direction are interchangeable characteristics of the local spatial activity property possessed by a texture. Other properties of gradient indexing include its translation invariance (which we think is clear without proof). We have also introduced a modified gradient indexing technique, local activity spectrum, having improved performance over those based on gradient operators. A great advantage of the gradient indexing technique is its computational efficiency. However we would like to point out that gradient indexing is not invariant to scale and rotation changes. More work is needed to achieve these two goals.

A texture retrieval system is introduced using local activity spectrum and image segmentation. A segmentation algorithm using LAS is proposed. The use of sum of minimum distance measure and spatial relationship measure allows a user to search an image database using templates of various sizes. This has several advantages. Notably, the user may find some particularly interesting region in some picture. By cropping out that region and using it as a search template, all database images possessing similar regions can be found.

Finally, we would like to make a few comments on the local spatial activity of texture as characterized by Laplacian histogramming. The Laplacian of the image function $f(x, y)$ is defined as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

with its discrete approximate commonly in the form of

$$\nabla^2 f = 4f(z0) - f(z2) - f(z4) - f(z6) - f(z8)$$

using the notation of Fig. 1. The Laplacian operator is often used for edge detection purpose because edge pixels correspond to the zero-crossing positions of the Laplacian. Althoug it is less clear than its gradient counterpart just how the Laplacian of a texture relates to its visual appearance, we have experimented with Laplacian histogramming as well. The performance is similar to gradient indexing using the three operators mentioned above (but is inferior to the local activity spectrum method). For example, by using 256 bins and 16 training samples for each class, the overall classification performance for Laplacian histogramming is 91 and 71%, when the image size is 64 × 64 and 32 × 32, respectively. We conclude from all the results presented in this work that local spatial activity is an effective characteristic for texture discrimination, as have been found in previous work by other researchers. Such local activity can be measured by gradient or even derivatives of higher orders.

REFERENCES

1. R. Haralick, Statistical and structural approaches to texture, *Proc. IEEE* **67**, 1979, 786–804.
2. W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin, The QBIC project: Querying images by content using color, texture and shape, in *Storage Retrieval for Image and Video Databases*, SPIE vol. 1908, pp. 173–187, 1993.
3. B. Tao, K. Robbins, and B. Dickinson, Image retrieval with templates of arbitrary size, in *Storage and Retrival for Image and Video Databases V*, SPIE vol. 3022, pp. 2–11, 1997.
4. R. Chellappa and S. Chatterjee, Classification of textures using Gaussian Markov random fields, *IEEE Trans. Acous. Speech Signal Proc.* **33**, 1985, 959–963.
5. F. Cohen, Z. Fan, and M. Patel, Classification of rotated and scaled textured images using Gaussian Markov random field models, *IEEE Trans. Pattern. Anal. Machine Intel.* **13**, 1991, 192–202.
6. S. Lu and S. Fu, Stochastic tree grammar inference for texture synthesis and discrimination, *Comput. Graphics Image Process.* **9**, 1979, 234–245.
7. S. Zucker, A. Rosenfeld, and L. Davis, Picture segmentation by texture discrimination, *IEEE Trans. Comput.* **24**, 1975, 1228–1233.
8. J. Weszka, C. Dyer, and A. Rosenfeld, A comparative study of texture measures for terrain classification, *IEEE Trans. Sys. Man Cybern.* **6**, 1976, 269–285.
9. R. Haralick, K. Shanmugam, and I. Dinstein, Textural features for image classification, *IEEE Trans. Syst. Man Cybernet.* **8**, 1973, 610–612.
10. T. Chang and C. Kuo, Texture analysis and classification with tree-structure wavelet transform, *IEEE. Trans. Image Process.* **2**, 1993, 429–441.
11. B. Tao and B. Dickinson, Texture classification on block-transformed data, in *Visual Communications and Image Processing*, SPIE vol. 3024, pp. 964–971, 1997.
12. A. Bovic, M. Clark, and W. Geisler, Multichannel texture analysis using localized spatial filters, *IEEE Trans. Pattern Anal. Machine Intel.* **12**, 1990, 55–73.
13. M. Unser, Local linear transforms for texture measurements, *Signal Process.* **11**, 1986, 61–79.
14. K. Laws, *Texture Image Segmentation*, Ph.D. thesis, University of Southern California, 1980.
15. L. Wang and D. He, Texture classification using texture spectrum, *Pattern Recognit.* **23**, 1990, 905–910.
16. F. Ade, Characterization of texture by eigenfilter, *Signal Process.* **5**, 1983, 451–457.
17. R. Gonzalez and R. Woods, *Digital Image Processing,* Addision-Wesley, Reading, MA, 1993.
18. R. Sutton and E. Hall, Texture measures for automatic classification of pulmonary disease, *IEEE Trans. Comput.* **21**, 1972, 667–676.
19. A. Rosenfeld and M. Thurston, Edge and curve detection for visual scene analysis, *IEEE Trans. Comput.* **20**, 1971, 562–569.
20. T. Caelli and D. Reye, On the classification of image regions by color, textrue and shape, *Pattern Recognit.* **26**, 1993, 461–470.
21. G. McLean and M. Jernigan, Indicator functions for adaptive image processing, *J. Opt. Soc. Amer. A* **8**, 1991, 141–156.
22. M. Swain and D. Ballard, Color indexing, *Internat. J. Comput. Vision* **7**, 1991, 11–32.
23. P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover, New York, 1966.

24. A. Pentland, R. Ricard, and S. Sclaroff, Photobook: Tools for content-based manipulation of image databases, in *Storage and Retrieval of Image and Video Databases*, SPIE vol. 2185, pp. 34–47, 1994.

25. H. Stone and C. Li, Image matching by means of intensity and texture matching in the Fourier domain, Technical report, NEC Research Institute, 1995.

26. K. Fukunaga, *Statistical Pattern Recognition*, 2nd ed., Academic Press, San Diego, 1990.

27. R. Wilson and E. Hancock, Structural matching by discrete relaxation, *IEEE Trans. Pattern Anal. Machine Intel.* **19**, 1997, 634–648.

28. R. Chellappa, Two-dimensional discrete Gaussian Markov random field models for image processing, in *Progress in Pattern Recognition* (L. Kanal and A. Rosenfeld, eds.), pp. 79–112, Elsevier, 1985.