PERGAMON

Pattern Recognition 34 (2001) 1841–1851

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Exploiting image indexing techniques in DCT domain

Chong-Wah Ngo, Ting-Chuen Pong*, Roland T. Chin

Department of Computer Science, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

Received 19 March 1999; received in revised form 3 March 2000; accepted 15 May 2000

Abstract

This paper is concerned with the indexing and retrieval of images based on features extracted directly from the JPEG discrete cosine transform (DCT) domain. We examine possible ways of manipulating DCT coefficients by standard image analysis approaches to describe image shape, texture, and color. Through the Mandala transformation, our approach groups a subset of DCT coefficients to form ten blocks. Each block represents a particular frequency content of the original image. Two blocks are used to model rough object shape; nine blocks to describe subband properties; and one block to compute color distribution. As a result, the amount of data used for processing and analysis is significantly reduced. This can lead to simple yet efficient ways of indexing and retrieval in a large-scale image database. Experimental results show that our proposed approach offers superior indexing speed without significantly sacrificing the retrieval accuracy.

Keywords: Image indexing and retrieval; DCT; Color histogram; Shape modeling; Texture recognition

1. Introduction

With the wide spread use of the WWW, future digital libraries are expected to manipulate huge amounts of image and video data. Due to the limitations of space and time, most of the data are represented in compressed forms. As a result, techniques used for editing, segmenting, and indexing images directly in the compressed domain have become one of the most important topics in digital libraries. In this paper, we investigate the use of DCT coefficients, which are the major components of JPEG and MPEG, in content-based image retrieval (CBIR).

In general, CBIR emphasizes rough image matching rather than exact matching. The DCT domain, to a certain extent, has unique scale invariance and zooming characteristics which can provide insight into object and texture identification, therefore, it is naturally considered to be a potential domain in mining visual features. We propose an approach for capturing the rough and global content of an image with very few DCT coefficients. Since other techniques such as relevancy feedback [1] and query expansion [2] can be used to fine tune the retrieval results, CBIR should not suffer greatly as long as a majority of the relevant images are retrieved.

In our proposed approach, only ten DCT coefficients from each 8×8 JPEG image block are extracted. By applying Mandala transformation [3], this approach groups these coefficients and forms ten blocks. Each block represents a particular frequency content of the original image. We apply the appropriate techniques to each block and generate features to describe the shape, texture, and color properties of the original image. Since the first block conveys color information, we use it to compute color histograms. To model object shape, we combine two blocks to compute its image gradient. With this, our approach tracks the contour of the underlying object and computes moments to estimate its global shape. Finally, the proposed approach calculates the intensity variances of nine blocks to describe their texture properties.

* Corresponding author. Tel.: 852-2358-6974; fax: 852-2358-1477.

E-mail addresses: cwngo@sc.ust.hk (C.-W. Ngo), tcpong@cs.ust.hk (T.-C. Pong), roland@cs.ust.hk (R.T. Chin).

0031-3203/01/$20.00
PII: S0031-3203(00)00111-4

2. Related work

Direct manipulation of the compressed images and videos offers low-cost processing of real time multimedia applications. To date, these efforts include algebraic operations [4], geometric transformation [5], image segmentation [6], feature extraction [7], indexing [8], and camera break detection [9]. Most of these works were done directly in the DCT domain.

Smith and Rowe [4] have shown how pixel addition, pixel multiplication, scalar addition and scalar multiplication can be implemented in the DCT domain. With these algorithms, one can dissolve a sequence of compressed images and overlay a subtitle on a compressed image. Chang and Messerschmitt [10] further proposed algorithms for manipulating compressed videos using the DCT coefficients with or without motion compensation. Shen and Sethi [5] described methods of performing geometric transformations such as rotation and diagonal flip by manipulating DCT coefficients.

Soltane et al. [6] suggested an adaptive edge operator selection scheme for image segmentation based on the mean, variance, and entropy of DCT coefficients. Shen and Sethi [7] further presented an edge detector with twenty times the speed of conventional methods. Their proposed approach determines edge strength and orientation through the pattern analysis of DCT coefficients, however, this method assumes the edge of each 8×8 block is a straight line. As a result, disconnected and broken edges, which are not suitable for contour extraction or segmentation, occur at the boundaries of some blocks. On the contrary, our proposed method generates a reduced, yet smooth edge map by manipulating two Mandala blocks. Since CBIR requires only rough matching, our method provides adequate visual cues and is suitable for further image analysis. These efforts somehow exploit the possibility of using DCT coefficients for describing object shape, and this is not well understood in the current literature.

In addition, Chang [8] reported several possible ways of extracting low level features from the compressed domain. For instance, the texture feature can be formed by computing the statistical measures of the DCT coefficients. To reduce the dimensionality of the feature space, the Fisher Discriminant technique is employed to maximize the separability among the known texture classes. Similarly, Seales et al. [11] employed a principle component analysis to obtain eigenvectors from DCT coefficients. Since DCT is linear and orthogonal, the distance in eigenspace is preserved. This method projects the DCT space to the first few principle axes and performs object recognition directly in the compressed domain. These approaches, nevertheless, may not be suitable for databases of large volume due to the training and updating costs, moreover, the retrieval accuracy may be degraded when new data arrives. Since future digital libraries are targeted to tackle the dynamic environment of the WWW Internet, re-training of huge amounts of data would not be feasible.

Earlier research into understanding the properties of DCT coefficients was reported by Hsu et al. [3], they proposed an approach for classifying man-made and natural images by computing 48 statistical features. Hou et al. [12] further demonstrated, in theory, that DCT behaves somewhat like subband filters and their impulse responses closely relate to wavelets.

Recently, Shneier and Abdel-Mottaleb [13] described a method of generating keys of JPEG images for retrieval, where a key is the average value of DCT coefficients computed over a window. During retrieval, images with similar keys are assumed to be similar, however, there is no semantic meaning associated with such similarities.

DCT coefficients also play a major role in video segmentation. Patel and Sethi [9] proposed approaches of detecting camera cuts by analyzing the DCT coefficients extracted from I-frames of MPEG. Detecting cuts in P and B frames is also possible since Yeo and Liu [14] have proposed method for estimating the DC sequences of P and B-frames. In addition, Ariki and Saito [15] applied DCT coefficients to cluster news video clips, and reported that AC coefficients are less sensitive to abrupt intensity change due to camera flushing.

3. JPEG DCT coefficients

3.1. Compression scheme

In JPEG the original image is divided into 8×8 blocks, then, each block is transformed independently by DCT. The transformed coefficients are quantized and Huffman coded. The only information loss is due to the quantization step. The quantization factors will perturb but not destroy the essential characteristics of the DCT coefficients. The DCT is defined as

$$F_{u,v} = \frac{1}{4} C(u)C(v) \sum_{i=0}^{7} \sum_{j=0}^{7} \cos\frac{(2i+1)u\pi}{16} \cos\frac{(2j+1)v\pi}{16} f_{i,j} \tag{1}$$

where $C(u), C(v) = 1/\sqrt{2}$ if $u, v = 0$, otherwise $C(u), C(v) = 1$. $F_{u,v}$ is the 2D DCT coefficient, and $f_{i,j}$ is the image spatial value. $F_{0,0}$ is normally called DC, while the rest are referred to as AC coefficients. The basis vectors of DCT are linear and orthogonal.

In this paper, we only extract $F_{0,0}$, $F_{0,1}$, $F_{1,0}$, $F_{2,0}$, $F_{1,1}$, $F_{0,2}$, $F_{0,3}$, $F_{1,2}$, $F_{2,1}$ and $F_{3,0}$ for indexing and retrieval. All these coefficients are quantized but not Huffman coded.

3.2. Basic properties

Denote $\mathbf{F}_j = \cos((2j+1)\pi/16)\sum_{i=0}^{7} f_{i,j}$ for $0 \leqslant j \leqslant 3$ and $\mathbf{F}_j = \cos((15-2j\pi)/16)\sum_{i=0}^{7} f_{i,j}$ for $4 \leqslant j \leqslant 7$, which is the sum of column j multiplied by a constant. Then, $F_{0,0} = \frac{1}{8}\sum_{i=0}^{7}\sum_{j=0}^{7} f_{i,j}$ and $F_{0,1} = \frac{1}{4}\{(\mathbf{F}_0 + \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3) - (\mathbf{F}_4 + \mathbf{F}_5 + \mathbf{F}_6 + \mathbf{F}_7)\}$. Notice that $F_{0,0}$ is actually 8 times the block intensity mean and $F_{0,1}$ is the horizontal block intensity difference. Similarly, $F_{1,0}$ is the vertical block intensity difference.

One can project these coefficients from the DCT domain to the Mandala domain. The Mandala transformation simply groups the coefficients with the same u, v as a block, where each block represents a particular frequency content of the original image. Denote $I_{x^u y^v}$ as a Mandala block with $u, v = [0,1,\ldots,7]$, we can have 64 blocks as $I, I_x, I_y, I_{y^2}, I_{xy}, I_{x^2}, \ldots, I_{x^7y^7}$ in the zig-zag order. I is normally called the DC image. We can express the image gradient ∇f, edge direction θ, and zero crossing $\nabla^2 f$ of I as,

$$\nabla f = \sqrt{I_x^2 + I_y^2}, \tag{2}$$

$$\theta = \tan^{-1}\frac{I_y}{I_x}, \tag{3}$$

$$\nabla^2 f = I_{x^2} + I_{y^2}. \tag{4}$$

Fig. 1 shows the resulting 64×64 images in the Mandala domain, computed from the 512×512 lena and airplane images.

Due to the linear orthogonal nature of DCT and its close relationship with Karhunen–Loève Transform (KLT), it is intuitive to employ DCT coefficients for image indexing and retrieval. DCT is asymptotically equivalent to KLT because of its stationary Markov-1 signals [16]. Its auto-covariance matrix, a Toeplitz matrix, is predetermined. In this case, selecting the first few DCT coefficients is approximately equivalent to selecting the KLT transformed values that exhibit significant variance.

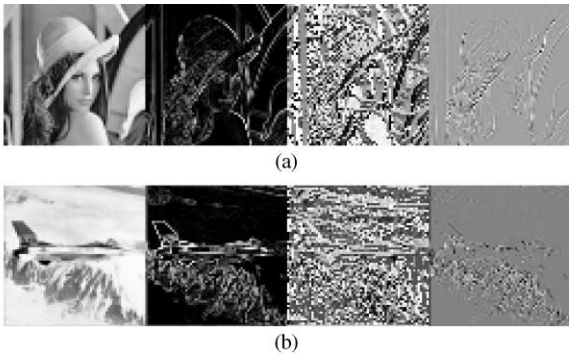

Fig. 1. Images constructed directly from DCT coefficients. (from left to right) DC, image gradient, edge direction and zero crossing images. (a) Lena, (b) Airplane.

4. Color histogram

The zero frequency, $F_{0,0}$, coefficients of a JPEG image are grouped to form a reduced and smooth DC image; the color space is then transformed from RGB (red, green, blue) to HSV (hue, saturation, brightness).[1] HSV is widely used in color histograms because of its uniformity, compactness, completeness and naturalness [18].

Let $v_c = (r, g, b)$ be the color triple in RGB space, and $w_c = (h, s, v)$ be the corresponding transformed triple in HSV space, where $r, g, b, s, v \in [0 \ldots 1]$ and $h \in [0 \ldots 6]$. Denote $\mathbf{T}$ as the transformation, then $w_c = \mathbf{T}(v_c)$. $\mathbf{T}$ is [19]

$$v = \max(r,g,b), \quad s = \frac{v - \min(r,g,b)}{v},$$

$$x = \frac{v - r}{v - \min(r,g,b)}, \quad y = \frac{v - g}{v - \min(r,g,b)},$$

$$z = \frac{v - b}{v - \min(r,g,b)},$$

$$h = \begin{cases} 5 + z; & r = \max(r,g,b) \text{ and } g = \min(r,g,b), \\ 1 - y; & r = \max(r,g,b) \text{ and } g \neq \min(r,g,b), \\ 1 + x; & g = \max(r,g,b) \text{ and } b = \min(r,g,b), \\ 3 - z; & g = \max(r,g,b) \text{ and } b \neq \min(r,g,b), \\ 3 + y; & b = \max(r,g,b) \text{ and } r = \min(r,g,b), \\ 5 - x; & \text{otherwise.} \end{cases} \tag{5}$$

The color histograms are obtained by summing up the number of pixels with similar values in the HSV components. To reduce the length of the histogram features, the color space is quantized to produce a compact set of colors. Because hue conveys the most significant characteristic of color, it is quantized to 18 levels. Saturation and brightness are separately quantized into 3 levels. The quantization provides 162 ($18 \times 3 \times 3$) distinct color sets. As stated in [18], such representations can yield greater perceptual tolerance while separating the hues so that red, green, blue, yellow, magenta, and cyan are each represented by three subdivisions.

We compare the performance of color histograms of DC images and uncompressed images. For simplicity, we

[1] Note that JPEG uses YCrCb color space. Through the software package in [17], one can obtain the RGB components directly from JPEG images.

refer to the former as the DC color histogram approach; and the latter as the color histogram approach. The indexing speed of the DC color histogram is approximately fourteen times faster than the color histogram. For an image of size 128×128, the DC color histogram takes less than 0.01 s, while the color histogram takes approximately 0.14 s of CPU time (excluding decompression time), to process on a Sun Sparc20 machine. Fig. 2 shows the retrieval results of four image queries in the VisTex [20] database of 228 images. The histogram intersection [21] is used as the color similarity measure. For both approaches, more than half of the top ten retrieved images are the same although their rankings are different. Since color retrieval, in general, is subject to human perception, as long as the top few retrieved images are similar to the query, the results can be improved by

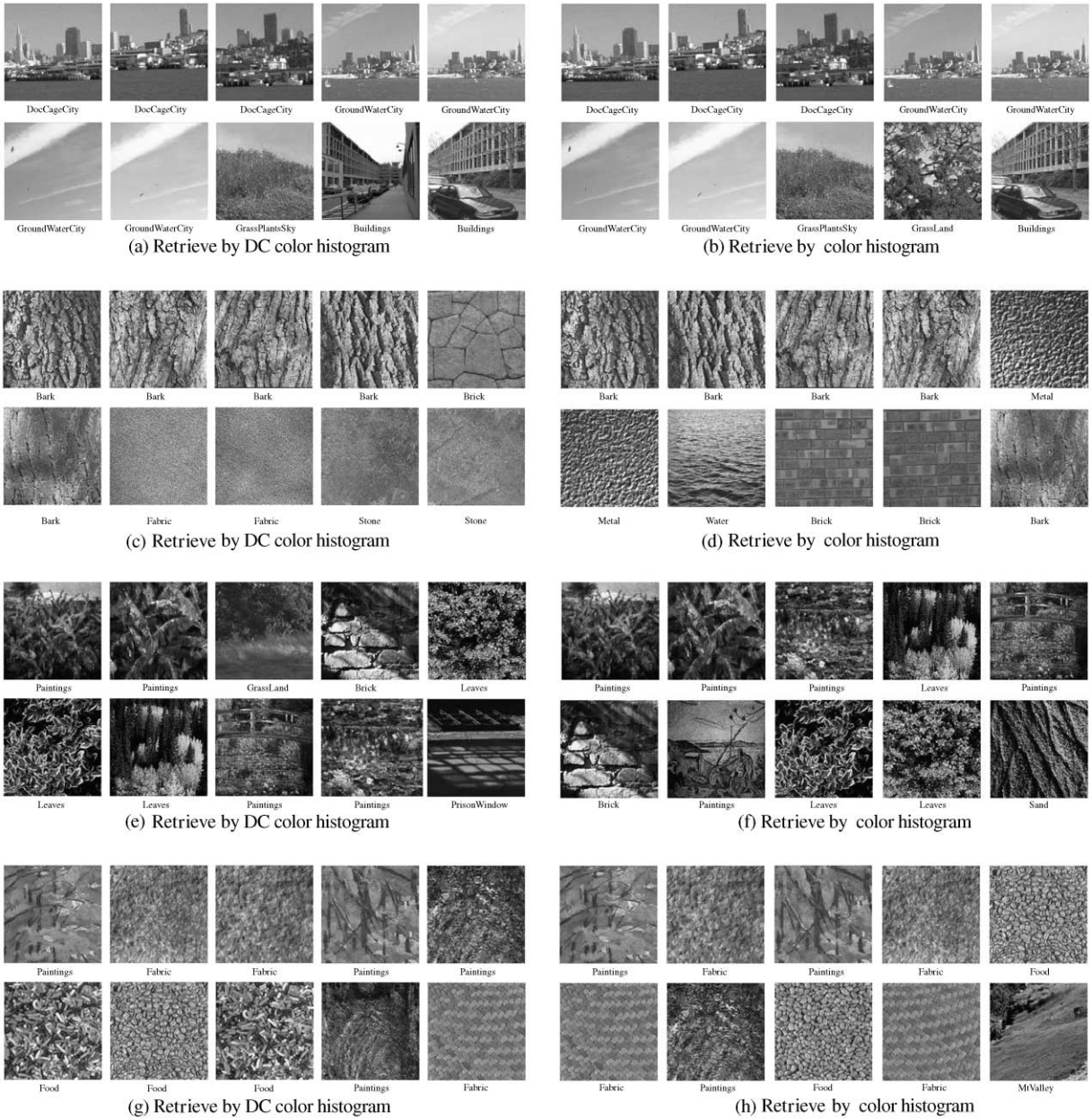

Fig. 2. Color retrieval of VisTex database comparing the DC color histogram (for JPEG images) to the color histogram (for uncompressed images). In each picture, the retrieved images are raster scan ordered by their similarities to the query image in the upper left.

involving human in the retrieval loop using relevancy feedback mechanisms.

5. Rough shape modeling

The basic idea of our shape modeling scheme is to generate an image gradient ∇f from the Mandala blocks I_x and I_y, track the contour of the underlying object, and then compute the seven invariant contour moment features for indexing. We refer to the indexed shape features as DCT moment features. Fig. 3 shows four sample images and their corresponding extracted contours. The size of a sample image is 128×128, while the contour image is only 16×16!

Given a contour $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]$, where $\mathbf{v}_i$ is defined on the finite grid: $\mathbf{v} \in \mathbb{E}^2 = \{(x, y)\colon x, y = 1, 2, \ldots, M\}$. Denote $\mathbf{g} = (\bar{x}, \bar{y})$ as the contour centroid. The central moment of the $(p + q)$th order is

$$\mu_{p,q} = \sum_{(x_i, y_i)} \sum_{\in \mathbf{V}} (x_i - \bar{x})^p (y_i - \bar{y})^q. \tag{6}$$

Notice that $\mu_{0,0}$ is the perimeter of a contour and $\mu_{0,1} = \mu_{1,0} = 0$. The central moments in (6) are invariant to translation. They can also be normalized to scale invariance by [22]

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad \text{where } \gamma = p + q + 1, \text{ for } p + q = 2, 3, \ldots \tag{7}$$

To be rotational invariance, Hu [23] derived seven moment invariants based on the 2nd- and 3rd-order moments:

$$\begin{aligned}
\phi_1 &= \eta_{20} + \eta_{02},\\
\phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2,\\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2,\\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{03} + \eta_{21})^2,\\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\\
&\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2],\\
\phi_6 &= (\eta_{20} - \eta_{02}[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}),\\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})\\
&\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2].
\end{aligned} \tag{8}$$

ϕ_1 to ϕ_7 which are invariant to translation, scaling and rotation, are frequently used in shape recognition and have been used in this section as shape features. Since the range of different ϕ_i can significantly vary, the covariance matrix of the seven moment invariants is computed and the Mahalonobis distance is used as the similarity measure between two shapes.

To evaluate the indexing and retrieval performances, we compare the DCT moment with deformable prototype [24] and moment (in uncompressed domain). We use the tropical fish image database [24] which consists of fifteen fish categories for experiments. Retrieval performance is evaluated by the measure AVRR/IAVRR [24,25]. AVRR is the average rank of all relevant images for a retrieval, while IAVRR is the ideal average rank when all n relevant images from a particular category appear in the first n position. They are formulated as

$$\text{IAVRR} = \frac{1}{m} \sum_{i=1}^{c} \frac{n_i^2}{2}, \tag{9}$$

$$\text{AVRR} = \frac{1}{m} \sum_{i=1}^{m} \frac{\sum_{k=1}^{m} (k \times d_k)}{p_i}, \tag{10}$$

where m is the total number of images in the database, c is the number of categories, n_i is the number of relevant images in ith category, and p_i is the number of relevant images that are in the same class as query i. The value $d_k = 1$ if the retrieved image in rank kth position belongs to a relevant image, otherwise $d_k = 0$. Perfect retrieval result is AVRR/IAVRR $= 1$.

Table 1 summarizes the experimental results. The speed of indexing the DCT moment is approximately 40 times faster than indexing the moment in the

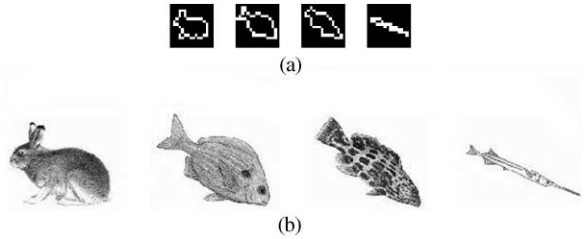

Fig. 3. The contours (a) extracted from the corresponding images in (b).

Table 1
Performance of various shape features (on a Sun Sparc20 machine). The indexing time of the deformable prototype approach is not listed here since it involves the manual selection of prototypes. The deformation prototype requires the deformation of each prototype with all images in the database during indexing, which is much slower than the moment based approach

	DCT Moment	Moment	Deformable prototype
Indexing time (s)	0.006	0.24	–
Feature vector length	7	7	5
AVRR/IAVRR	6.3	5.0	2.6

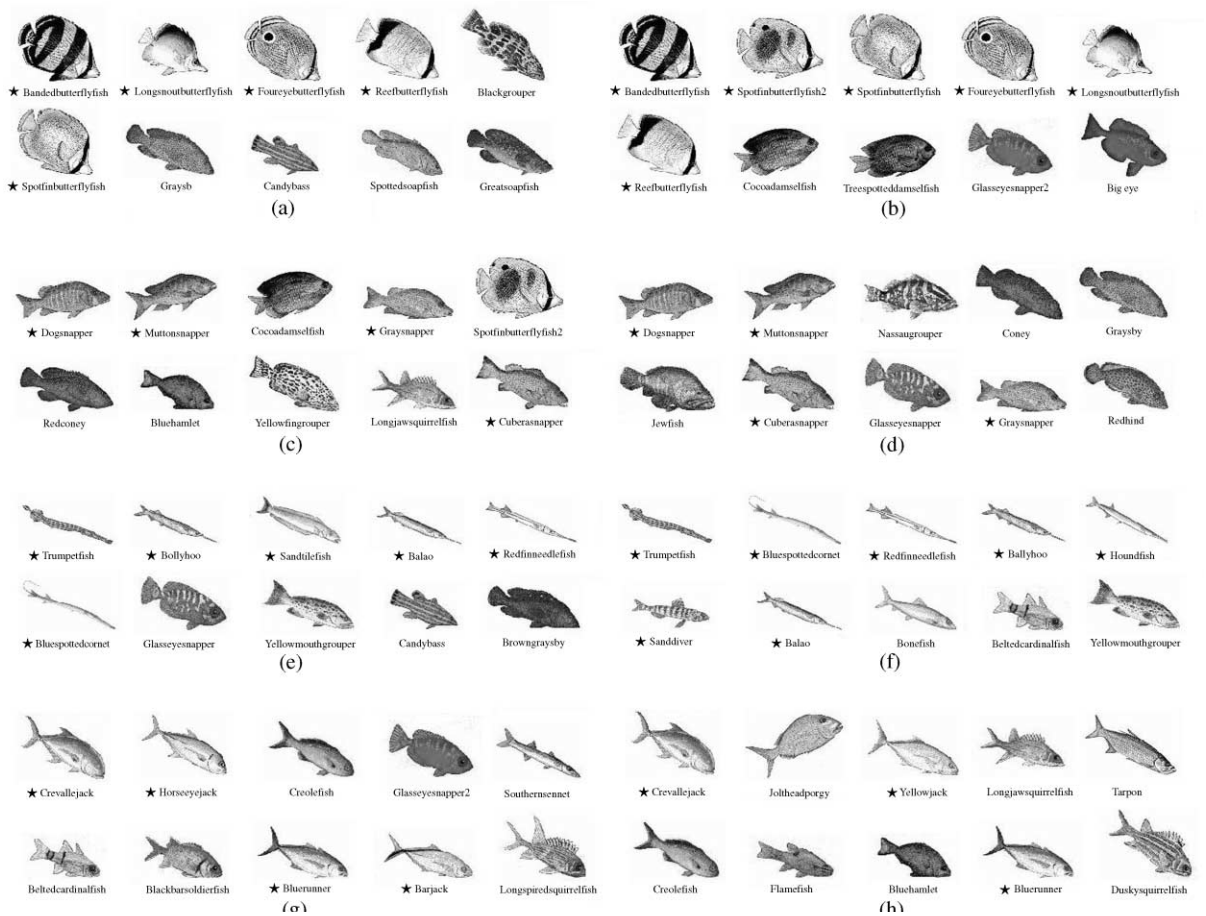

Fig. 4. Shape retrieval of the fish database comparing DCT moment and the deformable prototype. In each picture, the retrieved images are raster scan ordered by their similarities to the query image in the upper left corner. Relevant images are marked by red stars. (a), (c), (e) and (g) Retrieve by DCT moment; (b), (d), (f) and (h) Retrieve by deformable prototype.

uncompressed domain (excluding decompression time) with slight degradation in retrieval performance. The deformable prototype [24] requires a pre-processing step: the manual browsing of a database to select the representative prototypes, which is tedious when the size of database is large. The number of prototypes corresponds to the length of a feature vector. The indexing time of the deformable prototype includes solving the eigen problem for each shape, finding the point correspondence of a shape with every prototype and measuring the deformation energies.[2] Compared with the DCT moment, the deformable prototype is computationally expensive. In Fig. 4, four examples of DCT moment retrieval are shown together with the results given by the deformable prototype. In each picture, the top ten retrieved images of a query in the upper left corner are given. The relevant images are marked with stars. The relevant images being retrieved by DCT moment are quite similar to the deformable prototype,[3] except that their ranking is different. In general, the deformable prototype offers better ranking capability while the DCT moment gives superior indexing speed and does not

[2] In contrast to the traditional deformable-based retrieval, the deformable prototype performs deformation process off-line (during indexing) to omit the on-line shape matching during retrieval.

[3] In Figs. 4(a) and (c), the missed and false retrieval of spotfinbutterflyfish2 is due to the edge linking problem when tracking the contour.

require manual operation. In this case, the DCT moment can serve as a filtering mechanism in the initial stages of retrieval, while the deformable-based matching technique (for instance [26]) can serve at the later stage to improve ranking capability.

6. Texture description

Because DCT compresses the image energy into lower-order coefficients, we only consider the first nine AC coefficients. The texture feature is,

$$S_{u,v} = \mathbf{E}[I^2_{x^u y^v}] - \mathbf{E}[I_{x^u y^v}]^2 \tag{11}$$

where $S_{u,v}$ and $\mathbf{E}[I_{x^u,y^v}]$ are the variance and expectation of a Mandala block I_{x^u,y^v}, respectively, for $0 < u + v \leqslant 3$. The resulting texture feature vector is the variance of $\langle I_x, I_y, I_{x^2}, I_{xy}, I_{y^2}, I_{x^3}, I_{x^2y}, I_{xy^2}, I_{y^3} \rangle$ in the Mandala space. Similar to shape retrieval, the covariance matrix of the nine DCT texture features is computed and the Mahalonobis distance is used as the texture similarity measure.

To evaluate the effectiveness and efficiency of the DCT texture features, performance comparisons are made with Gabor wavelet features [27], tree-structured wavelet transform (TWT) [28], multiresolution simultaneous autoregressive model (MR-SAR) [29] and Tamura features (coarseness, contrast, directionality) [30] in term of indexing speed and recognition accuracy. The texture database used in the experiments is composed of 112 texture classes obtained from the Brodatz album [31]. Each texture class provides nine 128×128 images. The database contains a large variety of natural textures, including some inhomogeneous classes which are not usually included in studies. The performance is measured in terms of the recognition rate which is defined as the percentage number of retrieved images belonging to the same class as a query image in the top eight matches. We use every image in the database as query and calculate the average recognition rate.

Table 2
Performance of various texture features (on a Sun Sparc20 machine)

	DCT features	Gabor features	TWT	MR-SAR	Tamura features
Indexing time (s)	0.01	54.5	3.8	34.0	0.42
Feature vector length	9	48	84	15	3
Ave. Recog. Rate (%)	55.4	77.6	43.4	75.7	34.2

Table 2 gives the summary of the experimental results. It shows the trade-off of various texture features in terms of indexing speed (per image), feature vector length and recognition accuracy. Fig. 5 further illustrates the recognition performance as a function of the number of retrieved images. Throughout the experiments, Gabor and MR-SAR features give the best recognition rate with the expense of intensive indexing time. DCT features, by contrast, are computationally attractive. Moreover, on average, more than half of the relevant images appear in the top eight matches, which provide a good initialization for relevancy feedback mechanisms. In addition, DCT features constantly outperform TWT and Tamura features. TWT, which decomposes the high frequency band in the tree-structured representation, leads to instable features for some texture patterns. Tamura features, which consist of only three global texture features, are not robust for retrieval in large database.

In Fig. 6, three examples of texture retrieval are shown together with the results given by DCT and Gabor features. The query images consists of harmonic, evanescent and indeterministic texture patterns. In all cases, DCT features include one to two more irrelevant images than Gabor features. Some of these irrelevant images are perceptually similar to their query images, for instance D105 and D68 in Fig. 6(d). In Figs. 6(f) and (g), both features retrieve the same irrelevant images D108 and D67. In general, Gabor features offer better ranking capability while DCT features offer superior indexing speed. The choice of either feature is tailored to applications.

To test the effectiveness of DCT features in a more complicated dataset, we used VisTex [20] which consists of 228 texture classes and 3648 images for demonstration. Similar to the cases in Brodatz database, the average

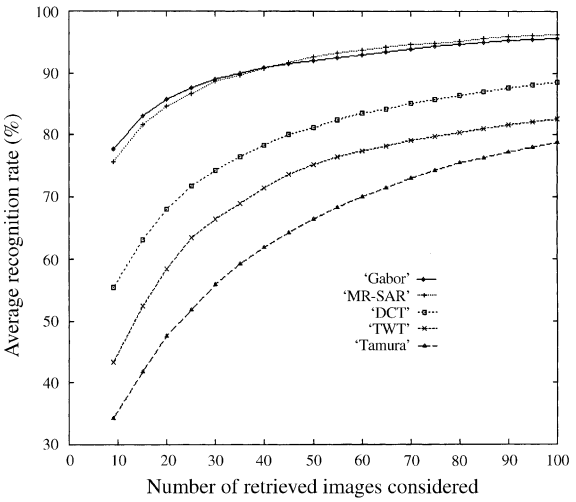

Fig. 5. Retrieval performance on the Brodatz database according to the number of top matches considered.

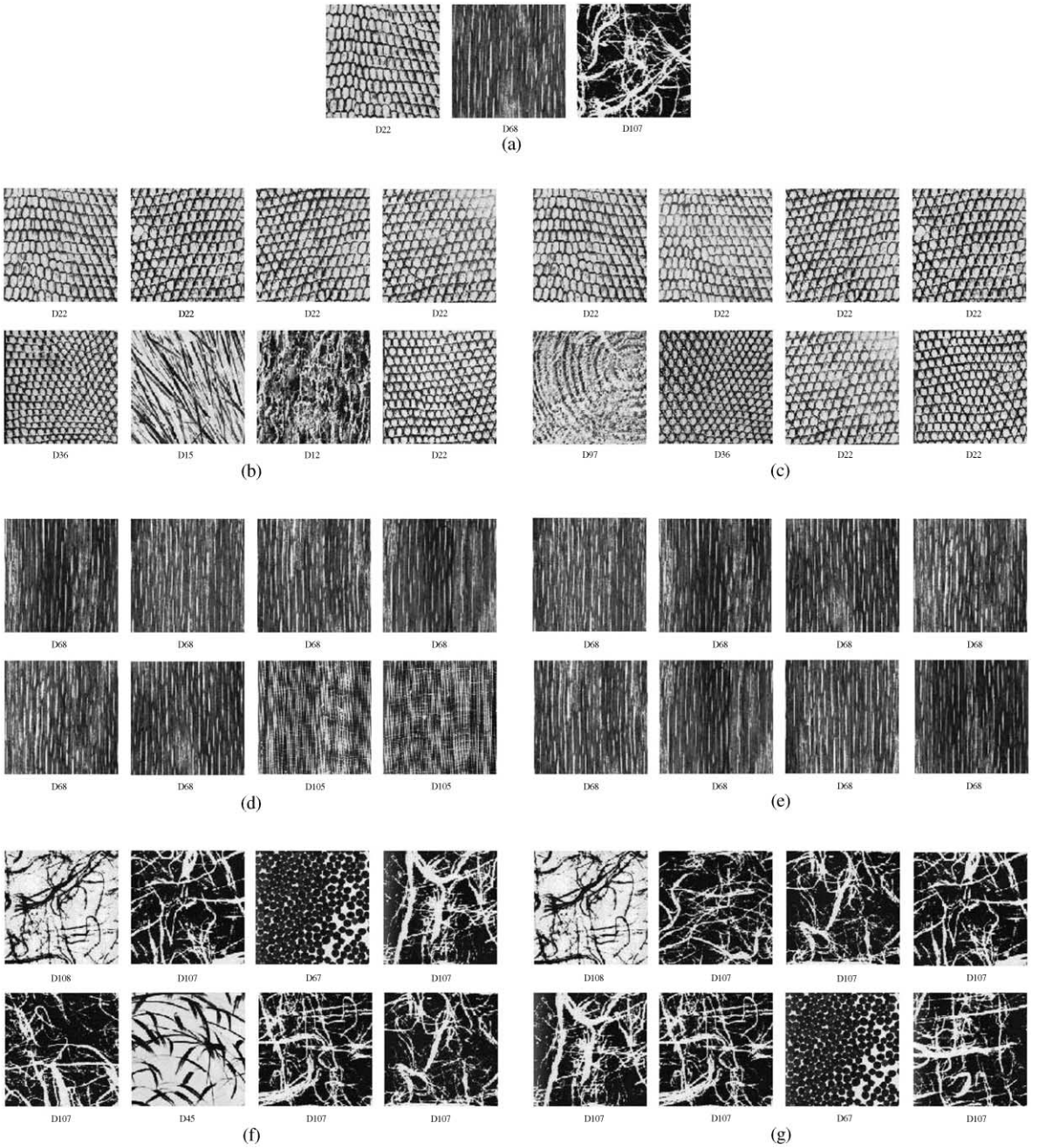

Fig. 6. Texture retrieval of the Brodatz database comparing DCT and Gabor features. The images are raster scan ordered by their similarity. (a) Query texture pattern, (b) and (c) Query D22 by Gabor features, (d) and (e) Query D68 by Gabor features, (f) and (g) Query D107 by Gabor features.

recognition rate of DCT features lies between Gabor and Tamura features as illustrated in Fig. 7(a). Figs. 7(b)–(e) further show two retrieval examples which demonstrate DCT features work reasonably well in the VisTex database.

7. Conclusion and future work

We have presented approaches for indexing shape, texture, and color features directly in the DCT domain by exploiting ten DCT coefficients. For color and shape

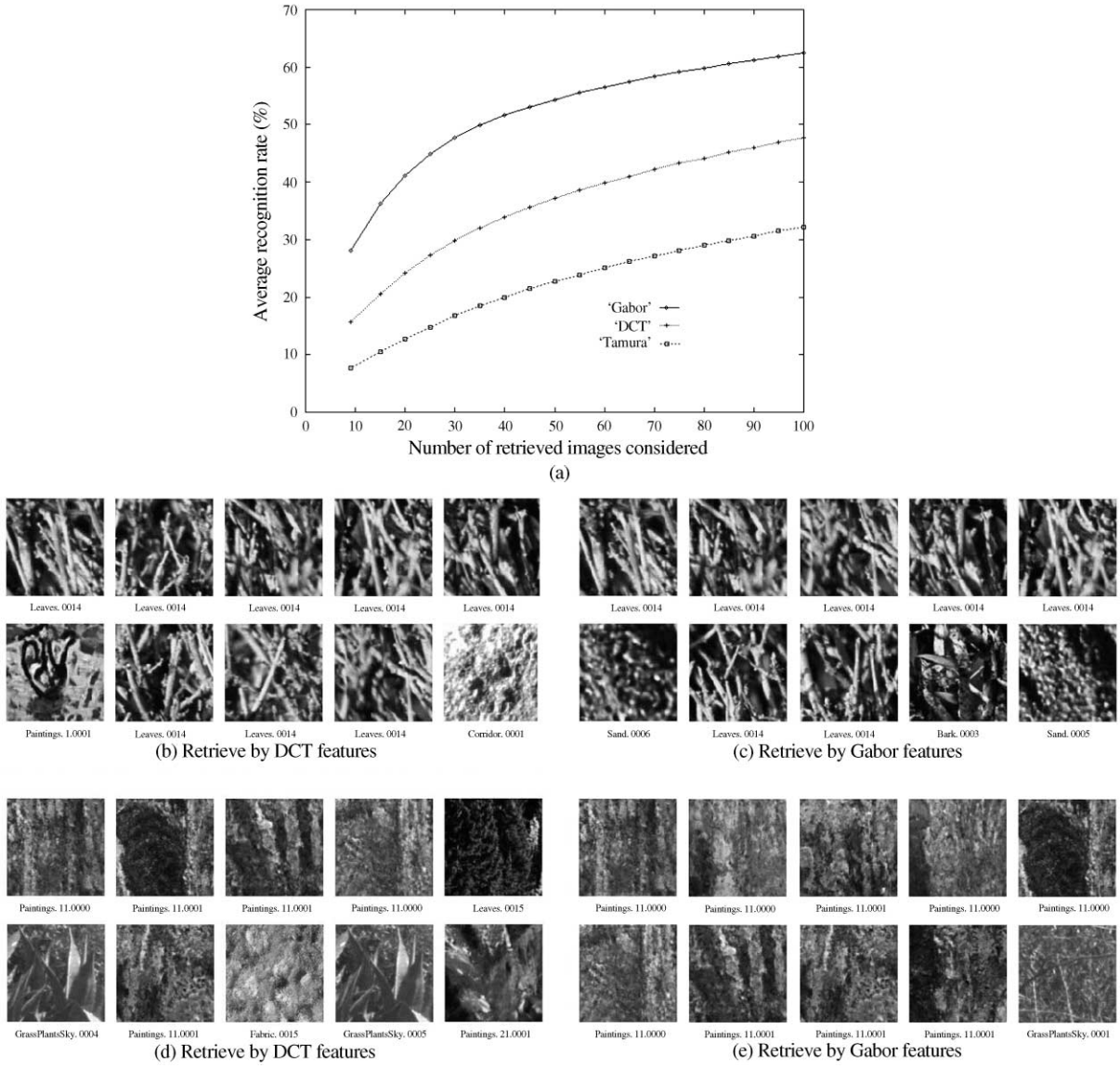

Fig. 7. Texture retrieval in the VisTex database. (a) Retrieval performance according to the number of top matches considered: (b)–(e) Retrieval of textures comparing DCT and Gabor features. The retrieved images are raster scan ordered by their similarity to the query image in the upper left corner.

indexing, the proposed methods achieve significant speed up compared to the same approach operating in the uncompressed image domain. Overall, the retrieval results are competent since most of the top retrieved images are relevant. For texture, the indexing speed of the DCT features is the fastest among the five tested texture features. Although the retrieval performance is not as good as Gabor and MR-SAR features, it does perform better than TWT and Tamura features. In general, DCT features are capable of retrieving similar images although the ranking of their similarity to a query is unpredictable. Nevertheless, the ranking can further be improved by using techniques such as the relevancy feedback mechanism where human plays a part in the retrieval loop. Image retrieval, in most cases, is subjective to human perception, for some applications we may just need a system that can provide rough image matching. DCT features comport with these types of applications, especially with their superior indexing speed.

Acknowledgements

This work is supported in part by RGC Grants HKUST661/95E, HKUST6072/97E, and HKUST6089/99E.

References

[1] Y. Rui, T.S. Huang, S. Mehrotra, Content-based image retrieval with relevancy feedback in mars, Proceedings of IEEE International Conference on Image Processing, 1997, pp. 815–818.
[2] H. Zhou, S. Chan, K.F. Lai, Query Expansion by text and image features in image retrieval, J. Visual Commun. Image Representat. 9 (4) (1998) 287–299.
[3] Y.S. Hsu, S. Prum, J.H. Kagel, H.C. Andrews, Pattern recognition experiments in the Mandala/cosine domain, IEEE Trans. Pattern Anal. Mach. Intell. 5 (5) (1983) 512–520.
[4] B.C. Smith, L.A. Rowe, Algorithms for manipulating compressed images, IEEE Comput. Graphics Appl. 13 (5) (1993) 34–42.
[5] B. Shen, I.K. Sethi, Inner-block operations on compressed images, Proceedings of the ACM International Conference on Multimedia'95, 1995, pp. 490–499.
[6] S. Soltane, N. Kerkeni, J.C. Angue, The use of two dimensional discrete cosine transform for an adaptive approach to image segmentation, Proceedings of the SPIE Image and Video Processing IV, 1996, pp. 242–251.
[7] B. Shen, I.K. Sethi, Direct feature extraction from compressed images, Proceedings of the SPIE Storage and Retrieval for Image and Video Database IV, 1996, pp. 404–414.
[8] Shih-Fu Chang, Compressed domain techniques for image/video indexing and manipulation, IEEE International Conference on Image Processing, 1995, pp. 314–317.
[9] N.V. Patel, I.K. Sethi, Compressed video processing for cut detection, IEE Proc. Visual Image Signal Process. 143 (5) (1996) 315–323.
[10] S.F. Chang, D.G. Messerschmitt, Manipulation and compositing of MC-DCT compressed video, IEEE J. Selected Areas Commun. 13 (1) (1995) 1–11.
[11] W. Brent Seales, C.J. Yuan, W. Hu, Content analysis of compressed video, Technical Report 265–96, University of Kentucky, 1996.
[12] H.S. Hou, D.R. Tretter, M.J. Vogel, Interesting properties of the discrete cosine transform, J. Visual Commun. Image Representat. 3 (1) (1992) 73–83.
[13] M. Shneier, M. Abdel-Mottaleb, Exploiting the JPEG compression scheme for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 849–853.
[14] B.L. Yeo, B. Liu, On the extraction of DC sequence from MPEG compressed video, IEEE International Conference on Image Processing, Vol. 2, 1995, pp. 260–263.
[15] Y. Ariki, Y. Saito, Extraction of TV news articles based on scene cut detection using DCT clustering, Proceedings of the International Conference on Image Processing, Vol. 3, 1996, pp. 847–850.
[16] K.R. Rao, P. Yip, Discrete Cosine Transform: Algorithm, Advantage, Applications, The Univeristy of Texas, Academic Press, New York, 1990.
[17] ftp.uu.net/graphics/jpeg/jpegsrc.v6a.tar.gz, The Independent JPEG Group's JPEG software.
[18] J.R. Smith, Integrated spatial and feature image systems: retrieval, analysis and compression, Ph.D Thesis, Columbia University, 1997 (Chapter 2).
[19] J.D. Foley et al., Computer Graphics Principles And Practice, Addison-Wesley, Reading, MA, 1990.
[20] www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html, Vision Texture.
[21] M.J. Swain, D.H. Ballard, Color indexing, Int. J. Comput. Vision 7 (1) (1991) 11–32.
[22] C.-C. Chen, Improved moment invariants for shape discrimination, Pattern Recognition 26 (5) (1993) 683–686.
[23] M.K. Hu, Visual pattern recognition by moment invariants, IEEE Trans. Inform. Theory 12 (1962) 179–187.
[24] S. Sclaroff, Deformable prototypes for encoding shape categories in image databases, Pattern Recognition 30 (4) (1997) 627–641.
[25] C. Faloutsos, R. Barber, M. Flickernew, J. Hafner, W. Niblack, D. Petkovic, W. Equitz, Efficient and effective querying by image content, J. Intell. Inform. Systems 3 (1994) 231–262.
[26] H.S. Ip, D. Shen, W. Wong, K.C. Law, An area-based shape representation for affine invariant content-based retrieval, IAPR International Workshop on Multimedia Information Analysis and Retrieval, 1998, pp. 132–142.
[27] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.
[28] T. Chang, C.-C.J. Kuo, Texture analysis and classification with tree-structured wavelet transform, IEEE Trans. Image Process. 2 (4) (1993) 429–441.
[29] J. Mao, A.K. Jain, Texture classification and segmentation using multiresolution simultaneous autoregressive models, Pattern Recognition 25 (2) (1992) 173–188.
[30] H. Tamura, S. Mori, T. Yamawaki, Textural features corresponding to visual perception, IEEE Trans. System Man Cybernet. 8 (6) (1978) 460–473.
[31] P. Brodatz, Textures: A photographic album for artists and Designers, Dover, New York, 1966.

About the Author—CHONG-WAH NGO received the B.S. degree with honors in 1994 and the M.S. degree in 1996, in Computer Engineering from Nanyang Technological University, Singapore. He is currently a Ph.D. Student in the Hong Kong University of Science and Technology. He was with Information Technology Institute, Singapore, in 1996, and was with Microsoft Research China as a summer intern from July to October in 1999. His current research interests include image and video indexing, computer vision, and pattern recognition.

About the Author—TING-CHUEN PONG received his Ph.D. in Computer Science from Virginia Polytechnic Institute and State University in 1984. In 1991, Dr. Pong joined the Hong Kong University of Science and Technology, where he is currently a Reader of

Computer Science, Director of the Sino Software Research Institute, and Associate Dean of Engineering. Before joining HKUST, he was an Associate Professor in Computer Science at the University of Minnesota, Minneapolis. Dr. Pong is a recipient of the Annual Pattern Recognition Society Award in 1990 and Honorable Mention Award in 1986. He has served as Program Co-Chair of the Third International Computer Science Conference in 1995 and the Third Asian Conference on Computer Vision in 1998. He is currently on the Editorial Board of the Pattern Recognition Journal.

About the Author—ROLAND T. CHIN received the B.S. degree with honors in 1975 and the Ph.D. degree in 1979, in electrical engineering from the University of Missouri, Columbia. From 1979 to 1981, he worked on remote sensing at NASA Goddard Space Flight Center, Maryland. He was on the faculty of Electrical and Computer Engineering at the University of Wisconsin, Madison, from 1981 to 1995; served as Associate Department Chair from 1986 to 1990. Since 1992, he has been on the faculty of the Computer Science Department of Hong Kong University of Science & Technology, and is Department Head of the Computer Science since 1996. Professor Chin was the recipient of the NSF Presidential Young Investigator Award in 1984. He has served on the editorial board of Asian Pacific Engineering Journal, the IEEE Transactions on Image Processing, and the IEEE Transactions on Pattern Analysis and Machine Intelligence.