A Block-Edge-Pattern based Content Descriptor in DCT Domain

Jianmin Jiang, Kaijin Qiu and Guoqiang Xiao

Abstract—**In this correspondence, we describe a robust and effective content descriptor based on block edge patterns extracted directly in DCT domain, which is suitable for applications in JPEG or MPEG compressed images and videos. This content descriptor is constructed by a run-length edge-block histogram with three patterns including horizontal edge, vertical edge and no-edge. In comparison with existing descriptors, the proposed features: (i) low-cost computing suitable for real-time implementation and high-speed processing of compressed images or videos; (ii) robust to orientation changes such as rotation, noise, reverse etc. (iii) directly operates in compressed domain. Extensive experiments support that the proposed content descriptor is effective in describing visual content. In comparison with existing techniques, the proposed descriptor achieves superior performances in terms of retrieval precision and recall rates.**

Indexing Terms — **Content descriptors, MPEG, Compressed domain video analysis.**

I. INTRODUCTION

Since many images and videos distributed or stored inside computers are now represented in compressed formats at the source, research on compressed domain image or video processing becomes increasingly important. Representative work can be summarized as follows. S. F. Chang [1] computed the statistical measures of DCT coefficients to form a texture feature. Chong-Wah Ngo, Ting-chuen Pong [2] described an image-indexing algorithm via reorganization of DCT coefficients and representation of color, shape and texture features in compressed domain. G.C. Feng and J. Jiang [3] extracted mean and standard deviation as a set of statistical features directly from DCT coefficients to retrieve JPEG compressed images. Sim, Kim [4] reported fast texture description and retrieval of DCT-based compressed images. J. Liu , H. Gu [5] developed a method, which can retrieve images from different kinds of domains by using wavelet coefficients. In [6], Coimbra and Davies proposed a fast implementation of optical flow estimation in MPEG-2 compressed domain, where the DCT coefficients, AC[1] and AC[8], are used to approximate a confidence matrix, leading to a fast estimation of optical flow via LK algorithm.

Extensive research on image processing for the past decades illustrate that edge is a strong feature for characterizing visual content. While many methods as highlighted above extract features in DCT domain, most techniques reported in the literature extract edge features in pixel domain, including those widely reported for image segmentations [7]. Recent trend on image processing is characterized by detecting edges and classifying their patterns at a block level either in pixel domain or compressed domain [8,9]. Kim and Lee [10] found out that the edge location is well captured by the polarities of the projection of DCT coefficients. Soltane et al.[7] suggested an adaptive edge operator selection scheme for image segmentation based on the mean, variance, and entropy of DCT coefficients. Shen and Sethi [11] further proposed a method to determine edge strength and orientation through pattern analysis of DCT coefficients. The latest work on edge extraction reported by Chang et al [12] proposed a fast and systematic scheme to classify the edge orientation of each block in DCT domain. In total, five directional edge patterns (no edge, 0-directional edge, π/4-directional edge, π/2-directional edge, and 3π/4-directional edge) were proposed in their paper. Their algorithm is similar to the one proposed in [13].

Based on the analysis of the existing work as described above, we propose a block-edge-pattern based content descriptor, in which only three block-edge-patterns, no-edge, horizontal-edge, and vertical edge, are extracted directly inside DCT domain. In comparison with the existing work, especially the major reference [12], our contributions and their novelty can be highlighted as: (i) a new run-length histogram to exploit the strength of the multi-dimensional histograms across all dimensions, yet avoid the high dimensional problems with conventional histograms; (ii) extraction of three edge patterns rather than five to further improve the processing speed and save computing cost without compromising the performance of content description.

The rest of the article is organized into four further sections, where Section II briefly revisit the work reported in [12] and present our argument that extraction of three edge patterns rather than five is sufficient for the proposed content descriptor, Section III describes our proposed run-length histogram to construct the content descriptor. Finally, experimental

J. Jiang is with the Faculty of Computer and Information Science, Southwest University, China; and the School of Informatics, University of Bradford, UK. Phone: +44 1274 233695; fax: 44-1274-233727; e-mail: j.jiang1@ Bradford.ac.uk)

K. Qiu and G. Xiao are with the Faculty of Computer and Information Science, Southwest University, Chongqing, 400715, China. (e-mail: qkjqkj@ swu.edu.cn; gqxiao@swu.edu.cn).

results and their analysis are given in Section IV to support the proposed design, and conclusions are given in Section V.

II. EXTRACTION OF BLOCK EDGE PATTERNS IN DCT DOMAIN

In 2005, Chang et al proposed a technique of extracting 5 edge patterns from DCT coefficients in terms of 8×8 block of pixels [12]. This technique operates in compressed domain by just analyzing the value of the first few DCT coefficients and thus only limited decoding operations are required instead of full decompression. Such block edge pattern classification is based on the principle that, when a block of 8×8 pixels is divided into four sub_blocks of 4×4 pixels, comparison of their average pixel value can be used to decide whether there exists an edge pattern inside the block or not. As an example, if the average pixel of the two sub_blocks on the left is significantly larger than the average pixel of the two sub_blocks on the right or vice versa, the block can be classified as a vertical edge pattern block. While the principle has been applied by many other researchers [13], the contribution of [12] specifies that such edge pattern classification can be carried out by extracting edge measure values in DCT domain, which are summarized in Table I, where *X(1,0), X(0,1)* and *X(1,1)* are the first three DCT coefficients (excluding the DC) along the zig-zag scanning order.

While it is much simpler to calculate the edge measure values in DCT domain, it still requires four additions and two multiplications to calculate $\delta_{\frac{\pi}{4}}$ and $\delta_{\frac{3\pi}{4}}$, respectively. Yet if we only use three edge patterns, as seen from Table I, significant savings on computing cost could be achieved since calculation of δ_0 and $\delta_{\frac{\pi}{2}}$ does not incur any addition or multiplication and only two DCT coefficients instead of three are needed. In addition, our further observation and analysis on its content application verifies that image edge distribution can be characterized by three edge patterns instead of five. In other words, we only need the edge patterns of no edge, vertical edge, and horizontal edge to formulate their content applications without compromising the quality of visual content description. This is because that an arbitrary edge inside an image can be generally approximated by combinations of vertical and horizontal edges. To support the argument, we reconstruct an edge pattern image based on the extracted edge patterns and then apply a distortion measure (DM) by following the spirit of PSNR (peak-signal-to-noise-ratio) with respect to the original image [17], where the pixel value of the reconstructed edge image is defined as the corresponding average pixel values inside the sub-block, which are used to determine the block edge patterns. Summary of all the DM values for six popular image samples is provided in Table II.

Table II indicates that the three edge pattern images only incur very small difference in PSNR values in comparison with that of five edge pattern images. Such measurements support that, when the edge pattern extraction technique reported in [12] is further developed to construct content descriptors, it could be sufficient to only use the vertical edge pattern and the horizontal edge pattern.

Table II Distortion Measure of Edge-Pattern Images against their Source Image

Image Samples	PR_5 (dB)	PR_3 (dB)
Lena	59．439892	59．180862
Peppers	57．447807	57．687534
Clown	50．144257	50．109631
Camera	48．590816	48．425625
Boat	55．761734	56．058308
Bridge	47．100334	47．334343

Table I Measure for Five Edge Patterns

Edges	Measure values
No edge	λ (set by user, $\lambda = 50$ in our design)
0	$\delta_0 = \lvert X(1,0) \rvert$
$\pi/4$	$\delta_{\frac{\pi}{4}} = \frac{3}{4}\max\left\{\left\lvert\frac{1}{2}(X(0,1)+X(1,0)+X(1,1))\right\rvert, \left\lvert\frac{1}{2}(X(0,1)+X(1,0)-X(1,1))\right\rvert\right\}$
$\pi/2$	$\delta_{\pi/2} = \lvert X(0,1) \rvert$
$3\pi/4$	$\delta_{\frac{3\pi}{4}} = \frac{3}{4}\max\left\{\left\lvert\frac{1}{2}(X(0,1)-X(1,0)+X(1,1))\right\rvert, \left\lvert\frac{1}{2}(X(0,1)-X(1,0)-X(1,1))\right\rvert\right\}$

Correspondingly, the three block edge patterns can be classified by:

$$block\_edge\_pattern = \begin{cases} no\_edge \quad if \ \max(\delta_0, \delta_{\pi/2}) < \lambda \\ vertical\_edge \quad if \ \delta_{\pi/2} \geq \delta_0 \\ horizontal\_edge \quad otherwise \end{cases} \tag{1}$$

However, with five block edge patterns, their classification is done by:

$$edge\_pattern = \begin{cases} no\_edge \quad if \ \max\left(\delta_0, \delta_{\frac{\pi}{4}}, \delta_{\frac{\pi}{2}}, \delta_{\frac{3\pi}{4}}\right) < \lambda \\ 0\_edge \quad if \ \delta_0 = \max\left(\delta_{\frac{\pi}{4}}, \delta_{\frac{\pi}{2}}, \delta_{\frac{3\pi}{4}}\right) \\ \frac{\pi}{4}\_edge \quad if \ \delta_{\frac{\pi}{4}} = \max\left(\delta_0, \delta_{\frac{\pi}{2}}, \delta_{\frac{3\pi}{4}}\right) \\ \frac{\pi}{2}\_edge \quad if \ \delta_{\frac{\pi}{2}} = \max\left(\delta_0, \delta_{\frac{\pi}{4}}, \delta_{\frac{3\pi}{4}}\right) \\ \frac{3\pi}{4}\_edge \quad if \ \delta_{\frac{3\pi}{4}} = \max\left(\delta_0, \delta_{\frac{\pi}{4}}, \delta_{\frac{\pi}{2}}\right) \end{cases} \tag{2}$$

As seen from (1) and (2), the 3 edge pattern classification would not only achieve computing cost savings on edge measure calculation, but also on edge pattern classifications.

III. CONTENT DESCRIPTOR DESIGN

In this section, we follow the general route to design a content descriptor based on histograms for the edge block patterns extracted [18]. Given blocks of 8×8 pixels, their classified edge patterns are firstly transformed into a set of three elements, $\psi = \{\#, -, |\}$, where # = no-edge, - = horizontal edge and | = vertical edge. When a histogram is constructed by calculating the number of occurrences for each edge pattern, the histogram will only have three elements, which is often referred to 1D edge-block-histogram (EBH). As the number of bins in such 1D edge-block-histogram is too small to characterize the image content, the occurrence of states can be counted instead by considering combination of edge patterns, leading to construction of high dimensional histograms. As an example, if we examine two edge patterns together as one state, the total number of elements inside the histogram can be increased to 9 (a 2D EBH). When *N* edge patterns are considered together as one state, the resulting histogram will have 3^N bins or elements.

Low dimensional EBH not only have small number of elements, but also contain little information for the location of those classified edge patterns inside the image, which are often useful to reflect the visual content. On the other hand, 5D EBH and 6D EBH incurs large number of states, which illustrate significant potential for high computing cost, especially when millions of compressed images need to be addressed via such content descriptors. To achieve an appropriate balance, we propose a run-length histogram following the spirit of run-length coding in the area of lossless data compression. Specifically, we raft scan all block edge patterns on both row and column basis to count the number of same patterns inside one single run. Every time a different edge block pattern is encountered, it is regarded as a broken run and thus a new run will be started. The output code for each run can be represented as:

$$R_i = \{\alpha_i, \eta_i\} \tag{3}$$

Where $\alpha_i \in [\#, -, |]$, and η_i stands for the number of α_i inside this single run.

R_i essentially represents the state of run-length EBH. To reduce the statistical modeling cost [14], we group those long run-length states together into one single state:

$$R_i = \begin{cases} \{\alpha_i, \eta_i\} & if \ \eta_i \leq T \\ \{\alpha_i, T\} & when \ \eta_i > T \end{cases} \tag{4}$$

In our implementation, we selected *T=4*. As a result, the histogram can be represented as follows:

$$H_{run} = \frac{\{C_1, C_2, \ldots C_M\}}{M} \tag{5}$$

Where C_i counts the number of the state R_i occurred inside the image after the run-length coding is finished, *M* is 28, including 12 states in run-length on row-based scanning, 12 states in run-length on column-based scanning, and two special states "_|" and "|_ on both row and column basis.

The run-length histogram given in (5) not only resolves the problem that histograms with high dimension increase their size in exponential order, but also provides non-redundant statistical analysis of the edge pattern distribution. This is achieved by the nature of run-length, where same edge patterns are grouped together to formulate states rather than individually fixed patterns. When the histogram is constructed by such states, additional advantage can be gained as such that grouped edge patterns are capable of describing connected edges rather than individual patterns. Such connectivity is described by components of: (i) horizontal run-length; (ii) vertical run-length, and (iii) individual run-length (run-length=1). In this way, the proposed run-length histogram not only describes the visual content in terms of edge pattern distribution, but also in terms of these connected edges, providing further potential information for shape, contour or internal texture of visual objects. To this end, the proposed content description is different from those edge-based descriptors [8,9], where the technique proposed in [8] relies on Canny edge detector to extract salient points and connect them into edge curves via a water-filling filter, and the technique in [9] extracts edge map to formulate global features through fuzzy compactness and combination of shape information for image retrieval.

In addition, the proposed run-length histogram is capable of preserving the strength of conventional histograms, as well as characterizing the connectivity of edges and the continuity of regional edge patterns. As seen in equation (5), the first three states with a run of 1 single edge pattern are exactly the same as those elements inside one dimensional histogram. These parts of run-length histogram will make the same contribution in describing the visual content as 1D conventional histogram. Further on, the states with 1 run of 2 edge patterns will function in the same principle as that of 2D conventional histogram. The only difference is that among all the possible different states inside conventional 2D histograms, only the states, where the two edge patterns are the same, are selected to construct bins inside the run-length histogram to describe the visual content. Such selection is based on the ground that: (i) due to the fact that the content descriptor is mainly based on edges, their connectivity and continuity is more important than others in describing the visual content; (ii) other states with different edge patterns are less likely to make useful contributions since most of them will be rejected in practical image content processing. One typical example is image segmentation, where detected edges need to be further processed to form elements of meaningful segmentations such as contours etc.; (iii) the role of such states with two different edge patterns is partially played by the previous three states with 1 single edge pattern. For other states with 3 different edge patterns or more, the principle is the same. Therefore, the proposed run-length histogram is essentially extracting those useful states from conventional histograms across all dimensions and consequently combining all their strengths together to describe the visual content, yet such a design overcomes the problem that high dimensional histograms generate high number of bins inside the histogram, making it computing intensive and non-manageable.

Finally, to describe the edge connectivity in two dimensions, the run-length histogram is constructed by scanning along rows as well as columns, which is commonly used in processing two dimensional signals (such as DCT). In this way, the edge connectivity can be described in two dimensions rather than in one dimension. Therefore, global connectivity such as diagonal edge connectivity will be described by collective contributions from both row-based edge connectivity and column-based edge connectivity.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

To evaluate the proposed content descriptor, we design two phases of experiments. The first phase is to use such a content descriptor to run content-based image retrieval and measure its effectiveness in terms of retrieval precision. For this purpose, a database of 14787 JPEG compressed images is established, all of which are manually classified into 10 categories in order to provide a ground truth for measuring the performances. The 10 categories include cars, flowers, bike, building, animal, people, mountains, sky, sunset, and textured. The second phase is to subject the randomly selected images inside the database to a range of orientation changes and then add them into the database to see how many such changed images can be retrieved when the original image is taken as a query. This is designed to test the robustness of the proposed content descriptors.

To provide a comparison with the proposed content descriptor, we select two existing algorithms [3,15] as our benchmark to compare the performances on the principle that the existing techniques are comparable and the relevant comparisons are fair in terms of computing cost and algorithm complexity. The first benchmarking technique [3] operates in compressed domain via direct extraction of mean and variance out of each block with 8×8 pixels. A histogram is then constructed from the space of mean and standard deviation by quantizing the mean into four regions and the standard deviation into seven regions. Therefore, this technique provides good comparability that it operates in compressed domain and constructs the content descriptor based on histograms. The second benchmark was drawn from an existing PicToSeek CBIR system [15], where color features invariant to viewpoints were described in pixel domain and the indexing key was constructed via histograms. The invariant color space is a direct result of the RGB space transformation, which is specified as follows.

Table III Summary of the first phase experiments measured in precision rates

Image Categories	Benchmark-1	Benchmark-2	RLEBH(3)	RLEBH(5)
Cars	0.42	0.41	0.92	0.55
Flowers	0.41	0.60	0.67	0.79
Bike	0.62	0.33	0.60	0.51
Building	0.38	0.29	0.45	0.73
Animal	0.30	0.28	0.42	0.41
People	0.62	0.34	0.82	0.27
Mountains	0.28	0.31	0.43	0.43
Sky	0.39	0.53	0.47	0.40
Sunset	0.50	0.58	0.53	0.52
Textured	0.81	0.27	0.61	0.65
Average	0.47	0.39	0.59	0.53

$$\{c_4, c_5, c_6\} = \begin{cases} c_4\{RGB\} = \dfrac{R-G}{R+G} \\ c_5\{RGB\} = \dfrac{R-B}{R+B} \\ c_6\{RGB\} = \dfrac{G-B}{G+B} \end{cases} \tag{6}$$

The histogram-based indexing key is constructed as follows after the invariant color space is quantized into 4 states indexed by k:

$$H(c_4 c_5 c_6) = \frac{\eta\{(c_4 \in k) \cap (c_5 \in k) \cap (c_6 \in k)\}}{W \times H} \qquad k \in [0,3] \tag{7}$$

where $\eta(.)$ represents a counting operation on the number of state occurrences.

To calculate the similarity between histograms, we adopt the simple L1 distance, which is chosen on the ground that: (i) it is widely used by the CBIR community, such as the two benchmarks [3,15]. (ii) it has lower computing cost than standard Euclidean distance, yet preserves its strength in similarity measurement.

For the first phase, we randomly choose 200 images from the database, 20 out of each category, as the query images to search the database. The retrieval performance was measured in terms of precision, which is the number of retrieved images deemed to be relevant divided by the first N retrieved images. In our experiments, we designed N to be 12 in line with the benchmarks [3,15] and other references [8,9]. All results in the first phase are summarized in Table III, where RLEBH(3) refers to the proposed run-length histogram with 3 edge block patterns and RLEBH(5) with 5 edge block patterns.

From the experimental results given in Table III, it can be seen that: (i) the proposed RLEBH(3) achieves the best performance among all the algorithms tested in terms of the total-average precision measurement; (ii) While the proposed run-length histogram significantly outperforms the first benchmark across most categories, it is inferior to the first benchmark for both the category of Bike and Textured. This is because that the first benchmark descriptor is strongly texture oriented, where the mean and the standard deviation extracted from DCT domain collectively captures the texture information quite well. Yet all images under this category are overwhelmingly texture dominant, and thus the precision rate achieved by benchmark-1 is better than RLEBH. However, for other categories with strong texture orientation, the proposed RLEBH remains competitive. Even for the category Bike, the benchmark-1 only slightly outperforms the proposed. (iii) In comparison with the second benchmark that is purely colour-based descriptor, the proposed algorithm is only beaten in color dominant categories, such as Sunset and Sky. Within these two categories, colour feature dominates the image content. For other categories, the proposed run-length histogram provides superior performances on the ground that it provides additional advantage in capturing connected edge information towards the combination of multiple feature strength, including texture of gradients, shape, contour and edges of visual objects and thus it performs well except for those colour dominated images, for which the second benchmark is well tailored; (iv) in comparison with 5 edge patterns, selection of 3 edge patterns achieves better performance in six categories, maintains the same performance in one category, slightly inferior in 2 categories and inferior in 1 category, where the category of building is edge-dominant. But REBH(3) is still better than REBH(5) in terms of the overall performance, which verifies that selection of three edge patterns is sufficient for content based applications.

The second phase of experiments aims at evaluating the robustness of the proposed content descriptor in retrieving images with orientation changes, such as rotation, zoom in or out, and noisy effects etc. In this experiment, fifty images were chosen at random from

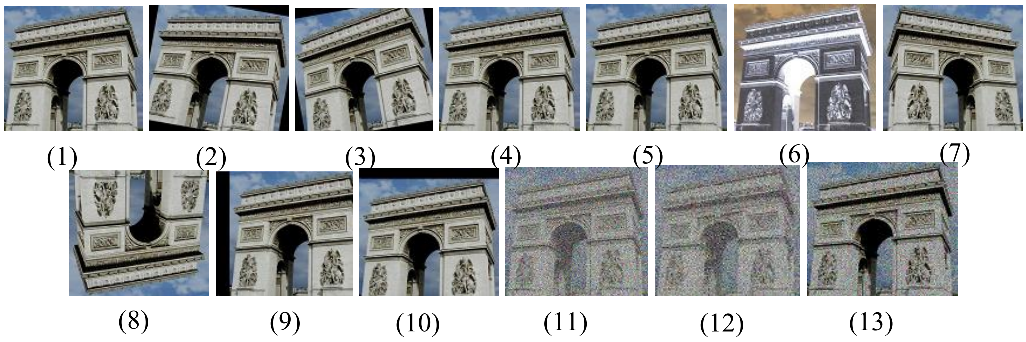

Fig. 1. Illustration of orientation changed images, where (1) original image; (2) clockwise rotated by $10°$; (3) anti-clockwise rotated by $10°$; (4) zoomed in by 20%; (5) zoomed out by 20%; (6) inverse coloured; (7) horizontally orientated; (8) vertically orientated; (9) horizontal displaced; (10) vertically displaced; (11) Gaussian white noise added; (12) salt & pepper noise added; (13) peckle noise added.

Table IV Summary of second phase experimental results measured by recall rates

Methods	Benchmark-1	RL EBH(3)	RLEBH(5)
Average recall after clockwise rotate by 10^{o}	0.78	0.50	0．48
Average recall after anti-clockwise rotate by 10^{o}	0.74	0.52	0．40
Average recall after zoom out 20%	0.96	0.72	0．76
Average recall after zoom in 20%	0.94	0.90	0．92
Average recall after inverse color	0.02	1.00	1．00
Average recall after horizontal upset	0.92	0.98	0．68
Average recall after vertical upset	0.92	0.94	0．88
Average recall after horizontal displacement 10%	0.82	0.50	0．48
Average recall after vertical displacement 10%	0.86	0.52	0．50
Average recall after adding Gaussian white noise	0.02	0.90	0．94
Average recall after adding salt& pepper noise	0.02	0.42	0．46
Average recall after adding speckle noise	0.08	0.60	0．66
Total average recall	0.59	0.71	0．68
ANMRR	0.1	0.08	0．09

the database. Specifically, each of the chosen images was processed as follows: clockwise rotation by 10 degrees, anti-clockwise rotation by 10 degrees, zoom out by 20%, zoom in by 20%, inverse color, horizontal reverse, vertical reverse, horizontal displacement by 10%, vertical displacement by 10%, adding Gaussian white noise (means equal to 0, variance equal to 0.04), adding salt & pepper noise (variance equal to 0.04) and adding speckle noise (variance equal to 0.04). Figure 1 illustrates examples of all the orientation changes. Following that, all such processed images are added into the database. When one of the originals is taken as a query image, we would like to see how many of those processed images can be retrieved to test the robustness of the proposed content descriptor. To this end, we measure the retrieved results in terms of the recall rate and the average of the normalized modified retrieval rank (ANMRR) over all queries as described in [16].

All the experimental results produced in the second phase are summarized in Table IV, which indicates that the proposed content descriptor with both 3 and 5 edge patterns is more robust than the first benchmark in terms of the total average recall rate, and achieves superior performances measured by ANMRR values. Between the 3 edge pattern and 5 edge pattern selections, RLEBH(3) still achieves better performance in terms of the overall recall rate, although there exist some places that RLEBH(3) is inferior than RLEBH(5).

The robustness of the proposed descriptor can be explained by: (i) the edge pattern is block based and extracted in compressed domain. Therefore, any regional corruption inside image will be measured in terms of blocks rather than pixels. While small region corruption is measured big in terms of pixels, it is much smaller or even ignorable in terms of blocks, as one block contains 64 pixels; (ii) corrupted pixels inside the same block may compensate with each other due to the properties of DCT transform and the fact that edge pattern is extracted in terms of sub-blocks with 16 pixels in total. In other words, the corrupted pixels may not necessarily lead to wrong edge pattern extraction.

V. CONCLUSIONS

In this paper, we proposed an effective content descriptor for both JPEG compressed images and MPEG compressed videos. In comparison with existing techniques, the proposed algorithm illustrates the following advantages and features: (i) robust to a range of orientation changes which are frequently encountered during transmission, processing, editing and storages; (ii) directly operates in compressed domain, which are suitable for processing large compressed image or video databases; (iii) extremely low computing cost with RLEBH(3) suitable for real-time implementation and fast image or video content management applications, which can be seen in Table-I and equation (1). Finally, extensive experiments also support that the proposed algorithm outperforms similar counterparts in terms of retrieval precision and recall rates. Therefore, the proposed algorithm would have significant potential for practical applications either as a standalone tool or as a component working with other tools towards efficient and effective visual content management and descriptions.

In addition, further research could improve the proposed algorithm, where run-length of edge pattern states rather than individual edge patterns could be considered. One example of such edge pattern state could be constructed as joining edges with connected vertical and horizontal patterns across more rows or columns. Other possibilities could include some popular elements inside contours, texture, or shapes etc. Finally, the authors wish to acknowledge the financial support from the EU IST FP6 Programme under the IP project: Live staging of media events (Contract No IST-4-027312).

REFERENCES

[1] S.F. Chang, "Compressed domain techniques for image/video indexing and manipulation", IEEE International Conference on Image Processing, pp.314-317, 1995.

[2] C.W. Ngo, T.C. Pong, "Exploiting image indexing techniques in DCT domain", Pattern Recongnition 34 pp.1841-1851, 2001.

[3] G. Feng, J. Jiang, "JPEG compressed image retrieval via statistical features", Pattern Recognition 36, pp.977-985, 2003.

[4] D.G. Sim, H.K. Kim, R.H. Park, "Fast texture description and retrieval of DCT-based compressed image", Electronic Letters, 37(1), pp.18-19, 2001.

[5] J.H. Liu, H.M. Gu, "Image retrieval in various domains", Computers & Graphics 27, pp.807-812, 2003.

[6] M. Coimbra, and M. Davies, "Approximating optical flow within the MPEG-2 compressed domain", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 15, No. 1, pp. 103-107, Jan. 2005.

[7] 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, pp.242-251, 1996.

[8] J.W. Han, L. Guo, "A shape-based image retrieval method using salient edges, Signal processing", Image communication 18, pp.141-156, 2003.

[9] M. Banerjee, M.K. Kundu, "Edge based features for content based image retrieval", Pattern Recognition 36, pp.2649-2661, Nov. 2003.

[10] D.S. Kim and S.U. Lee, "Image vector quantizer based on a classification in the DCT domain", IEEE Trans. Commun, Vol. 39, No.4, pp.549-556, Apr.1991.

[11] B. Shen and I.K. Sethi, "Direct feature extraction from compressed images", Proc. SPIE: Storage and Retrieval for Still Image and Video Databases IV, Vol.2670, pp.404-414, Mar.1996.

[12] H.S. Chang, K. Kang, "A compressed domain scheme for classifying block edge patterns", IEEE Transactions on image processing, Vol.14, No.2, pp 145-151, Feb. 2005.

[13] H. Li, G. Liu, and Y. Li, "An effective approach to edge classification from DCT domain", Proc. IEEE Int. Conf. Image Processing, pp.940-943, Sep.2002.

[14] J. Jiang, "A low cost content adaptive and rate controllable near lossless image codec in DPCM domain", IEEE Trans. On Image Processing, Vol 9, No. 4, pp543-554;

[15] G. Theo, W.M. Arnold, PicToSeek, "combining color and shape invariant features for image retrieval", IEEE Transactions on Image Processing, Vol 9, No 1, pp102-119, 2000;

[16] H.J. Bartsch Handbook of mathematical formulas, Academic press, 1974.

[17] I. Popovici and W.D. Withers, "Custom-built moments for edge location", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, No. 4, pp. 637-642, April 2006.

[18] K. Qiu, J. Jiang, G. Xiao, "An edge based content descriptor for content based image and video indexing", Lecture Notes in Computer Science, Image Analysis and Recognition, Springer, Vol 4141, No 1, pp673-684, 2006;

Jianmin Jiang (M'76–SM'81–F'87) received B.Sc degree from Shandong Mining Institute, China, in 1982, M.Sc degree from China University of Mining and Technology in 1984, and PhD from the University of Nottingham, UK, in 1994.

From 1985 to 1989, he was a lecturer at Jiangxi University of Technology, China. In 1989, he joined Loughborough University, UK, as a visiting scholar and later moved to the University of Nottingham as a research assistant. In 1992, he was appointed a lecturer of electronics at Bolton Institute, UK, and moved back to Loughborough University in 1995 as a lecturer of computer science. In 1997, he was appointed as a full professor at the School of Computing, University of Glamorgan, Pontypridd, UK. He joined University of Bradford in 2002 as a professor of Digital Media at the School of Informatics, University of Bradford, UK. In 2004, he was appointed as an adjunct professor at the Southwest China University, Chongqing, China. He is a fellow of IEE and fellow of RSA in the UK. His research interests include visual information retrieval, image/video processing, visual content management, Internet video coding, stereo image coding and neural network applications. He has published more than 180 refereed research papers.

Kaijin Qiu received B.Sc degree from Southwest University of Science and Technology, China, in 1994, M.Sc degree from Southwest University, Chongqing, China, in 2007. Since 2007, he has been with the Faculty of Computer and Information Science, Southwest University, Chongqing, China. His research interests include image analysis and processing, pattern recognition, and video compression techniques

Guoqiang Xiao received the Ph.D. degree in signal and information processing from University of Electronic Science and Technology of China, Chengdu, and B.E. degree in radio technology from Chongqing University, Chongqing, China, in 1999 and 1986, respectively. Since 1986, he has been with the Faculty of Computer and Information Science, Southwest University, Chongqing, China, where he is currently a Professor. From 2001 to 2003, he was with Department of Electrical & Electronic Engineering, the University of Hong kong, as a postdoctor. His research interests include image processing, pattern recognition, neural networks, and wireless network communication.