

Segmentation using Codebook Index Statistics for Vector Quantized Images

Hsuan T. Chang* and Jian-Tein Su
Photonics and Information Laboratory
Department of Electrical Engineering
National Yunlin University of Science and Technology
Douliu Yunlin, Taiwan

Abstract—In this paper, the segmentation using codebook index statistics (SUCIS) method is proposed for vector-quantized images. Three different codebooks are constructed according to the statistical characteristics (mean, variance, and gradient) of the codewords. Then they are employed to generate three different index images, which can be used to analyze the image contents including the homogeneous, edge, and texture blocks. An adaptive thresholding method is proposed to assign all image blocks in the compressed image to several disjoint regions with different characteristics. In order to make the segmentation result more accurate, two post-processing methods: the region merging and boundary smoothing schemes, are proposed. Finally, the pixel-wise segmentation result can be obtained by partitioning the image blocks at the single-pixel level. Experimental results demonstrate the effectiveness of the proposed SUCIS method on image segmentation, especially for the applications on object extraction.

Keywords—image segmentation; vector quantization; index image; adaptive thresholding; codebook statistics

I. INTRODUCTION

Vector quantization (VQ) has been a simple, efficient and attractive image compression scheme since the past three decades [1], [2]. The basic VQ scheme partitions an image into small blocks (vectors) and each vector is assigned by an index of the codeword in the codebook for encoding [3]. If the indices of the codewords are arranged in a specific form, they can effectively represent the image characteristics in terms of image blocks. For example, Yeh proposed a content-based image retrieval algorithm for VQ images by analyzing the indices of codewords [4]. Moreover, the correlations among these indices can be used to develop an efficient image segmentation scheme. Most of previous studies performed segmentation operations in the pixel level [5], [6]. However, block-based image segmentation schemes can provide several advantages [7], [8]. The block-based scheme is very suitable for the VQ-based image segmentation because VQ techniques usually divide an image into non-overlapping image blocks. Block-based segmentation schemes segment an image with blocks rather than pixels by considering the relative information of the neighboring image blocks. Large homogeneous regions can be detected by means of this technique. Moreover, the computational complexity can be saved by segmenting an image in the block level rather than in the pixel level.

In this paper, the segmentation using codeword index statistics (SUCIS) method is proposed for the vector quantized images. First, an image of size $N \times N$ is divided into non-overlapping image blocks of size $B \times B$. A training set of images is first used to generate an image codebook based on the well-known Linde-Buzo-Gray (LBG) algorithm [1]. The codewords in the image codebook are then sorted in an incremental order according to their mean, variance, and gradient to obtain three different codebooks. For a vector quantized image, the corresponding index from the sorting results is treated as a pixel's grayscale value to obtain the index image. Therefore, three corresponding index images can be obtained. An index image intrinsically demonstrate the correlations among image blocks, which can be used as the features to label image blocks into homogeneous, edge, or texture blocks. Several image-dependent threshold values are adaptively determined for region labeling. Then, a region number is assigned to all connected homogeneous blocks with the same label to obtain disjoint regions. Finally, the pixel-based image segmentation result can be obtained by partitioning the size of edge blocks further into the pixel level.

The rest of this paper is organized as follows: Section 2 deals with the proposed SUCIS method for VQ-coded images. Section 3 shows the experimental results for various test images. Conclusions are finally drawn in Section 4.

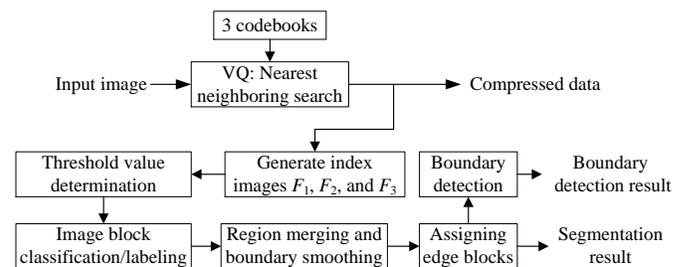


Fig. 1. The schematic diagram of the proposed SUCIS method

II. THE PROPOSED METHOD

Figure 1 shows the block diagram of the overall image segmentation system by using the proposed SUCIS method. The overall system consists of six stages: (1) codebook generation, (2) VQ encoding, (3) index image generation, (4) region labeling, merging, and boundary smoothing, (5)

boundary detection, and (6) edge block assignment for image segmentation. The first and second stages have been well studied in image compression framework based on VQ. The third stage describes how to rearrange the indices of the codewords in the codebook using three different mapping functions. Then, three different index images can be generated and are used to label all image blocks, which are classified into three types: (1) homogeneous blocks, (2) edge blocks, and (3) texture blocks. Figures 2(a) – 2(f) show six typical image blocks in the block labeling configurations. The fourth stage discusses two post-processing methods: the region-merging and boundary-smoothing schemes which can refine the boundaries of regions of special interest. In the region-merging process, many homogeneous regions are merged into certain regions and this achieves the rough image segmentation result. In the boundary-smoothing process, the boundaries of the image segmentation result can be more refined by smoothing the boundaries of regions of special interest. In the fifth stage, the result of boundary detection will be obtained by extracting the edge blocks in the image segmentation result. The final stage describes how to partition and assign the edge blocks from the block level to the pixel level. The segmentation results under the three different pixel levels will be obtained through assigning the edge blocks continuously.

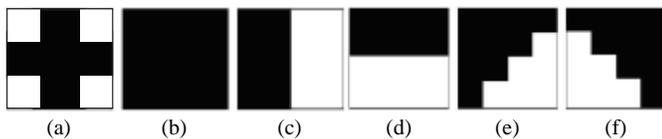


Fig. 2. Some typical image blocks in block labeling: (a) texture block; (b) monotone block; (c) vertical edge block; (d) horizontal edge block; (e) and (f) diagonal edge blocks

A. Codeword Index Assignment and Index Image Generation

VQ is a generalization of scalar quantization to quantize a vector and a mapping of q -dimensional positive Euclidean space R^q into a finite subset C of R^q . Suppose that there are L codewords in the codebook. Let c_i and C denote the codewords and codebook in VQ, respectively, where $C = \{c_i | c_i \in R^q, \text{ for } i = 0, 1, \dots, L-1\}$. The subscript i is the index of the codeword and is represented as a b -bit ($2^b = L$) binary word. It is usually written as a decimal integer for notational brevity.

Figure 3 shows the flowchart of the proposed scheme for codebook generation. If the codeword c_i is considered as a discrete random sequence $X_i = (c_{i,1}, c_{i,2}, \dots, c_{i,L})$ and every random variable $c_{i,k}, k=1, 2, \dots, L$ in X_i is uniform distributed between 0 and 255 (because of the 8-bit resolution in digital images), then the J^{th} moment of the random sequence X_i is expressed by [9]

$$E[(X_i)^J] = \frac{1}{B \times B} \sum_{k=1}^{B \times B} c_{i,k}^J \times p(c_{i,k}), \quad (1)$$

where $p(c_{i,k})$ is the probability density function of the random variable $c_{i,k}$ [7]. The codewords c_i can be sorted in an incremental order based on their J^{th} moment (where $J=1, 2$) to obtain $c_{j(J)}$ where $j(J)$ is the rank of c_i based on its J^{th} moment. From a given image codebook C , two mapping functions between indices i and $j(J)$ for mean $J=1$ and variance ($J=2$) are created. The first mapping function f_1 computes the mean value

of the codewords in the codebook, where f_1 is a linear function. The second mapping function f_2 computes the variance value of the codewords, where f_2 is a nonlinear function. Moreover, the third mapping function f_3 computes the edge density of the codewords. Here, the edge density δ_i represents the texture feature of the i^{th} 4×4 codeword. The edge density is defined as [10]:

$$\delta_i = \frac{|\{c_k | \text{Mag}_{\text{gradient}}(c_i) \geq T\}|}{B^2}, \quad (2)$$

where T denotes the average gradient magnitude of all codewords. First, the mapping function f_1 is a straight line with unit slope because the codewords in the codebook C have been already sorted in the incremental order with mean. Alternatively, the codebooks C_{F1} , C_{F2} , and C_{F3} are directly derived from image codebook C if codewords are sorted according to their mean, variance, and gradient, respectively. The codewords in the three codebooks represent the different meanings. In the first codebook C_{F1} , the indices of the codewords distribute from low mean values to high mean values. In other two codebooks C_{F2} , and C_{F3} , the high indices of the codewords represent the edge and texture blocks in the test image, respectively. We utilize the codewords in the codebook to label the total homogeneous blocks in the test image.

If the index of the codeword is treated as the grayscale value of a pixel, then an $N \times N$ image can be quantized into $N/B \times N/B$ indices and be converted into the index images F_1, F_2 and F_3 of size $N/B \times N/B$ by the use of the codebooks C_{F1}, C_{F2} , and C_{F3} , respectively, based on the three different mapping functions f_1, f_2 and f_3 . For the codebook whose size is larger than 256, to make the indices fall into the range between 1 and 256, the codewords in the codebook must be uniformly divided into 256 classes according to their incremental order. Therefore, numerous codewords can be represented by the same index.

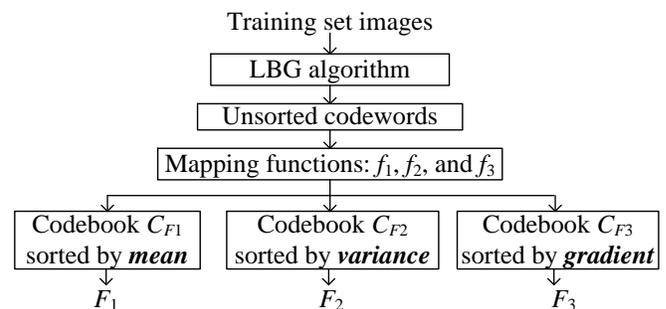


Fig. 3. Flowchart of the image codebook generation

The characteristics of all image blocks in a test image can be classified into three typical image blocks. According to the characteristics in three index images F_1, F_2 and F_3 , the regions and boundary of objects of our interest are extracted. The index image F_1 is a thumbnail version of the original image. It reveals the mean order of image blocks and is used to label all corresponding image blocks. The label of the image block is indeed the corresponding pixel value in the index image F_1 . Both the index images F_2 and F_3 strongly reveal the structural

arrangement (edge and texture information) in the image blocks and are used to determine whether the image blocks belong to the homogeneous blocks or not. The larger values in the index images F_2 and F_3 represent that the gray-level distribution of an image block is closer to the edge or texture block. If the index image F_1 is used to label all image blocks and a region number is assigned to all connected image blocks to obtain disjoint regions, the rough segmentation result can be obtained and can be used as the initial segmentation result for the following pixel-based image segmentation problem.

B. Adaptive Thresholding for Block Labeling

To classify the image blocks in a VQ-coded image, the threshold values are employed. Here, two threshold values T_M and T_F for the mean and fused index images, respectively, are determined at first. The histogram of the index image F_1 is used to determine the initial threshold value T_M . On the other hand, the histogram of fused index image (the fusion of index images F_2 and F_3) is used to determine the initial threshold value T_F . In the image fusion method, the pixel values with the same position in two index images F_2 and F_3 are compared. Then, the larger value in the two pixels is chosen as the pixel value of the fused index image. The image fusion equation is given by

$$F_f(i, j) = \begin{cases} F_2(i, j), & \text{if } F_2(i, j) > F_3(i, j) \\ F_3(i, j), & \text{otherwise} \end{cases} \quad (3)$$

Finally, the fused index image can be created by combining two index images F_2 and F_3 . The overall procedure of the parameter setting for the index image F_1 is as follows:

Step 1: Compute the histogram of index image F_1 and its average value as the initial threshold value T_M which is determined by

$$T_M = \frac{\sum_{i=0}^{255} i \times \text{Pixel number}(i)}{(N/B)^2} \quad (4)$$

Step 2: Determine the accumulated distribution of the histogram by gradually accumulating every five percent of total pixels in the index image.

Step 3: In order to obtain four regions with different grayscale distributions, two more threshold values T_{ML} and T_{MH} are calculated as follows:

$$T_{ML} = \frac{T_M}{2}; \quad T_{MH} = \frac{T_M + 255}{2} \quad (5)$$

Step 4: Calculate the difference values between the three average values T_M , T_{ML} , and T_{MH} and every grayscale level in the accumulated distribution, respectively.

Step 5: Separately update these average values by using the grayscale levels that have the minimum difference among them,

Step 6: Compute the differences between first and last three grayscale levels. Set T_L and T_H are equal to the first and last second grayscale levels, respectively, if the difference is the smallest.

If the pixel values are greater than T_{MH} , these pixel values in the index image F_1 represent the light homogeneous blocks. On the other hand, if they are smaller than T_{ML} , they represent the dark homogeneous blocks. The threshold values T_H and T_L are used to label the lightest and darkest homogeneous blocks, respectively. As mentioned above, we can employ these threshold values to label all homogeneous blocks in a VQ-coded image. After all the homogeneous blocks in a VQ-coded image are labeled, some similar image blocks can be merged into the same region. Thus, we find that many different homogeneous regions spread over a VQ-coded image. Finally, the rough image segmentation result can be obtained.

In order to make the segmentation result more accurate, it must distinguish the edge and texture blocks separately from all image blocks in a test image. The index images F_2 and F_3 are employed to determine which image blocks belong to the edge and texture blocks, respectively. Finally, the edge blocks in the index image F_2 and the texture blocks in the index image F_3 are fused and used to generate a new index image F_f by using Eq. (3). These pixel values in the index image F_f represent the bright texture blocks, if the pixel values are greater than T_F . On the other hand, they represent the dark texture blocks and are smaller than T_F . After all texture blocks in a test image are labeled, some similar image blocks can be merged into the same region. The overall procedure of the parameter setting for the fused index image is similar to that used for the mean index image except that two threshold values T_{MH} and T_{ML} are not used here.

After we remove the labels of these image blocks in the rough segmentation result and mark them as edge or texture labels, the rest of image blocks which have labels in the rough segmentation result are called the homogeneous blocks. Finally, all image blocks in a test image are classified into three classes including the homogeneous, edge, and texture image blocks. In the post-processing, the region-merging process and boundary smoothing will be used to modify the rough image segmentation result. The region-merging process can avoid the over-segmentation for a test image. More accurate contours of the objects can be obtained by means of boundary smoothing.

C. Region Merging and Boundary Smoothing

As a result of the block classification and labeling shown above, the labels for all image blocks can be obtained. A region number can be assigned to all connected blocks [11]. In numbering the connected blocks, we must check whether a candidate block and its neighboring blocks have the same label. In order to reduce the number of regions, the small-size and non-interested regions should be ignored such that the complete and large regions are left. For a given block, if the number of connected image blocks is less than a defined smallest region (SR) in a region, it will be merged into one of its neighboring regions. At the same time, the mean difference between a candidate region and one of its neighboring regions is also checked based on their corresponding codewords. A candidate region will be assigned to the neighboring region which has the smallest mean difference with the candidate region. Since only the large regions in a VQ-coded image are left, the number of regions can be reduced. Therefore, the

rough segmentation result can be obtained and the over-segmentation problem can be avoided.

After the operation of region merging has been finished, we find that the boundaries of the large and interested regions in a VQ-coded image are unsmoothed, which may lead to the erroneous segmentation. The final segmentation result might not be accurate. Thus, to obtain the higher accurate segmentation result, the operation of boundary smoothing is used to refine the boundaries of the large and interested regions. The procedure of boundary smoothing is described as follows: First, all regions in a VQ-coded image are divided into two different type regions. If a region whose label of the only neighboring region is different from that of it, it is called the independent region. On the other hand, if a region whose labels of many neighboring regions are different from that of it, it is called the dependent region. Let a mask of size 3×3 be mapped to all regions in a VQ-coded image. Then, the ratio of every image blocks in a mask is computed. If a number of image blocks of objects in a mask of size 3×3 are smaller than that of its neighboring regions, then the image blocks that lies in the center position of a mask are merged into one of its neighboring regions. Otherwise, the image blocks will not be changed. The above steps are repeatedly executed until the image blocks of all regions in a VQ-coded image are never changed again. Finally, the boundary smoothing results can be obtained.

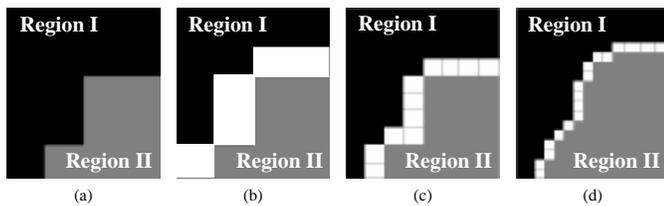


Fig. 4. Homogeneous regions and edge blocks labeling: (a) Homogeneous regions labeling. Edge blocks labeling in (b) Level 0; (c) Level 1; (d) Level 2

D. Assignments of Edge Blocks

Here we describe how to assign the edge blocks and to refine the segmentation results from the block level to pixel level. There are many homogeneous and texture regions in a test image and the image blocks in their boundaries are labeled as the edge blocks. Figure 4(a) shows two different regions, where Region I and Region II stand for the object region and the background region, respectively. Figure 4(b) shows the boundaries in the object region that is labeled as the edge blocks, where every edge block is of size 4×4 .

After all edge blocks have been labeled, they are partitioned into half size and assigned to one of its neighboring regions. The procedure of assignments of edge blocks is described as follows: First, the mean difference between a candidate edge block and each of its neighboring (4-connected neighbors [12]) regions is examined based on their corresponding codewords. Then, a candidate edge block will be assigned to the neighboring region which has the smallest mean difference with it. The size of edge blocks can be reduced to be its sub-level. As the size of edge blocks is reduced, the contour of objects can be more accurate.

Finally, the refined image segmentation result can be obtained when none of the edge block is left. The segmentation results for three different levels from the block level to the pixel level can be obtained. The basic units for assignment in the Level-0, Level-1, and Level-2 segmentation results are the blocks of size 4×4 , 2×2 , and single pixel, respectively.

TABLE I. THE PARAMETER LIST FOR TWO TEST IMAGES

Image	Parameters						
	T_M	T_{ML}	T_{MH}	T_L	T_H	T_F	SR
Lena	139	70	197	NA	NA	104	40
Tower	132	66	194	NA	NA	137	30

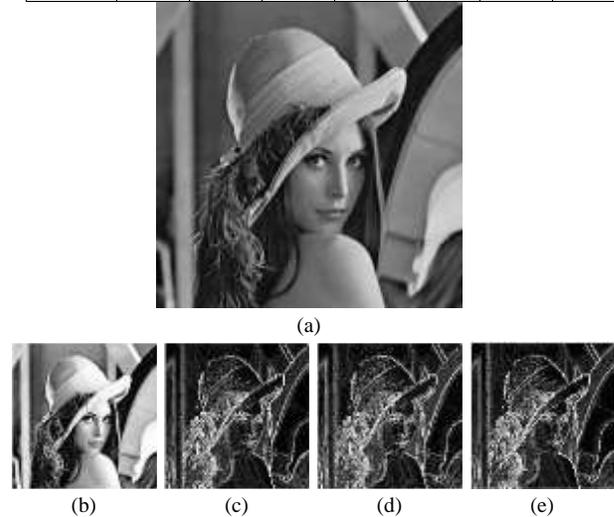


Fig. 5. (a) The original Lena image and three index images of size 128×128 : (b) F_1 , (c) F_2 , (d) F_3 , and (e) the fusion result of (c) and (d)

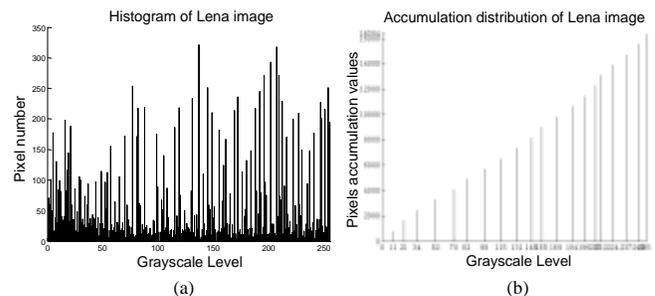


Fig. 6. (a) The histogram for the Lena index image (ordered by mean); (b) The accumulation distribution determined from (a)

III. EXPERIMENTAL RESULTS

In this paper, the Lena image of size 512×512 and the Tower image of size 360×288 are used to test the proposed SUCIS method. The unsorted codebook can be generated by using the LBG algorithm. Before the two test images are segmented, some parameters must be determined for them. Every image should have the different parameters because the complexity in each image is different. Table I shows the threshold values and the SR parameter for two test images. Both the threshold values T_L and T_H in these two images are not available because the conditions described in Section II.B

are not satisfied. Figure 5(a) shows the original Lena image. Figures 5(b)-5(d) shows the three index images F_1 , F_2 , and F_3 , which are generated by image codebooks C_{F1} , C_{F2} , and C_{F3} , respectively. The mean image shown in Fig. 5(b) looks very similar to the original image. As shown in Figs. 5(c) and 5(d), the brighter pixels represent higher variance and gradient values of the blocks in the original image. The fusion result of Figs. 5(c) and 5(d) is shown in Fig. 5(e). Figures 6(a) and 6(b) show the histogram and its accumulation distribution of the index image of the Lena image, respectively.

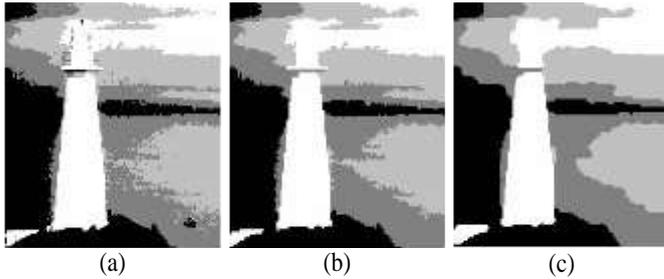


Fig. 7. Regions merging for the Tower index image F_1 (order by mean): (a) Before region merging; (b) After region merging with SR=30 and the 4-connected condition; (c) After boundary smoothing

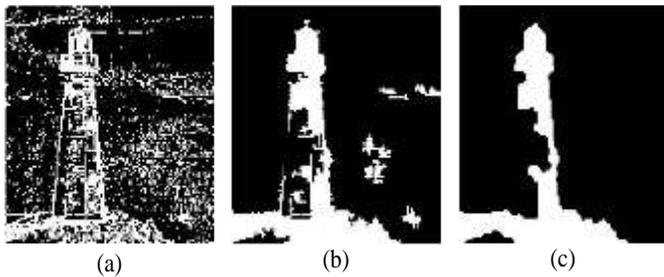


Fig. 8. Regions merging for the fused index image F_f of the Tower image: (a) Before region merging; (b) After regions merging with SR=30 and the 4-connected condition; (c) After boundary smoothing

Consider the Tower image for the proposed SUCIS method. Figures 7(a)-7(c) show some results of region merging with SR=30 and using the 4-connected condition for the mean index image. As shown in Fig. 7(c), a rough segmentation result can be observed in the smoothed regions. Figures 8(a)-8(c) show the boundary smoothing effects for the fused index image F_f . Finally, Figs. 9(a) and 9(b) show the boundary detection results for the test Lena and Tower images, respectively. According to these two figures, Figs. 10 and 11 show the segmentation results in three different levels for the Lena and Tower images which are VQ-coded. The different image segmentation results can be obtained. Obviously, the image segmentation results with better accuracy can be obtained with the edge block labeling in Level 2.

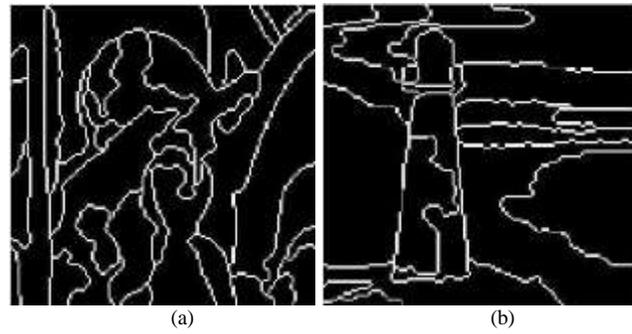


Fig. 9. Boundary detection results for the (a) Lena and (b) Tower images

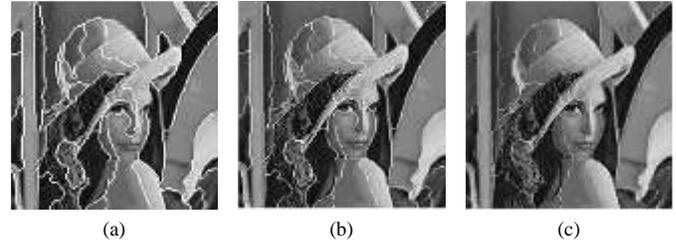


Fig. 10. Segmentation results for the Lena image of size 512×512: (a) Level 0; (b) Level 1; (c) Level 2

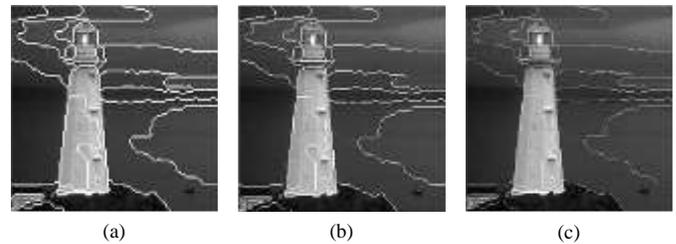


Fig. 11. Segmentation results for the Tower image of size 360×288: (a) Level 0; (b) Level 1; (c) Level 2

The proposed SUCIS method is compared with the benchmark watershed method [13] to verify the effectiveness especially for the foreground object segmentation. Four test images downloaded from the Berkeley Segmentation Dataset [14] are utilized in our experiments. Figures 12(a), 12(c), 12(e), and 12(g) show the segmentation results of the proposed SUCIS method. By using the watershed function provided in MATLAB, Figs. 12(b), 12(d), 12(f), and 12(h) show the segmentation results of the watershed method. As shown in both figures, the foreground objects can be successfully segmented. In Figs. 12(a) and 12(b), the proposed method also segments the tree branches in addition to the mother bird. The most difference results are shown in Figs. 12(e) and 12(f). The proposed method segment the foreground only in one region, while the watershed method can segment the more detail shapes for the owls.

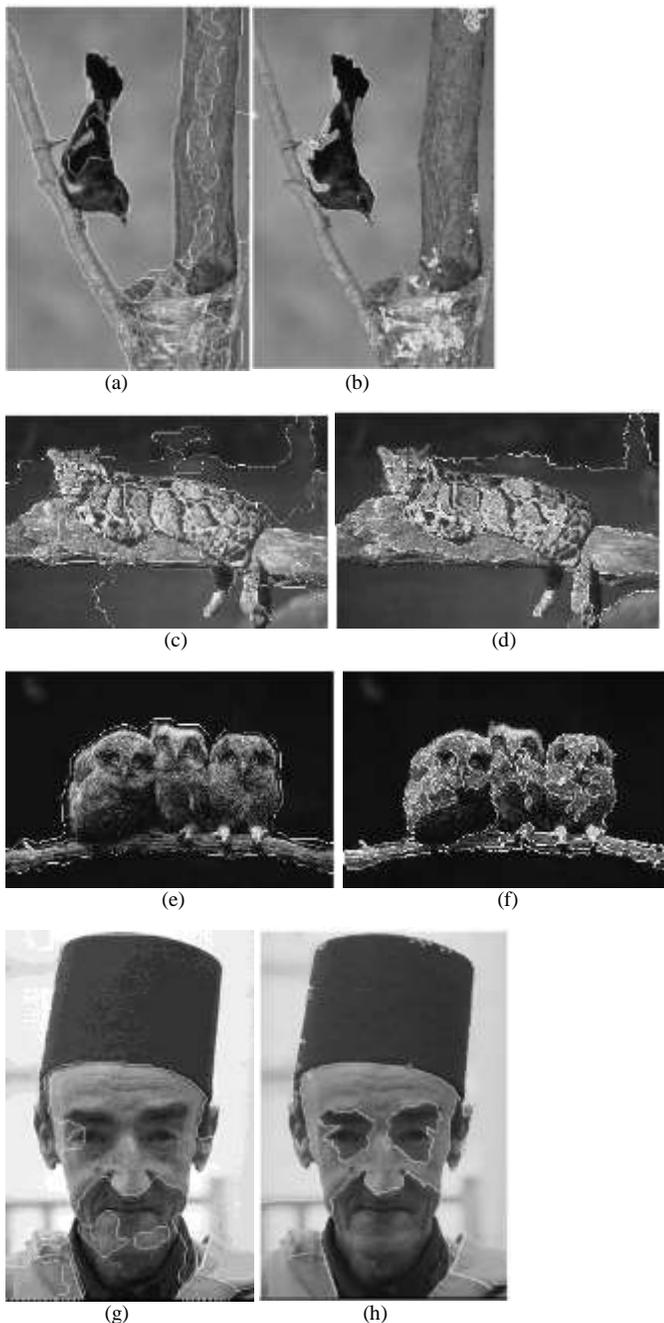


Fig. 12. Comparison of the segmentation results between the proposed SUCIS and the watershed methods: (a), (c), (e), (g) are the level-2 results of the proposed SUCIS method and (b), (d), (f), (h) are the results of the watershed method

IV. CONCLUSIONS

In this paper, the SUCIS method is proposed based on the index statistics of VQ-coded images to perform the image segmentation especially for the VQ-coded images. The SUCIS method uses three different index images to label and classify image blocks in a VQ-coded image and distinguish them into homogeneous, edge, and texture blocks to obtain disjoint regions such that the segmentation result can be obtained. Although our SUCIS method segments an image in terms of blocks rather than pixels, the pixel-based segmentation result

can be obtained by partitioning the edge blocks into the half size and assigning to one of their neighbor regions until to the pixel level. Moreover, the computational complexity can be reduced by segmenting images in the compressed domain and with blocks rather than pixels. In summary, the proposed SUCIS method is effective and efficient for image segmentation and the results can be applied on VQ-based image/video coding schemes. Our future work may focus on extracting the objects from the segmented results and/or describing the semantic meanings in the processed images. The proposed method may be extended by cooperating the local robust statistics [15] or the machine learning methods based on the artificial neural network [16] and fuzzy theory [17]. The effectiveness of applying the proposed SUCIS method onto the medical image processing [16], [18] can be investigated as well.

ACKNOWLEDGMENT

This work is partially supported by Ministry of Science and Technology (MOST), Taiwan under Contract Number MOST 105-2221-E-224-021-MY2.

REFERENCES

- [1] Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communication*, vol. 28, no. 1, pp. 84-95, Jan. 1980
- [2] R. M. Gray, "Vector quantization," *IEEE ASSP Magazine*, vol. 1, no. 2, pp. 4-29, 1984
- [3] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Boston, MA:Kluwer, 1992
- [4] C.-H. Yeh and C. J. Kuo, "Content-based image retrieval through compressed indices based on vector quantized images," *Optical Engineering*, vol. 45, no. 1, pp. 1-8, Jan. 2006
- [5] Y.T. Hsiao, C.L. Chuang, J.A. Jiang, and C.C. Chien, "A contour based image segmentation algorithm using morphological edge detection," *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, pp. 2962-2967, Oct. 2005.
- [6] H. Gao, W.C. Siu, and C.H. Hou, "Improved techniques for automatic image segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1273-1280, Dec. 2001
- [7] C.S. Won, "A block-based MAP segmentation for image compression," *IEEE Transactions on Circuit and System for Video Technology*, vol. 8, no. 5, pp. 592-601, 1998
- [8] X. Li and S. Lei, "Block-based segmentation and adaptive coding for visually lossless compression of scanned documents," *2001 IEEE International Conference on Image Processing*, vol. 3, pp. 450-453, Oct. 2001
- [9] W.K. Pratt, *Digital Image Processing*, John Wiley and Sons, Inc, 1991
- [10] G.C. Stockman and L.G. Shapiro, *Computer Vision*, Chapter 7, pp. 216, Prentice hall, New Jersey, 2001
- [11] Y.L. Leng and X. Li, "Adaptive image region-growing," *IEEE Transactions on Image Processing*, vol. 3, no. 6, pp. 868-872, Nov. 1994
- [12] Y. Deng and B.S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800-810, Aug. 2001
- [13] L. Vincent and P. Soille, "Watersheds in digital space: An efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583-597, Jun. 1991
- [14] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "Database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Proc. 8th International Conference on Computer Vision*, vol. 2, pp. 416-423, July 2001

- [15] C. Huang and L. Zeng, "Robust image segmentation using local robust statistics and correntropy-based K-meansclustering," *Optics and Laser in Engineering*, vol. 66, pp. 187-203, Mar. 2015
- [16] A. De and C. Guo, "An adaptive vector quantization approach for image segmentation based on SOM network," *Neurocomputing*, vol. 149, Part A, 3, pp. 48-58, Feb. 2015.
- [17] Y. Li, "Wavelet-based fuzzy multiphase image segmentation method," *Pattern Recognition Letters*, vol. 53, pp. 1-8, Feb. 2015
- [18] A. De and C. Guo, "An image segmentation method based on the fusion of vector quantization and edge detection with applications to medical image processing," *International Journal of Machine Learning and Cybernetics*, vol. 5, no. 4, pp. 543-551, Aug. 2014