An Automatic Key Video Object Plane Selection Scheme for MPEG-4 Video[+]

Jane-Du Huang [#] and Jin-Jang Leou, [#@]

[#]Department of Computer Science and Information Engineering

National Chung Cheng University

Chiayi, Taiwan 621, Republic of China

摘　　要

在本研究中，我們提出一種 MPEG-4 視訊的代表視訊物件平面自動選擇方法，其目的在於抽取 *L* 個代表視訊物件平面，藉此這一個物件可以簡潔的用這 *L* 個代表視訊物件平面及一些附屬的資訊加以表示。

在所提出的方法中，物件的形狀資訊和 DC 序列先從視訊序列中被擷取出來。在 DC 序列從 MPEG-4 的視訊序列擷取出來後，第一個和最後一個視訊物件平面會先被選為視訊物件的代表視訊物件平面。接著在每一個重複步驟中，和最近的左邊和右邊的代表視訊物件平面有最大距離的視訊物件平面會被選為新的代表視訊物件平面，以上步驟將一直重複執行直到獲得事先決定數量(*L*)的代表視訊物件平面為止。所提出用來計算兩個視訊物件平面的距離度量將結合視訊物件的 DC 序列中所包含的外形、邊緣、及亮度分佈資訊。根據實驗結果顯示，與兩個現有的方法比較，所提方法能從 MPEG-4 的視訊序列選出更好、更有代表性的代表視訊物件平面。

ABSTRACT

In this study, an automatic key video object plane selection scheme for MPEG-4 video is proposed. The objective is to extract a predetermined number (L) of key VOPs (video object planes) so that a video object can be described compactly by the L key VOPs and some auxiliary information.

In the proposed approach, the shape information and the DC sequence of the video object are first extracted from the video sequence. Initially, the first and last VOPs are selected as the key VOPs of the video object. Next, within each iteration, the VOP having the maximum distance with its nearest neighboring selected key VOPs is selected as the new key VOP of the video object. The above procedure is iterated until the predetermined number (L) of key VOPs are obtained. The proposed distance measure between two VOPs is obtained by combining the shape, edge, and intensity histogram information derived from the DC sequence of the video object. Based on the simulation results obtained in this study, the key VOPs selected by the proposed approach for the video object are better and more "representative" than that selected by the two existing approaches for comparison.

Index Terms: MPEG-4 video, key video object plane, DC sequence, Hausdorff distance.

1. INTRODUCTION

Multimedia information systems are becoming increasingly important with the advent of broadband networks, high-power PC's, audio/visual compression standards, and many applications such as digital libraries, and trademark and copyright databases. The advancements of several video compression standards such as MPEG-2, MPEG-4, and H.263 [1] have made it possible to have large digital video databases. To access these data efficiently, many indexing and retrieval techniques of digital video were proposed [2]-[5].

A structured collection of selected video frames, or key frames, is a compact representation of a video sequence and is useful for various applications [1]. In addition to visual summarization, key frames also provide salient visual features (color, shape, and texture) for video indexing and retrieval. Two types of key frame extraction approaches for a video sequence have been reported [6]-[9]. The first type of approaches segments a video sequence into shots and then extracts key frames from individual shots of a video sequence [6]-[7]. The second type of approaches, however, directly extracts key frames from a video sequence without shot boundary detection [8]-[9]. A shot is defined as an unbroken sequence of frames recorded from a single camera and consecutive frames within a shot exhibit temporal continuity. Once a video sequence is partitioned into shots, key frames can be extracted from individual shots.

Although key frames provide frame-based indexing and summary of a video sequence (bitstream),

[+] This work was supported in part by National Science Council, Republic of China under Grant NSC 90-2213-E-194-039.

[@] Author to whom all correspondence should be addressed. E-mail: jjleou@cs.ccu.edu.tw, Tel: 886-5-2720411 Ext. 33105, Fax: 886-5-2720859.

[+] Area: Multimedia, Computer Graphics, and Image Processing.

they do not provide an accurate description of individual video objects of the video bitstream. Object-based indexing/browsing/searching is essential and important for video databases that support object-based queries. In the object-based compression standard MPEG-4 [10], visual information is organized on the basis of the video object (VO) concept, which represents a time-varying visual entity with arbitrary shape that can be individually manipulated and combined with other similar entities to produce a scene. The information associated with a VO is represented as a set of video object layers (VOLs). Each VOL is considered as a sequence of video object planes (VOPs), which represent the information associated to given temporal instants and substitute the traditional video frames. Similar to key frames, key VOPs can be used for indexing and visual summarization of the video object contents in MPEG-4. In this study, an automatic key video object plane selection scheme for MPEG-4 video is proposed.

For key VOP selection, Ferman, et al. [11] suggested an algorithm that extracts key VOPs in an MPEG-4 compressed sequence based on the texture coding modes chosen by the encoder for the macroblocks of individual video objects. The proposed algorithm employs the percentage of intra-coded macroblocks as a measure for significant change in the contents. However, the accuracy of using the percentage of intra-coded macroblocks is too low for effective selection of key VOPs. In Gunsel, et al. [12], each video object is represented by an adaptive 2D triangular mesh. A mesh-based object tracking scheme is then employed to compute the motion trajectories of all mesh node points until the object exits the field of view. A similarity measure based on motion discontinuities and shape changes of the tracked object is defined to detect content changes and select key VOPs. However, the proposed algorithm is very computationally intensive. Erol and Kossentini [13] used the shape information of a video object retrieved from the MPEG-4 compressed video to select key VOPs. The shape of the video object is approximated by using the shape coding modes of I, P, and B video object planes (VOPs), without decoding the shape information in the MPEG-4 compressed bitstream. Two distance measures, the Hamming and Hausdorff distance measures, are modified to measure the similarities between the approximated shapes of the video objects. Although their method is computationally efficient, the approximated shapes are usually too coarse and sometimes inaccurate to the video object.

In this study, an automatic key video object plane selection scheme for MPEG-4 video is proposed. The objective is to extract a predetermined number (L) of key VOPs so that a video object can be described compactly by the L key VOPs and some auxiliary information. The algorithm extracts the key VOPs using the DC sequence of the video object, which can be readily extracted from the MPEG-4 compressed video, without full-VOP decompression.

The paper is organized as follows. The proposed automatic key video object plane selection scheme for MPEG-4 video is addressed in Section 2. Simulation results are given in Section 3, followed by concluding remarks.

2. PROPOSED AUTOMATIC KEY VIDEO OBJECT PLANE SELECTION SCHEME FOR MPEG-4 VIDEO

2.1 Extraction of DC Sequence from MPEG-4 Compressed Domain

DC images are spatially reduced versions of the original images [14]. If an image is divided into $N \times N$ blocks, the (i, j) pixel of the DC image is the average pixel value of the (i, j) block of the original image. Sequence formed in such a manner will be called DC sequence [14]. Although the DC image is much smaller than the original image, it still retains significant amount of information. Many applications performing on the original images can also be performed on the DC images. In MPEG-4, spatially reduced versions of VOPs can be extracted from the original VOPs in a similar way. The spatially reduced version of the VOP is hereafter referred to the DC VOP, as an illustrated example shown in Fig. 1. Because the shape of a video object (VO) in MPEG-4 is arbitrary, the average value of the boundary block should be derived according to the shape. That is, the average value of the block should be calculated only for the object pixels. In this study, the DC sequence of a video object is extracted from the MPEG-4 coded video sequence without full-VOP decoding. The DC sequence is then used to select the key VOPs of the video object.

A. Relationship between DC image and Discrete Cosine Transform (DCT)

For the IVOPs of MPEG-4, the 2-D (2-dimensional) DCT is applied on 8×8 blocks for reduction of spatial redundancy. The DC term, $c(0, 0)$, of the 2-D DCT becomes:

$$c(0,0) = \frac{1}{8}\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y), \qquad (1)$$

which is 8 times the average intensity of the block.

B. Extraction of DC sequence from MPEG-1 coded sequence

Yeo and Liu [14] proposed a method to extract the DC images from MPEG-1 coded sequence. Extraction of the DC image from an I-frame in MPEG-1 is simple since the (i, j) pixel of the DC image is just 1/8 the DC term of the (i, j) 2-D DCT block. However, inverse motion compensation in the MPEG-1 compressed domain is necessary to extract the DC image from the P- or B-frame in MPEG-1. In Fig. 2, P_{ref} denotes the current block of interest and P_1, P_2, P_3, and P_4 are the four neighboring blocks from which P_{ref} is derived. The shaded regions in P_1, P_2, P_3, and P_4 are moved by (Δx, Δy). Due to the linearity of DCT, the DC coefficient of P_{ref} is given by:

$$DC(P_{ref}) = \sum_{i=1}^{4}\left\{\sum_{m=0}^{7}\sum_{l=0}^{7} w_{ml}^{i}\left[DCT(P_i)\right]_{ml}\right\}, \quad (2)$$

where $[DCT(P_i)]_{ml}$ denotes the (m, l) DCT coefficient of P_i. The factor w_{ml}^{i} weights the contribution of $[DCT(P_i)]_{ml}$. Here w_{ml}^{i} is given by:

$$w_{ml}^{i} = \left[DCT(S_{i1})\right]_{0m} \times \left[DCT(S_{i2})\right]_{l0}. \quad (3)$$

S_{i1} is a matrix of the form

$$\begin{pmatrix} 0 & 0 \\ I_{h_i} & 0 \end{pmatrix} \text{ or } \begin{pmatrix} 0 & I_{h_i} \\ 0 & 0 \end{pmatrix}$$

and S_{i2} is a matrix of the form

$$\begin{pmatrix} 0 & 0 \\ I_{w_i} & 0 \end{pmatrix} \text{ or } \begin{pmatrix} 0 & I_{w_i} \\ 0 & 0 \end{pmatrix},$$

where I_n is an identity matrix of size n. The factors h_i and w_i in these equations are the height and the width of the overlap of P_{ref} and P_i, respectively. P_{ref} can be divided into four subblocks of interest lying in P_1, P_2, P_3, and P_4. Combinations of S_{i1} and S_{i2} for different subblocks are tabulated in Table 1. It can be shown that the weight w_{00}^{i} of $[DCT(P_i)]_{00}$ is $(h_i \times w_i)/64$ and Eq. (3) becomes:

$$\left[DCT(P_{ref})\right]_{00} = \sum_{i=1}^{4} \frac{h_i w_i}{64}\left[DCT(P_i)\right]_{00} + c, \quad (4)$$

where $c = \sum_{i=1}^{4} \sum_{(m,l)\neq(0,0)} w_{ml}^{i}\left[DCT(P_i)\right]_{ml}$.

Because c in Eq. (4) is small, the first term can be viewed as an approximation of $[DCT(P_{ref})]_{00}$ (DC coefficient of P_{ref}) by the weighted sum of the DC's of P_1, P_2, P_3, and P_4. In practice, $(h_i w_i)/64$ is just the fraction of area occupied by the subblock in P_i. That is, the approximated DC of P_{ref} can be obtained by just calculating the weighted sum of the DC's of P_1, P_2, P_3, and P_4 according to the fractions of areas occupied by the four subblocks.

C. Extraction of DC sequence from MPEG-4 coded sequence

Because the shape of a video object in MPEG-4 can be arbitrary, it is impossible to get the DC VOP without knowing its shape information. That is, the shape information of a VOP has to be decoded before the extraction of the DC VOP. After decoding the shape of the VOP, a kind of temporary VOP is produced. The pixels within the temporary VOP are set to 1/8 the DC values of their corresponding blocks. Fig. 3 shows an illustrated example of a temporary VOP. Because extraction of the DC term from an IVOP is simple, we can obtain the temporary IVOPs from an MPEG-4 coded sequence easily. Furthermore, the inverse motion compensation technique developed by Yeo and Liu [14] can be used to obtain approximated temporary PVOPs or BVOPs. Dividing Eq. (4) by 8, we have:

$$\frac{1}{8}\left[DCT(P_{ref})\right]_{00} = \sum_{i=1}^{4} \frac{h_i w_i}{64}\left\{\frac{1}{8}\left[DCT(P_i)\right]_{00}\right\} + c', \quad (5)$$

where $c' = \frac{1}{8}c$. Eq. (5) shows that the simple inverse motion compensation technique developed by Yeo and Liu [14] can also be performed on the temporary VOPs to derive approximated temporary PVOPs or BVOPs when all referenced blocks are interior blocks, i.e., an approximated pixel value of a block in PVOPs or BVOPs can be derived by the weighted sum of the pixel values of its four referenced blocks according to the fractions of areas occupied by the four subblocks when all referenced blocks are interior blocks. However, macroblock-based repetitive padding is required for the region outside the temporary VOP before inverse motion compensation is applied on the temporary VOP.

As the LPE (low pass extrapolation) padding is required for the intra-coded blocks at each boundary of each VOP before performing 2-D DCT, the DC coefficients extracted from the intra boundary DCT blocks are actually approximated ones. However, the correct DC coefficients of the residual boundary blocks of the VOP can be obtained by $DC = \frac{64}{k} DC'$ because they are padded with zeroes. The factor k is the number of object pixels in the block and DC' is the original DC value.

Once the temporary VOPs and its binary alpha planes are extracted, the DC VOP and the reduced binary alpha planes can be obtained by a simple downsampling. The non-transparent blocks in the VOP are viewed as the pixels inside the downsampled VOP so that the reduced binary alpha planes can be derived easily. In addition, downsampling performing on the temporary VOP is also simple since the object pixels of the (i, j) block of the temporary VOP is set to the approximated value of the (i, j) pixel of the DC VOP.

2.2 Proposed Distance Measure Between Two DC VOPs

Because key VOPs have to reflect significant changes in shape and contents of a video object (VO), the proposed distance measure uses three kinds of information, namely, shape, edge, and intensity histogram information, to measure the distance between two VOPs. In the proposed distance measure, the distance related to shapes and edges between two VOPs (D_1) is measured using the Hausdorff distance [15]. The intensity histogram distance between two VOPs (D_2) is derived from their intensity histograms [16]. After these two distances are derived, the proposed distance measure can be obtained by combining them.

Before measuring the distances related to shapes and edges, the shapes and edges of the DC VOPs have to be extracted from the DC sequence. Extraction of object boundary points from the reduced binary alpha plane is simple. To extract the edge information from the DC

VOP, the Sobel filter [16] is employed, which is a type of derivative filter. Consider a 3×3 image region shown in Fig. 4, where the z's denote the gray values. The gradient magnitude can be approximated at point z_5 by:

$$\nabla f \approx |(z_7+2z_8+z_9)-(z_1+2z_2+z_3)|+|(z_3+2z_6+z_9)-(z_1+2z_4+z_7)|. \quad (6)$$

An edge point is declared if its approximated gradient ∇f is above a given threshold T_e. Once the edge points of the DC VOP are extracted, the object boundary points and the edge points are combined to a new set of points. Although the DC VOP may lose some edge information of the original VOP, the key and important edge information is usually preserved. The Hausdorff distance measure [15] is then applied on the two sets (boundary and edge) of points to derive the shape and edge distance between two DC VOPs.

The Hausdorff distance measure is defined as a "maxmin" function between two sets of points. Given two finite sets $A = \{a_1, a_2, a_3, \ldots a_p\}$ and $B = \{b_1, b_2, b_3, \ldots, b_q\}$, the directed Hausdorff distance is defined as:

$$h(A,B) = \max_{a_i \in A}\{\min_{b_j \in B}\{d(a_i, b_j)\}\}, \quad (7)$$

where $d(a_i, b_j)$ is the Euclidean distance between two points a_i and b_j. Eq. (7) measures the furthest distance from the points in the set *A* to the points in the set *B*. The Hausdorff distance is asymmetric, i.e., $h(A,B)$ is not necessarily equal to $h(B,A)$. Therefore, the more general definition of the Hausdorff distance is given by:

$$H(A,B) = \max(h(A,B), h(B,A)). \quad (8)$$

For two DC VOPs, VOP_i and VOP_j, the distance D_1 related to shape and edge between VOP_i and VOP_j is given by:

$$D_1 = H(P_i, P_j) \Big/ \sqrt{\min(A_i, A_j)}, \quad (9)$$

where P_i and P_j are the corresponding sets of points of VOP_i and VOP_j, whereas A_i and A_j are the areas of VOP_i and VOP_j, respectively. Based on Eq. (9), D_1 is inversely proportional to the area of the smaller DC VOP.

Based on the fact that the shape of a video object in MPEG-4 can be arbitrary, alignment (registration) is required for DC VOPs before measuring the Hausdorff distance between two DC VOPs. Because the shape of the original VOP has been decoded, the centroid can be computed using the original shape with more accuracy. Here the centroids are used to align two DC VOPs. Based on our experimental results, aligning two VOPs using their centroids provides a reasonable alignment between two DC VOPs.

To obtain the intensity histogram distance between two DC VOPs, the normalized intensity histograms of the DC VOPs are calculated first. Let *B* be the total number of bins in the intensity histogram and $p(x,y)$ be the pixel value at (x,y). The normalized intensity histogram of the *k*th bin within VOP_i is defined as:

$$H_i(k) = \frac{1}{N}\sum_{x,y \in VOP_i} d(x,y) \quad \text{for } k = 0,1,2,3,\ldots,B-1, \quad (10)$$

where

$$d(x,y) = \begin{cases} 0.5 & \text{if } \frac{256}{B}k \le p(x,y) < \frac{256}{B}(k+1) \text{ and } p(x,y) \text{ is a boundary pixel,} \\ 1 & \text{if } \frac{256}{B}k \le p(x,y) < \frac{256}{B}(k+1) \text{ and } p(x,y) \text{ is an interior pixel,} \\ 0 & \text{otherwise,} \end{cases}$$

and $N = \sum_{x,y \in VOP_i} c(x,y)$,

where

$$c(x,y) = \begin{cases} 0.5 & \text{if } p(x,y) \text{ is a boundary pixel,} \\ 1 & \text{if } p(x,y) \text{ is an interior pixel.} \end{cases}$$

The intensity histogram distance between VOP_i and VOP_j is given by:

$$D_2 = \sum_{m=0}^{B-1} H_i(m) \sum_{n=0}^{B-1} |H_i(n) - H_j(n)| \left(m - n/B-1\right)^2 + \sum_{m=0}^{B-1} H_j(m) \sum_{n=0}^{B-1} |H_j(n) - H_i(n)| \left(m - n/B-1\right)^2. \quad (11)$$

Based on the fact that $\sum_{k=0}^{B-1} H_i(k) = 1$, D_2 will take the value between 0 and 2. Finally, the proposed distance measure between two VOPs is given by:

$$D = D_1 + D_2. \quad (12)$$

2.3 Proposed Key VOP Selection Scheme for MPEG-4 Video

After DC sequence is obtained from MPEG-4, key VOPs can be extracted using DC images. First, the first and last VOPs are selected as the key VOPs of the video object (VO). Let C_i denote a candidate key VOP, and L_i and R_i be the nearest neighboring left and right key VOPs of C_i, respectively, as an illustrated example shown in Fig. 5. Assume that the distance between C_i and L_i is D_{L_i} and the distance between C_i and R_i is D_{R_i}. For a candidate key VOP C_i, D_i is defined as:

$$D_i = \min\left(D_{L_i}, D_{R_i}\right). \quad (13)$$

The candidate key VOP having the maximal D_i value is selected as the next key VOP of the video object.

Within the proposed approach, to extract the *L* key VOPs from the video sequence, the first and the last VOPs are initially selected as the key VOPs of the video object (VO). Next, within each iteration, the VOP having the maximum distance with its two nearest neighboring VOPs is selected as the new key VOP of the video object. The above procedure is iterated until the predetermined number (*L*) of key VOPs are obtained. The proposed approach to extract *L* key VOPs from the DC sequence containing *S* DC images is summarized as Fig. 6. Because the proposed approach builds a hierarchical structure of key VOPs, the user can get the desired key VOPs at different levels of details once the VOPs have been extracted from the DC sequence.

3. SIMULATION RESULTS

Two test video sequences, "Bream" and "Weather,"

are used to evaluate the performance of the proposed approach. Each video frame of the two test video sequences is 352×288 in size and the color space used is Y, C_B, and C_R. Because human is more sensitive to luminance than chrominance, edge information and intensity histograms are only derived from the Y component of each DC VOP.

To compare with other existing key VOP selection approaches, two existing compressed domain approaches, namely, Ferman, et al.'s method [11] and Erol and Kossentini's method [13], are implemented in this study. Ferman, et al. [11] used the percentage of intra-coded macroblocks of PVOPs to select key VOPs, whereas Erol and Kossentini [13] used the shape information of the video object retrieved from the MPEG-4 compressed domain to select key VOPs (using the Hausdorff distance). Ferman, et al.'s method [11] will be compared with the proposed approach based on the IPPP structured video object bitstreams. The key VOPs selected by Ferman et al.'s method and the proposed approach for the Weather video object bitstream are shown in Fig. 7, where $L = 4$ and $S = 300$. Erol and Kossentini's method will be compared with the proposed approach based on the IBBBPBBBPBBB structured video object bitstreams. The key VOPs selected by Erol and Kossentini's method and the proposed approach for the Weather video object bitstream are shown in Fig. 8, where $L = 4$ and $S = 300$. The distance measures D between two successive VOPs selected by Ferman, et al.'s method and the proposed approach for the Weather video object bitstream are listed in Table 2, whereas that selected by Erol and Kossentini's method and the proposed approach for the Weather video object bitstream are listed in Table 3. The simulation results for the Bream video object bitstream can be found in [17] and thus omitted here.

4. CONCLUDING REMARKS

Based on the simulation results (Figs. 7-8 and Tables 2-3) obtained in this study, several phenomena can be observed. The key VOPs selected by the proposed approach are better than that selected by the existing approaches for comparison. Ferman, et al.'s method [11] selects a redundant key VOP (see VOP 267 in Fig. 7(a)), which does not provide good visual summaries of the Weather video object. Even most key VOPs selected by Erol and Kossentini's method [13] are similar to that selected by the proposed approach, however, Erol and Kossentini's method selects a redundant key VOP (VOP 239 in Fig. 8(a)), which does not provide a good visual summary of the Weather video object. Based on the results shown in Tables 2-3, the distance measures D between two successive key VOPs selected by the proposed approach are usually more "uniform" than that selected by the two existing approaches for comparison. That is, the key VOPs selected by the proposed approach are better and more "representative" than that selected by the two existing approaches for comparison.

In this study, an automatic key video object plane selection scheme for MPEG-4 video is proposed. The objective is to extract a predetermined number (L) of key VOPs so that the video object can be described compactly by the L key VOPs and some auxiliary information. The algorithm extracts the key VOPs using the DC sequence of the video object, which can be readily extracted from the MPEG-4 compressed video object bitstream, without full-VOP decompression.

Within the proposed approach, the shape information and the DC sequence of the video object are first extracted from the video sequence. To extract the L key VOPs from the video sequence, the first and the last VOPs are initially selected as the key VOPs of the video object (VO). Next, within each iteration, the VOP having the maximum distance with its two nearest neighboring selected key VOPs is selected as the new key VOP of the video object. The above procedure is iterated until the predetermined number (L) of key VOPs are obtained. The proposed distance measure between two VOPs is obtained by combining the shape, edge, and intensity histogram information derived from the DC sequence of the video object.

Based on the simulation results obtained in this study, the key VOPs selected by the proposed approach for the video object are better and more "representative" than that selected by the two existing approaches for comparison. This shows the feasibility of the proposed approach.

REFERENCES

[1] A. Hanjalic, et al., *Image and Video Databases: Restoration, Watermarking and Retrieval*. Amsterdam, the Netherlands: Elsevier, 2000.

[2] S. W. Smoliar and H. Zhang, "Content-based video indexing and retrieval," *IEEE Multimedia*, vol. 1, 1994, pp. 62-72.

[3] J. Lee and B. W. Dickinson, "Hierarchical video indexing and retrieval for subband-coded video," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 10, no. 5, 2000, pp. 824-829.

[4] S. Dagtas, W. Al-Khatib, A. Ghafoor, and R. L. Kashyap, "Models for motion-based video indexing and retrieval," *IEEE Trans. on Image Processing*, vol. 9, no. 1, 2000, pp. 88-101.

[5] ISO/IEC, JTC1/SC29/WG11, "MPEG-7: requirements document version 11.0," N2723, March 1999.

[6] H. J. Zhang, J. H. Wu, D. Zhong, and S. W. Smoliar, "An integrated system for content-based video retrieval and browsing," *Pattern Recognition*, vol. 30, no. 4, 1997, pp. 643-658.

[7] A. D. Doulamis, N. Doulamis, and S. Kollias, "Non-sequential video content representation using temporal variation of feature vectors," *IEEE Trans. on Consumer Electronics*, vol. 46, no. 3, 2000, pp. 758-768.

[8] A. Hanjalic and H. Zhang, "An integrated scheme

for automated video abstraction based on unsupervised cluster-validity analysis," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, no. 8, 1999, pp. 1280-1289.

[9] Y. Gong and X. Liu, "Video summarization using singular value decomposition," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, 2000, pp. 174-180.

[10] ISO/IEC JTC1/SC29/WG11 N3093, "MPEG-4 video verification model version 15.0," 1999.

[11] A. M. Ferman, B. Gunsel, and A. M. Tekalp, "Object-based indexing of MPEG-4 compressed video," in *Proc. of IS\&T/SPIE Symp. on Electronic Imaging*, vol. 3024, 1997, pp. 953-963.

[12] B. Gunsel, A. M. Tekalp, and P. J. L. Van Beek, "Content-based access to video objects: Te mporal segmentation, feature extraction and visual summarization," *IEEE Trans. on Signal Processing*, vol. 46, 1998, pp. 261-280.

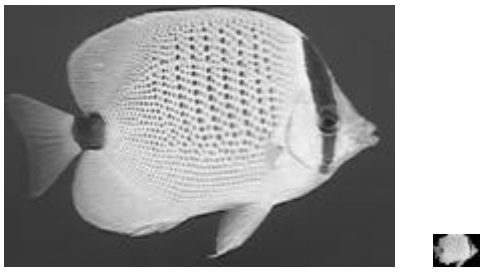
[13] B. Erol and F. Kossentini, "Automatic key video object plane selection using the shape information in the MPEG-4 compressed domain," *IEEE Trans. on Multimedia*, vol. 2, no. 2, 2000, pp. 129-138.

[14] B. L. Yeo and B. Liu, "Rapid scene analysis on compressed domain," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 5, no. 6, 1995, pp. 533-544.

[15] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, 1993, pp. 850-863.

[16] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, Massachusetts: Addison-Wesley, 1992.

[17] J. D. Huang, "An automatic key video object plane selection scheme for MPEG-4 video," *Master Thesis*, Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan, R.O.C., 2001.



(a) (b)

Fig. 1. An illustrated original VOP and the corresponding DC VOP: (a) original VOP, (b) the corresponding DC VOP.

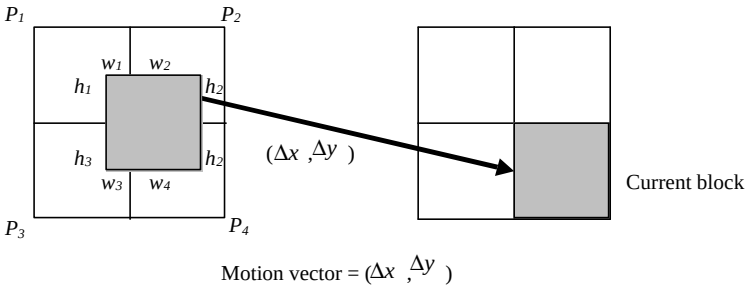


Fig. 2. Reference block (P_{ref}), motion vectors, and original blocks.

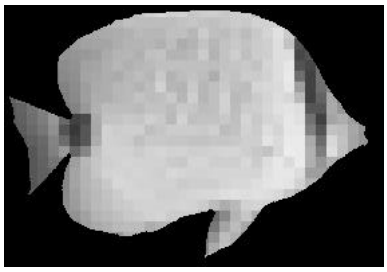


Fig. 3. An illustrated example of a temporary VOP.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Fig. 4. A 3×3 region of an image.

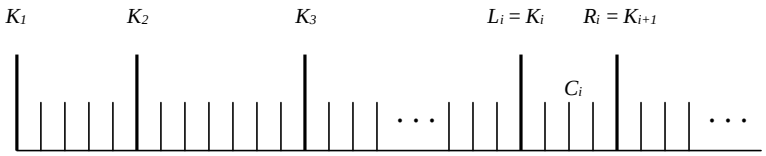


Fig. 5. The relationship between C_i, L_i, and R_i, where K_i is the ith selected key VOP.

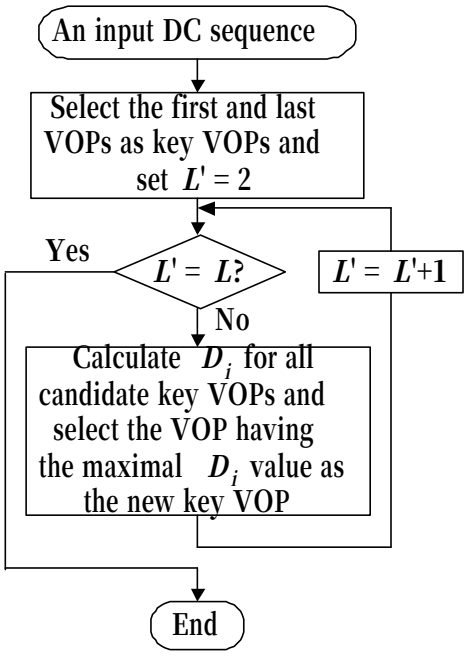
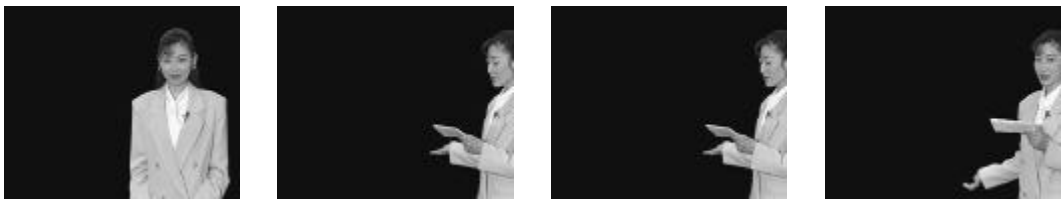


Fig. 6. The proposed approach to extracting L key VOPs from the DC sequence containing S DC images.



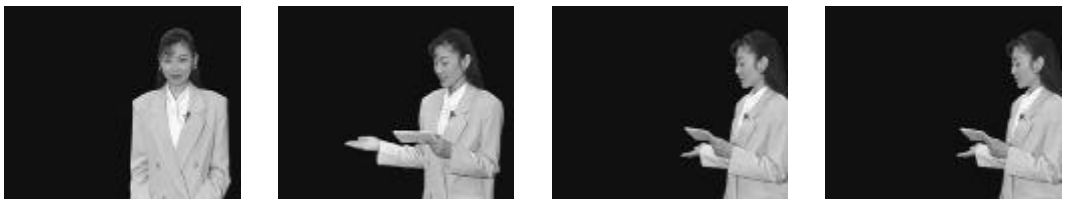
VOP 0 VOP 266 VOP 267 VOP 299

(a)

VOP 0 VOP 198 VOP 257 VOP 299

(b)

Fig. 7. The key VOPs selected by Ferman, et al.'s method (a) and the proposed approach (b) for the IPPP structured Weather video object bitstream, where L=4 and S=300.



VOP 0 VOP 195 VOP 238 VOP 239

(a)

VOP 0 VOP 198 VOP 257 VOP 299

(b)

Fig. 8. The key VOPs selected by Erol and Kossentini's method (a) and the proposed approach (b) for the IBBBPBBBPBBB structured Weather video object bitstream, where L=4 and S=300.

Table 1. Matrices S_{i1} and S_{i2}.

Subblock location	Position	S_{i1}	S_{i2}
P_1	lower right	$\begin{pmatrix} 0 & I_{h_1} \\ 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 \\ I_{w_1} & 0 \end{pmatrix}$
P_2	lower left	$\begin{pmatrix} 0 & I_{h_2} \\ 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & I_{w_2} \\ 0 & 0 \end{pmatrix}$
P_3	upper right	$\begin{pmatrix} 0 & 0 \\ I_{h_3} & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 \\ I_{w_3} & 0 \end{pmatrix}$
P_4	upper left	$\begin{pmatrix} 0 & 0 \\ I_{h_4} & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & I_{w_4} \\ 0 & 0 \end{pmatrix}$

Table 2. The distance measure D between two successive VOPs selected by Ferman, et al.'s method and the proposed approach for the Weather video object bitstream.

Methods	D_{12}	D_{23}	D_{34}
Ferman, et al.'s Method	0.920	0.107	0.503
Proposed Approach	0.620	0.911	0.572

Table 3. The distance measure D between two successive VOPs selected by Erol and Kossentini's method and the proposed approach for the Weather video object bitstream.

Methods	D_{12}	D_{23}	D_{34}
Erol and Kossentini's Method	0.604	0.833	0.156
Proposed Approach	0.620	0.911	0.572