

Motion Detection and Segmentation in H.264 Compressed Domain For Video Surveillance Application

Khushbu Patel
Dept. VLSI & EMBEDDED SYSTEM
Ganpat University
Mehsana, India

Abstract— In this paper, a novel algorithm for the real-time, unsupervised motion detection in compressed-domain sequences is proposed. The goal is to develop algorithm which may be useful in a real-life industrial perspective by facilitating the processing of large numbers of video streams on a single server. The focus of the work is on using the information in coded video streams to reduce the computational complexity and memory requirements, which translates into reduced hardware requirements and costs. The devised algorithm detects and segments activity based on motion vectors embedded in the video stream without requiring a full decoding and reconstruction of video frames. To improve the robustness to noise, an unreliable motion field is removed by processing MV field and DCT energy. The algorithm was tested on surveillance H.264 sequences.

Keywords---H.264, Compressed domain, Video Surveillance, Segmentation, MPEG, DCT.

I. INTRODUCTION

IP video surveillance systems are growing in size, complexity and capacity. Introduction of advanced coding standards like the H.264 [6] means that better compression is achieved, but also that more resources have to be allocated to decoding of the video. Higher resolution images delivered by newer cameras require more processing time for motion detection and segmentation algorithms based on background subtraction or frame-to-frame-change detection. Performance of pixel based processing in large systems may be challenged by the load of decoding multiple streams and processing of vast amounts of video. So algorithm based on pixel processing cannot fulfill requirement of real time application.

Motion detection is often the very first step of the analysis of the video, used either for triggering alarms or to determine which video sequences have to be stored. This task is performed continuously on all video streams in the system. In such a set-up, the motion detection and extraction must be fast and accurate, to avoid omission of important parts of the video, which may later be used in a legal case. The accuracy and sensitivity must be good enough to detect the events but not to trigger many false alarms. A solution for reduction of the amount of computing power required is to use the data from the compressed video.

Motion detection algorithm in compressed domain usually relies on two types of features in terms of macro block (MB): motion vector (MV) and DCT (Discrete cosine transform) coefficients. MVs are obtained in the motion compensation between the current frame and its reference

frames block by block. MV presents the temporal correlation between two one

frames and provides the displacement of the block. All MVs in frame can be treated as a sparse motion vector fields. On the other hand, the DCT coefficients of an MB carry the image information. For the inter-coded block, DCT coefficients contain the residues of the motion compensation. For the intra-coded block, DCT coefficients are transformed signal of the original image. Therefore, the block DCT coefficients can be used to reconstruct the DC (Discrete Cosine) image or treated as the texture feature to measure the similarity of blocks. In H.264/AVC, the intra coded block is spatial intra-predicted according to its neighbor pixels. So, the DCT coefficients provide the spatial prediction residues information for blocks now. On the other hand, H.264/AVC supports variable block-size motion compensation. An MB may be partitioned into several blocks and has several MVs.

As a result, the MV field for H.264/AVC compressed video consists of MVs with variant block size. Therefore, there is a requirement of efficient motion detection technique for the H.264/AVC compressed video. This paper proposes a novel motion detection and segmentation algorithm. The method can extract motion block efficiently and robustly. This

paper is organized as follows. First, Section 2 briefly reviews some related work of motion detection segmentation over MPEG compressed domain. Then, an overview of the proposed algorithm is presented in section 3. This algorithm consists of three stages: the MV processing, texture measure and measure of residue energy. Clustering algorithm is described in section 4. Experimental results are presented in section 5 and conclusion and future scope are provided in section 6.

II. RELATED WORK

The H.264/AVC standard is the most widely used codec in the present time. It can achieve higher compression and better quality than any of the previous standards, which benefit online multimedia applications, such as HDTV delivery, online video uploading and downloading as well as video sharing and recording. Most of motion detection methods for H.264/AVC video come from previous MPEG-2-based algorithm, which can be classified into two groups: a

motion vector based method and a residual information based method. Motion vector based methods uses only motion vector information. H. Zen in [3] detected motion based on the magnitudes of motion vectors. Direction of motion vector is used for grouping objects. Ashraf et al. [4] detected motion blocks based on the value of motion vectors after Gaussian and median filtering. Liu [12] uses a median filter to remove the noise and smooth the input MV field. A binary partition tree filtering is used to segment motion blocks. Zeng et al. [7] classified motion vectors into foreground, background and noisy MVs to detect moving objects in H.264/AVC compressed domain. Residual information based methods uses DCT coefficient or color information which partially decoded from I frame. Schonfeld and Lelescu method [1] acquired objects from I frame by using template matching. While Manerba et al. [2] also detected motion by removing global motion vectors on P-frames and additionally use DCT coefficients to obtain motion blocks on I-frames. However, because of the intra-prediction in I frame for H.264/AVC video, a prediction block is formed based on previously encoded and reconstructed blocks. H.264/AVC will vary with the changes of compression rate, thus I frame is not reliable for motion detection. Some researchers are using the features of H.264/AVC to achieve motion detection. Wonsang et al. [8] uses skip MB to remove background MBs, and uses spatial filtering and temporal filtering to remove noisy foreground MBs. However the number of skip MB is dependent on the resolution of the video and also the type of frames. Since P frames contain less number of skip MBs than B frames do, and skip MB is also used to encode homogeneous color area within a large size of moving object, the method by Wonsang et al. [8] cannot be directly used for main profile H.264/AVC video. Size (in bits) of MB and transform coefficients are used in Chris [9]. It can achieve detection in compressed domain, but it needs several predefined threshold, which depend on the resolution of each video. The algorithm uses fuzzy logic and allows describing position, velocity and size of the detected regions, demanding high computational complexity. Noticing that previous detection algorithm for H.264/AVC can not meet the need of accurate results for all type profiles of H.264/AVC video, in this paper a detection method using motion compensation block size and motion vector and DCT energy information is proposed.

III. THE PROPOSED ALGORITHM

The proposed method can detect motion by simply examining and processing the motion information and residue information of a macro block directly in the compressed domain. The Block Size (Inter Mode) of a block, its corresponding Motion Vector and DCT coefficients can be obtained by entropy decoding the H.264 bit stream. The proposed method consists of the following phases:

- A. Motion vector processing
- B. Measure of motion texture
- C. Measure of residue energy
- D. Segmentation

A. Motion vector processing

1. Motion vector normalization

In order to obtain a uniformly sampled and temporally normalized MV field for the segmentation purpose, the raw MVs of inter-frames are normalized as follows. First, the raw MV field is uniformly sampled at each 4x4 block, which is the minimal block size supported by H.264. The MV and the Reference frame index of each partition whose size is larger than 4x4 are directly assigned to all its covered 4x4 blocks. Then, the MV of each 4x4 block is normalized according to the temporal distance indicated by the reference frame index. For example, assume that the MV and reference frame of 4x4 block B in current P-frame c is denoted by MV(B) and r, respectively, then the normalized MV will be calculated as

$$N_{mv} = \frac{MV}{c - r}$$

Thus the equivalent reference frame for each inter-frame is its previous frame in the resultant normalized MV field.

2. Vector median filtering

The vector median filtering can eliminate the isolated vector noise and smoothen the difference of motion vector between the adjacent blocks. Sliding window approach is used for median filtering. First, the difference d_i among all elements in the NxN window function is defined:

$$d_i = \sum_{j=1}^N ||v_i - v_j|| \quad i = 1, 2, \dots, N$$

where: v_i and v_j are the motion vector in the NxN window. $\{i\}$ descending order is mapped into $\{i\}$ v. Finally, v_{out} is taken as $v_{(n+1)/2}$ for mean vector greater than threshold otherwise v_1 . Threshold can be set as half of maximum motion vector in frame.

3. Forward block vector accumulation

Reference [11] uses the backward motion accumulation approach, which not only increases the reliable motion information like object motion and video camera global motion, but also inhibits the noise generally existing in the single frame motion vector field. It actually accumulates the vectors appearing on the location of the current block and the same location of the following two frames. The forward block vector accumulation approach used in this paper accumulates the vectors of the same current frame and its previous two frames. This approach can effectively solve the problem that it is not easy to perform subsequent treatment when stationary block exists in the current frame motion object and obtain more obvious vector field by the accumulation. Though it increases the large light and shadow noise vector that can't be eliminated by the vector median filtering, the macro-block characteristics can effectively distinguish the difference between them. The motion fields in the current frame t and previous frame $t-1$ are used to reconstruct the predicted motion field for $t-1$ frame. 4x4 block is a basic processing unit. Based on block's location, its predicted motion vector is calculated. The accumulated motion fields are grouped into 16x16 macro-block. The macro-block strength and macro-block inner vector

difference are taken as the macro-block characteristic of the accumulated motion field. The macro-block strength and macro-block inner vector difference are represented respectively by I and D . I and D are calculated respectively from the formula, where q is relative threshold and a , b are x and y co-ordinates.

$$I = 1/16 \sum_{i=b}^{16} q\sqrt{a^2 + b^2}$$

$$D = \frac{1}{15} \text{diff}(MB)$$

Due to the characteristics of the motion estimation in the H.264 coding process, the accumulated motion field shows the following features:

(1) The macro-block strength of the background is 0. The macro-block strength of the motion object and intense light and shadow noise approaches 1. The common noise approaches 0.

(2) The motion object has larger macro-block inner vector difference. The macro-block inner vector difference of the background and noise approaches 0.

B. Measure of motion texture

Besides the motion vector (m_x , m_y), we take the partitioned block size as an important cue for motion analysis, which is a measure of motion texture. It is observed that background either is static or has regular motion which simply caused by the camera movement, whereas foreground (or motion blocks) may has motion details caused by both camera movement and motion. Therefore blocks with smaller partitioned block size and detailed motion are more likely to attract viewer's attention. The motion texture of a 4×4 block is set to be inversely proportional to the size of partition where the block locates

$$W(B) = 1 \text{ if the size of } P(B) \text{ is } 16 \times 16$$

$$2 \text{ if the size of } P(B) \text{ is } 16 \times 8 \text{ or } 8 \times 16$$

$$4 \text{ if the size of } P(B) \text{ is } 8 \times 8$$

$$8 \text{ if the size of } P(B) \text{ is } 8 \times 4 \text{ or } 4 \times 8$$

$$16 \text{ if the size of } P(B) \text{ is } 4 \times 4$$

C. Measure of residue energy

Motion vector are not true indication of motion blocks. So we consider residue energy as another motion cue for motion detection. DCT coefficients can reflect the prediction residual energy indicating the complexity of the video content in terms of texture and motion. First, the sum of squared DCT coefficients for each 4×4 block $B_{i,j}$ is calculated as a feature, where i and j are the horizontal and vertical indices of the block in each video frame indexed by t respectively. Then log operation and histogram equalization are applied sequentially in order to enhance the contrast. Thus blocks that have higher energy than threshold value are identified as motion blocks where threshold is set as half of the highest energy in the frame.

D. Segmentation

The blocks those belong to motion after motion vector processing are processed for measure of texture motion and residue energy. Motion blocks are the blocks which contain energy higher than threshold and contain higher macroblock strength and inner larger difference. All the motion blocks are

then clustered by clustering algorithm to form motion area or motion object.

IV. BLOCK LEVEL CLUSTERING

After detection of motion blocks in the frame, blocks are merged to form unique clusters. Merging of blocks in frame is performed using the proposed block clustering method in compressed domain. A sliding window of size 3×3 blocks is used for clustering. The window slides in horizontal and vertical direction within a frame. A block, whose cluster number needs to assign, is placed at the center of the window. The proposed block clustering method is illustrated in Fig.1 where a sliding window of size 3×3 blocks is shown in the pink color box. The brown color box at the center of the window is the block whose cluster number needs to decide.

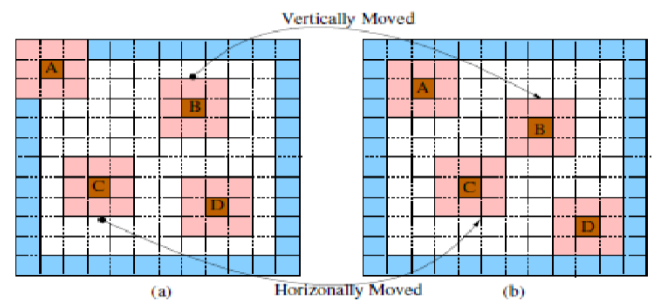


Fig.1 Block Clustering Method: Part (a) and (b) figure illustrate the diagonal (horizontal and vertical) movement of window containing blocks A and D, vertical movement of window containing block B, and horizontal movement of window containing block C to form clusters.

The rest of the blocks in the window are 8-neighbor blocks of the center block. Blue color boxes are zero blocks padded at the boundaries to include original boundary blocks in the center of the window. In part (a) of the figure, the block A is a boundary block. It can be included in the center of the window after only after padding zero blocks. Then the window slides to check the rest of the blocks in part original boundary blocks in the center of the window. In part (a) of the figure, the block A is a boundary block. It can be included in the center of the window after only after padding zero blocks. Then the window slides to check the rest of the blocks in part (b) of the figure1. The windows containing blocks B and C in part (a) of the figure, slide vertically and horizontally, respectively in part (b) of the figure. Similarly, the window containing block D slide horizontally and vertically over the frame is shown in part (a) and part (b) of the figure.

V. EXPERIMENTAL RESULTS

The proposed segmentation algorithm is implemented in C and evaluated on several H.264 compressed video sequences. The hardware platform for the testing is a Pentium 4 2.4 GHz PC with 512 M RAM. The encoder configuration is set as follows: baseline profile, group of picture (GOP) is IPPP, the interval of I-frames is 50, five reference frames, the MV search range is 32, the quantization parameter (QP) is 29 and coding frame rate is 10 f/s. The total processed frame number is 5039. The segmentation results are presented for the sequence video. Fig.2 is the segmentation experiment of

fruit and vegetable store 17th frame. Fig.2(a) shows original frame. Fig.2(b) is the result of motion detection algorithm. As it shows, for the macro-block level only needs to cluster the two-dimensional characteristic data of 9x6 macro-blocks in left side area. The result is that only 30 macro-blocks are in motion. Fig 2(c) shows Result of clustering. Fig.2(d) shows motion area in frame. Fig.3 Segmentation results of super store 228th frame. Fig.3(a) is original frame. Fig.3(b) is result of motion detection algorithm. Fig.3(c) shows result of clustering. Fig.3(d) shows area formed by motion blocks.



Fig.2 Fruit and vegetable store (a) Original Frame (b) Result of motion detection (c) Result of clustering algorithm (d) Motion area in frame.



Fig.3 Super store (a) Original Frame (b) Result of motion detection (c) Result of clustering algorithm (d) Motion area in frame.

VI. CONCLUSION AND FUTURE WORK

This paper presents a new motion detection approach based on the motion field in compressed domain for the latest video coding standard H.264. Several experimental results demonstrated the good performance and efficiency of the proposed approach, which can be directly applied to many video applications, either at encoder side or decoder side. In

our approach, the motion vector and partitioned block size are used as segmentation cues, which have the resolution at a 4x4 block level. In this approach, treatments such as vector median filtering and forward block vector accumulation are used on the motion field. Then, measure of residue is used to detect true motion blocks and finally clustering algorithm is applied to extract motion area. Algorithm provides high accuracy in detecting motion blocks compare to other algorithms of motion detection but timing of this algorithm is very high. For real time application optimization can be done by applying DCT energy measurement before MV processing. This paper just provides a fundamental investigation on video segmentation in H.264 compressed domain, some further analysis, such as tracking, recognition will be the future focus.

ACKNOWLEDGEMENT

I would like to express my deep gratitude to Mr. Sudhir Bhaduria, project supervisor and internal guide Professor Bhavesh Soni, Asst Prof at Ganpat University, Mehsana, India for their patient guidance, enthusiastic encouragement, assistance in keeping my progress on schedule and useful critiques of this research work.

REFERENCES

- [1] Dan Schonfeld and Dan Lelescu, "VORTEX: Video retrieval and tracking from compressed multimedia databases—multiple object tracking from MPEG-2 bit stream," *Journal of Visual Communication and Image Representation*, vol. 11, no. 2, pp. 154–182, 2000.
- [2] Francesca Manerba, Jenny Benois-Pineau, Riccardo Leonardi, and Boris Mansencal, "Multiple moving object detection for fast video content description in compressed domain," *EURASIP J. Adv. Signal Process.*, vol. 2008, no. 1, pp. 1–13, 2008.
- [3] H. Zen, T. Hasegawa, and S. Ozawa, "Moving object detection from MPEG coded picture," *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, vol. 4, pp. 25–29 vol.4, 1999.
- [4] Ashraf M.A. Ahmad, Duan-Yu Chen, and Shu-Yin Lee, "Robust object detection using cascade filter in mpeg videos," in *Proceedings of the IEEE 5th International Symposium on Multimedia Software Engineering (ISMSE)*, 2003, pp. 196–203.
- [5] Lu Y, ZHANG Z Y, Liu Z, Han Z M. Motion characteristic based object segmentation in the H.264 compressed domain[J]. *Journal of Optoelectronics.Laser*, 2009, 5(20): 668–671.
- [6] ITU-T Rec. H.264 (05/2003) – Series H: Audio-visual and Multimedia Systems.
- [7] W.Zeng, J.Du, W.Gao, Q .Huang, "Robust moving object segmentation on H.264 compressed video using the block-based MRF model ", *Real-Time Imaging* 11, 290-299,(2003)
- [8] Wonsang You, M.S.Houari Sabirin, Munchurl Kim, "Real-time detection and tracking of multiple objects with partial decoding in H.264/AVC bitstream domain", *Proceedings of SPIE(2009)*, Vol.7244, (2009)
- [9] Chris Poppe, Sarah De, Bruyne, Tom Paridaens, Peter Lambert, Rik Van de, Walle, "Moving object detection in the H.264/AVC compressed domain for video surveillance application", *Journal of Visual Communication and Image Representation*, Vol.20, Issue 6, Page 428, (2009)
- [10] C.Solana-Cipres, G.Gernandez-Escribano, L.Rodriguez-Benitez, J.Moreno-Garcia, L.Jimenez-Linares, "Real-time moving object segmentation in H.264 compressed domain based on approximate reasoning ", *International Journal of Approximate Reasoning*, Vol.51, Issue.1, Page 99-114, (2009)
- [11] Lu Y, ZHANG Z Y, Liu Z, Han Z M. Motion characteristic based object segmentation in the H.264 compressed domain[J]. *Journal of Optoelectronics.Laser*, 2009, 5(20): 668–671.
- [12] Zhi Liu, Zhao yang, and Liqun Shen, "Moving object segmentation in the H.264 compressed domain," *Opt. Eng.*, Vol. 46, 2007.

- [13] Iain E. Richardson, "H.264 and MPEG-4 Video Compression Video Coding for Next-generation Multimedia", 2003.
- [14] Fei HAO, Zhenjiang MIAO, Ping GUO, Zhan Xu, "Real Time Multiple Object Tracking Using Tracking Matrix", Proceedings of the 2009 International Conference on Computational Science and Engineering, Vol. 02, 2009.

IJERT