Volume 4, Issue 4, April 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Real Time Moving Object Detection and Tracking in H264 Compressed Domain for Video Surveillance

Mansi Patel
Dept.VLSI & EMBEDDED SYSTEM
Ganpat University, Mehsana, INDIA

Abstract-A real-time moving object detection and tracking algorithm on H.264 compressed video streams for IP video surveillance systems. The goal is to develop algorithms which may be useful in a real-life industrial perspective by facilitating the processing of large numbers of video streams on a single server and to reduce the computational complexity and memory requirements by extraction information directly from coded video stream .The proposed algorithm detects and segments regions having motion based on motion vectors embedded in the video stream without full decoding process and reconstruction of video frames. It includes spatiotemporal filtering. Spatial filtering detect moving object and temporal filtering tracks the object. The algorithm was tested on indoor surveillance H.264 sequences.

Keywords-H.264 compressed domain, video surveillance, segmentation and tracking , partial decoding

I. INTRODUCTION

The wide applications of video surveillance call for automated and efficient analysis, where detection and tracking of moving objects are the essential steps and can be performed in either pixel domain or compressed domain. Since most videos captured nowadays are compressed for storage and transmission, compressed domain techniques have recently drawn increasing research attention. In video analytics application in surveillance, motion detection can be base for many other application of analyzing video stream including object recognition, face detection and recognition etc.

For MPEG compressed domain, moving object segmentation algorithms usually rely either on motion vectors (MVs), residual information (DCT coefficients), or both. However, for H.264 video stream, full decoding is necessary to get residual information, so nothing but MVs is used to segment moving object in all known algorithms. Several methods has been proposed for H.264/AVC compressed domain moving object detection in literature as it provides information of motion in compressed form in terms of motion vector. Those techniques have distinct advantages over their pixel domain counterparts in convenience and efficiency, since they can use the information explored by the compressing process and avoid full video decoding. According to the required degree of partial decoding, those methods can be grouped into three categories, i.e. entropy decoding level (ED level) methods [1, 2], macroblock decoding-level (MD level) [3] and frame decoding-level (FD level) methods [4].

Entropy decoding methods includes only decoding of entropy encoding like cavlc, exp-golomb. Macroblock level decoding includes decoding of whole macro block, that is one step ahead in decoding and requires more time. Frame decoding includes decoding of whole frames like Intra frames.

The proposed algorithm is a real-time ED level algorithm for the segmentation and tracking of moving objects in the surveillance videos encoded under the H.264 baseline profile with only one slice per frame. The spatial filtering filters out macroblock having motion based on information like total number of DCT coefficient, MV combined with temporal filtering which tracks for the same object in consecutive frames from MV and DCT correlation. The last process to segregate the real object from the false object is to analyze occurrence frequency. The real object appears more frequently and hence by thresolding the frequency it can be detected.

II. DIFFERENT APPROACHES

The majority of work on change detection and motion detection for surveillance has naturally been performed in the pixel domain based on reconstructed images. Only a minor part of the papers have been devoted to the possibilities of doing this in a compressed domain.

A. *Pixel domain approaches*

A comprehensive survey of change detection algorithms in pixel domain was provided by Radke [5]. This survey presents a wide range of change detection algorithms along with methods for pre-processing of input images and post processing of the resultant change masks. The survey covers a wide area of applications of change detection, including video surveillance. Another recent study of pixel domain change detection is provided by Parks and Fels [6], where the focus was on background subtraction algorithms.

B. *Compressed domain aprroaches*

Compressed domain approaches are new and less explored compared to pixel domain processing. One example is method presented by You et al. [3] minimizes the dissimilarity energy and partially decodes frames for tracking the moving object specified by users. Due to the uniformity assumption on the velocity within one group of pictures (GOP), this method requires the intra period, which is defined as the distance between two consecutive I-frames, to be relatively small.

C. *Comparison of compressed domain and pixel domain approaches*

Both approaches to video processing have their advantages and disadvantages. Change detection or motion detection on reconstructed images is conceptually appealing and in simple implementations can be fast, usually at some cost of reliability. The methods most commonly applied in video surveillance systems seem to be the background subtraction methods based on various statistical background models. Pixel domain approaches are independent of the video coding standard used and there is a great potential for very reliable detection using advanced pre- and post processing. Although this method can be made reliable it has a serious disadvantage in a video surveillance system where all the video is stored and transmitted in a compressed format, and where video reconstruction might handicap the system performance. Compressed domain processing avoids the full decoding and reconstruction of the video, which provides a potential for real time processing of multiple video streams on one server. Compressed domain processing also has the advantage of extracting video stream data, which has been generated using the original non-compressed data, which will not be available when processing a decoded stream. Thus the lossy video coding introduces a noise component, which will have to be dealt with.

III. SPATIOTEMPORAL FILTERING

The Fig 1 shows the system block diagram which shows the probabilistic spatiotemporal macroblock filtering (PDMF) process. It is the process of filtering background macroblocks on the basis of their spatial and temporal properties, in order to rapidly segment object regions in the macroblock unit and track each object roughly in real time. The process is organized as spatial filtering, blocks clustering and temporal filtering.

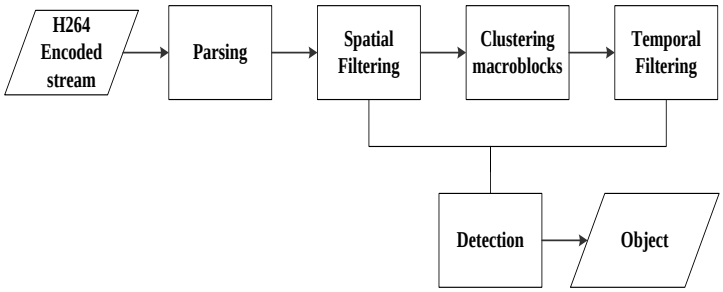

Fig.1. System block diagram.

A. *Spatial filtering and block clustering*

We assume that a video is encoded with the H264 baseline profile. It is observed in P-frames that most parts of the background tend to be encoded into skip macroblocks since they theoretically have no residual error between the motion-compensated prediction block and the source macroblock. Contrary to background macroblocks, most parts of objects which prominently move in a scene tend to be encoded into non-skip macroblocks since most of macroblocks inside the moving objects are split into several sub-blocks or have residual errors in the motion prediction of encoding process due to dynamic change in shape or color. Thus, we can reduce search space by filtering off all skip macroblocks which are considered as potential parts of the background. Then, the remaining macroblocks are naturally clustered by their mutual connectivity. In other words, we obtain several fragments, called block groups, which consist of non-skip macroblocks connected in the horizontal, vertical, or diagonal directions. The process is called block clustering. We need to further eliminate erroneous block groups which substantially belong to the background, and to merge homogeneous block groups of the foreground into one independent object. We have observed that most block groups, which consist of just one isolated macroblock or do not contain any non-zero IT coefficient, belong to the background. In this reason, we additionally filter off such block groups; the process is called spatial filtering and the surviving block groups are the area which has motion. The erroneous blocks are eliminated and tracked by temporal filtering further

B. *Temporal filtering*

Temporal filtering further filters out the erroneous blocks and tracks the object in consecutive frames. Which includes two steps, first is to project the motion vector on the previous frame and filter out the true motion vector and by projecting true motion vector on previous frame the object can be tracked. Second step includes correlating the DCT of two matched object and further reducing the false alarm .Followed by last step of frequency analysis which distinguish real object from false alarm.

1) Motion vector projection and estimating motion vector reliability.

Since motion estimation is performed from a coding point of view, MVs do not always capture actual motion, but can contain a lot of noise. To reduce the effect of noisy MVs, we propose to estimate the reliability of MVs based on

the temporally co-located MVs in surrounding frames. In particular, by projecting MVs from surrounding frames to the current frame, multiple vectors representing the motion in the current frames are obtained, and thereafter compared. First, the MVs belonging to the next frame in display order ft+1 are backward projected to the current frame ft by moving each block bt+1,i,j according to its corresponding MV, as depicted in Fig. 2. By calculating the weighted average of the projected MVs for each block based on the size of the overlapping area, a projected MV bmvt,i,j is obtained. Next, the similarity of this projected bmvt,i,j and the co-located MV mvt,i,j of the current frame is calculated using the following formula, which is partially based on sarah[8].

$$SimB(t,i,j) = e^{-\frac{|mv_{t,i,j} - bmv_{t,i,j}|^2}{(|mv_{t,i,j}| + |bmv_{t,i,j}|)^2}}$$

When the projected and co-located MV are similar, which typically corresponds to vectors representing true motion, SimB(t, i, j) will obtain a value close to one. However, in case one of these vectors is noisy, the similarity will be much lower. Note that as the sizes of both vectors are incorporated in the denominator, a difference between small vectors will reduce the similarity more than when this difference is encountered for vectors having a large magnitude. In the next step these obtained similarities are used to localize the noisy vectors, and to diminish their effect.

After this process we get the Motion vector which shows the actual motion. Hence from this motion vector we can track the location of object of the current frame in previous frame. The true motion vectors are average out for each object and projected on previous frame to locate the current object. The accuracy can be further increased by using below process. It should be noticed that there is a trade-off between performance and computational complexity, which may be controlled according to whether to carry out the following process or not.

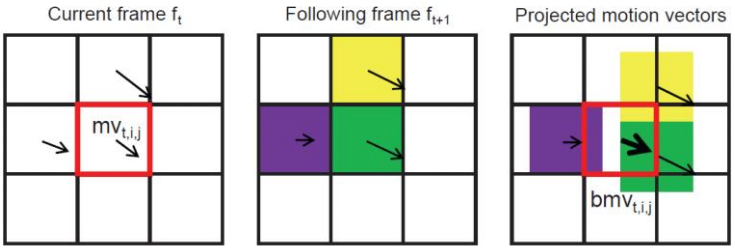

Fig.2. Backward projection of MVs.

2) Correlation between DCT values of object detected in previous and current frame

By correlating the values of object in current frame with that of in previous frame we can compare the texture similarity energy as DCT is nothing but the luma samples of the object. Hence the energy difference between the same object in current and previous frames should be least and energy difference between different objects would be higher.

$$E_c = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(\sum_{i=1}^{n}(x_i - \bar{x})\text{^}2)(\sum_{i=1}^{n}(y_i - \bar{y})\text{^}2)}}$$

Here, xi = Current objects ith block
yi = ith block of the projected object on previous frame
n = total number of macro block in object in current frame
Ec=Correlation energy.
If Ec<ɗ it is matched with project object in previous frame. If it is not than its not matched with the projected object. where ɗ = Maximum allowed difference in object and projected object.
Temporal filtering not only tracks the object but it filter out the erroneous blocks and detect only real object.

3) Frequency analysis

It has been seen that the real object appear continuously in the consecutive frame rather than false alarm. False alarm which have part of background object mainly due to luminance changes appear and disappear in the consecutive frame. Probability analysis of the appearance of object in particular observation period will distinguish false object from real object. Here observation period is taken equal to GOP size and minimum frequency is taken 50% for implementation. Minimum threshold frequency can be varied and is trade off between reducing false alarm and capturing smallest activity across the observation period. The real object can further decoded and by applying canny edge detectio006E and other feature detection algorithm can be recognized.

IV. EXPERIMENTAL RESULT

The proposed method has been tested on several indoor video sequences encoded using H.264 encoder. The encoder configuration set as follows: baseline profile (including non B frame), the interval of I-frames is 32, and quantization parameter (QP) is 26.

Fig 3 shows the result of spatial filtering. The areas which do not have motion has been filtered out at great extent. But still there are areas which have high luminance change and have been detected as moving object inspect of no object. Fig 4 shows its consecutive frame in which by temporal filtering the area with false alarm has been eliminated and the real object is tracked by projecting it on previous frame. Fig 5 shows the same object has been tracked for consecutive frames eliminating the false alarm due to luminance changes.

Table shows timing analysis of different domain approaches. First approach is foreground background subtraction technique[9], second uses Gradient difference to detect object boundary, both are on raw domain which is compared with the third method .It is the proposed method on compressed domain.

Fig.3. Result after spatial filtering (shows macroblock having motion)

Fig.4. Result after temporal filtering (erroneous blocks are eliminated)

Fig.5. Same object tracked in consecuitive frames

Table.1.Timing analysis

Method Used for object detection	**Time required per frame including decoding/partial decoding(ms)**	**Performance cost**
Forground/backgraound subtraction(raw domain)	70	High
Gradient difference method(raw domain)	55	High
Proposed method(compressed domain)	15	Very low(Real time)

Timing analysis shows how time efficient the proposed method is and due to less time required for computation, this method can be used for real time application

V. CONCLUSION

This paper present a fast and efficient algorithm for moving object detection and tracking for H264 surveillance video for real time applications, in which motion vector and DCT coefficient are used to detect moving object and track it in consecutive frames. the experimental result and timing analysis shows that the proposed method can detect and track object in real time and very much less time compare to pixel domain approaches..

ACKNOWLEDGEMENT

I would like to express my deep gratitude to Mr. Sudhir Bhadauria, my research supervisor, and Professor Bhavesh Soni, internal guide at Ganpat University for their patient guidance, enthusiastic encouragement and useful critiques for this research work.

REFERENCES

[1] Z. Liu, Y. Lu, and Z. Zhang, "Real-time spatiotemporal segmentation of video objects in the H.264 compressed domain," Journal of Visual Communication and Image Representation,vol. 18, no. 3, pp. 275–290, 2007.

[2] C. Poppe, S.D. Bruyne, T. Paridaens, P. Lambert, and R.V.d.Walle, "Moving object detection in the H.264/AVC compressed domain for video surveillance applications," Journal of Visual Communication and Image Representation, vol. 20, no. 6, pp. 428–437, 2009.

[3] W. You, M.S.H. Sabirin, and M. Kim, "Moving object tracking in H.264/AVC bitstream," in International Workshop on Multimedia Content Analysis and Mining (MCAM), 2007, pp.483–492.

[4] C. K¨as, M. Brulin, H. Nicolas, and C. Maillet, "Compressed domain aided analysis of traffic surveillance videos," in Third ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC), August 2009, pp. 1–8.

[5] R. J. Radke, , S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: A systematic survey. IEEE Trans. Image Proc., 14(3):294–306, 2005.

[6] D. H. Parks, , and S. S. Fels. Evaluation of background subtraction algorithms with post-processing.

[7] S.D. Bruyne, C. Poppe, T. Paridaens, P. Lambert, and R.V.d. Walle, "Estimating Motion Reliability To Improve Moving Object Detection In The H.264/Avc Domain" in Multimedia and Expo, 2009. ICME 2009.

[8] Thi Thi zin," A Novel Probabilistic Video Analysis for Stationary Object Detection in Video Surveillance Systems" in IAENG International Journal of Computer Science.

