Event Detection by Spatio-Temporal Indexing of Video Clips

Shan Du, Choudhury A. Rahman, Saika Sharmeen, and Wael Badawy, *Member, IACSIT*

Abstract—**With the continuous recording of video data in current surveillance systems, it is almost impossible to quickly identify frames where events of interest did occur within a camera scene. This paper presents the concept of Spatio-temporal Indexing, which is a novel video indexing and retrieval technique that can be used for event detection. Spatio-temporal Indexing allows the users to rapidly retrieve the video clips that contain events of interest from a given video library. The proposed indexing technique analyzes the video and stores the Spatio-temporal Indexes that will be further processed to retrieve video clips queried by users. The proposed technique was tested on hours of video recordings. The results obtained provide 99.9% accuracy for event detection and retrieval. The average processing time is 3 seconds to create index for 10 minutes of videousing Intel i5-2400 processor.**

Index Terms—**Incident investigation, video indexing, video summarization, video retrieval.**

I. Introduction

With the continuous recording of video in current surveillance systems, large amounts of video data are produced. Effective indexing and retrieval from surveillance video databases are very important. Traditionally, the content of video has been represented either by simple textual techniques or low-level image features such as color histogram, shape representation, transform domain features and visual summary information based on segmentation of video into smaller units. In surveillance video, we are more interested in dynamic contents. We want to rapidly retrieve the video clips that contain events of interest. Thus, we need event- or activity- based representation.

Video retrieval system using audio features was introduced in [1] to detect human's reaction and realize a browsing function for the recorder. It enables completely automatic detection of sports highlights. A novel technique was used for constructing the video table of content (ToC), video highlights and video index as well as integrating them into a unified framework for video summarization and retrieval. This technique can be used in surveillance to identify special events such as gun shots or hits. But for normal behaviour, it lacks accuracy. Moreover, recording sounds violates the privacy act in Canada [2], which allows video recording butwith no audio recording. Also, cameras on the top of a building cannot record sounds from the street due to the long distance.

Temporal slice model was introduced in [3] for detecting three essential types of camera breaks, namely cuts, wipes, and dissolves. It is based on the analysis of temporal slices which are extracted from the video by slicing through the sequence of video frames and collecting temporal signatures. In [4], the authors also used the video slice extraction approach accompanied by the pixel difference to detect the shot breaks. Their proposed framework generates video skimming that guarantees both the balanced content coverage and the visual coherence.

There are some surveillance systems using automatic image understanding to extract information from the surveillance data such as the IBM smart surveillance system. It provides capability to automatically monitor a scene, manage the surveillance data, perform event based retrieval, receive real time event alerts through standard web infrastructure and extract long term statistical patterns of activity [5].

The use of low-level features of frames (color histogram) for highlights extraction in sports video was introduced in [6]. An algorithm to detect salient motion in complex environments by combining temporal difference image and a temporal filtered motion field was introduced in [7]. It distinguishes between interesting (salient) motion (e.g., a person) and uninteresting motion (e.g., swaying branches). Another technique introduced in [8], [9] is a line-scan technique to build image signature to detect the defects.

The concept of TimeLine was introduced in [10] where the spatio-temporal line is used to create a storyboard to visualize the video content and identify events using change detection algorithm. However, this approach is sensitive to the location of the sampled line. In this paper, we overcome this problem by using a new approach that depends on more than one sampled lines and performs the same procedure for each line, which will increase the computational time but it will enhance the results and guarantee better performance.

This paper presents a novel video indexing and retrieval technique where the video frames are processed at certain indexes to identify the events and generate compressed index patterns. Furthermore, these patterns are used together to retrieve video clips queried by users.

The user may query Spatio-temporal Index in several ways depending on the application area of the surveillance system and also depending on the user specific interest. For example, if the surveillance camera is used in a speed bump to monitor cars entrance and exit in a parking lot, the user in this case canquery the Spatio-temporal Index to get the number of cars during a specific time using a simple query like "how many cars passing from time t_1 to time t_2". Another example could be when the surveillance cameras are installed to monitor indoor hallways. The user can use queries regarding people's activities in certain areas of the building. This can be queries

Manuscript received February 4, 2013; revised April 8, 2013.

The authors are with IntelliView Technologies Inc., 327-41 Avenue NE, Calgary, Alberta, Canada T2E 2N4 (e-mail: du@intelliview.ca, rahman@intelliview.ca, sharmeen@intelliview.ca, badawy@intelliview.ca).

DOI: 10.7763/IJCTE.2014.V6.831

such as “show me the persons with blue-shirts who enter the building today” or “show me the persons taller than two meters crossing the door”.

The remaining of the paper is organized as follows. Section II describes the proposed video indexing technique - Spatio-temporal Indexing. Section IIIand Section IV present the performance of the proposed technique. Section V concludes the paper.

II. The Proposed Method

A. Overview

The proposed technique retrieves the video contents based on the dynamic contents of the video data. The index is always triggered by the dynamic behaviour. The proposed indexing scheme will address queries on the dynamic contents of video clips but not on the static contents, i.e., the system will be more efficient to answer a question such as “retrieve a video clip of red cars” assuming that these cars are moving or stopped from moving. The system can also answer partial query such as “if there was a red car at a specific time”. But the system cannot answer questions such as “retrieve a video clip which has flowers photos” since flowers are stationary object.

To explain the proposed Spatio-temporal Indexing, we consider a speed bump camera looking at a street as shown in Fig. 1. In the top part of the figure, we show a selected sampling position at the center of the image (the vertical or the horizontal line at the center). We extract one line from the selected position in every frame and put them together to form an image that is the index image (shown in the lower part of the figure) such that each vertical (or horizontal) line in the index image is a spatio-temporal line of the video. In case of surveillance cameras with fixed parameters (pan-tilt-zoom) and fixed background, the resulting index image can tell the users where the events happened.

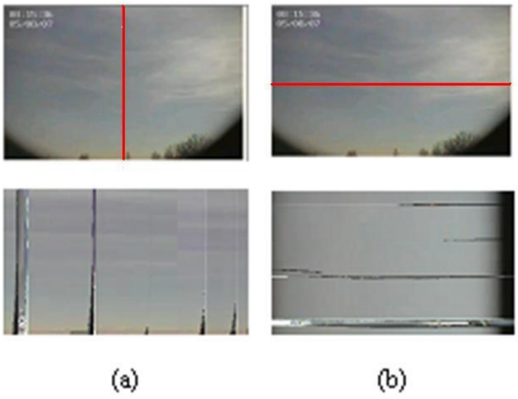


Fig. 1. Samples of index patterns: (a) Video vertically sampled at center line over the temporal axes. (b) Video horizontally sampled at center line over temporal axes.

Fig. 1 shows the index patterns of two minutes video recorded from speed bump camera where the distinct changes in the lower images represent an event (cars or persons crossing). It becomes obvious now that users can use these indexes to distinguish between the video scenes that do not contain events and the ones that have motion or crossing.

Based on that concept, our proposed tool provides the users with the ability of retrieving the scenes which contain events. Fig. 2 shows three corresponding frames of three different lines extracted from an index pattern using the scrollbar on the left of the index image. When the line is inside a slice of image that has different rhythm, it is corresponding to a car crossing (Fig. 2(a)). When the line is on the no defect portion, it is corresponding to a frame without any information (Fig. 2(b)). When the line is on small defects that are on the left of the index image, it is corresponding to a bike crossing from the left of the camera (Fig. 2(c)). Fig. 3 shows the use of multi-indexes. We place multiple sampling lines to generate multiple indexes.

B. Index Generation

The index has a representation of every frame but it only stores data when there is an event of interest. Here, we show the extended version of the index to make it easier for the readers to understand the operation. The index as illustrated in this paper will include both active (i.e., event) and not-active (i.e., no-events) data. However, in the real implementation the stored index will include only active data.

Let’s consider one shot video. Hence, all frames have the same background. We assume that the video sequence is sampled vertically at center line such that $X = FrameWidth/2$.

Our approach considers the first frame as a key frame and takes its sampled line pixels as a reference array$RefLine(X, y, 1)$. Here, $t = 1$.

$$RefLine(X, y, 1) = \sum_{y=1}^{y=frame\ height} FrameArray(X, y) \quad (1)$$

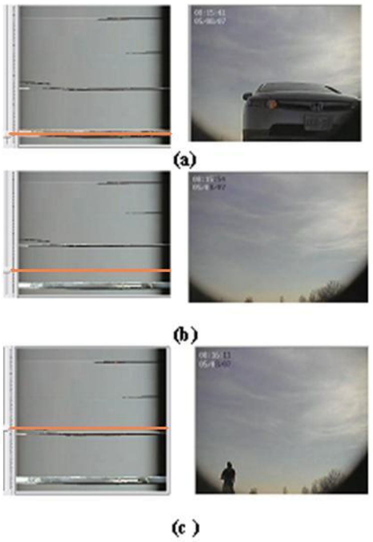


Fig. 2. Three frames correspond to different positions of the index image: (a) Line inside a slice of image that has different rhythm corresponds to the car crossing. (b) Line on the no defect portion corresponds to a frame without any information. (c) Line on small defects on the left of the index image corresponds to a bike crossing from the left of the camera.

Then we compute the pixel differences $RefPixDiff$ between sampled key frame line and the next ten frames lines and obtain the maximum to eliminate the camera noise.

$$RefPixDiff = MAX(RefLine(X,y,1)\text{-}RefLine(X,y,t))_{t=2 \text{ to } 11} \quad (2)$$

Based on $RefPixDiff$ value, we decide two thresholds $LoThreshold$and $UpThreshold$. The first threshold is to detect small changes in the scanned line and the second one is to detect the huge changes. Experimental results show that $LoThreshold$ should be one and a half of $RefPixDiff$ value and $UpThreshold$ should be 3 times of $RefPixDiff$.

If the pixel difference between any next frame line $Line(X, y, t)$ and $RefLine(X, y, 1)$ exceeds the lower threshold, we say that it is a small object crossing through. And if it exceeds the upper threshold, we say it a big object.

$$PixDiff(X, y, t) = \sum_{y=1}^{y=frame\ height} RefLine(X, y, 1) - LineX,y,t \qquad (3)$$

$$IFPixDiff(t) > LoThreshold, \quad smallobject$$
$$IFPixDiff(t) > UpThreshold, \quad bigobject$$

Moreover, we sample the video sequence at a constant rate of 10 frame/sec. We assume that there is no sudden change in frames, so we neglect any sudden object detection that lasts less than 10 frames. Hence, our method is immune to sudden changes of illumination and contrast. Also, if small object is detected over long time (more that 120 sec, 1200 frame) we consider that it is a normal change in illumination and take another key frame and update the reference value $RefPixDiff$ and as well as the thresholds.

C. Multi-Indexing Scheme

To be able to effectively index video clips, we use a multi-indexing scheme where the proposed index is computed at different locations. Many attributes of a crossing object can be easily obtained. For example, we can easily tell the speed of an object by knowing the number of frames occupied by this object. It also can be used to indicate the object size and its location with respect to camera position. Fig. 3 illustrates the multi-indexes generation and the corresponding index pattern for each index.

Moreover, the system is also capable to compute on-demand (based on a user's query request to save time from the pre-processing stage) with on-demand processing; the system will have more indexes with more different queries.

D. Index Format

The index key consists of *n*-tuple (CLIP_ID, TIME_STAMP, PIXDIFF, INDEX_ID, QORM), where CLIP_ID is a pointer to the Video File, TIME_STAMP is the time stamp of the Key, both CLIP_ID and TIME_STAMP are used to play the part of different clips that satisfy the user query, IINDEX_ID is a pointer to the Index Image.

QORM are extra fields that the user can select to populate the index such as the dominant color, the disturbance size or the numeric value of a license plate of a vehicle which can be computed through a third party interface. It also can be a user defined annotation.

III. Experiments

To illustrate the processing of object retrieval, we will only process the indexes to retrieve the following queries.

A. Relative Object Size

In Fig. 3(e) and Fig. 3(h), we can notice that there are disturbances repeated in both index patterns (the left part of each image). It means that the indexes have same data at this time. And some disturbances only appear on the right part of Fig. 3(h) pattern. It means the indexes on the right side of the pattern have different data corresponding to the events that only occurred on the right side of the camera view.

The first disturbances are caused by an object that has a relatively big size that affects both indexes of the left and the right scene. The disturbances that appear only on Fig. 3(h) pattern are caused by objects that have a small size relatively and cross the camera from the right side.

Fig. 4 illustrates the frames corresponding to the repeated disturbances in both left and right indexes and the disturbances occurred in right index only.

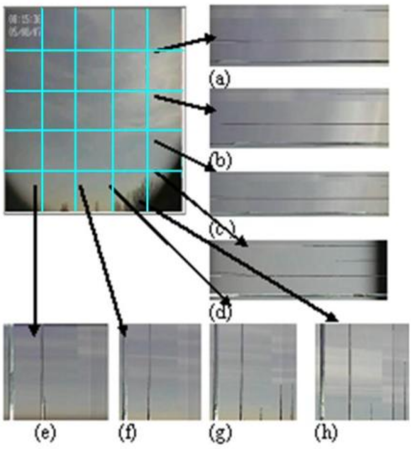


Fig. 3. Generation of multi-indexes using multiple sampling lines.

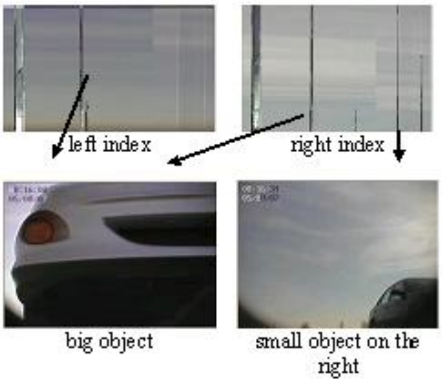


Fig. 4. Relative object size estimation.

B. Relative Object Speed

In Fig. 5, we show an index pattern that has five disturbances. Some of them last longer than others. It means that the less the width of the disturbance, the fast the object is moving.

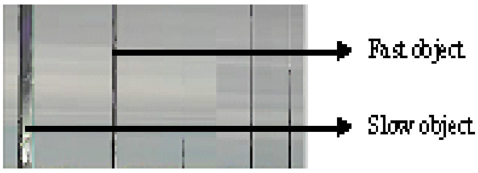


Fig. 5. Index pattern with five disturbances.

C. Color-Based Retrieval

For video surveillance systems, queries like "how many events occurred" or "how many times a person with black suit crossed" are possibly answered by analyzing the stored data in the indexes.

Fig. 6 illustrates an index pattern of indoor video surveillance camera where there are 3 crossing occurred and all of them made by a person wearing a black suit.

D. License Plate Shot Extraction of Crossing Vehicles

For the automatic license plate recognition (ALPR), we need to extract a clear license plate image of a passing vehicle first. Using the proposed index patterns, we can easily do so. When a car passing, the lower portion of the video frames will change gradually with respect to key frames until the car completely cover the camera view (the underneath car frames), where there is a sudden change. Then, the value of $PixDiff(t)$ will exceeds the value of the $UpThreshold$. We will take this frame as a second reference array $RefLine\_2(X, y, t)$ and the difference between next frames and this frame will be called $PixDiff\_2(t)$.

When the car start to move away from the speed pump, the value of $PixDiff\_2(t)$ will increase (because the underneath car frames is gone) until we have both $PixDiff\_2(t)$ and $PixDiff(t)$ values more than $UpThreshold$. At this moment, we have a frame which is not close to underneath car frames nor original frame which means that this frame represents a car back.

Finally, the car moves away when the value of $PixDiff(t)$ decreases below the $LoThreshold$.

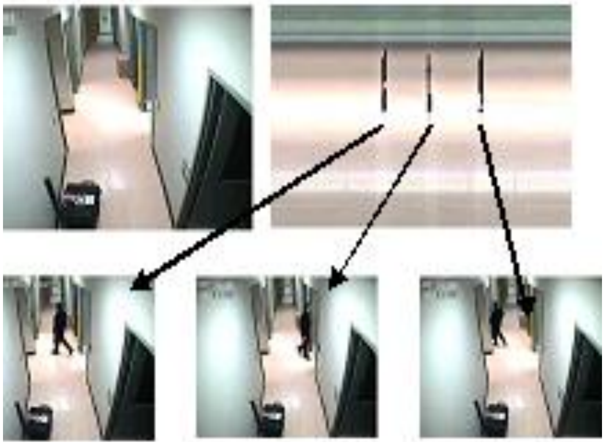


Fig. 6. Index pattern with three disturbances due to black object motion.

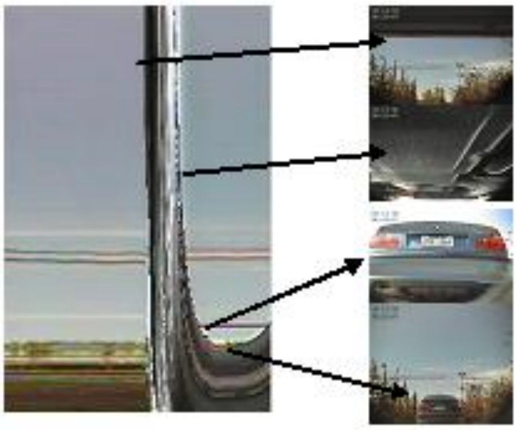


Fig. 7. Four frames of this special pattern with four status of car crossing.

Fig. 7 shows the four frames of this special pattern with four status of car crossing. Note that in figure number (3) the license plate of the car is pretty clear which means that we can use this technique to identify the frames that contain the car license plate. Fig. 8 shows another example of moments of index pattern and their corresponding events.

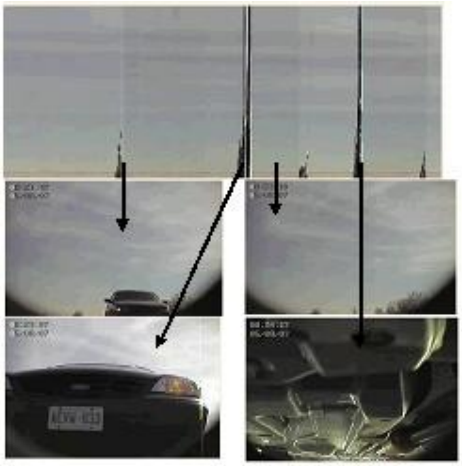


Fig. 8. Four moments of index pattern and their corresponding events.

E. Query Once and Retrieve Many (QORM)

QORM is used to store the attributes of a video that are obtained in user-requested queries. For example a user requests a query based on the blue color of an object. Whenever this query finds a match in the indexes, it will store the blue color attribute in the QORM. The next time when the user requests a query with a blue color object and whose size is more than *x*, the query will search only in indexes where the QORM has an attribute of blue stored in them and it does not have to do image processing search for the whole indexes again. In this way, the system retrieval efficiency is increased.

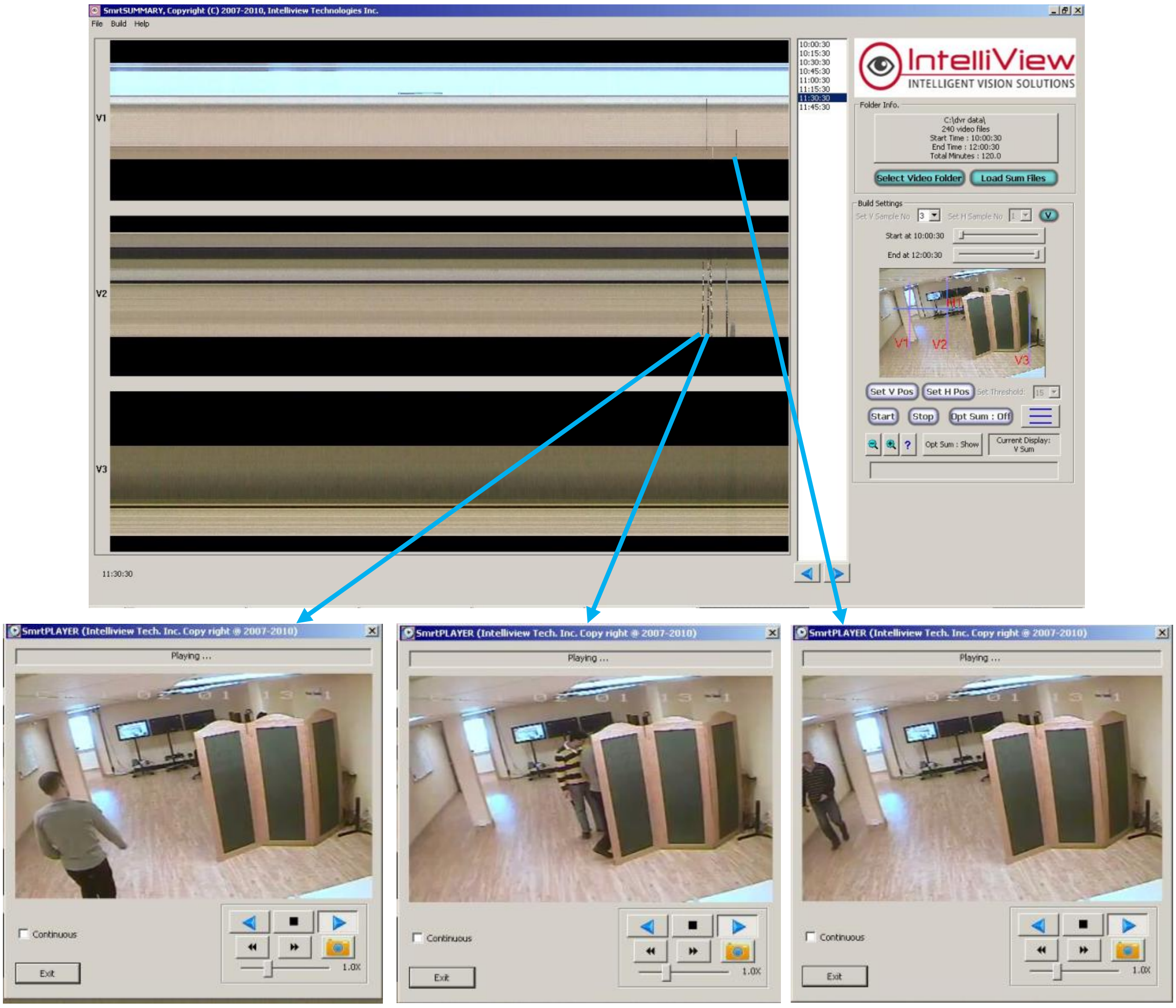
IV. Prototype andPerformanceAnalysis

The Spatio-temporal Indexing is implemented in C++ and tested on stored videos at different conditions (lighting, speed pump camera, inside door camera, outdoor camera). The video aspect ratio is 320×240 and its sampling rate is 10 frame/sec. So it needs 76.8 Kbytes (if we sample vertically) to generate index pattern of 32 secondsand then stores it as a bitmap file. The file size of the index is a function of the dynamics of the video.

The software systemimplemented by IntelliView(as shown in Fig. 9) is user friendly. It is capable of browsing the indexes in a storyboard that describes the video content. By just clicking on any point on the storyboard, the software displays the corresponding moment of video. So it provides users with easy way to retrieve the moments that contain events.

The accuracy rate of event detection is 99.9% in case of using multi-indexes and the size of the object is bigger than the difference between two indexes.

Using Intel i5-2400 CPU and2.73GB RAM with Windows XP, the system computes indexes in 2 second for creating 8 indexes for 1 minute video clip (i.e., 600 frames). The process time will increase if we want to analyze certain clips to extract more information about objects attributes or to identify special pattern as mentioned in the previous section. (4 second for 1 minute video in case of the car license plate frames).



SmrtSUMMARY, Copyright (C) 2007-2010, Intelliview Technologies Inc.

Folder Info.: C:\dvr data\ 240 video files Start Time : 10:00:30 End Time : 12:00:30 Total Minutes : 120.0

Build Settings: Set V Sample No 3; Start at 10:00:30; End at 12:00:30; Set Threshold: 15; Opt Sum : Off; Current Display: V Sum

SmrtPLAYER (Intelliview Tech. Inc. Copy right @ 2007-2010) — Playing ...

Fig. 9. An incident query.

Compared with other approaches, our proposed approach has the following advantages:

1) Visualizing the video contents by only scanning the spatio-temporal line.
2) By analyzing the index image, we can automate the event detection process and reduce the whole video to only the scenes that contain events.
3) By analyzing the index image, we can find out certain attributes like size, speed, or color.
4) By sending the index to the client over network, users can specify the certain scenes of interest for the video to be sent. Hence, traffic overhead required for sending a whole video is reduced.
5) Quick and efficient retrieval of video clips of interest can be achieved using the Query Once and Retrieve Many (QORM) method that was developed for Spatio-temporal Indexing.

V. Conclusion

This paper presents the Spatio-temporal Indexing for video retrieval based on dynamic events. The system performance showed that it is very efficient for indexing and retrieving video data, especially for surveillance purpose. Since it focuses only on the moments of events and stores them as indexes, the proposed method is fast and does not require a huge storage capacity to store all the video data.

Acknowledgment

We would like to thank Dr. Mohamed ShehataofBenha University, Cairo, Egypt for his help on this paper.

References

[1] I. Otsuka, K. Nakane, A. Divakaran, K. Hatanaka, and M. Ogawa, "A unified framework for video summarization, browsing and retrieval," Technical Report TR2004-115, Mitsubishi Electric Research Laboratory Inc., Sep. 2004.

[2] *Personal Information Protection and Electronic Documents Act*, 2000.

[3] C.-W. Ngo, T.-C. Pong, and R. T. Chin, "Video partitioning by temporal slice coherency," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, no. 8, pp. 941-953, Aug. 2001.

[4] S. Lu, I. King, and M. R. Lyu, "A novel video summarization framework for document preparation and archival Application," in *Proc. IEEE Conf. on Aerospace*, 2005, pp. 1-10.

[5] A. Hampapur, "S3-R1: The IBM smart surveillance system - Release 1,"in *Proc. the ACM SIGMM Workshop on Effective Telepresence*, 2004, pp. 59-62.

[6] B. Li and M. I. Sezan, "Event detection and summarization in sports video," in *Proc. IEEE Workshop on Content-based Access of Image and Video Libraries*, 2001, pp. 132.

[7] Y.-L. Tian and A. Hampapur, "Robust salient motion detection with complex background for real-Time video surveillance," in *Proc. IEEE Workshop on Motion and Video Computing*, vol. 2, 2005, pp. 30-35.

[8] C. Baykal and G. A. Jullian, "Detection of defects in textures with alignment error for real-time line-scan web inspection systems," in *Proc. the 45th Midwest Symposium on Circuits and Systems*, vol. 3, pp. 292-295, 2002.

[9] C. Baykal and G. A. Jullian, "On the use of hash function for defects detection in textures for in-camera web inspection systems," in *Proc.*

IEEE International Symposium on Circuits and Systems, 2002, vol. 5, pp. 665-668.

[10] M. Nunes, S. Greenberg, S. Carpendale, and C. Gutwin, “What did I miss? Visualizing the past through video traces,” Report 2007-855-07, Dept. of Computer Science, University of Calgary, 2007.

Shan Du received the M.S. degree in electrical and computer engineering from the University of Calgary, Calgary, Alberta, Canada in 2002 and the Ph.D. degree in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada in 2008.

She has been a research scientist with IntelliView Technologies Inc., Calgary, Alberta, Canada since 2009. She has authored more than 20 international journals and conferences papers.

Dr. Du’s interests include pattern recognition, computer vision and image/video processing.

Choudhury A. Rahman received the Ph.D. degree in electrical engineering from the University of Calgary, Calgary, Alberta, Canada in 2008. Since then he has been involved with the R&D team at IntelliView Technologies Inc., Calgary, Alberta, Canada for building smart analytic based video surveillance systems. He is a member of the SC29 and SC6 of the CAC/JTC1 committees. Dr. Rahman was also a contributor towards the development of the ISO/IEC MPEG-4/H.264 industrial standardization.

Saika Sharmeen received the M.Sc. degree in electrical engineering from the University of Calgary, Calgary, Alberta, Canada in 2008. Her area of research includes wireless location technologies, design and optimization of real time algorithms and communication protocols for industrial applications. She has been involved with IntelliView Technologies Inc., Calgary, Alberta, Canada since 2009 for developing intelligent industrial video surveillance systems. Ms. Sharmeen is a professional member of APEGA and a member of the American Society for Quality (ASQ).

Wael Badawy received the B.Sc. and M.Sc. degrees from Alexandria University, Egypt in 1994 and 1996, respectively. He received the M.Sc. and Ph.D. degrees from the Center for Advanced Computer Studies, University of Louisiana, Lafayette, LA, USA in 1998 and 2000, respectively.

Currently, he is the president of IntelliView Technologies Inc., Calgary, Alberta, Canada. He was a professor and iCore chair associate at the University of Calgary, Calgary, Alberta, Canada. Dr. Badawy is a world leading researcher in video surveillance technology and he published more than 400 peer-reviewed technical papers, 50+ contributions to develop the ISO standards, which is more than 75% of the hardware reference model for the H.264 compression standard. He is listed as "Primary contributor" in the VSI Alliance™ developing the "Platform-Based Design Definitions and Taxonomy, (PBD 11.0), 2003". He has 13 books and conference proceedings. He is a coauthor to the International video standards known as MPEG4/H.264. He represents Canada in ISO/TC223 - Societal Security Chairman of the Canadian Advisory Committee (CAC) on ISO/IEC/JTC1/SC6 "Telecommunications and Information Exchange Between Systems" and head of the Canadian Delegation. He received over 60 international and national awards for his technical, commercialization, innovation and contribution to the industry, academia and society. Among so many things he enjoys giving back and as a mentor in the Canadian Youth Business Foundation, he supports Canadian below 34 years to start and build their businesses. He has 8 patents and 13 patents applications in the areas of video systems and architectures.