Abnormal Spatial Event Detection and Video Content Searching in a Multi-Camera Surveillance System

Chueh-Wei Chang Ti-Hua Yang Yu-Yu Tsao
Department of Computer Science and Information Engineering,
National Taipei University of Technology, Taiwan
cwchang@ntut.edu.tw s4598024@ntut.edu.tw s4598029@ntut.edu.tw

Abstract

In a traditional multi-camera surveillance system, it's hard to find the routes of the suspect objects, and search for those video clips related to the suspect objects from the surveillance database. In this paper, we present a framework for spatial relationship construction, abnormal event detection and video content searching for visual surveillance applications. This system can automatically detect the abnormal events from monitoring areas, and select the representative key frame(s) from the video clips as an index, then store the color features of the suspect objects into the surveillance database. A graph model has been defined to coordinate the tracking of objects between multiple views, so that the surveillance system can check the route of objects whether go into a critical path or not. A variety of spatio-temporal query functions can be provided by using this spatial graph model. To achieve the content-based video object searching, a kernel-based approach is employed as a similarity measure between the color distribution of the suspect object and target candidates in the surveillance database.

1. Introduction

The most widely used video-based surveillance systems [3] generally employ two or more cameras that are connected to the monitors. This kind of systems needs the presence of a human operator, who interprets the acquired information and controls the evolution of the events in a surveyed environment. As the number of cameras increase, event monitoring by personnel is rather tedious, and easy to cause error. The automatic preprocessing of the video information by a surveillance system can greatly help person to improve validation of the events. Each camera must be capable of detecting and tracking moving objects of interest, and recording the video of the event into a surveillance database [2]. Video processing and understanding can be considered as a fundamental modality for surveillance applications.

A real-time visual surveillance system in [5] employs a combination of shape analysis and tracking, and constructs models of people's appearances in order to detect and track groups of people as well as monitor their behaviors even in the presence of occlusion and in outdoor environments. A single person tracking system in [8] detects moving objects in indoor scenes using motion detection, tracks them using first-order prediction, and recognizes behaviors by applying predicates to a graph formed by linking corresponding objects in successive frames. But most of the surveillance systems, just like the two systems we mentioned above, don't provide the query functions for spatio-temporal relations and similar object searching in video databases.

In this paper, we design a framework for the abnormal spatial event detection and the suspect video object searching in a multi-camera surveillance system. We will describe in the following sections the details of the design of our surveillance systems that utilize a large set of cameras and more extended and flexible processing strategies. In the next section, we describe the design concepts and process flowcharts of our surveillance system. In Section 3, we describe the spatial relationship between the surveillance areas. In Section 4, we show the mechanism of video content searching. The system interface and similar measure experiments are presented in Section 5. Some conclusions are drawn in Section 6.

2. System Design

In our multi-camera surveillance system, video data are acquired by distributed cameras and then are transmitted to the remote control center. It has the capability of observing and recording videos from distant places. It's necessary to design more sophisticated video processing algorithms for spatial event handling and suspect object searching. The architecture of the proposed surveillance system is shown in Figure 1. In this section, the main characteristics of the 3 modules are presented.

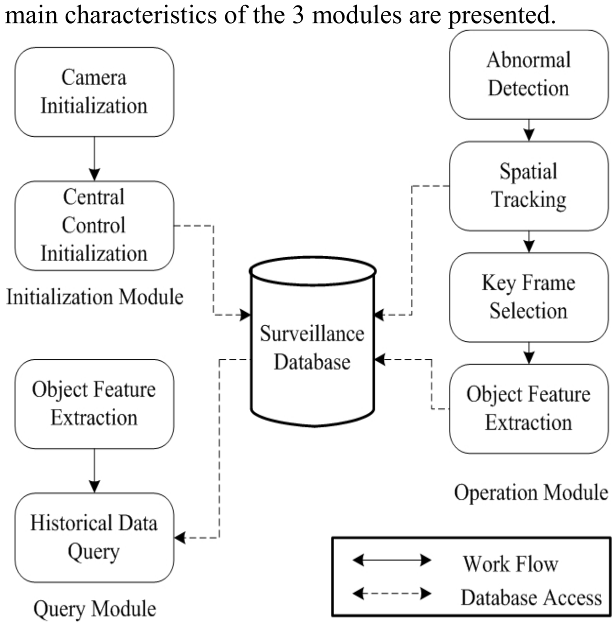


Figure 1. Architecture of the proposed system.

(1) Initialization Module

Several system initialization processes must be done before this system start to operation, including the background, the alert areas and alert types for each camera, the critical paths and places, the space zone and camera ID in this surveillance area, the spatial relationship between areas, and a number of threshold values for the key frame extraction and similarity measure processes. The alert area can be set to have several low to high level alarm priority.

(2) Operation Module

Once finishing the initialization step, the system can start monitoring from each camera. An abnormal event can be defined as some large object goes into the alert area. This detection procedure can be easily implemented by a sequence of video processing algorithm from video frame capture, background subtraction [7], thresholding, morphological noise removal, and abnormal object size comparison.

After confirming a suspect object in an abnormal event, the system will record the places where this object visit, and check whether its route is a critical path or not. The video clips of abnormal events will be recorded and stored in the center control surveillance database. Several key frames will be selected from these video clips. The color features of suspect object will be extracted and stored as an index also. This index will allow one to retrieve particular sequences in a fast and efficient way.

(3) Query Module

Every kind of alerts type can be monitored from the system central control. Operator of this center control then can select the color feature of some suspect object and ask the surveillance database many kind of historical queries by content-based similarity measure method [1], such as, "when and where this object arrived ?", "what kind of alert it caused ?", "where it have ever been visited in a specific time interval?".

3. Spatial Event Detection

In most of a building, we can transfer the surveillance area into a topological graph by partitioning the room space physically or logically, as shown in Figure 2. A node in this graph represents a well divided area (also called a "Zone"), and an edge stands for that two separate zones can be connected by a door or a corridor. More than one camera can be installed in a Zone. The shooting ranges of these cameras need to cover all of the entrance and exit path ways to keep track of the abnormal object. Therefore, we can define the spatial abnormal event as a special route (path) that an alert will

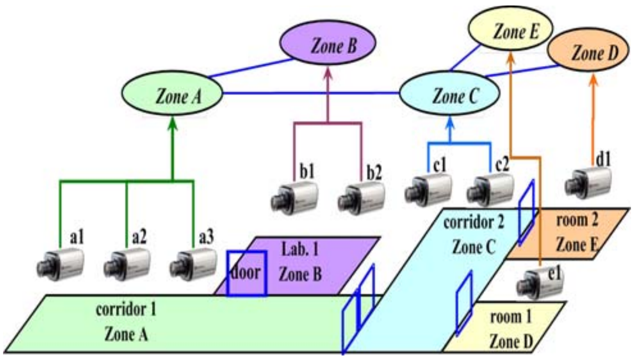


Figure 2. Transfer from an indoor space into a graph.

be caused if some abnormal object goes by this path, such as the sequence {Zone A → Zone B → Zone A → Zone C → Zone E} in Figure 2. We call this system specified route a "Critical Path".

Once a suspect object entered a monitored area, the places it visited will be tracked until it left this area. By using this graph model, we can check the route by traveling the graph node, and the system can retrieve any route information about the suspect object.

4. Video Content Searching

To keep as the evidence, we need to store those video clips when an abnormal event occurred. Then, we select several key frames and color feature information of suspect object form those video clips as an index. When we search an object from the surveillance database, we can also use this color information as the example of the content-based retrieval.

4.1. Key frame extraction

A key frame extraction is a technique to automatically extract from a video sequence one or several salient frames representative of its contents. We want to get at least one key frame from each abnormal event video clip. A number of techniques have been published in the literature [6] to extract key frames. However, these techniques address the problem of extracting all the key frames of a video with the goal of producing a storyboard representation for video browsing. We need a new methodology to extract the color information of the suspect object in the abnormal event video clips, and also use this key frame as an index to speed up the search time when query the suspect object video from database. We choose 6 criteria to evaluate those abnormal video frames, and the key frame is chosen based on (1) large size, (2) solid, (3) human shape, (4) low motion activity, (5) high contrast and sharpness, and (6) has plenty of color information. Some of the frame with the highest score (quality) will be selected as the key frame(s), and then stored in the surveillance database. The description about these 6 criteria will be described as follows.

(1) Object size

The suspect object size is defined as the total number of pixels in the selected key frame. After applying the pixel-wise background subtraction process between nth captured abnormal video frame and the background image, the suspect object is the region of pixels that their intensity difference are larger than a threshold value. We define

$$Size(n) = \sum_{x,y} I^n_{x,y} \quad , \tag{1}$$

where $Size(n)$ denotes the area size of the suspect object in the nth abnormal video frame, $I^n_{x,y}$ represents the binary value of the pixel at location (x,y) in frame n after the thresholding process, $I^n_{x,y} \in \{0,1\}$. We prefer a suspect object is large enough, not just some small scatter objects like noises.

(2) Object density

The object density of the *n*th video frame is simply computed as

$$Density(n) = Size(n) / (Width_n \times Height_n), \quad (2)$$

where $Width_n$ and $Height_n$ are the width and height of the minimal bounding box that contains the abnormal object, respectively. We prefer the suspect object is solid and quite different from the background.

(3) Object aspect ratio

The object aspect ratio of the *n*th video frame is defined as

$$Ratio(n) = \begin{cases} 1 & if\ 0.25 \le (Width_n / Height_n) \le 0.75 \\ 0 & otherwise \end{cases}, \quad (3)$$

where $Width_n$ and $Height_n$ are the width and height of the minimal bounding box, respectively. We prefer the suspect object is like a human shape, not as a dog or cat.

(4) Edge change ratio

We use the Edge Change Ratio (ECR) [9] algorithm to measure the edge changing between frame *n* and frame *n+1.* The edge change ratio can show how fast an object is moving in the video sequences.

In the ECR algorithm, two kinds of edge change ratios are defined, one is the Exiting Edge Pixel (Outgoing Edge Pixel), and the other one is the Entering Edge Pixel (Incoming Edge Pixel). This approach is based on a simple observation: during a fast move object, new intensity edges appear far from the locations of old edges. Similarly old edges disappear far from the location of new edges. We define an edge pixel that appears far from an existing edge pixel as an entering edge pixel and an edge pixel that disappears far from an existing edge pixel as an exiting edge pixel. By counting the entering and exiting edge pixels we can realize how fast this object is moving.

The ECR algorithm takes as input two consecutive images I_n and I_{n+1}. It first performs a Sobel edge detection step and then thresholding step, resulting in two binary images E_n^{thresh} and E_{n+1}^{thresh}.

Next, the Exiting Edge Pixel X_n^{out} and Entering Edge Pixel X_{n+1}^{in} are defined as

$$X_n^{out} = E_n^{thresh}\ AND(NOT(E_{n+1}^{dilation})), \quad (4)$$

$$X_{n+1}^{in} = E_{n+1}^{thresh}\ AND(NOT(E_n^{dilation})), \quad (5)$$

where $E_n^{dilation}$ and $E_{n+1}^{dilation}$ are the results after the morphological dilation operation from E_n^{thresh} and E_{n+1}^{thresh} respectively. The AND and NOT functions are the binary logic operations.

The Edge Change Ratio (ECR) of the *n*th video frame is computed as

$$ECR(n) = \max(X_n^{out} / E_n^{thresh}, X_{n+1}^{in} / E_{n+1}^{thresh}). \quad (6)$$

We prefer the suspect object is a slow changing one because it will have a better stable and sharpness image quality.

(5) Frame clarity

Because we prefer the suspect object is contrast and sharpness. We use the overall edge count to evaluate the focus value and also clarity of a frame. The frame clarity of the *n*th video frame is computed as

$$Edges(n) = \sum_{x,y} E_n^{thresh}(x, y) \cdot \quad (7)$$

(6) Histogram entropy

How much color information in a frame can be provided by the histogram entropy. The histogram entropy is defined as

$$H(n) = -\sum_x p(x,n) \log_2(p(x,n)), \quad (8)$$

where $p(x,n)$ is the probability of the grayscale value *x* in the luminance histogram of frame *n*. We prefer more color in the frame.

For each abnormal event, a score function *KeyFrameScore(n)* combining the different measures is computed as

$$\begin{aligned} KeyFrame(n) = {} & w_{size} \times Size^{Normal}(n) + w_{Ratio} \times Ratio(n) \\ & + w_{Density} \times Density^{Normal}(n) + w_{ECR} \times ECR^{Normal}(n), \\ & + w_{Edges} \times Edges^{Normal}(n) + w_H \times H^{Normal}(n) \end{aligned} \quad (9)$$

where W_{Size}, $W_{Density}$, W_{Ratio}, W_{ECR}, W_{Edges}, W_H, are the weighting factors of each criterion which can be adjusted heuristically. Finally, some of the key frames with the highest score are selected.

4.2. Similarity measure

As mentioned before, the similarity measure we develop is highly dependent on the object color information. We modify some of the approach for the real-time tracking of non-rigid objects in [4].

The feature *c* representing the color of the suspect object is assumed to have a density function q_c, while the target candidate *n* in the key frame of a surveillance database has the feature distributed according to $p_c(n)$.

The problem is then to find the target candidate *n* whose associated density $p_c(n)$ is the most similar to the suspect object density q_c. We calculate the similarity between two densities according to the Bhattacharyya coefficient, whose general form is defined by

$$\rho(n) \equiv \rho[p(n), q] = \int \sqrt{p_c(n) \cdot q_c}\, dc \quad (10)$$

Due to the computational complexity we use the density estimates derived from a simple histogram formulation. The discrete density $q=\{q_u\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} q_u = 1$) is estimated from the *m*-bin histogram of the suspect object, while $p(n)=\{p_u(n)\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} p_u = 1$) is estimated at a given target candidate *n* from the *m*-bin histogram. Therefore, the sample estimate of the Bhattacharyya coefficient is given by

$$\rho(n) \equiv \rho[p(n), q] = \sum_{u=1}^{m} \sqrt{p_u(n) \cdot q_u} \quad (11)$$

5. Experimental Results

In our system interface, as shown in Figure 3, there has 8 major functions area, including the Camera Display & Control, the Graph Node Relation List, the Critical Path, the Region Setup, the Event List, the Detected Motion, the Historical Record Query, and the Query Result.

In the Camera Display & Control area, we can watch, replay, and switch the videos between different cameras in the surveillance area. Once there has an abnormal event occurs in the real-time monitoring mode, the camera ID will change color depends on the type of alert. The route of this abnormal object where it has been visited will be displayed in the Event List area. We can setup the Zone, the camera(s) in the Zone, the connectivity between Zones, and the specified Critical Path, by using the Graph Tool in the Graph Node Relation List area. The query result match the query constrains, no matter by the time interval or by the critical path, will be listed in the Query Result area. The query constrains can be set to the time, date range, specific zone, camera, or a critical path in the Historical Record Query area.

A set of process steps and results are presented in Figure 4 and Figure 5, demonstrating the suspect object segmentation and similar candidate selection from the surveillance database. The suspect object can be specified from any video frame by the system operator manually. Those qualified target candidates will be listed according to the rank of similarity measure by a searching of the database, and the related routes of target candidates can also be listed.

6. Conclusions

Indoors wide range surveillance and monitoring using multiple cameras is a challenging task. In this paper, we design a multi-camera surveillance system that can detect the abnormal spatial events and provide the function of a color-based suspect object searching for the surveillance video database query. This surveillance system is able to tracking the routes of suspect objects between multiple camera views. It allows for filtering and retrieval of the relevant path or similar object events. The availability of our automated real-time event detection method would greatly facilitate the monitoring of large sites with numerous cameras.

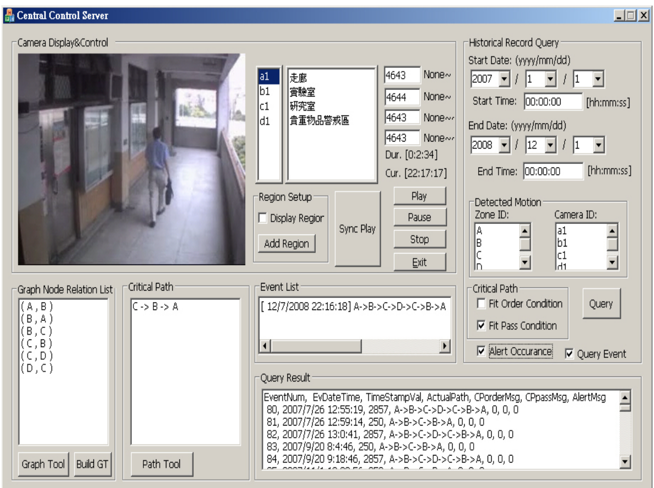
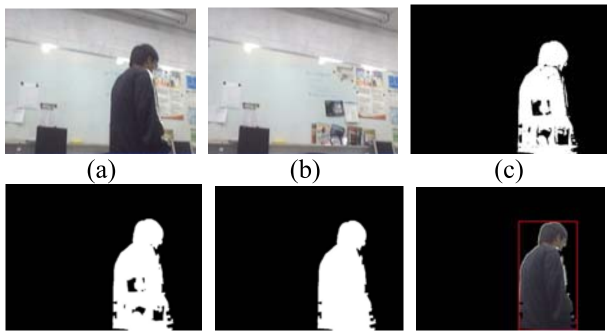


Figure 3. The system interface.



(a) (b) (c)
(d) (e) (f)

Figure 4. Suspect object segmentation steps. (a)current captured frame, (b)background image, (c)background subtraction and thresholding, (d) noise removal by morphological operation, (e)area filling, (f)result of this suspect object segmentation.

(a) (b) (c) (d) (e)

Figure 5. Similarity query results. (a)the suspect object, (b) the Bhattacharyya similarity coefficient $\rho(y)$= 0.945, (c) $\rho(y)$= 0.8, (d) $\rho(y)$= 0.75, (e) $\rho(y)$= 0.65.

References

[1] S. Antani, R. Kasturi, and R. Jain, "A Survey on the Use of Pattern Recognition Methods for Abstraction, Indexing, and Retrieval of Images and Video," *Pattern Recognition*, Vol. 35, No. 4, pp. 945-965, 2002.

[2] J. Black, D. Makris and T. Ellis, "Hierarchical Database for Multi- Camera Surveillance System," *Pattern Analysis Application*, pp. 430-446, 2005.

[3] R. T. Collins, A. J. Lipton and T. Kanade, "Introduction to the special section on video surveillance," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, pp. 745 - 746, Aug. 2000.

[4] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel Based Object Tracking," *IEEE Trans. Pattern Analy. Machine Intel.*, Vol. 25, No.*5*, pp. 564-557, May 2003.

[5] Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time Surveillance of People and their Activities," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 22, pp. 809–830, Aug. 2000.

[6] T. Liu, H. J. Zhang and F. Qi, "A Novel Video Key-Frame-Extraction Algorithm based on Perceived Motion Energy Model," *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 13, No. 10, pp. 1006- 1013, Oct. 2003.

[7] Neri, S. Colonnese, G. Russo and P. Talone, "Automatic Moving Object and Background Separation," *Signal Processing*, Vol.66, No. 2, pp. 219 - 232, Apr.1998.

[8] T. Olson and F. Brill, "Moving Object Detection and Event Recognition Algorithms for Smart Cameras," *Intl. Conf. on Image Understanding Workshop*, 1997, pp. 159–175.

[9] R. Zabih, J. Miller,and K. Mai. ,"A Feature Based Algorithm for Detecting and Classifying Scene Breaks," *Proc. of ACM Multimedia 95*, pp. 189-200, Nov. 1995.