

Automatic Motion Trajectory Analysis for Dual Human Interaction Using Video Sequences

Yuan-Hsiang Chang, Pin-Chi Lin, Li-Der Jeng

Abstract—Advance in techniques of image and video processing has enabled the development of intelligent video surveillance systems. This study was aimed to automatically detect moving human objects and to analyze events of dual human interaction in a surveillance scene. Our system was developed in four major steps: *image preprocessing*, *human object detection*, *human object tracking*, and *motion trajectory analysis*. The adaptive background subtraction and image processing techniques were used to detect and track moving human objects. To solve the occlusion problem during the interaction, the Kalman filter was used to retain a complete trajectory for each human object. Finally, the motion trajectory analysis was developed to distinguish between the *interaction* and *non-interaction* events based on derivatives of trajectories related to the speed of the moving objects. Using a database of 60 video sequences, our system could achieve the classification accuracy of 80% in interaction events and 95% in non-interaction events, respectively. In summary, we have explored the idea to investigate a system for the automatic classification of events for interaction and non-interaction events using surveillance cameras. Ultimately, this system could be incorporated in an intelligent surveillance system for the detection and/or classification of abnormal or criminal events (e.g., theft, snatch, fighting, etc.).

Keywords—Motion detection, motion tracking, trajectory analysis, video surveillance.

I. INTRODUCTION

TECHNIQUES of image processing, video processing, and computer vision have been extensively studied by many researchers in the past decades. These techniques have also been used in a wide range of applications, e.g., smart-phone applications, digital multimedia systems, computer games, and human-computer interaction, etc. One major application of the image/video processing techniques is the video surveillance system. Traditional video surveillance systems have several drawbacks, such as hardware configuration and cost of storage space. An obvious problem of traditional video surveillance systems is that human observers are required to monitor the surveillance environments in a timely manner for potential threats to people's security (e.g., violence, snatch, theft, etc.). However, long-term human surveillance could result in fatigue and negligence, leading to a late response to potential threats. With the advance of image/video processing techniques, the intelligent video surveillance systems have the potentials to offer an alternative solution for the problem. The video sequences as obtained using surveillance cameras, while being

recorded, can be automatically analyzed by computers for potential threats.

In recent years, techniques for the development and evaluation of the intelligent video surveillance systems have been extensively studied. Typical techniques include *motion detection*, *motion tracking*, and *motion trajectory analysis*.

The objective of the *motion detection* is aimed to detect moving objects in the surveillance environments. Typical techniques include the *background subtraction* and *temporal difference*. For example, [1] presented a method for detecting and tracking multiple human objects. Kuralkar and Gaikwad [2] presented a novel algorithm for detecting moving objects from a static background scene. The techniques of motion estimation and detection, background subtraction, and shadow removal were applied in both [1] and [2]. Porikli and Tuzel [3] presented an automatic, real-time human tracking, and observation system. They improved the system by integrating a mean-shift based model update technique with an adaptive change detection method. Banerjee and Sengupta [4] presented an automated video surveillance system for tracking an object in motion and classifying it as a human or non-human entity. The system employed techniques of adaptive background modeling and histogram of oriented gradients for the classification. Rakibe and Patil [5] presented a new algorithm for detecting moving objects from a static background scene based on background subtraction. In addition, contour projection analysis was combined with the shape analysis to remove the effect of shadows such that the moving human bodies could be accurately and reliably detected. Kim and Lee [6] developed a useful segmenting and tracking tool for rigid and non-rigid (i.e., deformable) objects using the active contour model, snake. Comaniciu et al. [7] proposed a new approach toward target representation and localization. The feature histogram-based target representations were regularized by spatial masking with an isotropic kernel. In addition, a metric derived from the Bhattacharyya coefficient was used as similarity measure. They claimed the method could successfully cope with camera motion, partial occlusions, clutter, and target scale variations.

The objective of the *motion tracking* is aimed to identify the motion trajectory of a moving object in the surveillance scene. The main obstacle of the tracking arises if occlusion of moving objects occurs. Chang et al. [8] and Gabriel et al. [9] used the Kalman filter to predict the trajectory of a moving object. The Kalman filter was shown to strengthen the tracking accuracy even though the moving objects could be occluded.

The objective of the *motion trajectory analysis* is aimed to analyze the motion trajectory for trajectory similarity, event detection, or behavior analysis in the surveillance scene. Li et al.

Yuan-Hsiang Chang, Ph.D. (phone: 886-3-265-4713; fax: 886-3-265-4799; e-mail: changyh@ice.cycu.edu.tw), and Pin-Chi Lin, M.S. is currently with the Information and Computer Engineering Department, Chung-Yuan Christian University, Chung Li, Tawian, R.O.C

Li-Der Jeng, Ph.D. is with the Electronic Engineering Department, Chung-Yuan Christian University, Chung Li, Taiwan.

[10] presented a method to compare the curve of trajectory using the minimum-projection distance-based metric algorithm. Liao et al. [11] presented a video-based surveillance system that could perform real-time event detection. They used techniques of mixture of Gaussian and color blob-based tracking to track foreground objects. A tracking module was designed to track multiple trajectories and the Douglas-Peucker algorithm [12] was used to approximate a trajectory. By comparing two arbitrary trajectories, real-time event detection could be achieved. Zhou and Huang [13] proposed a novel method to represent motion trajectory, namely *bag of segments*. In this representation, the trajectories were transformed to a vector in the codeword space. The codebook was generated by clustering a large amount of trajectory segments using the Expectation Maximization (EM) algorithm. The bag of segments was shown to be effective in trajectory similarity search and classification. Suk et al. [14] proposed a novel method of analyzing human interactions based on the walking trajectories of human subjects. Their principle assumption was that each interactive episode was composed of meaningful small unit interactions, called *sub-interactions*. The hidden Markov model (HMM) was then used to model each sub-interaction. In addition, a network of Dynamic Probabilistic Models (DPMs) was used to represent the complete interaction. Habe et al. [15] presented a method to analyze interaction between pedestrians based on their trajectories obtained using cameras. They proposed a set of features that measures the interaction between pedestrians. In addition, a method to extract transition points of a walking pattern was presented, which was then used to measure the strength of the influence between pedestrians.

II. METHOD

The objective of this study is to develop an intelligent video surveillance system that could be used to automatically analyze the surveillance scene in which events of dual human interaction (i.e., *interaction* vs. *non-interaction*) between two pedestrians are of particular interest.

Fig. 1 shows the flowchart of our system for the *automatic motion trajectory analysis for dual human interaction using video sequences*. Our method includes four major steps: *image preprocessing*, *human object detection*, *human object tracking*, and *motion trajectory analysis*. Detail technical approaches of our system are described herein.

A. Image Preprocessing

The objective of the *image preprocessing* is to detect and segment moving objects from the static background in video sequences. The *gray-level transform* is used to acquire the gray-level images for further processes. Then, the *background modeling* is used to build the background model as the static background information for motion detection.

Gray-Level Transform - To convert color images to gray-level images for further processes, we used the equation:

$$Y = 0.299R + 0.587G + 0.114B \quad (1)$$

where R , G , B are the trichromatic color values, Y is resulting gray-level values.

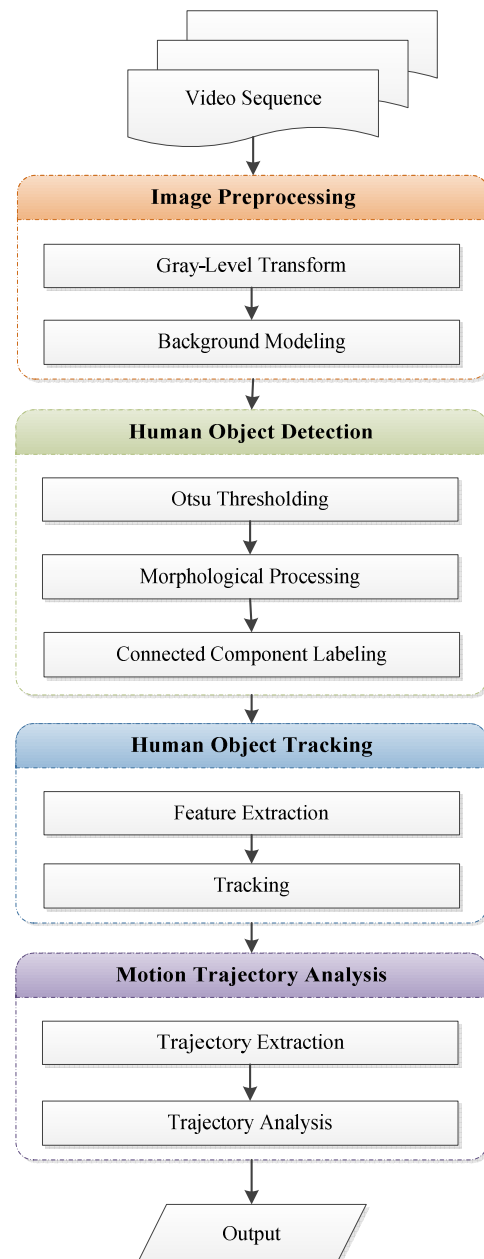


Fig. 1 System flowchart for the *automatic motion trajectory analysis for dual human interaction using video sequences*

Background Modeling—The *background modeling* is an important technique for motion detection. The method is used to build and maintain a background model which is actually an image for the static background without any moving objects.

To adapt to illumination variations in the surveillance environment, the *adaptive background modeling* is used in our system by:

$$B_i(x, y) = (1 - \alpha) \cdot B_{i-1}(x, y) + \alpha \cdot I_{i-1}(x, y) \quad (2)$$

where B_{i-1} is the previous background model, I_{i-1} is the previous gray-level image (frame), and B_i is the background model for the current frame in the video sequence. In addition, (x, y) is the coordinate of pixel in images (frames), α is an adaptive constant between 0 and 1. Therefore, the background model is constantly updated as time proceeds.

For the motion detection, an initial background model is obtained. After the adaptive background model is acquired, the *adaptive background subtraction* is then applied to detect moving objects in the surveillance scene by:

$$S_i(x, y) = |I_i(x, y) - B_i(x, y)| \quad (3)$$

where B_i is the adaptive background model, I_i is the gray-level image, and S_i is the result of the adaptive background subtraction for the i -th frame.

An example is shown in Fig. 2. Two human objects walked through the surveillance scene and passed by each other without any interaction. Fig. 2 (a) is the original image (frame), (b) is the adaptive background model, and (c) is result after the adaptive background subtraction in which moving pixels (objects) can be detected.

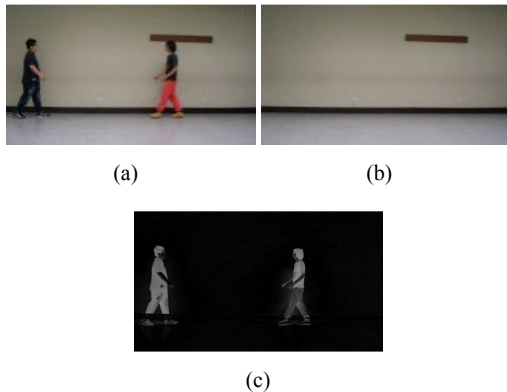


Fig. 2 An example of the *image preprocessing*: (a) original image; (b) adaptive background model; (c) adaptive background subtraction result image

B. Human Object Detection

The objective of the *human object detection* is to group the moving pixels and to detect each moving human object as an independent *region of interest* (ROI) for posterior motion analysis. The method is based on the *Otsu thresholding*, *morphological processing*, and *connected component labeling* as described below:

Otsu Thresholding – To achieve a relatively robust detection of moving objects, the *Otsu thresholding* [16] is an adaptive thresholding method with respect to various illuminations and is applied by:

$$\begin{cases} S_i(x, y) \geq T_{otsu} & \text{if } (x, y) \in Obj \\ S_i(x, y) < T_{otsu} & \text{otherwise} \end{cases} \quad (4)$$

where Obj is the moving object and T_{otsu} is an adaptive threshold using the Otsu thresholding method. Therefore, a pixel is considered as a candidate foreground pixel in a moving object if its subtraction result exceeds the Otsu threshold.

Morphological Processing–After the *Otsu thresholding*, the ROIs for the moving objects may still contain incomplete contours (shapes) of the objects and certain amount of noise. The *morphological processing* [17] (i.e., the morphological closing) is applied such that the contours (shapes) representing each moving object become more complete and well-defined.

Connected Component Labeling–To isolate and extract each moving object, the *connected component labeling* [18] is used to label set of pixels in the ROIs by eight-connectivity. Therefore, each connected component is assigned a unique label and represents a ROI for the moving human object.

An example of the *human object detection* is shown in Fig. 3, where (a) is the result after the image preprocessing, (b) is the image after the Otsu thresholding, (c) is the image after the morphological processing, and (d) is the image after the connected component labeling. As seen, the moving human objects in the surveillance scene were detected accordingly.

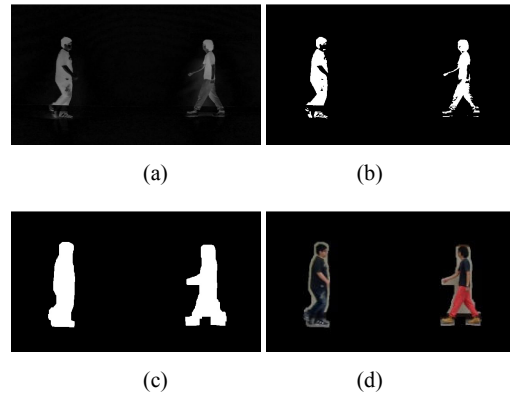


Fig. 3 An example of the *human object detection*: (a) result image after the adaptive background subtraction; (b) result image after the Otsu's thresholding; (c) result image after the morphological processing; (d) detected moving objects after the connected component labeling

C. Human Object Tracking

The objective of the *human object tracking* is to process the images (frames) in the temporal video sequence and identify the motion trajectories for the moving objects. In our system, the techniques used can be described as *feature extraction* and *tracking*.

Feature Extraction – The objective of *feature extraction* is to extract features (i.e., location and color properties) of moving objects for tracking. Feature extraction includes skeleton extraction, color histogram analysis, and centroid extraction. The chain code [17] is used to represent the contour of a moving object (connected component).

In the skeleton extraction, the distance transform [19] is used to extract the main skeletons of moving objects. Using the binary image, the distance transforms measures the nearest distance from an object pixel to background pixel and replaces the original pixel value by the Euclidean distance value.

The human objects are with the larger regions in the distance transform image. A threshold is thus selected to distinguish the larger regions as:

$$Skeleton(x, y) = \begin{cases} 1 & \text{if } DT(x, y) > T_D \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where the $DT(x, y)$ is pixel value of the distance transform image, T_D is the threshold, and $Skeleton(x, y)$ is the pixel value of main skeleton image.

An example of the feature extraction is shown in Fig. 4, where (a) is the original image, (b) is the result image after the human object detection, (c) is the result image after the distance transform, and (d) is the result image for the skeletons of two human objects.

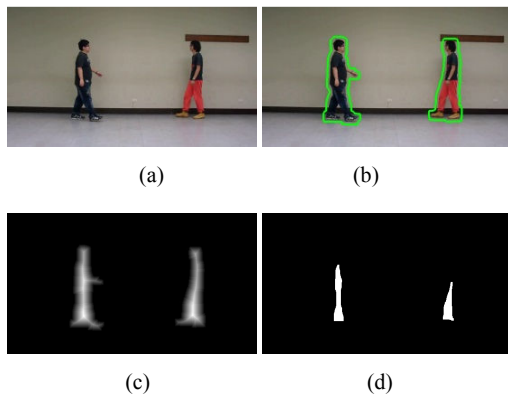


Fig. 4 An example of the *feature extraction*: (a) original image; (b) result image after the human object detection; (c) result image after the distance transform; (d) result image of the skeletons for the two pedestrians

To increase the robustness of tracking human objects, the color histogram analysis is used. Our assumption is that the moving object in a short period of time (consecutive frames) has similar color properties. Here, the HSV color model, including *Hue*, *Saturation*, and *Value*, is used for comparing color properties for regions in two consecutive frames. In addition, the Bhattacharyya coefficient is used for the color similarity in regions of two consecutive frames and can be defined by:

$$d_{Bhatta}(H_1, H_2) = \sqrt{1 - \frac{\sum_i \sqrt{H_1(i) \cdot H_2(i)}}{\sqrt{\sum_i H_1(i) \cdot \sum_i H_2(i)}}} \quad (6)$$

where H_1 and H_2 are the two corresponding hue histograms, and d_{Bhatta} is the similarity between 0 and 1.

An example of the color histogram analysis is shown in Fig. 5, where (a) are the color histograms for the first human object; (b) are the two consecutive frames; (c) are the color histograms for the second human object. As seen, in consecutive frames, the moving human object has similar color properties.

To track the location of human objects, the region centroid of each human object can be calculated by:

$$(\bar{x}, \bar{y}) = \left(\frac{\sum_{(x,y) \in R} x}{\sum_{(x,y) \in R} 1}, \frac{\sum_{(x,y) \in R} y}{\sum_{(x,y) \in R} 1} \right) \quad (7)$$

where R is the region of moving human object, and (\bar{x}, \bar{y}) is the region centroid to represent the current location of the moving human object.

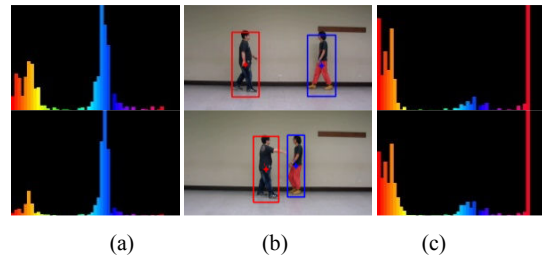


Fig. 5 An example of the *color histogram extraction*: (a) are the color histograms of the first object in consecutive frames; (b) are the two consecutive frames; (c) are the color histogram of the second human object in consecutive frames

Tracking – The objective of the *motion tracking* is aimed to identify the motion trajectories of the moving objects in the surveillance scene. The main obstacle in technical development is that occlusions may occur during the dual human interaction. Examples of occlusions are shown in Fig. 6. During the occlusion, two moving objects were often detected as one region only.

A set of rules is used to determine the *state* of tracking defined by:

$$State = \begin{cases} Occlusion & \text{if } \sum I_{obj} < \sum Obj \\ New Object & \text{if } \sum I_{obj} > \sum Obj \\ Unchanged & \text{if } \sum I_{obj} = \sum Obj \end{cases} \quad (8)$$

where $\sum I_{obj}$ is the number of moving objects in current frame, $\sum Obj$ is the number of objects in previous frame. As a result, our system can identify the *state* as *occlusion*, *new object*, or *unchanged*, respectively, in the temporal sequence.

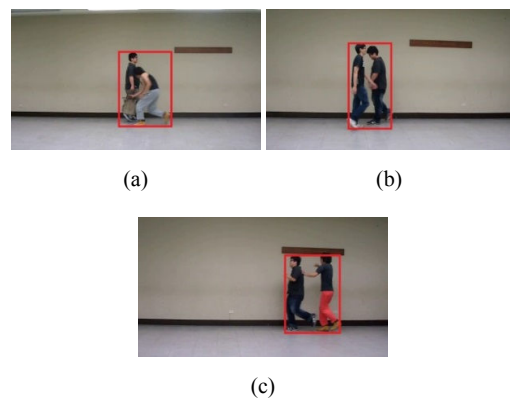


Fig. 6 Examples of occlusions in dual human interactions

Because the motion trajectory of a moving object is lost during occlusion, a method to retain the complete trajectory for the moving object is required for reliable event detection.

The Kalman filter was proposed by Kalman in 1960 [20]. It is usually used in the detection and tracking of moving objects. The Kalman filter is a recursive linear filter and can be divided into two parts: prediction and correction. The Kalman filter can be defined by:

$$\begin{aligned} x_t &= F x_{t-1} + B u_t + w_t \\ z_t &= H_t x_t + v_t \end{aligned} \quad (9)$$

where x_t is the state vector, u_t is the control vector, w_t is the process noise, F is the transfer matrix, and B is the control model. In addition, z_t is the measurement vector, v_t is the observation noise, and H_t is the observation model. According to (9), the state $\hat{x}_{t,t-1}$ is predicted from the state $\hat{x}_{t-1,t-1}$ as:

$$\hat{x}_{t,t-1} = F \hat{x}_{t-1,t-1} + B u_t \quad (10)$$

Then, by combining the predicted value $\hat{x}_{t,t-1}$ in previous time and the current measurement z_t , the best estimation can be obtained by:

$$\hat{x}_{t,t} = \hat{x}_{t,t-1} + K_t (z_t - H_t \hat{x}_{t,t-1}) \quad (11)$$

where K_t is the Kalman gain and can be solved using:

$$K_t = P_{t,t-1} H_t^T (H_t P_{t,t-1} H_t^T + R_t)^{-1} \quad (12)$$

In our system, the Kalman filter is used to predict the location of the moving objects when the occlusion occurs. Therefore, our system is able to track the moving objects and retain a complete trajectory for each moving object, even though the predicted location of the moving object during occlusion may subject to minor error.

D. Motion Trajectory Analysis

The objective of the *motion trajectory analysis* is aimed to analyze the motion trajectory for dual human interaction. In this study, we are particularly interested in the classification of *interaction* or *non-interaction* events. Our definitions of the two events are:

- (1) *Interaction events*: Two human objects walk into the surveillance scene and certain types of interaction between the two human objects occur (e.g., hand shaking, greeting, chatting, and/or giving stuff, etc.);
- (2) *Non-interaction events*: Two human objects walk through the surveillance scene and simply pass by each other.

In this step, technical approaches include *trajectory extraction* and *trajectory analysis*. After tracking each moving object, the trajectory of each human object can be extracted as a complete trajectory. Our assumption is that if the two human objects walk through the surveillance scene and pass by each other (i.e., *non-interaction* event), they may maintain approximately constant speed through the scene. However, if

the any types of interaction occurs (e.g., hand shaking, greeting, chatting, etc.), the locations of the human objects may become static or change locally, resulting in speed variations.

An example of the motion detection and tracking in a frame is shown in Fig. 7. In our system, each image (frame) is with the 640×360 pixel resolution and the image coordinate system is shown accordingly for motion detection and tracking. As seen, two human objects are detected and their locations (centroids) are tracked (as marked).

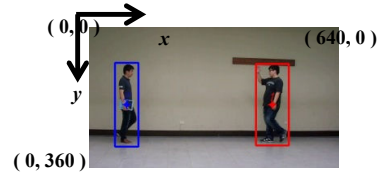


Fig. 7 An example of the motion detection and tracking in a frame. The image coordinate system is shown accordingly

Fig. 8 is the video sequence in which dual human interaction occurs in the surveillance scene. Our system was able to track each human object and the trajectory for each human object is recorded. During occlusion, the trajectory could be retained using the Kalman filter, thus achieving a complete trajectory.

We assume that $x(t)$ and $y(t)$ are the image coordinates (locations) in the x and y directions, respectively, and t is the frame index. Therefore, the trajectories for the two human objects can be defined as functions of t in either the x or y direction, respectively. An example is shown in Fig. 9 for the video sequence in Fig. 8.

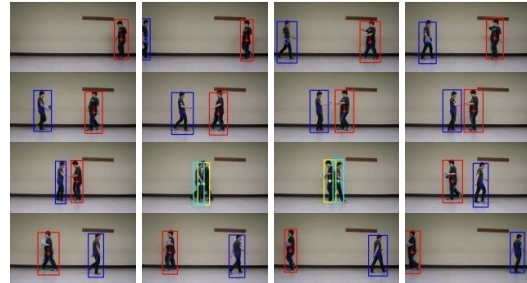


Fig. 8 An example of the motion detection and tracking for dual human interaction in a video sequence: Two human objects walked into the surveillance scene, in which interaction occurred (greeting and giving stuff)

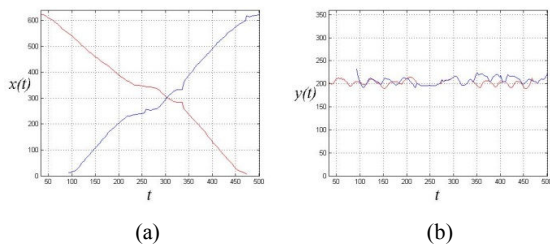


Fig. 9 Example of the trajectories for the video sequence (Fig. 8): (a) trajectory of the two human objects in the x direction; (b) trajectory of the two human objects in the y direction

As the trajectory of a human object is acquired, the derivatives of the trajectory $x(t)$ and $y(t)$ with respect to the frame index t are computed by:

$$\begin{aligned}\frac{dx}{dt} &= x(t) - x(t-1) \\ \frac{dy}{dt} &= y(t) - y(t-1)\end{aligned}\quad (13)$$

Here, numerical differentiations are used to approximate the derivatives (i.e., difference between the current location and the previous location). The derivatives are related to the *speed* of the moving objects. An example of the derivatives of the trajectory $x(t)$ is shown in Fig. 10. As seen, when the two human objects walked into the scene, they both maintained approximately constant speed initially. As the two human objects interacted, the speed of each object was associated with larger variances.

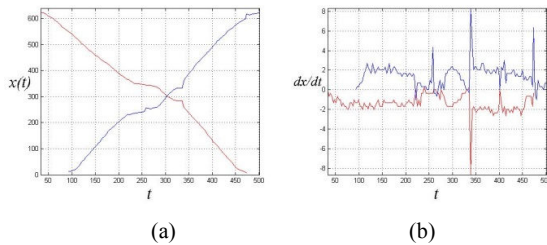


Fig. 10 An example of the derivatives of the trajectory in x direction for the video sequence in Fig. 8: (a) trajectory in the x direction (same as Fig. 9 (a)); (b) derivative of the trajectory

To further classify if the surveillance scene contains the two human objects in the events of interaction or non-interaction. The means of the derivatives for the two human objects (*obj1* and *obj2*) in the video sequence are computed by:

$$\begin{aligned}m_{obj1} &= \frac{1}{n_1} \sum_{t=1}^{n_1} \frac{dx}{dt}(t) \\ m_{obj2} &= \frac{1}{n_2} \sum_{t=1}^{n_2} \frac{dx}{dt}(t)\end{aligned}\quad (14)$$

where n_1 , n_2 are the number of frames when the two human objects appear in the video sequence. The means of the derivatives are related to the average speed for each human object during the time they appear in the surveillance scene. The average between the two means for the two human objects can also be obtained by:

$$\bar{m} = \frac{1}{2} (m_{obj1} + m_{obj2}) \quad (15)$$

A simple classifier for the *interaction* and *non-interaction* events is defined as:

$$Interaction = \begin{cases} Positive & \text{if } \bar{m} < T_I \\ Negative & \text{otherwise} \end{cases} \quad (16)$$

where T_I is a threshold for the classification of interaction events. The threshold is trained using a training database as described below.

In this study, a database of video sequences with the *interaction* or *non-interaction* events was collected. The database was first classified as *interaction* or *non-interaction* video sequences, denoted I and NI , based on visual inspection of the video sequences. Given the known *truth* of the events, we calculate the mean values according to the different types of video sequences by:

$$\begin{aligned}\bar{m}_I &= \frac{1}{N_1} \sum_{Videos=I} \bar{m} \\ \bar{m}_{NI} &= \frac{1}{N_2} \sum_{Videos=NI} \bar{m}\end{aligned}\quad (17)$$

where N_1 and N_2 are the total number of *interaction* and *non-interaction* video sequences, respectively, in the training database. Based on the two mean values, the threshold for the classification of interaction events can be defined by:

$$T_I = \frac{1}{2} (\bar{m}_I + \bar{m}_{NI}) \quad (18)$$

where the two mean values are computed using (17).

III. RESULTS

In this section, experimental results of system are presented. Research environment is summarized in Table I. Our system software was developed using C/C++ programming with the Open Source Computer Vision (OpenCV) library. The image resolution was with 1280×720 pixels, and all the images (frames) were reduced by half for computer processes.

TABLE I
RESEARCH ENVIRONMENT

CPU	Intel® Core™ i3 550 3.20GHz
Main Memory	4GB
Operating System	Microsoft Windows 7
Integrated Development Environment	Microsoft Visual Studio C++ 2010 With OpenCV2.4.8
Camera	BenQ DV-M1
Resolution	1280 × 720 (60fps)

A database of 60 digital video sequences was collected in our laboratory for system development and evaluation. Each digital video contains the scenario in which two human objects walk through the surveillance scene and may (or may not) interact. Table II summarizes the database used.

TABLE II
THE DATABASE FOR SYSTEM DEVELOPMENT AND EVALUATION

Database	
Interaction video	40
Non-Interaction video	20

During the system evaluation, the *Round-Robin* method was used. For each testing cycle, a video sequence was removed from the database, and the remaining database was used for

training. Therefore, the threshold for the classification of interaction and non-interaction events could be calculated accordingly and used to classify an interaction event as either *positive* or *negative*. Table III summarizes the system evaluation results. In the interaction events, our system achieved a classification accuracy of 80% in the *interaction* events and 95% in the *non-interaction* events.

TABLE III
SYSTEM EVALUATION RESULTS

	Classification	
	Positive	Negative
Interaction Videos	32	8
Non-Interaction Videos	1	19

Figs. 11 and 12 demonstrate the results of our system for the *non-interaction* events. In both events, two human objects walked through the scene and past by each other. Figs. 13 and 14 demonstrate the results of our system for the *interaction* events. In Fig. 13, an interaction event, i.e., *hand shaking*, was observed. In Fig. 14, an interaction event, i.e., *greeting and chatting*, was observed.

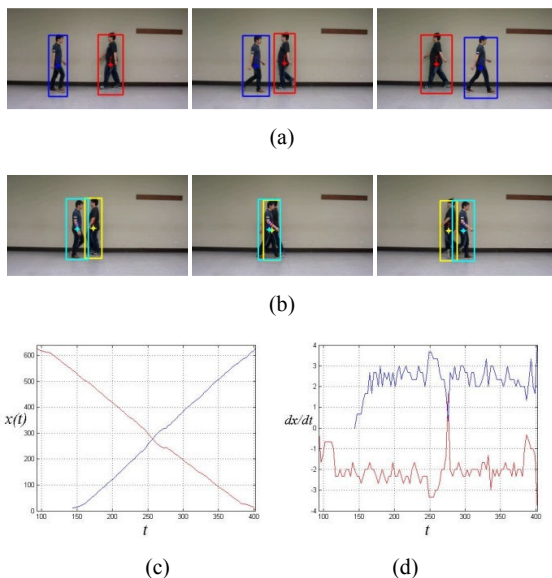


Fig. 11 An example of the system results for the *non-interaction* event: (a) result of motion detection; (b) result of motion tracking during occlusion; (c) trajectory of the two human objects; (d) derivatives of the two trajectories

IV. CONCLUSION

In this study, we presented a system for the *automatic motion trajectory analysis for dual human interaction using video sequences*. Our method included four major steps: *image preprocessing, human object detection, human object tracking, and motion trajectory analysis*. Our system was able to successfully detect moving human objects under various illumination conditions. In addition, the Kalman filter was found to provide reliable prediction and correction for the motion tracking in retaining the complete trajectories during

occlusion. These trajectories were then analyzed for the classification of *interaction* and *non-interaction* events.

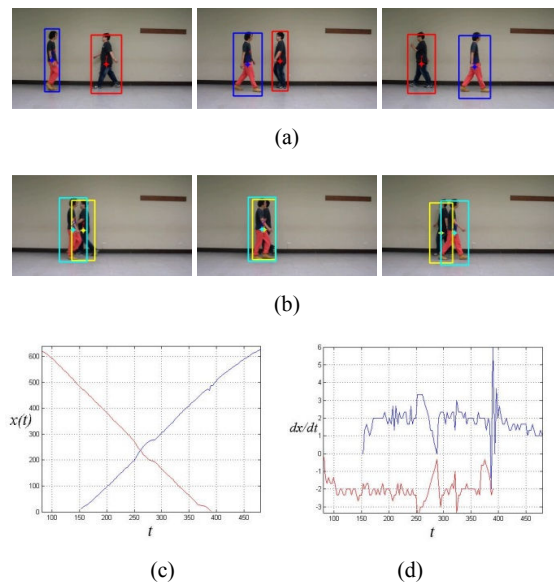


Fig. 12 An example of the system results for the *non-interaction* event: (a) result of motion detection; (b) result of motion tracking during occlusion; (c) trajectory of the two human objects; (d) derivatives of the two trajectories

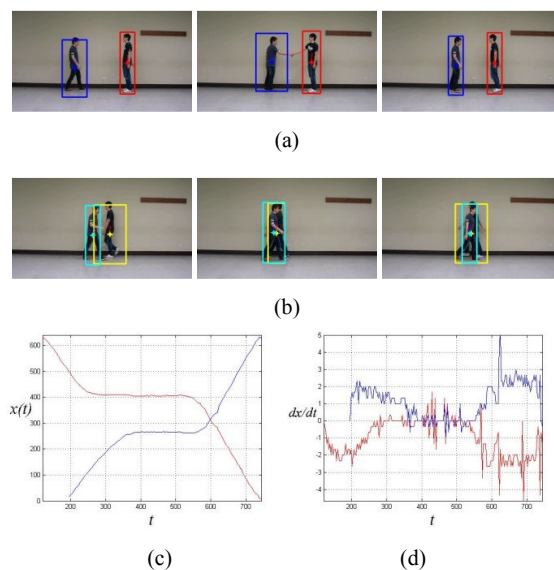
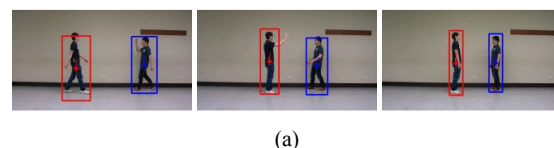


Fig. 13 An example of the system results for the *interaction* event *hand shaking*: (a) result of motion detection; (b) result of motion tracking during occlusion; (c) trajectory of the two human objects; (d) derivatives of the two trajectories



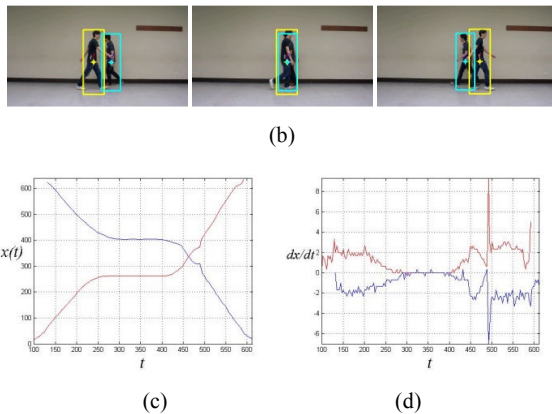


Fig. 14 An example of the system results for the *interaction event greeting and chatting*: (a) result of motion detection; (b) result of motion tracking during occlusion; (c) trajectory of the two human objects; (d) derivatives of the two trajectories

In a database of 60 video sequences with 40 *interaction* events and 20 *non-interaction* events, our system could achieve the classification accuracy of 80% in interaction video sequences, and 95% in non-interaction video sequences, respectively. The overall classification accuracy of our system was 87%. The results demonstrated that our system could effectively distinguish the interaction and non-interaction events involving two pedestrians in a surveillance scene.

We have explored the idea to investigate a system that could be used to automatically classify interaction events when two human objects were seen in the surveillance scene. Ultimately, our system could be integrated in an intelligent surveillance system to classify abnormal or criminal events (e.g., theft, snatch, fighting, etc.).

REFERENCES

- [1] S. Saravanakumar, A. Vadivel, and C. G. Saneem Ahmed, "Multiple human object tracking using background subtraction and shadow removal techniques," *International Conference on Signal and Image Processing (ICSIP)*, pp. 79-84, 2010.
- [2] P. P. Kuralkar, and V. T. Gaikwad, "Human object tracking using background subtraction and shadow removal techniques," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 3, March 2012.
- [3] F. Porikli, and O. Tuzel, "Human body tracking by adaptive background models and mean-shift analysis," *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2003.
- [4] P. Banerjee, and S. Sengupta, "Human motion detection and tracking for video surveillance," *Proceedings of the national Conference of Communications, IIT Bombay, Mumbai*, pp. 88-92, 2008.
- [5] R. S. Rakibe, and B. D. Patil, "Background subtraction algorithm based human motion detection," *International Journal of Scientific and Research Publications*, vol. 3, no.5, 2013.
- [6] W. Kim, and J. Lee, "Visual tracking using Snake based on target's contour information," *Industrial Electronics, Proceedings. ISIE 2001. IEEE International Symposium on IEEE*, vol. 1, pp. 176-180, 2001.
- [7] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel based object tracking," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 25, no.5, pp. 564-557, 2003.
- [8] F. Chang, L. Ma, and Y. Qiao, "Target tracking under occlusion by combining Integral-Intensity-Matching with multi-block-voting," in *Proc. of Lecture Notes*, vol. 3644, pp. 77-86, 2005.
- [9] P. F. Gabriel, et al., "The state of the art in multiple object tracking under occlusion in video sequences," *Advanced Concepts for Intelligent Vision Systems*, pp. 166-73, 2003.
- [10] Z. Li, A.K. Katsaggelos and B. Gandhi, "Fast video shot retrieval based on trace geometry matching," *IEE Proc.-Vis. Image Signal Process.*, vol. 152, no. 3, June 2005.
- [11] H.-Y. M. Liao, D.-Y. Chen, C.-W. Su and H.-R. Tyan, "Real-time event detection and its application to surveillance systems," *Proceedings of IEEE International Symposium on Circuits and Systems, ISCAS*, pp. 509-512, 2006.
- [12] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *The Canadian Cartographer*, vol. 10, no. 2, pp. 112-122, 1973.
- [13] Y. Zhou and T. S. Huang, "'Bag of segments' for motion trajectory analysis," *15th IEEE International Conference on Image Processing, ICIP*, pp. 757-760, 2008.
- [14] H.-I. Suk, A.K. Jaim and S.-W. Lee, "A Network of Dynamic Probabilistic Models for Human Interaction Analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 7, July 2011.
- [15] H. Habe, K. Honda and M. Kidode, "Human Interaction Analysis Base on Walking Pattern Transitions," *Third ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC*, pp. 1-8, 2009.
- [16] N. Otsu, "A threshold selection method from gray-Level histogram," *IEEE Trans. Systems, Man and Cybernetics*, vol. 9, no.1, pp. 62-66, 1979.
- [17] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd Edition. New York: Prentice Hall, 2002.
- [18] H. Freeman, *Boundary Encoding and Processing*, in *Picture Processing and Psychopictorics*, B S Lipkin and A Rosenfeld, Editors, Academic Press: New York. pp. 241-266, 1970.
- [19] P. F. Felzensz and D. P. Huttenlocher, "Distance transforms of sampled functions," *Technical Report TR2004-1963, Computer Science Department, Cornell University*, September 2004.
- [20] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35-45, 1960.