A SKELETON-BASED PAIRWISE CURVE MATCHING SCHEME FOR PEOPLE TRACKING IN A MULTI-CAMERA ENVIRONMENT

Chien-Hao Kuo[a], *Shih-Wei Sun*[b,c], *and Pao-Chi Chang*[a]

[a] Department of Communication Engineering, National Central University, Taoyuan, Taiwan
[b] Department of New Media Art, Taipei National University of the Arts, Taipei, Taiwan
[c] Center for Art and Technology, Taipei National University of the Arts, Taipei, Taiwan

ABSTRACT

In this paper, we propose a pairwise curve matching scheme in a multi-camera environment to handle the mis-tracking issue caused by occlusion problem happened in a single camera. According to the skeleton/joints of a human subject analyzed from a depth camera (e.g., Kinect), based the foot points (joints) used for people tracking in a field of view, we apply homography transformation to project the foot points from different views to a virtual bird's eye view, using Kalman filter to achieve people tracking with a pairwise curve matching. The contribution of this paper is trifold: (a) the proposed pairwise curve matching scheme can handle the occlusion problem happened in one of the cameras, (b) the complexity of the proposed scheme is low and affordable to be implemented in a realtime application, and (c) the implementation on a Kinect camera can provide satisfactory tracking results in a bright or extremely dark environment due to the skeletons/joints analyzed by the coded structured light-based infra-red (IR) sensor.

1 Introduction

Tracking multiple people using a multi-camera system is a challenging issue in recent years. In a surveillance application, when a suspicious human subject walks in an environment monitored by a multi-camera surveillance system, the cooperation among different cameras becomes important. However, a human subject occluded by another in a single camera environment would cause the blind spot effect. On the other hand, in a interactive application, a multi-camera environment can monitor a user in different observation point of views to compensate the blind spot from a single camera. For example, a human subject put his/her hand on the back, a front camera cannot observe the hand for gesture/pose recognition, providing a following interactive response from a system. Therefore, a multi-camera environment tends to be a important research issues for the possible daily life applications.

According to the literature [1, 2, 3], multi-camera tracking techniques have shifted from the monocular approaches [4, 5, 6, 7, 8, 9] toward the multi-camera approaches [1, 2, 3]. The tracking approaches using monocular camera aim to track people by a single camera. To name a few, most of the existing systems adopted blob-based [4, 5, 6] and color-based [7, 8, 9] approaches to perform tracking. In the conventional approaches, the foreground detection for a human subject plays the critical role for people tracking, either for blob-based or for color-based approaches. Recently, with the launching of Kinect camera [10] developted by Microsoft, the human subject in a indoor small distance environment (0.5m-3.5m from camera to a human subject) can be reliably obtained in an official software development kit (sdk) [11] environment for a single camera condition in the front view.

To track a human subject in a single camera environment, Kalman filtering [13, 14] and particle filtering [15, 16] are proposed to predict motions when occlusion occurs. In the Kinect sdk, the joints and a human subject of a single human subject can be tracked with satisfactory results when no occlusion occurs. However, when an occlusion from one person to another, the user ID of the occluded person would be given a new one, causing a mis-tracking problem, as shown in the right of Fig. 1 (a), the right human subject is given by user ID with blue color, but after occlusion, as shown in Fig. 1 (c), a another user ID with yellow color is uncorrected assigned by the Kinect sdk.

However, the conventional predictive filtering methods, e.g., Kalman filter, can only handle a short-term occlusion problem. To handle a long-term occlusion problem, other approaches need to be investigated. Among different potential solutions, utilizing multiple cameras to work together as a team is one of the best solutions to this problem. Nevertheless, a multi-camera tracking is still a challenging research issue due to: fusion of data extracted from multiple cameras, illumination difference at different locations, camera placement problem, and so forth.

To utilize the people tracking capability in bright/dark in the same system using the skeleton/joint analyzed results, there is few results to use multiple Kinect cameras to track human subjects. To name a few, Luber et al., [17] proposed to track people from a public space with non-overlapped field of views. The occlusion issue handling can directly adopt the single camera approaches, providing the similar occlusion handling capability. However, in this paper, we target on handling the occlusion problem in the overlapped field of view with much more complicated issues to be dealt with. There

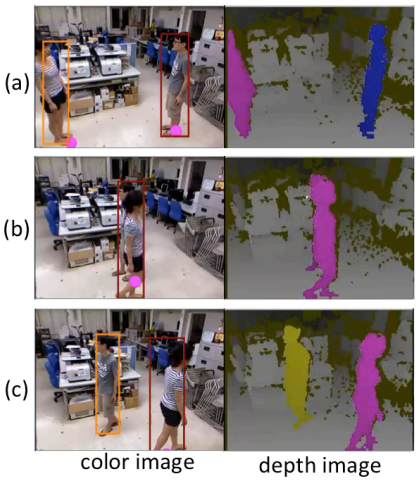

Fig. 1. The occlusion problem observed in color images and depth images (the foreground area in depth image, i.e., the right column, with the same color represents the same given user ID by Kinect sdk): (a) before occlusion, (b) during occlusion, and (c) after occlusion.

are three contributions in this paper: (a) we propose a pairwise curve matching scheme can handle the occlusion problem from one of the multi-cameras, (b) the proposed scheme has low complexity for further implementation in realtime applications, and (c) by adopting the proposed scheme to a multi-Kinect environment, the human subjects can be properly tracked both in bright or dark situations.

The rest of this paper is organized as follows. In Sec. 2, the framework of the proposed people tracking system, the proposed pairwise curve matching scheme, and the corresponding occlusion detection are presented. The experimental results are reported in Sec. 3. Finally, the conclusions and future work are given in Sec. 4.

2 Proposed People Tracking System

Fig. 2 shows the flowchart of the proposed people tracking system. The system is consist of three major parts: foreground detection, multi-object Kalman filter tracking, and multi-curve with occlusion handling, which will be described in detail in the following subsections.

Fig. 2. The flowchart of the proposed people tracking system.

2.1 Foreground Detection

In this paper, Kinect cameras are adopted for capturing the human subjects in a field of view. Meanwhile, the official Kinect sdk [11] is applied for foreground detection with reliable human detection results, providing the skeletons and joints tracking [12] results for human subjects, as shown in Fig. 3.

Fig. 3. The obtained results from Kinect sdk: (a) color image, (b) depth image with foreground detection, and (c) skeleton of a human subject.

We define a joint $j_{i,c,t} = (x_{i,c,t}, y_{i,c,t}, z_{i,c,t})$ to represent the 3D position in a real space of the i-th joint from the c-th camera at time t locating at the position of x, y, and z, respectively. For the right foot joint $i = rf$, left foot joint $i = lf$ are the most concerned joints in this paper. Hereafter, we use $j_{rf,c,t}$ to represent $j_{i=rf,c,t}$. Similarly, $j_{lf,c,t}$ to represent $j_{i=lf,c,t}$. The representative foot joint $j_{f,c,t}$ is assigned by the average of the 3D position from the two obtained left foot point and right foot point, as calculated by:

$$j_{f,c,t}(x_{f,c,t}, y_{f,c,t}, z_{f,c,t}) = (j_{rf,c,t} + j_{lf,c,t})/2, \quad (1)$$

for further people tracking.

Although the Kinect sdk can provide the tracking results for short-term occlusion from a single camera, the long-term occlusion problem happened from different cameras would cause the integrating for the results from the different tracks and different cameras difficult. Therefore, we apply the Kalman filter to track multiple object, as described in Sec. 2.2. Furthermore, the fusion issue to deal with the tracks from the different cameras will be described in detail in Sec. 2.3.

2.2 Kalman Filter for Multiple Object Tracking

According to the foot point obtained in Eq. (1), the Kalman filter is utilized for people tracking in a single camera, containing three sub-modules: motion model, assignment, and model update. Given a foot point: $j_{f,c,t}(x_{f,c,t}, y_{f,c,t}, z_{f,c,t})$, the z factor (depth from a camera, not concerned for tracking in this paper) is omitted. Therefore, hereafter, a foot point at the c-th camera is notated by: $j_{f,c,t}(x_{f,c,t}, y_{f,c,t})$.

According to the definition of a conventional Kalman filter [18], based on the given foot point $j_{f,c,t}(x_{f,c,t}, y_{f,c,t})$, the true state vector at time t is defined by as $[x^t_{f,c}, y^t_{f,c}, v^t_{f,x}, v^t_{f,y}]^T$. The variable $x^t_{f,c}, y^t_{f,c}$ represent the foot positions, and $v^t_{f,x}, v^t_{f,y}$ represent the velocities respectively. At time t, the measurement model of the true state is simply the foot positions $[x^t_{f,c}, y^t_{f,c}]$. After the states and measurement equations of motion model defining, the Kalman filter can be utilized to estimate the object's location and its trajectory in the next frame.

In the multiple object tracking for the assignment module, the Munkres algorithm (i.e. Hungarian algorithm)[19] was used to computes the optimal assignment for tracked foot points. After the assignment progress, the Euclidean distance

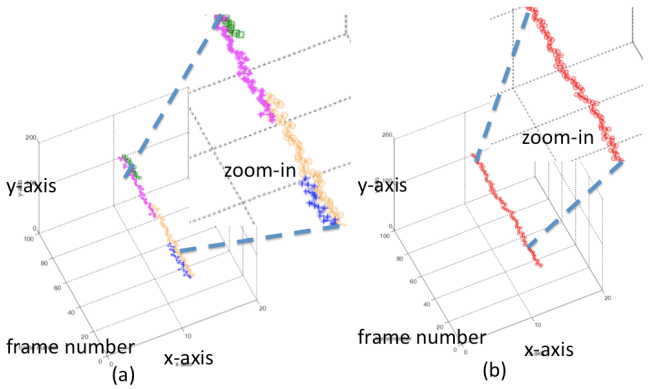

Fig. 4. The trajectory of an occlusion example shown in the virtual bird's eye view: (a) the pink dots and blue dots are the foot points detected belonging to the same person with Kalman filter tracking (occluded by another person at frame 30-50, the missing segment with red dashed line) in one view, and green dots and yellow dots are the foot points detected of the same person having occlusion effect (at frame 60-80, the missing segment with red dashed line) in another view, and (b) result of the proposed pairwise multi-curve matching (red dots) calculated from the uncorrected tracking results by Kalman filter(the dots in (a) with four different colors).

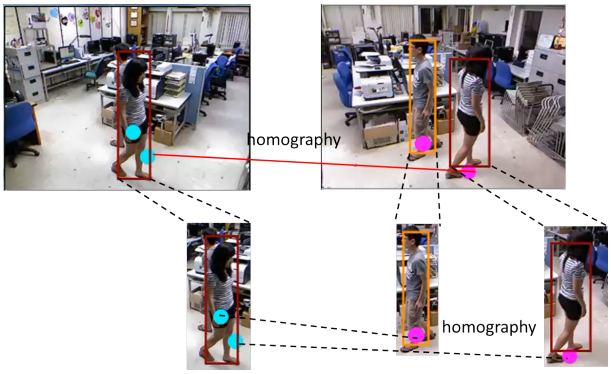

Fig. 5. Top: two example views of the human subject detection results by our system. The detected human subjects are bounded by rectangles. The calculated foot locations of different people are illustrated by different colored circles. Bottom: an occluded event is detected in the left example view where the two circles lie within the same rectangular region. In contrast to that, people associated to these two circles do not occlude each other in the example view on the right.

calculating is utilized to reject the points as the outliers. For the points without track assignment, a new tracker will be started by the Kalman filter.

As shown in the Fig.4 (a), directly using the Kalman filter for tracking would cause mis-tracking problem due to long-term occlusion. The dots assigned by different colors are the tracking results with multi-object Kalman filtering. The human subject in this camera is occluded by another twice for periods of time. To solve mis-tracking problem, we propose a multiple curve matching scheme to handle occlusion problems in a multi-camera environment, as described in section 2.3.

2.3 Multi-Curve Matching With Occlusion Handling

To deal with the mis-tracking problem, at first, the foot points should be transformed from different cameras to a virtual bird's eye view. Next, the occlusion event should be detected from each camera to clearly identify the missing segment, as shown in Fig. 4 (a). Finally, the tracks from different cameras should be merged as a single track, alleviating the mis-tracking problem.

2.3.1 Multi-Camera Projection

In the proposed system, the homography [20, 21] technique plays the role of matching corresponding objects among different views. Similar to the multi-camera people tracking systems [1, 2, 3], the object correspondence can be determined among different cameras using homography matrices H.

Let the captured foot point in a camera-1 at time t be $j_{f,1,t}$, and its corresponding point in another camera-2 be $j_{f,2,t}$. The corresponding foot point can be calculated from $j_{f,1,t}$ and H as:

$$[w(j_{f,2,t})^T; w] = H[(j_{f,1,t})^T; 1], \quad (2)$$

where w is a scalar. The homography matrices can be obtained by supplying the corresponding landmark points in different camera views at the initial state to the virtual bird's eye view.

2.3.2 Occlusion Detection: Multiple Points to One Region Relationship (M-to-One)

In this paper, we adopt our previous research result, the M-to-One relationship proposed by Sun et al. [22], to handle the occlusion issue in the pairwise cameras.

The upper-left part of Fig. 5 illustrates an occlusion event, in which the yellow square and the red square that belong to two different human subjects fall into a same object region. However, from the view observed by another camera (upper-right of Fig. 5), the two corresponding human subjects do not occlude each other. For a homography transform from one camera to another, if there are multiple foot points fall into the same foreground object area, an occlusion event is detected at that view of camera. Therefore, the M-to-one relation can be utilized to detect an occlusion event by fusing the information obtained from multiple cameras.

2.3.3 Proposed Pairwise Curve Matching

When an occlusion event is detected in Sec. 2.3.2, the boundary of the missing segments along the time axis can be clearly determined. The points falling into the missing segment can be treated as the outliers or noisy points for each track from the multiple-object Kalman filter tracking mentioned in Sec. 2.2. To alleviate the mis-tracking problem in Fig.4 (a), we pro-

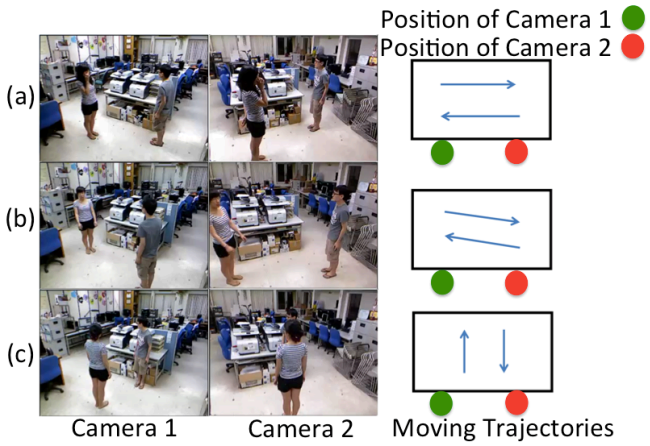

Fig. 6. The camera setting in the testing space and the corresponding human subject moving trajectories: (a) horizontal moving, (b) diagonal moving, and (c) vertical moving.

pose a pairwise curve matching scheme to merge the uncorrected assigned tracks from different views as a single track with reliable result.

Given a point in the virtual bird's eye view from time t and camera c, p_t^c, the merged point P_t is calculated from the points without occlusion events by:

$$P_t = (1/C) \sum_{c=1}^{C} p_t^c, \tag{3}$$

where C is total number of cameras that the foot points are detected without occlusion event at the c-th camera.

As shown in Fig.4 (b), according to the proposed operation in Eq. (3) for the detected foot points with Kalman filtering, the point merging results are shown by the red points as a single track (curve), which provides a reliable tracking result.

3 Experimental Results

In the experimental results, the official Kienct sdk 1.7 is adopted for implementation, providing 20 joints for each person, with at most 2 persons having skeleton/joints tracking and displaying results at the same time. Based on the limitation of the Kinect sdk, in our experiments, we tested at most two persons staying in the overlapped field of view at the same time. In addition, we setup two Kinect cameras in our lab for testing, with the angle between the observation point to the two cameras is set by $45°$ with overlapped area about $1.5m \times 1.5m$. The human moving trajectories and the camera setting in the testing space are depicted in Fig. 6.

As shown in Fig. 7, under bright lighting condition, the human subjects are visible in *Camera 1* and *Camera 2*, respectively. At first, the Kalman filter is applied for people tracking. As shown in the third column of Fig. 7, the occluded human subjects in the same view is uncorrected assigned as two independent tracks, leading the mis-tracking effect. However, by adopting the proposed scheme, the mis-tracking effect is properly alleviated. As shown in the fourth column of Fig. 7,

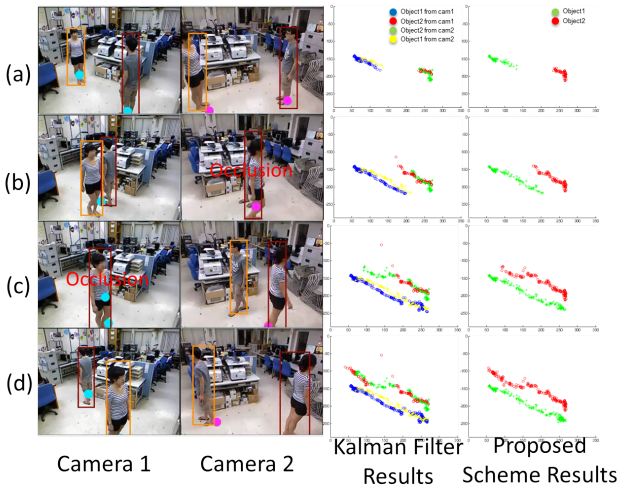

Fig. 7. The results for the bright scene: (a) before occlusion, (b) occlusion in *Camera 2*, (c) occlusion in *Camera 1*, (d) after occlusion.

red dots represent the proposed tracking result for the person standing in the right of Fig. 7 (a), and green dots represent the tracking results for another person. Even the person is occluded in Fig. 7 (b) and (c), the proposed pairwise curve matching can still properly provide satisfactory tracking results, as shown by the red dots and green dos in the fourth column of Fig. 7 .

Besides, in a dark scene test, the subjective results has the similar manner, as shown in Fig. 8. We should notice that the rectangles in the first and second column of Fig. 7 and Fig. 8 are the bounding rectangle of the detected human subjects in a single camera. Furthermore, the colored circles are the detected foot points projected from the other camera according to the obtained homography matrix. The close result from the circle to the bottom part of the corresponding rectangles represents that the given homography matrix at the initial state is proper for transformation between different cameras.

In the computational complexity results, as shown in Table 1, *Video-1* to *Video-3* are the corresponding video sequences under bright lighting conditions, as shown in Fig. 6. In addition, *Video-4* to *Video-6* are the testing video in dark lighting conditions with corresponding human subject moving patterns. The overall results of the computational complexity results is *0.0022* sec. (in an intel-i7 CPU with 8GB RAM for a Matlab implementation) for processing each frame. As a result, the proposed scheme is proper to be adopted in realtime applications.

4 Conclusions and Future Work

In this paper, we proposed a skeleton-based pairwise curve matching scheme for people tracking in a multi-camera environment. The proposed scheme can track multiple human subjects in a multi-camera environment with occlusion problem alleviated from the proposed pairwise curve matching based

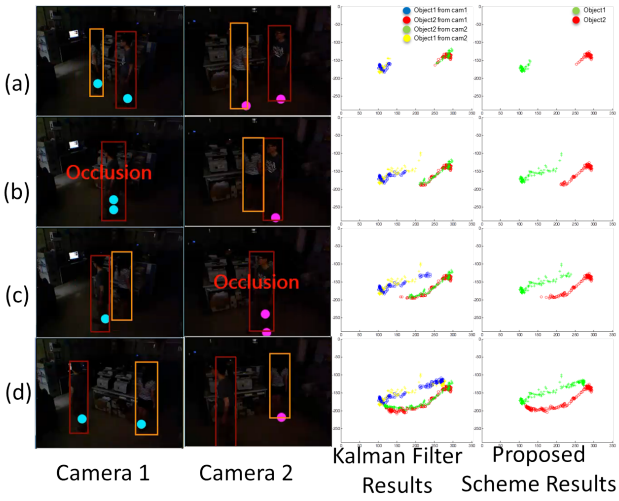

Fig. 8. The results for the dark scene: (a) before occlusion, (b) occlusion in *Camera 1*, (c) occlusion in *Camera 2*, (d) after occlusion.

Table 1. Computational complexity results for the test videos.

	Total Frame Number	Exe. Time (s)	Avg. Exe. Time (s)
Video-1	299	0.625	0.0021
Video-2	270	0.631	0.0023
Video-3	435	0.900	0.0021
Video-4	295	0.646	0.0022
Video-5	257	0.604	0.0024
Video-6	273	0.637	0.0023
Avg.	304.83	0.674	0.0022

on M-to-one occlusion detection. By integrating the proposed scheme to a Kinect official sdk, the obtained skeleton/joints of a human subject can provide reliable people detection results both in bright and dark situations. The proposed pairwise curve matching with low complexity can be efficiently adopted to the realtime applications. In the experimental results, in bright environment and dark environment verified that the proposed scheme can successfully track the human subjects in a multi-camera environment under different lighting conditions.

5 References

[1] W. Hu, M. Hu, X. Zhou, T. Tan, J. Lou, S. Maybank, "Principal Axis-Based Correspondence between Multiple Cameras for People Tracking", IEEE TPAMI, 2006.

[2] F. Fleuret, J. Berclaz, R. Lengagne, P. Fua, "Multicamera People Tracking with a Probablistic Occupancy Map", IEEE TPAMI, 2008.

[3] S.M. Khan, M. Shah, "Tracking Multiple Occluding People by Localizing on Multiple Scene Planes", IEEE TPAMI, 2009.

[4] I. Haritaoglu, D. Harwood, L. Davis, "Who, When, Where, What: A Real-Time System for Detecting and Tracking People", IEEE FG, 1998.

[5] R.T. Collins, "Mean-Shift Blob Tracking through Scale Space", IEEE CVPR, 2003.

[6] M. Han, W. Xu, H. Tao, Y. Gong, "An Algorithm for Multiple Object Trajectory Tracking", IEEE CVPR, 2004.

[7] D. Comaniciu, V. Ramesh, P. Meer, "Real-Time Tracking of Non-Rigid Objects Using Mean Shift", IEEE CVPR, 2000.

[8] S. Khan, M. Shah, "Tracking People in Presence of Occlusion", ACCV, 2000.

[9] Q. Cai, J. Aggarwal, "Automatic Tracking of Human Motion in Indoor Scenes Across Multiple Synchronized Video Streams", IEEE ICCV, 1998.

[10] http://www.microsoft.com/en-us/kinectforwindows/

[11] http://www.microsoft.com/en-us/kinectforwindows/Develop/developer-downloads.aspx

[12] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, "Real-time Human Pose Recognition in Parts From Single Depth Images", IEEE CVPR, 2011.

[13] I. Mikic, S. Santini, R. Jain, "Video Processing and Integration from Multiple Cameras", Proc. Image Understanding Workshop, 1998.

[14] J. Black, T. Ellis, P. Rosin, "Multi-view Image Surveillance and Tracking", IEEE WMVC, 2002.

[15] J. Giebel, D. Gavrila, C. Schnorr, "A Baysian Framework for Multi-cue 3D Object Tracking", ECCV, 2004.

[16] K. Smith, D. Gatica-Perez, J.-M Odobez, "Using Particles to Track Varying Numbers of Interacting People", IEEE CVPR, 2005.

[17] M. Luber, L. Spinello, and K.O. Arras, "People tracking in RGB-D data with on-line boosted target models", IEEE IROS, 2011.

[18] Kalman and R. Emil, "A New Approach to Linear Filtering and Prediction Problems", Transactions of the ASME–Journal of Basic Engineering, 1960.

[19] J. Munkres, "Algorithms for the Assignment and Transportation Problems", Journal of the Society for Industrial and Applied Mathematics, 1957 .

[20] K.J. Bradshow, L.D. Reid, D.W. Murray, "The Active Recovery of 3D Motion Trajectories and Tieir Use in Prediction", IEEE TPAMI, 1997.

[21] L. Lee, R. Romano, G. Stein, "Monitoring Activities from Multiple Video Streams: Establishing a Common Coordinate Frame", IEEE TPAMI, 2000.

[22] S.W. Sun, H.Y Lo, H.J. Lin, Y.S. Chen, F. Huang, and H.Y. M. Liao, "A Multi-Camera Tracking System That Can always Select A Better View to Perform Tracking", APSIPA ASC, 2009.