

Object Matching Across Multiple Non-overlapping Fields of View Using Fuzzy Logic

LOKE Yuan Ren¹ KUMAR Pankaj² RANGANATH Surendra¹ HUANG Wei-Min²

¹(Electrical and Computer Engineering, National University of Singapore,
4 Engineering Drive 3, Singapore 117576, Singapore)

²(Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613, Singapore)
(E-mail: yrloke@nus.edu.sg, pankaj.kumar@adelaide.edu.au, elesr@nus.edu.sg, wmhuang@i2r.a-star.edu.sg)

Abstract An approach based on fuzzy logic for matching both articulated and non-articulated objects across multiple non-overlapping field of views (FoVs) from multiple cameras is proposed. We call it fuzzy logic matching algorithm (FLMA). The approach uses the information of object motion, shape and camera topology for matching objects across camera views. The motion and shape information of targets are obtained by tracking them using a combination of ConDensation and CAMShift tracking algorithms. The information of camera topology is obtained and used by calculating the projective transformation of each view with the common ground plane. The algorithm is suitable for tracking non-rigid objects with both linear and non-linear motion. We show videos of tracking objects across multiple cameras based on FLMA. From our experiments, the system is able to correctly match the targets across views with a high accuracy.

Key words Multi-camera tracking, fuzzy logic, camera topology, particle filter, CAMShift

1 Introduction

Multi-target tracking across multiple cameras with non-overlapping field of views (FoVs) is a challenging task. The clues are the behaviors of the targets before they disappear in an FoV and after they reappear in another FoV. An approach based on fuzzy logic, fuzzy logic matching algorithm (FLMA) is proposed to find the correspondence of multiple targets in multi-camera non-overlapping FoV network. Such a system is useful for analysis of behaviors under multi-camera system, which can be applied for surveilling large area such as highways, shopping malls, office buildings *etc.*

An overview of the system is showed in Fig. 1. It consists of several intra-camera tracking systems and an inter-camera tracking system. Intra-camera tracking system is used for tracking targets in the FoV of each camera. The first few frames of the video is used to build the background model, that is used to detect the foreground objects. These foreground objects are initialized and tracked by using ConDensation^[1] and CAMShift^[2] tracking algorithms. Condensation algorithm is used for estimating the states of the targets and CAMShift is used for data association. We have used ConDensation algorithm for multiple target tracking in single camera FoV, where the data association is done using the search window approximated by the CAMshift filter for each target. It takes care of errors in tracking which may arise due to the splitting of targets, a common phenomena in detection of articulated objects. This has lead to significant improvement in tracking of articulated objects compared to previous work^[3], which used Kalman filter for tracking rigid targets with linear motion. The present work uses particle filter and CAMshift filter to track the targets even when they split into more than one measurement. Splitting of targets is especially true for articulated targets, with non linear motion.

Once a target leaves the FoV, inter-camera tracking system takes over the tracking task. In the inter-camera tracking system, the FoV of each camera is projected onto a common ground plane view (CGPV) by using projective transformation, which provides the position and motion of the targets in CGPV. By manually matching the points in the common ground plane with those in the image plane we compute the perspective transform of each view on to the common ground plane view (CGPV). A target's position and motion in CGPV and its shape information are used to match a new initialized target with the targets in the missing target list. Missing target list is a list that stores the states of the targets that have left the FoV's of other cameras in the camera network. Thus the objective of the inter-camera tracking system is to determine whether a new target in the FoV of a camera is a new target in the whole camera network or a missing target from other views. To address the challenging task of matching targets across views with no overlap, we have developed a system based on fuzzy logic to match targets across the FoV's of the different cameras. The system proposed here is able to track

both articulated and non articulated targets with non-linear or linear motion. Further more the FoVs of different cameras can be arbitrarily located to view different patches in the common ground plane.

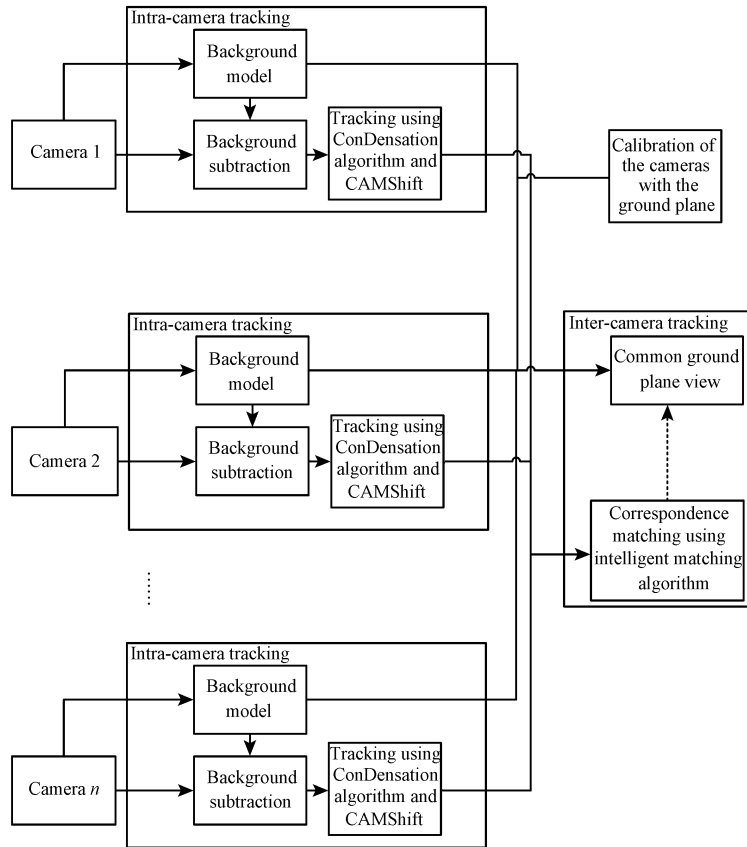


Fig. 1 Schematic block diagram of the multi-camera tracking system

2 Related work

In [4], Greg *et al.* showed a number of related works done by Kalman filtering for linear dynamic systems. Bar Shalom *et al.* attempted a mixture representation by extended Kalman filter (EKF) for tracking a nonlinear dynamic system. However, M. Isard *et al.*^[1] showed that Bar Shalom's EKF might not work because the mixture representation varies over time and the algorithm is not able to work on real-time system because of its high computational cost. They proposed ConDensation algorithm, a robust tracking algorithm by using finite number of samples to estimate the states of the targets.

Camera topology is the pattern of interconnection between cameras. Knowledge of camera topology in a camera network is very helpful in tracking targets across camera views. It can be used to track the targets from one camera to another. Javed *et al.*^[5] assumed linear arrangement and small gap between two cameras. This facilitated easy prediction of the reappearance of a target in a camera when the target disappeared from another camera. They used Kalman filter to track the targets in a single camera view. Huang *et al.*^[6] proposed a Gaussian probability distribution function (pdf) model for matching the targets between two camera view based on targets position, velocity, size and HSV color information. Chilgunde *et al.*^[3] used probabilistic formulation for intra camera target matching by modeling the state variables as Gaussian. T. Ellis *et al.* proposed an unsupervised learning algorithm^[7,8] to establish the topology of multiple camera networks. It can achieve the more sophisticated connectivity between cameras by using expectation-maximization.

3 Intra-camera multi-target tracking

The foreground objects are segmented out as blobs using background subtraction. Target tracking is initialized by the graph based initialization algorithm discussed in [9]. The solution of multi-target

tracking involves data association and state estimation. Each target will be either associated with a measurement in the previous frame or assigned as a new target status by data association. State estimation filter is used to predict the next state of each target and also to estimate the current state of the target. In this paper, CAMShift algorithm^[2] is applied for data association. ConDensation algorithm^[1] is used for state estimation.

3.1 Data association

Nearest-neighbor^[10] looks for the nearest-neighboring objects. This leads to errors in data association when two targets and their measurements are close to each other. Multiple hypothesis tracking (MHT)^[11] associated the data based on the probabilities of all possible combinations of matches between current observations and previous observations. The computation of MHT is expensive and is not suitable for real-time system. Joint Probabilistic Data Association (JPDA)^[12] assumes that the number of targets are fixed and known but the number of targets is changing from time to time in the data set we use for testing. CAMShift Algorithm is a non-parametric iterative technique to find the center of mass of a target and its search window based on likelihood of the color distribution model of the target. The proposed CAMShift in this paper is different from the original work^[2]. The search window, predicted by CAMShift filter, of the binary segmentation image obtained from background subtraction is applied with a Gaussian kernel, whose parameters are obtained from the above search window. The Gaussian kernel image is obtained as follows.

$$I_{ij}^{kernel} = I_{ij}^{Segment} \times e^{-\left(\frac{(i-i_c)^2}{\sigma_i} + \frac{(j-j_c)^2}{\sigma_j}\right)} \quad (1)$$

where $I_{ij}^{Segment}$ is the pixel value at i^{th} column and j^{th} row of the binary image; $e^{-\left(\frac{(i-i_c)^2}{\sigma_i} + \frac{(j-j_c)^2}{\sigma_j}\right)}$ is the Gaussian kernel, where i_c, j_c is the center of the search window and σ_i and σ_j are the variance of the Gaussian kernel in x and y direction respectively. The variance of the Gaussian kernel depends on the height and width of the search window. From experiments, σ_i is chosen to be $\frac{\text{width of the search window}}{1.58}$ and σ_j is $\frac{\text{height of the search window}}{1.58}$. The Gaussian kernel image, I_{ij}^{kernel} is used as input to the CAMShift filter. The application of Gaussian kernel, I_{ij}^{kernel} , reduces the instability in the computation of the center of mass of the target, which is common in the case of articulated targets. It also solves the problems of data association which arise due to splitting of targets into multiple blobs during segmentation.

3.2 State estimation filter

In the ConDensation algorithm, each object is represented by its state samples. Each sample consists of three components, $\mathbf{S}_t = \{S_t^n, \pi_t^n, c_t^n\}$, where S_t^n is the n^{th} sample states at time t ; π_t^n is the probability of n^{th} sample state at time t and c_t^n is the cumulative probability of n^{th} sample states at time t . The pdf of the states is obtained by the weights of the samples. In our experiment, the number of samples is 64. The position, x, y and velocity, V_x, V_y are used as the state of the target.

The ConDensation Algorithm is an iterative algorithm. The new states will be resampled for predicting the next new states. ConDensation state estimation consists of three steps: selection, prediction, and measurement. In selection step, N new samples are randomly selected based on the old sample-set pdf (See Appendix). The high probability samples will be drawn more frequently than the low probability samples. Each sample is propagated with a dynamic model as follow.

$$\begin{bmatrix} S_x^{(n)} \\ S_y^{(n)} \\ S_{V_x}^{(n)} \\ S_{V_y}^{(n)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} S'_x{}^{(n)} \\ S'_y{}^{(n)} \\ S'_{V_x}{}^{(n)} \\ S'_{V_y}{}^{(n)} \end{bmatrix} \quad (2)$$

where $S_x^{(n)}, S_y^{(n)}, S_{V_x}^{(n)}$, and $S_{V_y}^{(n)}$ are the n^{th} sample state of x, y, V_x , and V_y , respectively. $S'_x{}^{(n)}, S'_y{}^{(n)}, S'_{V_x}{}^{(n)}$ are the n^{th} sample state before applying the dynamic model. After that, the weight of each sample (π^n) is updated based on the measurements obtained from CAMShift. The update function is defined as follows.

$$\pi^n = e^{-\left(\left(\frac{S_x^n - x}{\phi_x}\right)^2 + \left(\frac{S_y^n - y}{\phi_y}\right)^2 + \left(\frac{S_{V_x}^n - V_x}{\phi_{V_x}}\right)^2 + \left(\frac{S_{V_y}^n - V_y}{\phi_{V_y}}\right)^2\right)} \quad (3)$$

where $\varphi_x, \varphi_y, \varphi_{V_x}$ and φ_{V_y} are the variance of the measurements in x, y, V_x , and V_y , respectively. In the experiments we choose the following values of these parameters $\varphi_x = \frac{width}{1.5}$, $\varphi_y = \frac{height}{1.5}$, $\varphi_{V_x} = \frac{width}{3.0}$, and $\varphi_{V_y} = \frac{height}{3.0}$; where $width$ is the width of target and $height$ is the height of target obtained from the CAMShift filter. Finally, the new state of the target is obtained from the expectation value of as $\mathbf{X}_t$:

$$E(\mathbf{X}_t) = \sum_{n=1}^N \pi_t^n S_t^n \quad (4)$$

4 Fuzzy logic matching algorithm

Fuzzy systems are capable of representing diverse, non-exact, uncertain, and inaccurate knowledge or information^[13]. It is a set of theory to express knowledge or information such as brightness, speed etc. as humans do. The expression based on the fuzzy sets and fuzzy membership functions can be used to make decisions. It has been widely used in control systems since 1980. The inter-camera correspondence matching problem is similar to some of the problems in control system where sometimes system information is not available due to limitation of the sensors. For example the problem of balancing inverted pendulum, due to limitation of the sensors and complexity of the system, it is a very difficult to balance an inverted pendulum. Similarly in our problem once a target moves into the blind region, none of the information about the target is directly available from the video sensors. Fuzzy logic has been successfully used to solve the problem of balancing inverted pendulums. Inspired by this application of fuzzy Logic we propose a fuzzy logic matching algorithm (FLMA) to match targets across-cameras with non-overlapping FoVs. Fig.2 shows the block diagram of our basic fuzzy logic based matching system.

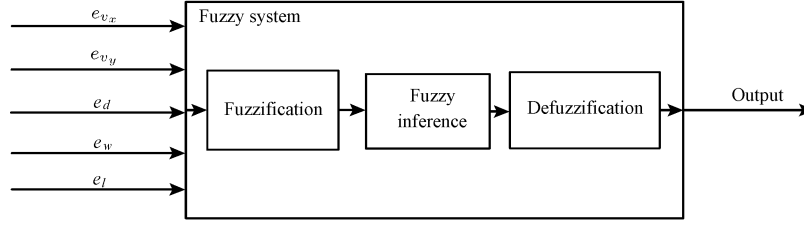


Fig. 2 The schematic block diagram of a basic fuzzy system. e_{v_x} and e_{v_y} are the error of speed in x and y direction, respectively. e_d is the error of direction. e_w and e_l are the error of size. The output of the system is the matching degree between the new target and the missing targets

4.1 Input of the Fuzzy Logic Matching Algorithm

When a target disappears from an FoV, its state, $O = [t_{out}, X, Y, V_x, V_y, W, L, ID]$ is saved in a list called missing target list. t_{out} is the frame number at which the target leaves the FoV of a camera. X, Y and V_x, V_y is the global position and velocity where the target disappears from the camera FoV. W and L are the width and height of the target respectively. ID is a number to identify the target. Every time a new target appears in the FoV of a camera then its states $O_{new} = [t_{in}, X', Y', V'_x, V'_y, W', L']$ is computed by target tracking initialization algorithm as a part of the intra-camera tracking system. To decide whether it is a new target in the entire camera network or an instance of reappearance of a target which has disappeared in another camera's FoV we use fuzzy logic to intelligently match the states of this target with the states of the targets in the missing target list.

The states of the new target and the states of target in missing target list are used to compute the following error functions: The error functions of speed are defined as

$$e_{v_x} = \left| \frac{X' - X}{t_{in} - t_{out}} - V_x \right|, \quad e_{v_y} = \left| \frac{Y' - Y}{t_{in} - t_{out}} - V_y \right| \quad (5)$$

The error function of velocity direction is defined as

$$e_d = \left| \tan^{-1} \frac{V'_y}{V'_x} - \tan^{-1} \frac{V_y}{V_x} \right| \quad (6)$$

The error functions of size are defined as

$$e_w = |W' - W|, \quad e_l = |L' - L| \quad (7)$$

4.2 The Structure of the Proposed Fuzzy System

Three steps are required to find the best match from the missing list by using fuzzy logic. They are (1) Fuzzification, (2) Fuzzy inference, and (3) Defuzzification^[14]. Fuzzification is the process of making a crisp quantity fuzzy. Crisp quantity is a clearly defined quantity. There is no uncertainty in crisp quantity. In contrast, fuzzy set is a representation of imprecise data. In the fuzzification module, each input has a corresponding membership function. Fig. 3 shows an example of a fuzzification membership function. The crisp set inputs such as $e_{v_x}, e_{v_y}, e_d, e_w$ and e_l etc. are represented as a fuzzy set for example

$$Fuzzy(e_{v_x}) = \begin{cases} z & \text{where } z = Z(e_{v_x}) \\ ps & \text{where } ps = PS(e_{v_x}) \\ pb & \text{where } pb = PB(e_{v_x}) \end{cases} \quad (8)$$

where z, ps and pb are called the degree of truth in Z, PS and PB membership functions respectively. Fuzzy ($\bullet$) is somehow describing the distance of $\bullet$ from the state zero, positive small and positive big.

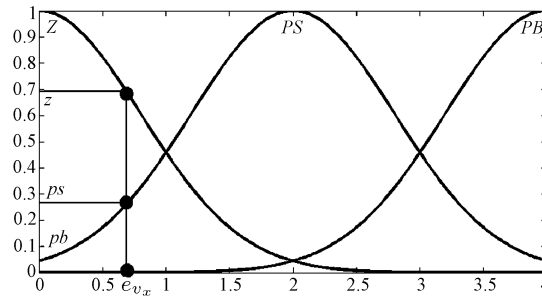


Fig. 3 A graphical plot of the membership function used for fuzzification of crisp variables. In this example, the states zero, positive small and positive big are 0, 2, and 4, respectively

Fuzzy inference computes fuzzy propositions with fuzzy set, which are input from the fuzzification module. Basically, there are M fuzzy if-then statements in the fuzzy inference module. Fuzzy logic is a set of functions based on Boolean algebra and it combines different fuzzy sets to form an if-else statements and returns an output based on the fuzzy sets. The following is an example showing definitions of three basic fuzzy logic connectives used in computing a match of a missing target with a new target:

Negation (NOT):

$$Fuzzy(\overline{e_{v_x}}) = 1 - Fuzzy(e_{v_x}) \quad (9)$$

Disjunction (OR):

$$Fuzzy(e_{v_x} \vee e_{v_y}) = \max(Fuzzy(e_{v_x}), Fuzzy(e_{v_y})) \quad (10)$$

Conjunction (AND):

$$Fuzzy(e_{v_x} \wedge e_{v_y}) = \min(Fuzzy(e_{v_x}), Fuzzy(e_{v_y})) \quad (11)$$

After the fuzzy inference module generates M outputs from M fuzzy if-then statements, the M outputs are used as inputs to the defuzzification module. The defuzzification module aggregates the inputs and returns a scalar value. The scalar value is named as the matching degree which is used for making matching decision here. There are a lot of methods, such as center-of-area defuzzification, center-of-sums defuzzification, center-of-largest-area defuzzification *etc.*, to defuzzify the inputs to a scalar value.

4.3 The Implementation of the proposed fuzzy system

First, the state is used to compute the error functions in (5), (6) and (7). These crisp variables are fuzzified by their corresponding membership functions. Next, the fuzzy inference module computes

the fuzzy proposition ($\{e_{v_x} \text{ is } Z\} \text{ AND } \{e_{v_y} \text{ is } Z\}$) based on fuzzy logic $\min(Fuzzy(e_{v_x}), Fuzzy(e_{v_y}))$. Then, Sugeno inference^[14] is applied to return a scalar value based on the output membership function, $output = F(input)$. The result is showed in Fig. 4. In the fuzzification module, the fuzzy sets are obtained by the Gaussian distance function. Therefore, the output membership function in the proposed fuzzy system here is an inverse of Gaussian function. It is defined as

$$Output = 1 - \sqrt{-0.2171 \times \ln(input)} \quad (12)$$

The defuzzification method is applied to aggregates these output into a scalar value. The best well-known defuzzification method which is known as the Center-of-Area is used here. This method computes the center of area below the output membership function as the *Matching Degree*. It is defined as

$$Matching\ Degree = \frac{\sum_{m=1}^M input_m \times F(input_m)}{\sum_{m=1}^M input_m} \quad (13)$$

Finally, the target in the missing target list with the highest matching degree to the new target is considered as the identical target if and only if the matching degree is greater than a predefined threshold value. The threshold value is obtained using a training data set where the matching targets have been manually labeled. In the results the same color is used to draw the targets' path in different views whose IDs are same.

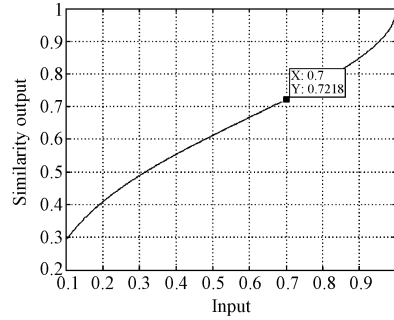


Fig. 4 Plot of the similarity output against input

5 Results

The proposed multiple-camera tracking system is implemented and tested on two and three-camera network systems. Fig. 5 shows the full ground plane view of the monitored area from a three camera system. The top image is from camera $C1$, the right image is from camera $C2$ and the left image is from camera $C3$. Fig. 6 shows the tracking result of a human in $C2$'s FoV, where the background is shown in black and foreground in grey, and tracking results in CGPV and in the image is shown by light gray track for this person. Fig. 7 shows the tracking result of another person in $C2$'s FoV and CGPV, its track is dark gray. Fig. 8 shows the reappearance of target labeled with light gray track or light track viewed in gray image in camera $C1$ it has been correctly matched with its previous occurrences and thus a light gray track is assigned to it. This figure also shows the track of the another target in camera $C3$ with a dark gray track. The track paths are drawn on the CGPV.

Another similar case is showed on Fig 7. Once the target left the camera view, the path in the tracking view will be removed but the path on CGPV remains unchanged. It can be clearly seen on Fig 7. The light gray track refers to the first target and dark gray track refers to the second target. In Fig 8, it shows that the first target was moved from one camera to another but its light gray path on the CGPV still remains unchanged. The matching degree of some experiment results are showed in Table 1.

In the experiments, the system can process 15fps (the frame size is 352×288) for a three-camera system with a P4 3.2 GHz. From the experiment, the system is able to correctly match targets across cameras with very high accuracy.

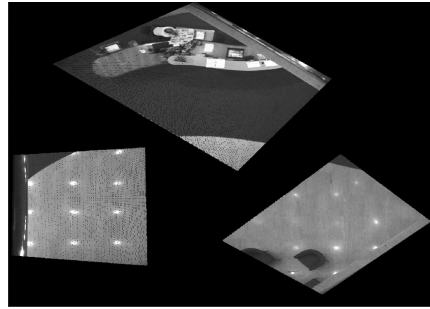


Fig. 5 The common ground plane view (CGPV) of a three camera system monitoring a reception area. The top image is from camera $C1$, the right image is from camera $C2$ and the left image is from camera $C3$

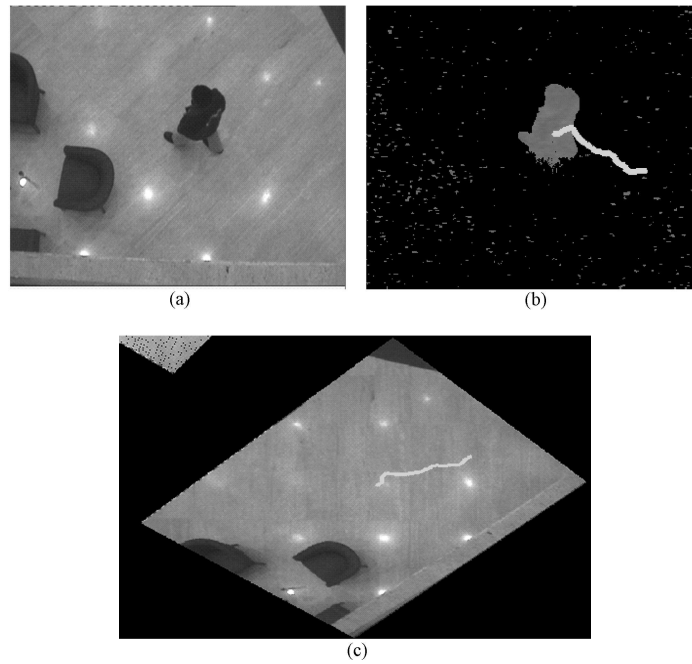


Fig. 6 Image (a) shows a frame from camera $C2$ with one target. Image (b) shows the segmentation and tracking result of this target in the image plane. Image (c) shows the tracking of this target in the CGPV. Note that the track of this target is light gray, as same color will be used when reappearance of this target is detected in other camera views

Table 1 The matching values of the fuzzy logic matching algorithm for multiple missing targets and multiple new targets

New target	Frame in	Camera No	Missing target	Frame out	Camera No	Matching degree
T_2	6186	3	T_1	4868	3	0.52
T_1	6550	1	T_1	4868	3	0.979
T_1	6550	1	T_2	6186	-	0
T_2	7762	1	T_2	6746	3	0.868
T_1	8362	2	T_2	8293	1	0.6217
T_1	8362	2	T_1	7217	1	0.91

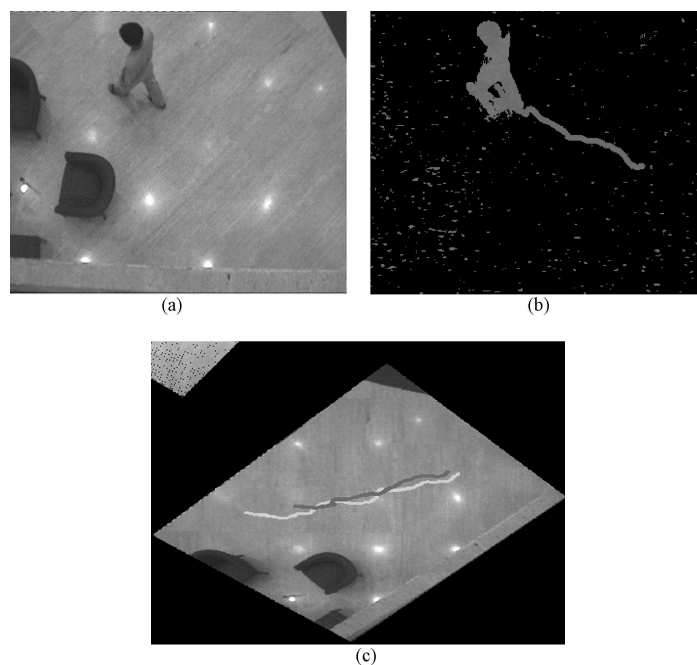


Fig. 7 Image (a) shows a frame from camera C2 with another target. Image (b) shows the segmentation and tracking result in the image plane. Image (c) shows the tracking result of this target in CGPV. Note that track of this target is labeled dark gray

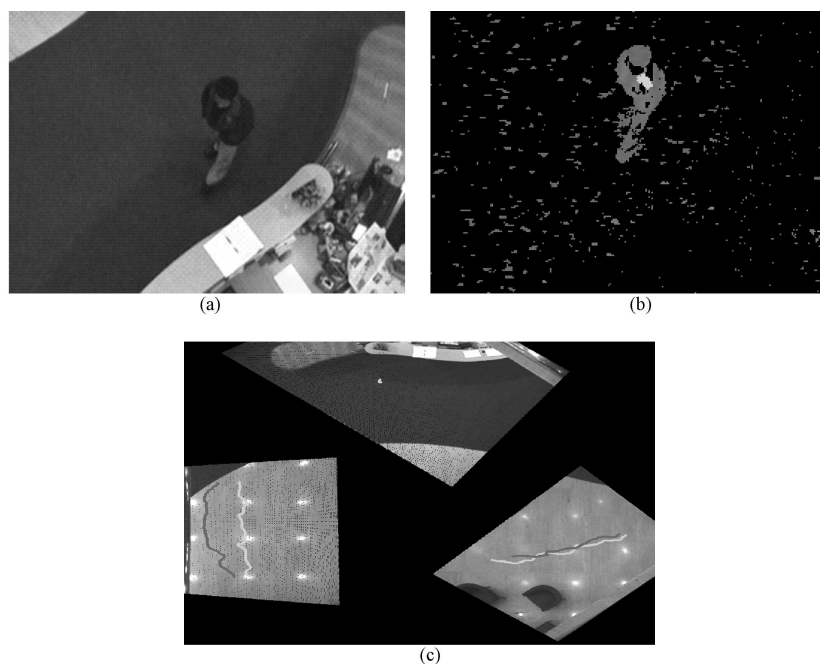


Fig. 8 Image (a) shows a frame with light gray labeled target in camera C1. Image (b) shows that correct light gray label is assigned to this target after Fuzzy Logic matching. Image (c) shows the tracking and correct matching result of the targets in the CGPV of camera C3

6 Conclusions and future work

We have successfully implemented a multiple camera system which can track objects across the views of non overlapping cameras with very high accuracy, using fuzzy logic to match the targets based on their states. The information of camera topology is used by obtaining the projective transforms of each camera with the ground plane view. We have developed a robust intra camera tracking system for articulated objects by a unique combination of Condensation and CAMShift based multi-target tracking. The combination has been able to successfully solve the problem of data association when a single target splits into multiple blobs which is common in segmentation of articulated objects. Thus we have obtained better tracking of articulated targets with non-linear motion.

At present the system works 15 fps for frame size (352×288). This can be further improved with the use of parallel programming techniques and multithreading. Application of distributed computing and grid computing techniques can also be applied to achieve 25 fps processing on each of the streams from the different cameras in the camera network. Since cameras of the multiple-camera tracking system in practice may be located far away among each other, efficient distribution of resources is required to save the transmission time of the information. Although the tracking system has been improved that allows the cameras to be arbitrarily positioned, it is limited on the same level of a building. However, the flexibility of FLMA allows the system to be improved for matching targets at different planes with a good color model and color-based similarity function as discussed in our paper^[11].

Appendix: Sampling algorithm

- 1) Given sample states, S_t^n and its corresponding probabilities, π_t^n and cumulative probability c_t^n .
- 2) Generate a random number r from $[0,1]$ (uniform distribution)
- 3) Map r to S_t^n as show in Fig. 9
 - Find the smallest n such that $c_t^n \geq r$
 - Return the corresponding S_t^n

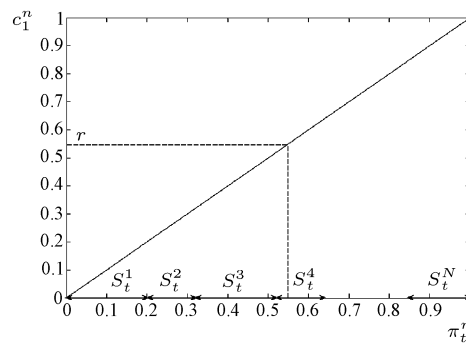


Fig. 9 Sampling algorithm for ConDensation

References

- 1 Isard M, Blake A. Contour tracking by stochastic propagation of conditional density. In: Proceedings of European Conference Computer Vision, Cambridge. UK: Springer, 1996. 1: 343~356
- 2 Gary R. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intelternational Technology Journal*, 1998, 2: 1~10
- 3 Chilgunde A, Kumar P, Ranganath S, Huang W. Multi-camera target tracking in blind regions of cameras with non-overlapping fields of view. In: Proceedings of British Machine Vision Conference, Kingston University, London, UK: IEE Press, 2004. 10: 397~406
- 4 Greg W, Bishop G. An introduction to the Kalman filter. In: Proceedings of SIGGRAPH. USA: Course 8. 2001. 81
- 5 Javed O, Rasheed Z, Shafique K, and Shah M. Tracking across multiple cameras with disjoint views. In: Proceedings of IEEE International Conference on Computer Vision. Nice, France: ACM Press, 2003. 165~171
- 6 Huang T, Russell S. Object identification: A Bayesian analysis with application to traffic surveillance. *Artificial Intelligence*, 1998, 103: 1~17
- 7 Ellis T J, Makris D, Black J. Learning a multicamera topology. In: Proceedings of IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. Nice, France: 2003. 165~171

- 8 Makris D, Ellis T, Black J. Bridging the gaps between cameras. In: Proceedings of Computer Vision and Pattern Recognition. 2004. **2**: 205~210
- 9 Kumar P, Ranganath S, Sengupta K, Huang W. Co-operative multi-target tracking and classification. In: Proceedings of European Conference on Computer Vision. Prague, Czech Republic: Springer, 2004. **1**: 376~389
- 10 Bar-Shalom Y, Fortmann T E. Tracking and Data Association. Academic press, 1988.
- 11 Kumar P, Ranganath S, Huang W, Gandhi S. Color calibration of multiple cameras for multi-target tracking. In: Proceedings of International Workshop on Sensor Networks and Applications, In conjunction with ICCV 2005. Beijing, China: IEEE Press, 2005. 65~70
- 12 Fortmann T E, Bar-Shalom Y, Scheffe M. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 1983, **8**(3): 173~183
- 13 Zadeh L A. Fuzzy Sets. Information and Control. New York, USA: Academic Press, 1965. 338~353
- 14 Takagi T, Sugeno M. Fuzzy identification of system and its application to modeling and control. In: Proceedings of International Workshop on Sensor Networks and Applications, USA: IEEE Press, 1985. 116~132

Loke Yuan Ren Ph.D. candidate at the National University of Singapore. His research interests include multi-camera multi-target-tracking and computer vision.

KUMAR Pankaj Received his bachelor degree in electrical and computer engineering from the Indian Institute of Technology, Delhi, the master and Ph.D. degrees from National University of Singapore. He worked as associate scientist at the Institute of Infocomm Research, Singapore for 3 years. Currently he is working as research fellow at the School of Computer Science, University of Adelaide. His research interests include computer vision, artificial intelligence, behavior analysis, multi-camera tracking, and surveillance.

RANGANATH Surendra Received his bachelor degree in electrical engineering from the Indian Institute of Technology (Kanpur), the master degree in electrical communication engineering from the Indian Institute of Science (Bangalore) and the Ph.D degree in Electrical Engineering from the University of California (Davis). From 1982 to 1985, he was with the Applied Research Group at Tektronix, Inc., Beaverton, OR where he was working in the area of digital video processing for enhanced and high definition TV. From 1986 to 1991, he was with the medical imaging group at Philips Laboratories, Briarcliff Manor, NY. In 1991, he joined the Department of Electrical and Computer Engineering at the National University of Singapore, where he is currently an Associate Professor. His research interests include digital signal and image processing, computer vision, pattern recognition and neural networks with focus on human-computer interaction, and video understanding applications.

Huang Weimin Received his bachelor in automation, master and Ph.D. degrees in computer engineering from Tsinghua University, Beijing, China, in 1989, 1991, and 1996, respectively. He is a research scientist at the Institute for Infocomm Research, Singapore. He has worked on research of handwriting signature verification, biometrics authentication and audio/video event detection. His research interests include pattern recognition, image processing, computer vision, human computer interaction, and statistical learning.