MNK Publication
International Journal of Latest Research in Science and Technology **ISSN (Online):2278-5299**
Volume 3, Issue 2: Page No.66-72 ,March-April, 2014
https://www.mnkpublication.com/journal/ijlrst/index.php

MULTI-VIEWPOINT TRACKING FOR VISUAL SURVEILLANCE USING SINGLE DOME CAMERA

[1]Malik Muhammad Khan, [2]Intaek Kim, [3]Youngsung Soh
Myongji University, Yongin, 449-728, Korea

Abstract- ***Efficient surveillance systems crave to track multiple objects throughout the scene with unique tracking IDs. Tracking multiple objects robustly requires invariant attributes matching. If objects have to be tracked across viewpoints, then a technique is required to reassign those IDs. Such an approach is proposed which combines extended Kalman filter with past information for isolated object tracking. And whenever objects partially or fully overlapped, features like color, size and object class are used to distinguish between them. To track objects across viewpoints, color matching and expected position is find out. The proposed method is also robust to background modeling technique. The system was tested in real world application and successful results were obtained.***

Keywords - Multi-viewpoint tracking; Inter-viewpoint tracking; Multi-target Tracking; Kalman Filtering; Robust Object Tracking.

I. INTRODUCTION

Today with the advancement of computer speed, the overall deployed systems are becoming smaller. So, handful apparatus could help you to attain desired results. Such an approach is used in this paper. Single dome camera is installed at road junction to track the objects across viewpoints with same tracking ID.

Owing to the appearance variability, motion blur and occlusion, obtaining the locations and trajectories of targets become significantly challenging. Various methods have been purposed and improved to deal this problem. Most of them either track single object or a camera is installed at each viewpoint. Therefore we intend to track multiple objects using single dome camera.

This paper is divided into four parts. Firstly, object extraction using spatio-temporal Gaussian mixture model (STGMM) technique proposed by Soh *et al.* [1]. The proposed technique eliminates the noise from the foreground and is robust to uniform and non-uniform motions in the background.

At second stage, extended Kalman filter (EKF) is exploited to track non-linear moving objects. The dome camera is installed at road junction, where cars could move much faster than walking people, so linear Kalman filter was replaced with EKF.

To track objects under mild or extreme occlusion, color information is utilized at third stage. Histogram matching was found best for comparing color information. The color of an object is an invariant attribute, also used to reassign tracking IDs across the viewpoints.

Finally, extracted features of tracked objects are saved in separate data files for later surveillance. The extracted features include track, color, time of appearance and leaving the scene and object class.

II. RELATED WORKS

Various techniques have been proposed and improved for single viewpoint tracking (SVT). However, only few techniques are available to track objects across viewpoints i.e. multi-viewpoint tracking (MVT). We would briefly review only few well-known techniques in both aspects.

1. SINGLE VIEWPOINT TRACKING

Kim *et al.* [2] robustly track multiple moving objects for intelligent video surveillance system. The proposed method uses red-green-blue (RGB) color background modelling to precisely extract foreground. The predicted direction and velocity of objects are also taken into consideration.

A good approach to track human under occlusion is bring forth by Chu *et al.* [3]. They propose use of adaptive multiple kernels with projected gradient. They also use Kalman filter to enable automatic tracking. Therefore, an object giving less information because of being overlapped by object can be still tracked efficiently.

Another method is proposed by Chitra *et al.* [4]. Objects are tracked efficiently by proposed method even they are overlapped completely. Histograms of oriented gradients are compared to decide between occluded objects using support vector machine (SVM).

Tracking objects precisely when they are occluded is a challenging task. Dang *et al.* [5] not only accept it but also accomplish it. They propose a robust data association with filtering procedure to handle occlusion and temporal lost problem. They use census transform technique and block-by-block strategy is employed to generate the confidence map. Then, the prior motion cues and observations are used for inferring the state of the occluded face.

Tracking not only humans but also vehicles is becoming hot topic nowadays. A method is proposed by Sheng *et al.* [6], where they use multiple visual features to distinguish one vehicle from another. Particle filter is also used to make tracking robust. To avoid incorrect results caused by model drift, model updating is controlled precisely.

Publication History

Manuscript Received	:	*25 April 2014*
Manuscript Accepted	:	*28 April 2014*
Revision Received	:	*29 April 2014*
Manuscript Published	:	*30 April 2014*

2. MULTI-VIEWPOINT TRACKING

A new method for multi-viewpoint tracking is proposed therefore finding related works which use one camera to track more than one viewpoint was hard task. Therefore, problems faced by researchers when they use set of cameras to track an object are reviewed here. It is somehow similar to use data taken from one viewpoint and matching it to another viewpoint.

A network of co-operative cameras for visual surveillance of parking lots is presented by Micheloni *et al.* [7]. The proposed system tracks multiple targets simultaneously throughout the controlled area. The positions of objects are computed from different sensors to get robust results. Active camera system compensates background changes; hence the proposed method autonomously tracks the interested objects.

Waist *et al.* [8] propose an approach to track several subjects captured by multiple cameras in real time. Each human subject is represented by a parametric ellipsoid. The likelihood of each subject is computed using their positions, heights and velocities. Moreover, particle filter is implemented on a graphic processing unit to increase the overall processing speed of the system.

To overcome multi-target occlusion problem, Wei *et al.* [9] proposed a distributed Bayesian framework using multiple collaborative cameras for robust and efficient multiple-target tracking in crowded environments with significant and persistent occlusion. The likelihood for multi-target occlusion is calculated using their views in different cameras.

Tracking becomes more complicated when the video is low-resolution. It becomes more severe when one has to track multiple objects using multiple cameras. The method proposed by Cristian *et al.* [10] tackles the above mentioned problems. They propose a method which generates 3D color representation of the scene. The Bayesian approach is used to efficiently create and destroy tracks. The likelihood function is based on local neighbourhoods in particle filter to decrease the computational time. Specifically, they propose the camera collaboration likelihood density by using epi-polar geometry with sequential Monte Carlo implementation.

III. PROPOSED METHOD

The proposed method tracks multiple objects across the viewpoints with same tracking ID efficiently. Single dome camera is used to capture all four viewpoints at road junction. Therefore, less apparatus can give good results using the proposed method.

1. OBJECT DETECTION

First and foremost important aspect in robust tracking is efficient foreground extraction. We reviewed the STGMM proposed by Soh *et al.* [1] for background modeling. The proposed technique deals with temporal behavior as well as spatial relations. Comprehensive description can be reviewed in [11]. A notable thing is, while doing multi-viewpoint tracking the background should be settled first for each viewpoint.

2. TRACKING

The proposed method is divided into parts. First is single viewpoint tracking which exploits extended Kalman filtering for isolated object tracking and color matching for confused situations i.e., occlusion. Second part explains multi-viewpoint tracking. It consists of intra-viewpoint tracking and inter-viewpoint tracking. For intra-viewpoint tracking it is same as single viewpoint tracking, whereas, inter-viewpoint tracking matches the color, velocity and expected position of objects in the present viewpoint to the objects in the previous viewpoints. A comparison is also carried out between single viewpoint tracking and multi-viewpoint tracking.

3. SINGLE VIEWPOINT TRACKING

1) Extended Kalman Filtering:

For tracking objects, we used EKF instead of LKF because mostly objects do not move at constant speed; cars and waking people. So, the state variables and measurements no longer remains linear combination of state variables. The key variables used in EKF were state estimate (x_k) and measurement (z_k) whose relation can be depicted from Eq. 1 and Eq. 2. This is advance work of our earlier research, so detailed description of EKF can be seen in [11].

$$x_k = Ax_{k-1} + w_{k-1} \tag{1}$$

where, A represents the transition matrix and x_k the state at time $k-1$ to k. Vector w_{k-1} is the Gaussian process noise with normal probability distribution. We preferred, $A = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix}$, where dt = 0.01 and $x_{k-1} = \begin{bmatrix} xPosition \\ yPosition \end{bmatrix}$ at time $k-1$. w_{k-1} was neglected because we were working on video.

$$z_k = Hx_k + v_k \tag{2}$$

where, H is the measurement matrix and z_k is the measurement observed at time $k-1$ to k respectively. v_k is also the Gaussian process noise with normal probability distribution, so neglected. $z_k = \begin{bmatrix} xPosition \\ yPosition \end{bmatrix}$ at time k and $H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

2) Feature Extraction and Matching:

The foreground extracted from STGMM was labeled and color, size and object class attributes were extracted. These features were well utilized to reassign object IDs at separation after merging or overlapping. Brief explanation of these features follows:

a) Color The RGB values of each object were stored in three independent histograms as they appear in the scene. These histograms are matched at separation after being merged using Bhattacharyya distance. Detailed explanation of color matching is given in our prior work [12].

b) Size The size of object was taken as area of bounding box i.e., Length*Width. It was very important feature while deciding between objects while they merged for few frames only. Also, object size helped to distinguish a person, group of people and vehicles in the scene.

c) Object Class The object class was helpful attribute for robust tracking. Because algorithm might consider group of people as single vehicle when they were far from the camera, as their size were almost similar. And probably we would have lost the track.

Algorithm: SVT

```
For Each Image
    For Each Foreground
        Find Center-point, Color, Size and Object Class
        Store Attributes and Find Kalman Prediction
    End of For loop
End of For loop
For Each Object X
    If (New Color & Size (after separation) – Previous Color & Size (before merging)) < Threshold
        Same Object X
    End If
    Else
        New Object Y
    End Else
End of For loop
```

Fig. 1 The proposed SVT Algorithm in pseudo code

4. MULTI-VIEWPOINT TRACKING

1) Intra-viewpoint Tracking:

Intra-viewpoint tracking is same as single viewpoint tracking. It uses extended Kalman filter and attribute matching. Every object has its unique tracking ID and tracked information. The proposed method for single viewpoint tracking or intra-viewpoint tracking is explained in Fig. 2.

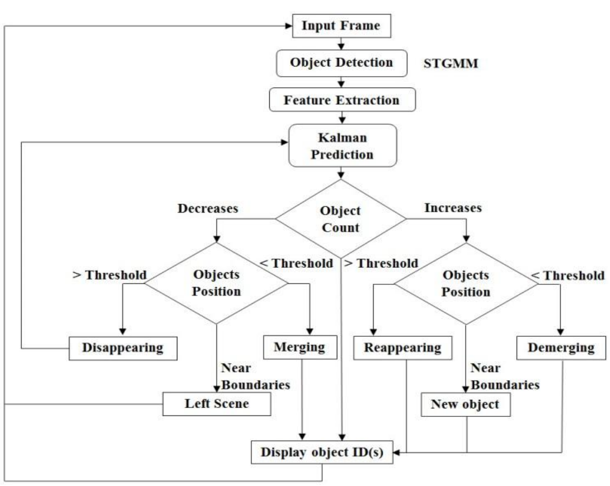


Fig. 2 The flowchart of proposed method

Merging is called when one or more objects overlapped each other and the resulted foreground appears as single blob, while, Demerging/Separation is the division of one blob into two or more, when the overlapped objects get isolated. The objects may be disappeared due to poor background modelling, so this problem is also addressed, to do robust tracking.

2) Inter-viewpoint tracking:

Tracking objects in one viewpoint with the same ID with which it had been already tracked in some other viewpoint is called Inter-viewpoint tracking. It is a challenging task when single camera is used, because objects are not being tracked frame by frame. The objects may or may not appear current viewpoint. Moreover, they could appear as merged object. Also, the objects could accelerate or slowdown while moving from one viewpoint to another. Such cases can be easily depicted from the scene given in Fig. 3. The objects are tracked in inside the area shown for each viewpoint. The area of dome camera transition i.e., moving from one viewpoint to another is called blind area. No tracking is done under blind area.

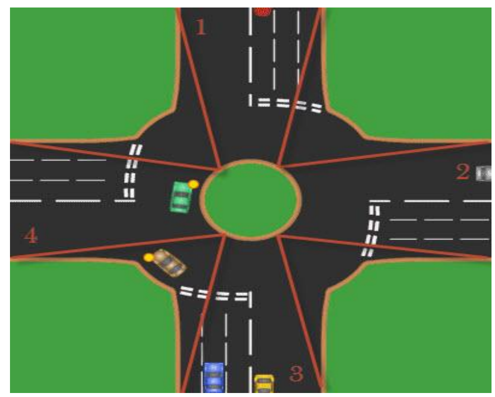


Fig. 3 The flowchart of proposed method

The proposed method robustly tracks objects across viewpoints with same tracking ID efficiently. The technique is very simple and effective. First of all, in each viewpoint, intra-viewpoint tracking is done. All objects are assigned unique IDs. Then, each object's features such as color, size, expected speed and velocity are matched to the objects in previous viewpoints. If matched, the ID and tracking attributes are assigned. The proposed algorithm can be seen in Fig. 4.

Algorithm: MVT

```
For rotation <= 2
    For (current viewpoint: starting ID to ending ID)
        For (last viewpoint: starting ID to ending ID)
            If {[Distance Threshold < (Object expected position in current viewpoint – Object position in last viewpoint)] && Color Similarity < Color Threshold}
                Same Object X
            End If
        End of For loop
    End of For loop
End of For loop

where:
    rotation = 1 cycle (4 viewpoints)
    If current viewpoint = 1 then last viewpoints are 4,3 and 2
```

Fig. 4 The proposed MVT Algorithm in pseudo code

IV. EXPERIMENTAL RESULTS

1. SINGLE VIEWPOINT TRACKING

The proposed method was tested and robust results were obtained for isolated and occluded objects tracking. The experiment was carried out on a data set of 24bit 1047 bmp files having dimensions 480*320. The testing platform was C++ using Visual Studio 2010 installed on Intel Core i5-3470 CPU @ 3.20 GHz with 8 GB RAM and NVIDIA GeForce GTX 550 graphic accelerator card installed. Only notable results for isolated and merged cases are mentioned below.

a) Isolated Object Tracking: EKF with past information of objects were exploited to track isolated moving objects. Fig. 5(b) shows a scenario where object with tracking ID 0 breaks in the STGMM but still neither we lost track nor new ID is assigned, that is shown in Fig. 5(c).

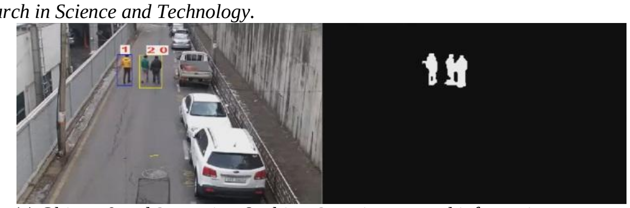
(a)
(a) Objects detected

(b)
(b) Object broken in STGMM

(c) (c) Objects with same tracking ID

Fig. 5 Experimental results of isolated object tracking; poor background modeling case

b) Overlapped Object Tracking: When the interested object is overlapped by other object(s), keeping precise track of interested object is quite hard task. It requires invariant attribute matching to recognize merging and separation cases. Fig. 6(a) shows two objects are merged i.e., three objects in the scene but two blobs, the merged IDs are shown in output. In Fig. 6(b) all of the objects in the scene merged hence as shown in output. Fig. 6(c) represents a case when the object 2 which had merged information of other objects as shown in Fig. 6(b), itself separated leaving behind object 0 and 1 merged. Here, color alone was insufficient to name merged object as “0 1” and stored the merged information. Therefore, size and number of objects near the separation were calculated. So, when object 0 and 1 were separated, all objects got their correct IDs using color matching as shown in Fig. 6(d).



(a) Objects 0 and 2 merging & object 2 storing merged information

(b) All of three objects merged and object 2 storing all merged information

(c) Object 2 separating & passing merged information to object 0

(d) Re-tracking all objects with original tracking IDs

Fig. 6 Experimental results of overlapped object tracking; the object with merged information separated case

2. MULTI-VIEWPOINT TRACKING

Tracking the same object in different viewpoints with same ID was the goal of the proposed method. To track objects across viewpoints, their color, expected position and velocities were precisely matched. Experiment was done on a data set of 24 bit 942 bmp files with dimensions 320*180 using C++ platform with Visual Studio 2010 installed on Intel Core i5-3470 CPU @ 3.20 GHz with 8 GB RAM and NVIDIA GeForce GTX 550 graphic accelerator card installed. Robust tracking results were obtained. Only few notable cases are mentioned below. The multi-viewpoint tracking is divided into two steps: Intra-viewpoint tracking and Inter-viewpoint tracking.

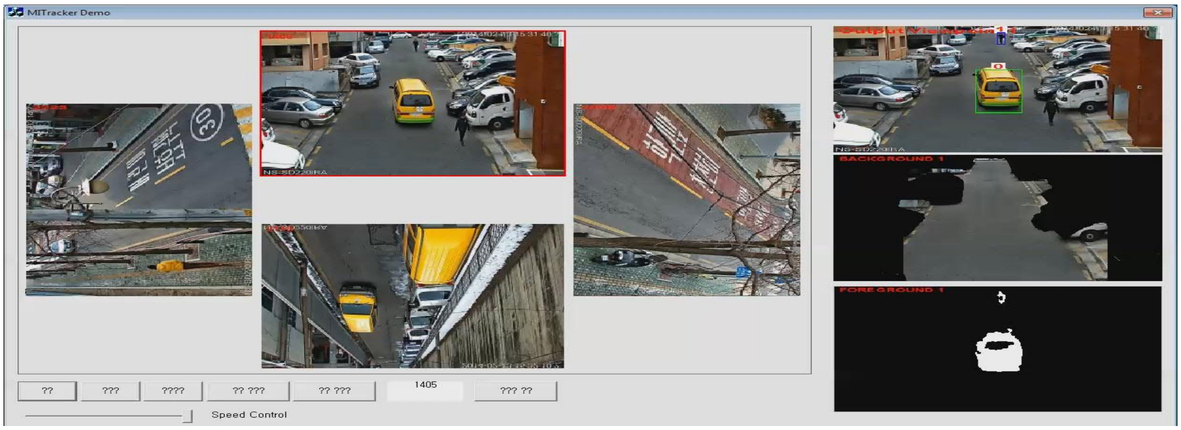
a) Intra-viewpoint Tracking: The procedure for intra-viewpoint tracking is same as SVT. The detected objects were assigned IDs. EKF was applied to position of objects and key attributes like size, color and object class, were extracted and stored. Each figure in Fig. 7(a) to 7(g) represents intra-viewpoint tracking for four viewpoints, when the objects appeared.

b) Inter-viewpoint Tracking: The algorithm explained in Fig. 4 was implemented for inter-viewpoint tracking to accomplish MVT. The detected objects were assigned IDs and then they were matched using their color, expected position in current viewpoint and velocity. The tracking of objects were not done under blind area i.e., dome camera switching between one viewpoint to another. First iteration of data set, the tracking was not begun to set the STGMM. After that, two objects were detected in first rotation and first viewpoint as shown in Fig. 7(a). Later third and fourth objects were detected as they appeared in STGMM but we only quoting here notable results.

In Fig.7(b) object with tracking ID 4 is tracked in second viewpoint of first rotation which was one of our two interested objects, which would later appear in other viewpoints. IDs were given to the objects detected in Fig. 7(c). The object with tracking ID 7 in rotation# 1 viewpoint# 3 was matched with object ID 4 in rotation# 1 viewpoint# 2 on basis of color and expected position calculated by velocity and time taken to leave viewpoint 2 and enter viewpoint 3.

So, ID and other attributes were returned. Hence, the yellow guy in the scene retained its ID i.e., 4. Another interested object in the scene is object with tracking ID 6. We tracked both objects 4 and 6, with same IDs in each and every viewpoint of all dataset.

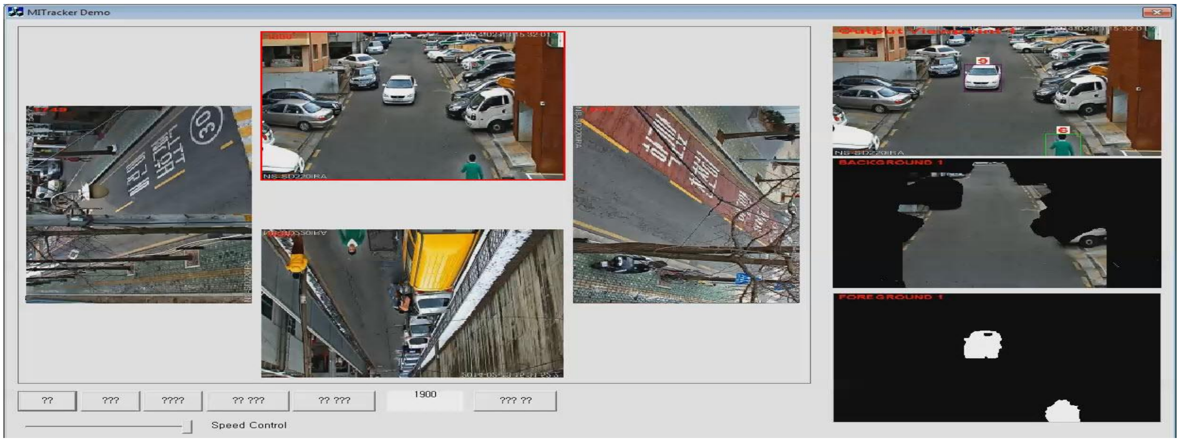
Robust results are shown from Fig. 7(d) to Fig. 7(g). The objects were not only checked in past three viewpoints but also past three rotations, that means we checked each object in 4*3-1 = 11 previous viewpoints.



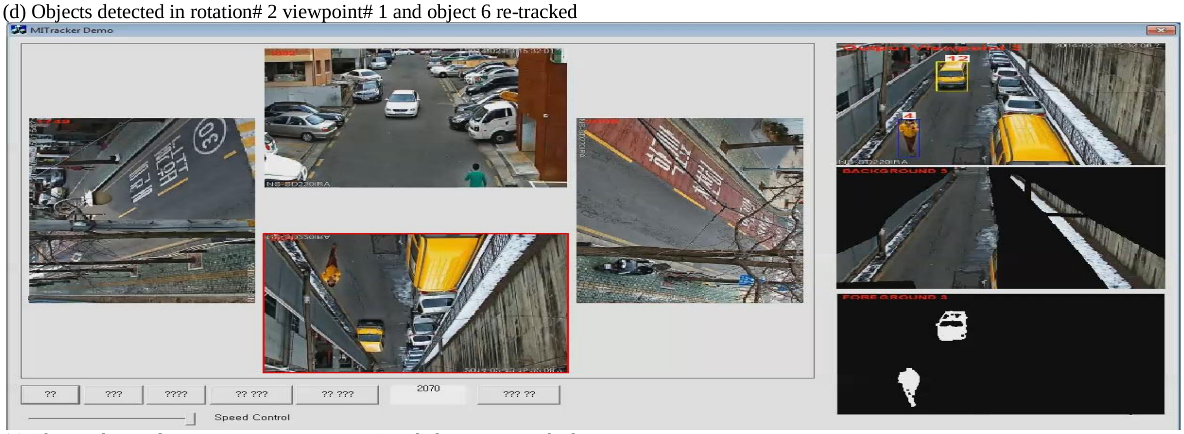
(a) Objects detected in rotation# 1 viewpoint# 1

(b) Objects detected in rotation# 1 viewpoint# 2

(c) Objects detected in rotation# 1 viewpoint# 3 and object 4 re-tracked

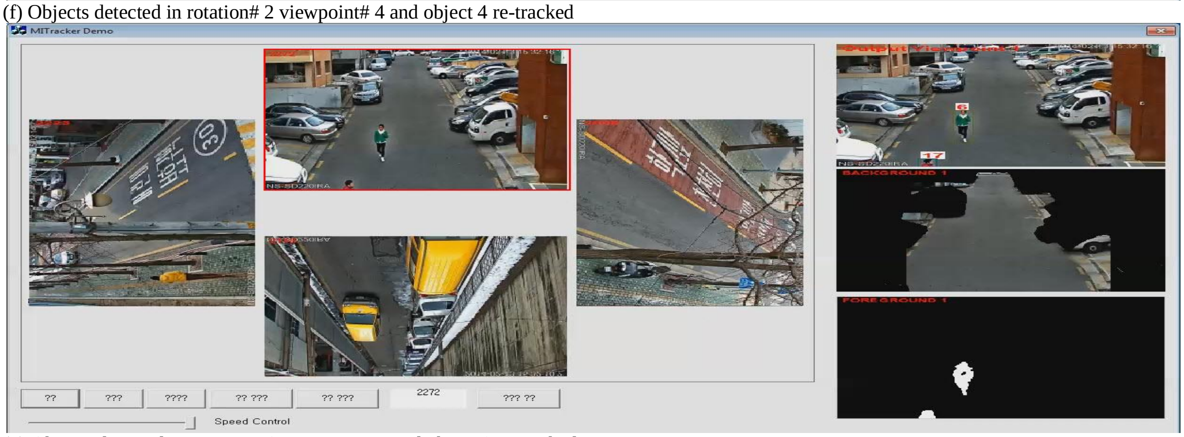


(d) Objects detected in rotation# 2 viewpoint# 1 and object 6 re-tracked



(e) Objects detected in rotation# 2 viewpoint# 3 and object 4 re-tracked

(f) Objects detected in rotation# 2 viewpoint# 4 and object 4 re-tracked



(g) Objects detected in rotation# 3 viewpoint# 1 and object 6 re-tracked

Fig. 7 Experimental results of multi-viewpoint tracking; the object retaining their IDs across viewpoints

V. CONCLUSIONS

Robust object tracking requires invariant attributes matching. Color and size of object play vital role in this aspect. Such a method was proposed and satisfactory results were obtained. Tracking objects across viewpoint with same tracking ID was a challenging task which was accomplished successfully. The proposed method combined the single viewpoint and multi-viewpoint features therefore, merged and separated cases could also be addressed easily across viewpoint.

In future we like to track objects in more complex scheme across the viewpoint using single rotating dome camera based on Kalman prediction and object features. Segmentation of horizontal objects in STGMM would also be considered.

ACKNOWLEDGMENT

This work (Grants No. C0005448) was supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded by Korea Small and Medium Business Administration in 2012.

We would like to thank Mr. Yongsuk Hae for providing dataset as input and graphical user interface (GUI) as output.

REFERENCES

[1] Y. Soh, Y. Hae, I. Kim, Spatio-temporal Gaussian Mixture Model for Background Modeling, IEEE ISM (2012) 360-363.

[2] J.S. Kim, D.H. Yeom and Y.H. Joo, “Fast and robust algorithm of tracking multiple moving objects for intelligent video surveillance systems,” IEEE Trans. Consumer Electronics, Vol. 57, pp. 1165-1170, 2011.

[3] C.T. Chu, J.N. Hwang, H.I.Pai and K.M. Lan, “Tracking Human Under Occlusion Based on Adaptive Multiple Kernels With Projected Gradients,” IEEE Trans. Multimedia, Vol. 15, pp. 1602-1615, 2013.

[4] M. Chitra, M.K. Geetha and L. Menaka, “Occlusion and Abandoned Object Detection for Surveillance Applications,” International Journal of Computer Applications Technology and Research Vol. 2, pp. 708-713, 2013.

[5] D.Xiaoyan, Z. Ya, W. Wei and W. Zhuo, “A robust multiple object tracking algorithm under highly occlusion,” Seventh International Conference on Computer Graphics, Imaging and Visualization, pp. 5-8, 2010.

[6] H. Sheng, Q. Wei, C. Li and Z. Xiong, “Robust multiple-vehicle tracking via adaptive integration of multiple visual features,” EURASIP Journal on Image and Video Processing, 2012.

[7] C. Micheloni, G.L. Foresti and L. Snidaro, “A network of co-operative cameras for visual surveillance,” IEEE Processings on Vision, Image and Signal Processing, Vol. 152, pp. 205-212, 2005.

[8] Wasit L, Andrew W. and M. Greg, “Real-Time People Tracking in a Camera Network,” IEEE Journal on Emerging and Selected Topics in Circuits and Systems, Vol. 3, pp. 263-271, 2013.

[9] Q. Wei, D. Schonfeld and M. Mohamed, “Distributed Bayesian Multiple-Target Tracking in Crowded Environments Using Multiple Collaborative Cameras,” EURASIP Journal on Advances in Signal Processing, 2007.

[10] Cristian C.F., Josep R.C., Montse P. and Enric M., “Multi-camera multi-object voxel-based Monte Carlo 3D tracking strategies,” EURASIP Journal on Advances in Signal Processing, 2011.

[11] I. Kim, Malik M. K., Tayyab W. A. and Y. Soh., “Multi-Target Tracking Using Color Information,” International Journal of Computer and Communication Engineering, Vol. 3, No. 1, 2014.

[12] Malik M. K., Tayyab W. A., I. Kim and Y. Soh, “Tracking Occluded Objects Using Kalman Filter and Color Information,” International Journal of Computer and Communication Engineering, Vol. 6, No. 5, 2014.