Distibuted Human Tracking in Smart Camera Networks by Adaptive Particle Filtering and Data Fusion

Fatemeh Rezaei
Department of Electrical Engineering
Sharif University of Technology
f_rezaei@ee.sharif.edu

Babak H. Khalaj
Department of Electrical Engineering
Sharif University of Technology
khalaj@sharif.edu

Abstract— **Human tracking is an essential step in many computer vision-based applications. As single view tracking may not be sufficiently robust and accurate, tracking based on multiple cameras has been widely considered in recent years. This paper presents a distributed human tracking method in a smart camera network and introduces a particle filter design based on Histogram of Oriented Gradients (HOG) and color histogram. The proposed adaptive motion model also estimates the target speed from the history of its latest displacement and improves the robustness of the tracker by decreasing the probability of missing targets. In addition, a distributed data fusion method is proposed which fuses the information from the cameras by an adaptive weighted average method. Each camera sends its own beliefs of the targets' states and the corresponding weights to other cameras in its communication range. The target fusion weights are determined by each camera, based on the certainty of the corresponding view for each target and an occlusion indicator which depends on the distance between detected targets. The performance of the proposed scheme is evaluated using the PETS2009 S2.L1 dataset. It is shown that the proposed data fusion method leads to more robust tracking among multiple cameras and improves handling of uncertainties and occlusions using multi-view information. In addition, the amount of data transferred in the network is significantly reduced in comparison with centralized methods.**

Keywords- distributed tracking; smart camera network; data fusion; human tracking; particle filter;

I. INTRODUCTION

In recent years, smart camera networks have been widely considered because of their key applications. One of these applications is tracking multiple targets among multiple cameras. Smart cameras combine video sensing, processing, and communication on a single embedded device [1]. A network of such cameras provides a large amount of information that has traditionally been processed in a centralized manner [2]. Optimization of network resources is a key issue in such designs. Distributed processing is one of the approaches used to optimally distribute processing power leading to advantages such as data transfer reduction, bandwidth efficiency, removing the need for large processing power at central nodes and higher security level.

Due to importance of human tracking and a large number of challenges facing it such as human pose variation, illumination changes, lack of specific moving behavior, occlusion and image noises, which affect the tracking robustness and performance, tracking among a camera network is a very active and highly investigated research area. Distributed tracking among a camera network requires a distributed data fusion strategy. One consequence of distributed tracking is that one target can be tracked by multiple sensors, independently. If it is known that multiple beliefs correspond to the same target, the problem of consolidating such beliefs is known as the distributed fusion problem [3].

Among different approaches for tracking such as Optical Flow [4], Cam Shift [5], and Kalman Filtering [6], Bayesian filtering through particle filters is popular in multimodal tracking because of its capability in combining complex (nonlinear, non-Gaussian) and heterogeneous observation models [7]. In addition, sequential Bayesian filtering is recursive and can be implemented in a distributed manner [3]. Because of the aforementioned advantages, this paper considers particle filter tracking in a distributed manner.

In this paper, a distributed human tracking method in a smart camera network is proposed. The contribution of the paper is to introduce a particle filter based on Histogram of Oriented Gradients (HOG) and color histogram which uses an adaptive motion model. The proposed motion model estimates target speed from the history of its latest displacement, therefore, improving the robustness of the tracker and decreasing the probability of missing targets. The proposed particle filter is applied in a distributed camera network. In such networks, camera nodes have different views and consequently different capabilities. Some cameras have zoom-in views and can observe the nearby targets with better separation. Others are in relative zoom-out mode and observe occlusion when targets come close to each other. The method proposed in this paper employs these various capabilities and uses the ability of zoom-out views for new target detection. In order to come up with a low complexity algorithm for data association, it is proposed that among all camera nodes in the network, the one which has zoom-out view and relatively reliable tracking is selected for new targets detection. Consequently, it detects the new targets entering its view and announces them to the other nodes. On the other hand, some of the zoom-in views can handle occlusion of the other views due to their ability in separating nearby targets. In order to increase tracking robustness and occlusion handling, an adaptive weighted average is proposed for data fusion among multiple cameras. In the proposed weighting method, each camera node announces its targets' states and weights to other cameras and receives their corresponding states and weights. The views which have a minimum distance offset among the targets, will detect no occlusion and have greater weights in data fusion and handle the occlusion situations in other camera views. In addition, by weighted averaging among multiple cameras, the uncertainties in single views can be handled.

The paper is organized as follows. In section II, related works are reviewed. In section III, the proposed method is explained. Section IV presents experimental results, followed by conclusions in section V.

II. RELATED WORKS

Single camera tracking is still an open issue in computer vision literature and a number of works such as [8, 9, 10] have been done in this field in recent years.

As single view tracking may not be sufficiently robust and accurate, tracking among multiple cameras has been widely considered in recent years. There are two main categories in multi-camera tracking, centralized and distributed algorithms. A number of references are based on centralized approaches [11, 12, 13, 14]. Among these references, [11, 12] employ a particle filter tracker. The advantage of the centralized approaches is that the entire information of the network is available on a central processor and consequently optimal decisions can be achieved by central information analysis. However, huge amount of data should be transferred due to transmission of camera frames to the central processor.

Distributed approaches constitute another class of algorithms for our target application. Some schemes such as the one proposed in [15] use particle filtering and perform simple local processing. The main disadvantage of this work is simplistic assumptions such as assuming that fixed cameras are aimed roughly horizontally around a room and the locations of the occluders are known. Other works present a distributed handover based algorithm [1, 16, 28]. For example, in [1], the main focus is on a fully decentralized handover procedure between adjacent cameras based on the master/slave paradigm using Cam Shift [5] tracking. In Cam Shift methods, if the tracker fails to track the target in one frame, it cannot follow the tracking in other frames. Also, [16] is built upon a master/slave approach, but the focus lies on Pan-Tilt-Zoom (PTZ) management issues arising in large smart camera systems rather than on computer vision algorithms.

There are some other works which consider the problem of distributed tracking in a more general case closer to our case of interest [2, 17-24]. Among them, [17- 22] are based on Kalman filtering. As we will explain later in this paper, Kalman filtering has lower tracking performance compared to the particle filtering. The advantage of [17] is to provide a message-passing version of the Kalman-Consensus filter that is capable of distributed tracking. A distributed approach, where each camera localizes targets on its own by communicating with its neighbors, is presented in [22]. Despite a number of advantages, this approach faces a number of challenges such as sole reliance on Kalman filter, lack of clear strategy for data fusion and proper numeric metric. A wireless embedded smart camera system for cooperative object tracking via multiple camera views is introduced in [23] which focuses on implementation issues and a power consumption analysis without presenting a specific tracker is introduced. Finally, in [24], an approach where each camera independently estimates local paths in its neighborhood is introduced. The conflicts on locally estimated paths among cameras are resolved by a voting algorithm, and the agreed local paths are finally combined into global paths. The disadvantage is that randomly generated points corresponding to each camera location are used for simulation of a distributed camera network.

Our goal in this paper is to introduce a novel multi-camera approach that combines the advantages of particle filtering with proper fusion algorithms among multiple cameras. The next section presents the details of our method proposed to achieve this aim.

III. PROPOSED METHOD

In this section, a distributed human tracking method in a smart camera network is proposed. First, the single camera tracking is considered and an adaptive particle filter based on HOG and color histogram is presented. The proposed scheme introduces an adaptive motion model which estimates the target speed from the history of its latest displacement. Second, a distributed data fusion method is introduced which fuses information from different cameras by an adaptive weighted average method.

A. Proposed Particle Filter for Single Camera Tracking

The first step in the proposed approach is performing particle filtering for the targets at each camera node. The particle filter contains samples and their corresponding weights which show the probability of each sample to be the centroid of the target and can be denoted as

$$S = \{(X^{(n)}, W^{(n)} \mid n = 1:N\} \qquad (1)$$

where S is the sample set, $X^{(n)}$ shows each sample, $W^{(n)}$ is the corresponding weight and N is the number of samples.

For particle filter modeling, the following three components are considered:

1) State Vector

The State vector of the filter is considered as

$$X_t = \{x_t, y_t, u_t, v_t\} \qquad (2)$$

where *x, y* are the centroid coordination and *u, v* are the estimated speed of each target at time step *t*.

2) Motion Model

In general tracking applications, human beings have no specific structure or equation for their motions. So, it is necessary to have a tracking algorithm that does not require target's equation of motion. In addition, if the tracking algorithm does not adapt itself with the changes in human direction and speed, it fails to track the targets. It is for this reason that the Kalman filter and its extensions such as EKF, which perform properly for objects with well-known motion equations, fail in human tracking applications.

In order to adapt the particle filter for human tracking, we propose the motion model as

$$\begin{aligned}(x,y)_t &= (x,y)_{t-1} + (u,v)_{t-1} * \Delta t + \varepsilon_{x,y} \\ (u,v)_t &= ((x,y)_t - (x,y)_{t-1}) / \Delta t + \varepsilon_{u,v}\end{aligned} \tag{3}$$

where Δt is the inter-frame time interval and ε is Gaussian random noise. The speed equation is the key part which adapts the motion model to human direction and speed by estimating target speed from the history of its latest displacement. For example, consider the situation where a human target enters the image from one side, continues walking to another side but suddenly encounters his friend. He, then stands in his place and starts talking and after some time changes his path to the other side. In such common situation, if the particle tracker does not adapt the estimated speed to the human moving, it fails to track the target. In section IV, it is shown that by applying the proposed motion model, the particle tracker will be significantly more robust to changes in human motions in comparison with traditional Kalman or extended Kalman filters.

3) *Observation Model*

The observation model proposed in this paper is a combination of HOG features and color histogram. Such approach provides a more robust performance against variations and noise in video sequences such as illumination changes, human pose variations and occlusions, because it simultaneously includes color information and human contour information of the target.

The HOG algorithm introduced in [25], is a well-known human detection algorithm. The HOG detector is a sliding window algorithm that for any given image, moves a window across all locations, then scales and computes a descriptor function. A pre-trained classifier is used to assign a matching score to the descriptor in order to decide whether there is a human in the image or not. The classifier is a linear Support-Vector Machine (SVM) classifier and the descriptor is based on the histograms of gradient orientations.

The color histogram is computed from RGB color space as follows

$$p(y) = \{p^{(u)}(y) \mid u = 1:m\},$$

$$p^{(u)}(y) = f \sum_{i=1}^{n} k(\|y - y_i\| / a)\delta(h(y_i) - u) \tag{4}$$

where y is the centre of the object, y_i denotes pixel locations of target centered at y, m is the number of histogram columns and K(x) is the kernel function, a is a scale factor, δ(x) is the Dirac function, n is the count of total pixels, and f is the normalization factor.

The proposed observation model is given by:

$$Wn = \alpha\left(\frac{1}{\sqrt{2\pi}\ \sigma} e^{-\frac{d^2}{2\sigma^2}}\right) + (1-\alpha)(W_{HOG}) \tag{5}$$

where

$$d^2 = (1 - \rho(p(X_n), q) \tag{6}$$

and ρ denotes the Bhattacharyya coefficient [26] for the template model comparing with the observations. W_{HOG} represents the observation weight obtained by HOG descriptor, and α is a coefficient controlling the effect of each observation on the total weight.

B. *Proposed Distributed Data Fusion Method*

In this section, a distributed data fusion method is introduced which combines information from different cameras by an adaptive weighted average algorithm.

Distributed tracking is achieved by cooperating all camera nodes in the network. We consider a network of partially overlapping cameras which observe an outdoor 3D scene. The distributed architecture proposed in [7] is considered which is based on three main constraints: (1) a single central fusion node is not present; (2) broadcast communication is not allowed, but only node-to-node communication is permitted; (3) the sensors have no global knowledge of the network, one node sees only its own neighborhood. These constraints make the system scalable and survivable to loss or addition of nodes.

In general, the targets can move in the world coordinate system $\{x_w, y_w, z_w\}$. As the considered targets are human, we know that they move at ground level. With such constraint, the image coordinates $\{u_i, v_i\}$ can be mapped to each other through the world coordinate system. As the proposed algorithm is a distributed one, each camera tracks the targets in its own view. By using the data fusion method, camera nodes are enabled to use the other camera track results to solve uncertainties and occlusions. Therefore, it is required to project one camera view to another, similar to what is for example done in [27] where homography information is used to map one view onto another. In order to estimate the homography, the required points are manually selected at the ground plane. Due to uncertainties of the homography methods, the calibration information provided by the data set is used in our method to prevent occurrence of such uncertainties. Thus, each camera node sends target states in its own coordinates to the other ones. The other camera nodes map the received coordinates to their own, through the world system coordinate provided by the calibration information as required.

An important issue in multi target-multi camera tracking is data association. In order to come up with a low complexity algorithm, we present the following approach. Among all camera nodes in the network, the one which has zoom-out view and relatively reliable tracking results is selected for new targets detection. The selected camera should detect the new targets entering its view and announce it to the other nodes. For new target detection, the border of the selected view should be controlled and the HOG human detection algorithm should be applied over that region. False positives of the detection algorithm are handled as follows: when the HOG algorithm detects a new target, it is tested for a specific frame number such as 3 frames. If in all these frames, the new target detection is reported, it is considered to be a new target and a new tracker is started. In addition, the new target detection announcement and its detected state are sent to other camera nodes. The other nodes then project the received state to their own coordinates and consequently, start tracking it.

In order to increase tracking robustness and occlusion handling, an adaptive weighted average is proposed for data fusion. In our framework, each camera sends its own beliefs of the target states and the corresponding weights to the other cameras in its communication range. Therefore, the camera nodes have two group of information. First, their sensor information derived from their own view, and second, the information from other camera views. Subsequently, each camera should combine all information in a single decision about the targets' tracks and states, which is the function of the fusion center in each camera node. The proposed fusion algorithm is an adaptive weighted average running on each camera node independently. The states fusion weights are determined by each camera, based on the certainty of its own view for each target and an occlusion indicator which depends on the distance between detected targets. Then, each camera node announces its target weights to other cameras and receives their weights. A weight is assigned to each camera-target pair corresponding to its accuracy and robustness. Each camera which has more robust tracking results will have a greater weight. If no additional information is available, all weights should be the same and equal to one. The fusion center of each camera can be modeled as

$$p_n(X_t / Z_t C) = \sum_{\substack{i=1 \\ i \neq n}}^{CN} CW_i \cdot p(X_t / Z_t C_i) \tag{7}$$

where $p_n(X_t/Z_tC)$ denotes fused beliefs of all camera sensors C for each target at the camera node n. CN denotes the number of the camera nodes in the communication range of the camera node n, $p(X_t/Z_tC_i)$ is the belief of each target at the camera-target tracker i, and CW_i is the corresponding weight of each camera-target pair which is sent from camera i to the camera tracker n. The weights are assigned such that

$$CW_i = \begin{cases} 0 & \textit{If the camera node i fails to track the target.} \\ \beta & \textit{If the camera node i detects occlusion.} \\ 1 & \textit{O. W.} \end{cases}$$

where β is a coefficient between 0 and 1. Some cameras have zoom-in views and can observe the nearby targets with a minimum amount of separation. Others are relatively in zoom-out mode and observe occlusion when targets come close to each other. As mentioned before, we use the ability of zoom-out views for new target detection. On the other hand, some of the zoom-in views can handle occlusion of the other views due to the ability of observing close targets separately. By employing the proposed weighting method, the views which have a minimum distance offset among the targets, will detect no occlusion and have greater weights in data fusion and handle the occlusion situations in other camera views.

Two situations are considered for multi camera data fusion. 1) If the current camera tracker fails to track a target, the weighted averages of the other *CN-1* cameras are considered as its observation at the current frame. If other camera nodes fail to track the target in the corresponding frame, its weight will be zero. 2) In applications where the single camera trackers are not sufficiently reliable or robust, the weighted average fusion can be done periodically after a specific number of frames, such as 10 frames. It should be mentioned that such process depends on the capacity of data transfer in the network and tolerable delays in each application. This method helps the whole network trackers be more robust and prevents them from missed tracking situations.

As shown in section IV, the proposed data fusion method leads to more robust tracking among multiple cameras and handles uncertainties and occlusions using multi-view information. In addition, the amount of data transferred in the network significantly decreases in comparison with centralized methods.

IV. Experimental Results

We have tested the performance of the proposed method on the publicly available data sets PETS2009 S2.L1. The Visual C++ 2008 and Open CV library is used for implementation. PETS2009 S2.L1 provides eight camera views, two of them are not useable due to lack of specific frame rate. Based on our simulations, two of these camera views are more sensitive to the small calibration misalignments and their results are affected by it. Thus, we have used the remaining four views to evaluate our proposed method. It should be noted that the used data set is a challenging set for human tracking, as there are many people who do not have significantly different color cloths and lots of occlusion occurs among people and objects. In [8], it is illustrated that the improvement of their approach is less critical on the PETS'09 pedestrian dataset than on the sport sequences. The reason is that people in this sequence are wearing dark clothes of a relatively uniform color.

The results of our multi view-multi target tracking are shown in Figure 1. For numeric analysis, we employ the metric used in [28] as follows. If the overlap of our bounding box and the ground truth bounding box is larger than 70% and the size of our bounding box is less than 1.5 times of the ground truth bounding box, we consider it to be a correct tracking case. The correct tracking percentage is the number of correctly tracked frames divided by the number of all frames in the video sequences. Figure 2 compares the single camera with proposed multi camera tracking results. As it is shown, the proposed distributed method increases tracking robustness by increasing the correct tracking percentages. Also, we have compared the performance of our method with two other particle filter trackers; one using only HOG and the other using only color histogram [11, 12]. As Figure 3 indicates, the proposed combination of HOG and color histogram improves the robustness of tracking.

In order to compare our results with other distributed schemes, which are based on Kalman filtering [17-22], we have simulated the Kalman filtering on PETS2009 S2.L1. However, the Kalman filter tracking failed to track the targets in the first few frames in this data set, due to the reasons mentioned earlier.

For performance comparison of our method with other schemes, in terms of amount of data transfer, we consider both centralized and distributed scenarios. As can be verified from Figure 4, the amount of data transferred throughout the network in our proposed distributed tracking method in comparison with the centralized methods such as [11-14] decreases significantly. The data transfer in a centralized method is considered as the size of each camera frame with 720 * 576 resolution, and the data transfer in our distributed method is considered as the average number of target per frames multiplied by the size of the files required to store belief vectors, which is equal to 5 * 127 = 605 Bytes per frame.

V. Conclusion

In this paper, a distributed human tracking method in a smart camera network was proposed. We introduced a particle filter based on HOG and color histogram with an adaptive motion model. The proposed motion model estimated target speed from the history of its last displacements. In addition, a distributed data fusion method was introduced which fuses information from the cameras by an adaptive weighted average method. The target fusion weights were determined by each camera, based on the certainty of the corresponding view for each target and an occlusion indicator which is a function of the distance between detected targets. It has been shown that the proposed data fusion method improves the robustness of tracking among multiple cameras and handles uncertainties and occlusions in comparison with single camera tracking. In addition, compared with the centralized methods, the amount of the data transferred in the network is significantly reduced.

Acknowledgement

The authors would like to thank National Elite Foundation for their support.

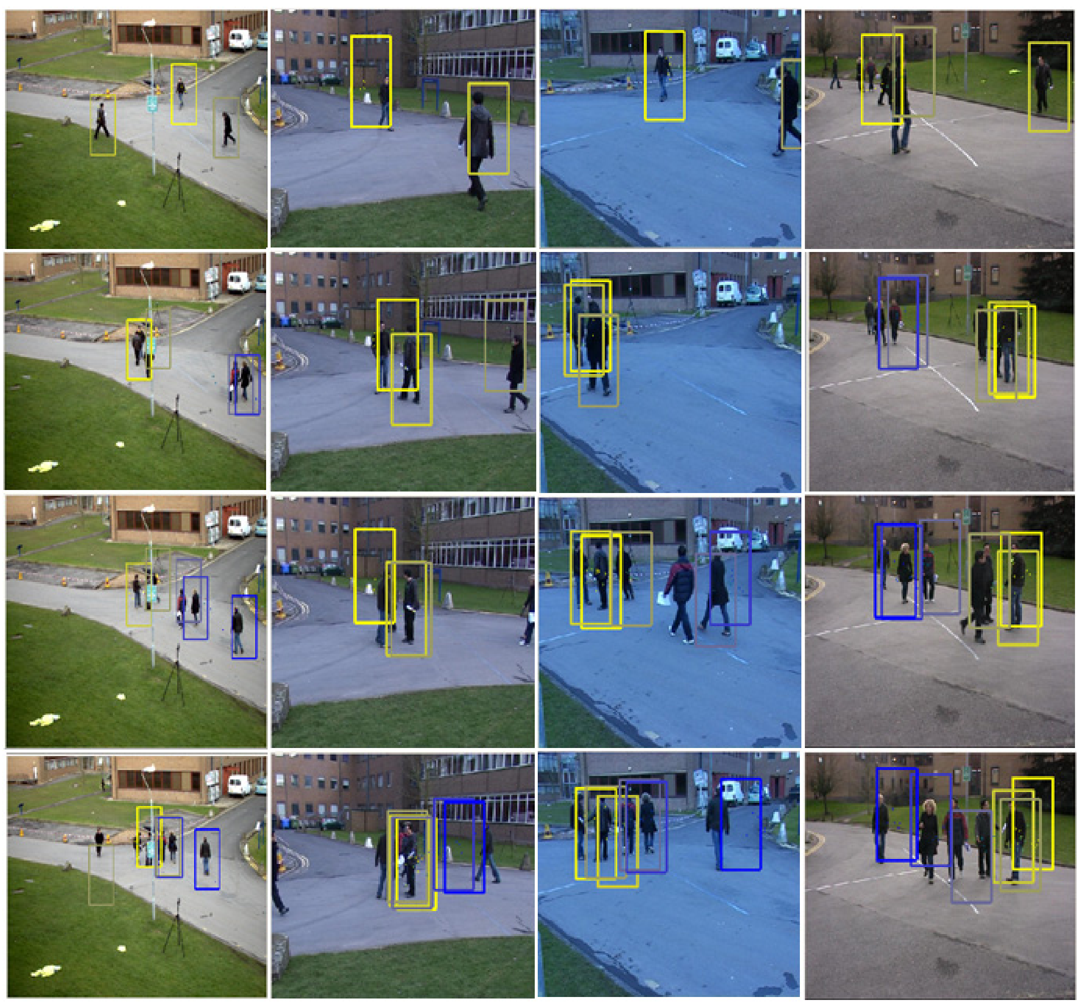


Figure 1. Results of multi view tracking by the proposed method. The columns from right present camera 1,camera 5, camera 6 and camera 8 results, respectively. The rows from top present frame 03, frame 26, frame 52 and frame 81, respectively.

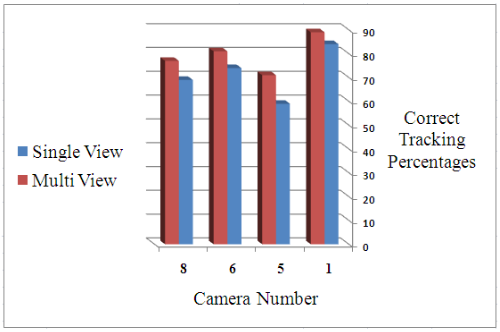


Figure 2. Performance comparison of the single camera with the proposed multi camera tracking results.

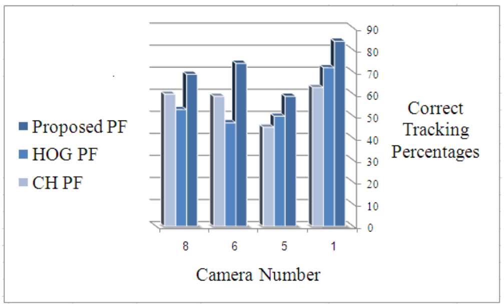
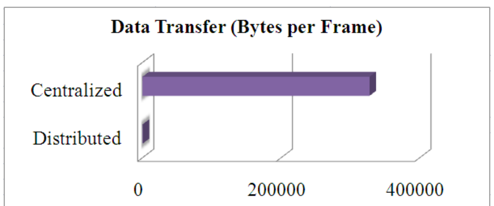


Figure 3. Tracking performance comparison of the proposed particle filter with 2 other particle filters in single view tracking.



	Distributed	Centralized
Series1	605	327680

Figure 4. Data transfer in distributed and centralized networks.

REFERENCES

[1] M. Quaritsch, M. Kreuzthaler, B. Rinner, H. Bischof, and B. Strobl, "Autonomous Multi camera Tracking on Embedde Smart Cameras", Hindawi Publishing Corporation, EURASIP Journal on Embedded Systems, Volume 2007, Article ID 92827.

[2] E. Lobaton, R. Vasudevan, R. Bajcsy, and S. Sastry, "A Distributed Topological Camera Network Representation for Tracking Applications", IEEE Transaction on Image Processing, Vol. 19, No. 10, October 2010.

[3] J. Liu, M. Chu, and J. E. Reich, "Multitarget Tracking in Distributed Sensor Networks,Implementing MTT with resource limitations", IEEE Signal Processing Magazine, May 2007.

[4] B. K. P. Horn and B. G. Schunck, "Determining optical flow" Artifi cial Intelligence, No. 17, Pages 185–203, 1981.

[5] G.R. Bradski, "Computer video face tracking for use in a perceptual user interface", Intel Technology Journal, 1998.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems", Journal of Basic Engineering, No. 82, Pages 35–45, 1960.

[7] A. Dore, M. Pinasco, C. S. Regazzoni, "Multi-Modal Data Fusion Techniques and Applications", Chapter 9, Multi-Camera Networks: Concepts and Applications, Elsevier, Edited by H. Aghajan, and A. Cavallaro, 2009.

[8] H. Ben Shitrit, J. Berclaz, F. Fleuret, P. Fua,"Tracking Multiple People under Global Appearance Constraints",IEEE International Conference on Computer Vision, ICCV 2011.

[9] I. Zuriarrainy, J. I. Aizpuruay, F. Leraslez, and N. Arana, "Multiple-Person Tracking devoted to distributed multi Smart Camera Networks", IEEE/RSJ International Conference on Intelligent Robots and Systems October 2010.

[10] P. Darvish Z. V, G. A. Bilodeau, "People Tracking Using a Network-based PTZ Camera", Machine vision and applications manuscript, No. 4, pages 671-690, July 2011.

[11] O. Lanz, P. Chippendale, and R. Brunelli, "An Appearance-based Particle Filter for Visual Tracking in Smart Rooms", Multimodal Technologies for Perception of Humans, Pages 57 – 69, Springer, 2008.

[12] M. Taj and A. Cavallaro, "Multi-camera track-before-detect", 3rd ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2009.

[13] X. Zhu, J. Liu, J. Wang, W. Fu, H. Lue, Y. Fang, "Global Trajectory Construction across Multi-Cameras via Graph Matching", 2011 Sixth International Conference on Image and Graphics.

[14] J. Yao, J. M. Odobez, "Multi-Person Bayesian Tracking with Multiple Cameras", Chapter 15, Multi-Camera Networks: Concepts and Applications, Elsevier, Edited by H. Aghajan, and A. Cavallaro, 2009.

[15] A. O. Ercan, A. E. Gamal, L. J. Guibas, "Object Tracking in the Presence of Occlusions via a Camera Network", 6th International Symposium on Information Processing in Sensor Networks, Pages 509–518, IPSN 2007.

[16] M. Hoffmann, M. Wittke, Y. Bernard, R. Soleymani and J. H¨ahner, "DMCTRAC: Distributed Multi Camera Tracking", 2nd ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2008.

[17] R. Olfati-Saber and N. F. Sandell, "Distributed Tracking in Sensor Networks with Limited Sensing Range", American Control Conference, ACC 2008.

[18] B. Song, A. T. Kamal, C. Soto, C. Ding, J. A. Farrell, and A. K. Roy-Chowdhury, "Tracking and Activity Recognition Through Consensus in Distributed Camera Networks", IEEE Transaction on Image Processing, Vol. 19, No. 10, 2010.

[19] A. A. Morye, C. Ding, B. Song, A. Roy-Chowdhury, J. A. Farrell, "Optimized Imaging and Target Tracking within a Distributed Camera Network", American Control Conference, ACC 2011.

[20] H. Medeiros, J. Park, " Cluster-Based Object Tracking by Wireless Camera Networks", Chapter 23, Multi-Camera Networks: Concepts and Applications, Elsevier, Edited by H. Aghajan, and A. Cavallaro, 2009.

[21] J. M. Sánchez-Matamoros, J.R. Martínez-de Dios and A. Ollero, "Cooperative localization and tracking with a camerabased WSN", IEEE International Conference on Mechatronics, 2009.

[22] B. Shirmohammadi, C. J. Taylor, "Distributed Target Tracking using Self Localizing Smart Camera Networks", 4th ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2010.

[23] Y. Wang, S. Velipasalar, and M. Casares, "Cooperative Object Tracking and Composite Event Detection With Wireless Embedded Smart Cameras", IEEE Transaction on Image Processing, Vol. 19, No. 10, 2010.

[24] H. Kim and M. Wolf, "Distributed Tracking in a Large-Scale Network of Smart Cameras", 4th ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2010.

[25] N. Dalal and B. Triggs, "Histogram of Oriented Gradients for Human Detection", IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2005.

[26] A. Bhattacharyya, "On a measure of divergence between two statistical populations defi ned by probability distributions," Bulletin of the Calcutta Mathematical Society, No. 35, Pages 99–109, 1943.

[27] P. M. Roth, C. Leistner, H. Grabner, H. Bischof," Online Learning of Person Detectors by Co-Training from Multiple Cameras", Chapter 9, Multi-Camera Networks: Concepts and Applications, Elsevier, Edited by H. Aghajan, and A. Cavallaro, 2009.

[28] Y. Li, B. Bhanu, "Fusion of Multiple Trackers in Video Networks", 5th ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2011.