MCMC-PF Based Multiple Head Tracking in a Room Environment

Ata-ur-Rehman[1]
a.ur-rehman@lboro.ac.uk
Syed Mohsen Naqvi[1]
s.m.r.naqvi@lboro.ac.uk
Raphael Phan[1]
r.phan@lboro.ac.uk
Wenwu Wang[2]
w.wang@surrey.ac.uk
Jonathon A. Chambers[1]
j.a.chambers@lboro.ac.uk

[1] Advanced Signal Processing Group, School of Electronic, Electrical and Systems Engineering, Loughborough University, Leicestershire UK.

[2] Center for Vision Speech and Signal Processing, Department of Electronic Engineering Faculty of Engineering and Physical Sciences, University of Surrey, Guildford UK.

Abstract

An improved multiple target tracking algorithm is proposed for tracking the heads of people in a room environment. The proposed algorithm focuses on mitigating the inter-target occlusion problem during complex interactions. This is achieved with the help of a particle filter, multiple video cues and a new interaction model. A Markov chain Monte Carlo particle filter (MCMC-PF) is used to track multiple targets while a colour and gradient histograms based framework is used for likelihood modeling. A new interaction model is also proposed to model the interactions of multiple targets, which is incorporated into the MCMC-PF to protect the tracker from failure when targets occlude each other. Performance of the proposed tracker is evaluated with natural video sequences including the AV16.3 corpus. Experimental results show that the proposed technique efficiently tracks the heads of multiple people and the tracker does not fail when such targets approach or cross each other, where the state of the art Markov random field (MRF) approach fails.

1 Introduction

Target tracking has a number of applications such as human computer interface, security, surveillance and video conferencing [16, 20]. Human tracking is generally a nonlinear and non-Gaussian tracking problem and Kalman and even Extended Kalman filters are commonly found to be unsuitable whereas particle filters [10] are well suited to the task. The general tracking problem has been intensively studied in recent years and different methods have been proposed in the literature to solve it. This includes "detect before track" technique [6] but we have used "track before detect" technique in this paper which is computationally less expensive. Most of the proposed methods are concerned with tracking a single target [5, 12, 22]. Tracking multiple targets and modeling the interactions between them is a complex tracking problem. Particle filter (PF) based solutions are presented in [8, 13] which include MRF and reversible jump Markov chain Monte Carlo (RJMCMC) for tracking multiple targets. However the main focus in [13] is on handling a variable number of targets and close interactions but not the occlusions. Their work is based on the idea of penalizing the particles with the help of a penalty function. This solution works well for tracking sequences where two or more targets do not occupy the same space, but it does not address the tracking failures caused by inter-target occlusions during target crossovers. In [19] a relatively complicated technique is proposed for tracking a large number of targets which move in a group, which is not a suitable technique for tracking a small number of targets in a room environment because of its high computational cost.

The main objective of this paper is to present a solution for handling occlusions while tracking multiple heads of people in a room environment. The importance sampling involved in generic particle filters [1] makes them inefficient for multiple target tracking [7]. The MCMC-PF [21] performs more efficiently in the multiple target tracking scenario. In our proposed tracking algorithm the MCMC-PF is used to simultaneously track heads of multiple people. Most tracking algorithms normally depend on a single video cue e.g. in [18] only a colour cue is used. A problem occurs however when there is an object around the target with a similar colour which may cause a tracking failure. Therefore to overcome this tracking failure two video cues are used i.e. colour and gradient. The advantage of colour cues is that they are object independent [5], while gradient information helps to differentiate between multiple targets especially when we are tracking multiple heads. Most of the proposed particle filter algorithms in the literature fail when two or more targets approach each other or cross. To overcome this problem a novel interaction model is proposed which protects the tracker from failure when targets cross each other. One of the main advantages of the proposed interaction model is that it is simple and computationally efficient.

Changes in the lighting conditions can affect the tracking results as discussed in detail by [17] and [11]. However the main focus of this research paper is on addressing the occlusion problems with the help of MCMC-PF, multiple video cues and interaction model under constant lighting conditions.

The paper is organized as follows. Section 2 explains the problem formulation. The sequential MCMC-PF and the proposed tracker are discussed in Section 3. The proposed interaction model is described in Section 4. Experimental results are shown in Section 50 and finally conclusions are drawn in Section 6.

2 Problem Formulation

We assume that it is required to track M targets. The state of each target n at discrete time k is represented as $\mathbf{x}_k^n = [x_k^n, \dot{x}_k^n, y_k^n, \dot{y}_k^n]$, where x_k^n and y_k^n are respectively the x and y coordinates of the state, while $\dot{x}_k^i$ and $\dot{y}_k^i$ are the respective velocities. The combined state of all the M targets is represented as $\mathbf{X}_k = [\mathbf{x}_k^1, \mathbf{x}_k^2, \ldots, \mathbf{x}_k^M]$. Similarly combined measurements of all the targets are represented as $\mathbf{Y}_k = [\mathbf{y}_k^1, \mathbf{y}_k^2, \ldots, \mathbf{y}_k^M]$.

In the case of Bayesian tracking the main objective is to calculate the posterior probability distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ of the state $\mathbf{x}_k$ at discrete time index k, given the observations $\mathbf{y}_{1:k}$ from time 1 up to k. A generic particle filter [1] makes use of the importance sampling technique in which independent weighted particles are taken from a known proposal distribution to build the required posterior distribution. Such importance sampling has limitations in high dimensions therefore we use an MCMC-PF to obtain better performance in a multiple target tracking scenario [15]. Multiple targets can be tracked with a single MCMC-PF while multiple generic particle filters are required to track multiple targets. In the MCMC-PF technique unweighted particles are taken from a known proposal distribution and each particle depends on the previously predicted particle. Monte Carlo estimation of the posterior distribution $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ can be represented as

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \frac{1}{N_s} p(\mathbf{y}_k|\mathbf{x}_k) \sum_{i=1}^{N_s} p(\mathbf{x}_k|\mathbf{x}_{k-1}^i) \tag{1}$$

where $p(\mathbf{y}_k|\mathbf{x}_k)$ is the likelihood which expresses the measurement model while $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is the prior which expresses the state model and N_s is the number of particles. The Metropolis-

Algorithm 1 MCMC-Based Particle Filter Algorithm

Input: 2-D positions of the center of the heads and reference patch for each head in the initial frame
Output: 2-D position of the heads in each frame

Initialize N_s particles for M number of heads $\{\mathbf{X}_k^i\}_{i=1}^{N_s}$
for $k = 2, \ldots, T$ **do**
    Randomly select a particle u from the posterior distribution of the state $\mathbf{X}_{k-1}$ and use this particle and the motion model $q(\cdot)$ to predict the initial state of all the targets at time step k
    $\mathbf{X}_k^1 \sim q(\mathbf{X}_k|\mathbf{X}_{k-1}^u)$
    for $i = 2, \ldots, N_s + B$ (where B is the number of burn in particles) **do**
        Randomly select another particle $\mathbf{X}_{k-1}^{'}$ from the posterior distribution at time $k-1$ $p(\mathbf{X}_{k-1}|\mathbf{Y}_{k-1})$
        Propose a new particle using the proposal distribution $Q(\cdot)$ and the randomly selected particle $\mathbf{X}_{k-1}^{'}$
        $\mathbf{X}_k^{'} \sim Q(\mathbf{X}_k^i|\mathbf{X}_{k-1}^{'})$
        Compute the measurement likelihoods $p(\mathbf{Y}_k|\mathbf{X}_k^{'})$ and $p(\mathbf{Y}_k|\mathbf{X}_k^{i-1})$ with respect to the proposed particle $\mathbf{X}_k^{'}$ and the previous particle $\mathbf{X}_k^{i-1}$ respectively
        Compute the acceptance ratio
        $\alpha = min\left(1, \frac{p(\mathbf{Y}_k|\mathbf{X}_k^{'})}{p(\mathbf{Y}_k|\mathbf{X}_k^{i-1})}\right)$
        Draw a point j from a uniform distribution
        if $j < \alpha$ **then**
            $\mathbf{X}_k^i = \mathbf{X}_k^{'}$
        else
            $\mathbf{X}_k^i = \mathbf{X}_k^{i-1}$
        end if
    end for
    Discard the first B particles and keep the remaining of N_s particles.
end for

Hastings (MH) algorithm [9] is the most basic MCMC algorithm and all the other algorithms including the Gibbs sampling algorithm are special cases of the MH algorithm. The basic MH algorithm is used in the proposed tracker to track the heads of multiple people.

3 Sequential MCMC Filtering

MCMC-PF filtering is a two step process, in the first step we predict a particle to estimate the posterior distribution of the next state and the second step is a refinement step in which we decide whether to accept or reject the predicted particle. The prediction step involves a state transition model and a suitable proposal distribution, while the refinement step requires a likelihood model. The MCMC-PF used in our work is summarized in Algorithm 1.

3.1 State Models

To estimate the translation motion of the moving targets, a constant velocity model [2] is used. The same model is used as a proposal distribution. A rectangular region (patch) which contains the head is manually selected in the initial frame. The pixel in the center of the patch is considered as a center of the head. Horizontal and vertical locations of this pixel are tracked in each frame. A two dimensional motion of a moving speaker can be described by the constant velocity model [5]

$$\mathbf{x}_{k+1}^{n} = \mathbf{A}\mathbf{x}_{k}^{n} + \mathbf{u}_{k} \tag{2}$$

where $\mathbf{u}_k$ is the measurement noise and the matrix $\mathbf{A}$ is defined as

$$\mathbf{A} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and T is the frame sampling interval.

3.2 Likelihood Models

In MCMC-PF it is very important to have a strong likelihood model. Predicted particles are accepted or rejected on the basis of acceptance ratio α. The likelihood model used in our work is based on the combination of colour and gradient histograms.

Colour histograms are widely used in the literature [5, 8, 22] to exploit the uniqueness of the skin colour to track the heads. Scaled versions of red (R), green (G) and blue (B) colours are used in our work. R-G and G-R are used to represent the chrominance information while R+G+B is used to represent the luminance information [4].

Reference histograms H_{ref} are created for all the target heads with the help of the patches selected in the initial frame. For the predicted particles, target histograms H_{target} are created by selecting a patch with the predicted state as its center. The Bhattacharyya coefficient ρ between the reference and the target colour histograms is calculated by their binwise multiplication

$$\rho(H_{ref}, H_{target}) = \sum_{j=1}^{E} \sqrt{H_{ref}^{j} \times H_{target}^{j}} \tag{3}$$

where E represents the number of histograms bins. Bhattacharyya distance [3] between two histograms is defined as

$$d(H_{ref}, H_{target}) = \sqrt{1 - \rho(H_{ref}, H_{target})} \tag{4}$$

The likelihood with respect to the colour cues, as in [5] is calculated as

$$L_c(\mathbf{y}_k|\mathbf{x}_k) \propto exp\left(-\frac{d(H_{ref}^{c}, H_{target}^{c})}{2\sigma^2}\right) \tag{5}$$

where σ^2 is the measurement noise variance.

Using only the colour histograms is insufficient for the tracking purposes because the colour based tracker fails when there is something else with a similar colour around the target. Integration of the gradient histograms helps to overcome such problems. Gradient histograms are created for reference and target patches for the purpose of edge detection. The likelihood with respect to these histograms is calculated by using the Bhattacharyya distance with the help of the following equation

$$L_g(\mathbf{y}_k|\mathbf{x}_k) \propto exp\left(-\frac{d(H^g_{ref}, H^g_{target})}{2\sigma^2}\right) \tag{6}$$

where the overall likelihood is then calculated as

$$p(\mathbf{y}_k|\mathbf{x}_k) = \nu L_c(\mathbf{y}_k|\mathbf{x}_k) + (1-\nu)L_g(\mathbf{y}_k|\mathbf{x}_k) \tag{7}$$

and ν is the weighting coefficient, which is used to weight the two video cues.

4 Proposed Interaction Model

In the case of multiple target tracking the simple MCMC-PF fails when a target is occluded by another. This is because the filter is unable to locate the target. When the target comes out of the occlusion the filter can not generally recover because the particle filter is unable to predict the particle on the target which was occluded and therefore the tracker starts tracking the target which is at the front because of the similarity between the face colours. An intelligent particle filter is required to predict particles in the desired region i.e. on either side of the visible target.

Therefore an improved interaction model is proposed here which overcomes tracking failures. The proposed interaction model includes automatic occlusion detection and reinitialization of target positions. In the automatic reinitialization step the algorithm calculates the probability of occlusion given position of targets i.e. $p(\theta|x)$. Probability of occlusion is based on the proximity of the targets. Pairwise probability of occlusion is calculated for every target with respect to its closest neighbor. It is assumed that only two of the targets can undergo occlusion at one time step. Hence the algorithm checks the maximum probability if it is higher than a predefined threshold level ω_1 then it means two of the targets have been occluded to a level which may cause a failure of the tracker. A Gaussian model is used to define the pairwise occlusion probability. The probability that the target x^n encounters occlusion at time k is defined as

$$p(\theta|x) = \mathcal{N}(x^n_k, \sigma^2_s) \tag{8}$$

The estimated position of the target x^n_k is taken as a mean and σ^2_s is the Gaussian measurement noise. This model results in a higher probability of occlusion when targets approach each other.

Occurrence of an occlusion automatically switches the tracker from the normal mode to the interaction mode. The goal of the interaction model is to search for the new location of the occluded target and to reinitialize the tracker with the new searched location when the occluded target comes out of occlusion. This search is conducted on either side of the visible target. The overall probability of the measurements taken from different sides of the visible target are modeled as a mixture of Gaussian, which is defined as

$$p(y|\mu,\Sigma) \approx \sum_{r=1}^{R} w_r \mathcal{N}(y|\mu_r, \sigma_r) \tag{9}$$

where elements r of the mixture model are actually sets of measurements taken on different sides of the visible target and w_r is the mixing coefficient. Total size R of elements of the mixture model is predefined depending on the size of the search. The actual goal of the model is to search for the location of the occluded target which corresponds to estimating the mean μ of mixture elements, this is achieved by maximizing the data log likelihood.

$$logp(y|\mu,\Sigma) \approx \sum_{p=1}^{P} log \sum_{r=1}^{R} w_r \mathcal{N}(y_p|\mu_r,\sigma_r) \tag{10}$$

If we consider a case where one of the targets is occluded by another target then the occluded target can appear only at one side of the visible target. With this assumption, all but one of the mixing coefficient will be equal to zero which results in the following simplified form.

$$logp(y|\mu,\Sigma) \approx \sum_{p=1}^{P} log\mathcal{N}(y_p|\mu_r,\sigma_r) \tag{11}$$

Number of measurements P is defined as a fixed sized square patch centered at the estimated location of the target. The same estimated location is used as mean μ_r of the respective mixture component with a predefined size σ_r. These patches are selected on either side of the visible target. Equation (11) holds only when the head of the occluded target comes out of occlusion and is visible on one of the sides of the visible target. To ensure this condition holds, the Bhattacharyya distance is calculated between the colour histograms of selected patches in the current frame and the reference patches selected in the first frame. If the distance drops below a threshold value ω_2 it means the tracker has found the head in one of these patches. The element of the Gaussian mixture which corresponds to the maximum likelihood is also based on the same Bhattacharyya distance. The mixing coefficient is 1 for the patch with the minimum Bhattacharyya distance and 0 for all other patches. The MCMC-PF is re-initialized with these new locations and it starts working again.

This technique helps to model different possible actions of targets. A few of them are shown in the next section i.e. when 1) targets cross over and follow their initial direction of motion 2) they go back after occlusion and follow the opposite direction 3) they cross and then change their direction. Another advantage of this technique is that it is simple to implement and computational efficient because we are searching for heads in a few small patches instead of searching for them in a whole frame.

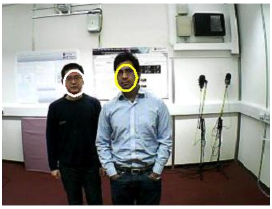
5 Experimental Results

The algorithm is compared with the state of the art MRF interaction model [13]. To model the interactions between targets, the algorithm introduces a potential function $\Psi(\mathbf{x}^n,\mathbf{x}^m)$ to define the interactions between targets n and m which is defined as

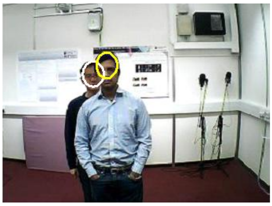
$$\Psi(\mathbf{x}^n,\mathbf{x}^m) \propto exp(-g(\mathbf{x}^n,\mathbf{x}^m)) \tag{12}$$

where $g(\mathbf{x}^n,\mathbf{x}^m)$ is a penalty function. To compare the proposed algorithm with [13], the penalty function depends on the distance between two targets.

The proposed algorithm is specifically for indoor tracking problems, hence both algorithms are tested to track the heads of two people in a room with the help of a single camera. All the parameters have been chosen empirically to yield best results. The total number of particles used is 400 with a burn in period of 100. From the experimental results it is observed that in most of the cases the colour cues perform better than the gradient cues, so we



(a) Frame 1

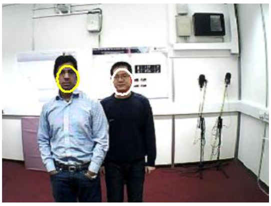


(b) Frame 60

(c) Frame 90

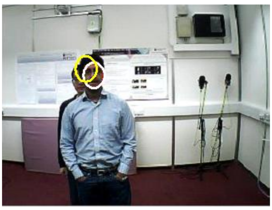
(d) Frame 180

Figure 1: Tracking the heads of moving targets with the MCMC-PF, multiple colour cues and a MRF interaction model [13]. The tracker performs well when targets are apart (a) and (b), and even when one of them is occluded (c) but the tracker fails when one of the targets goes back after occlusion (d).



(a) Frame 1

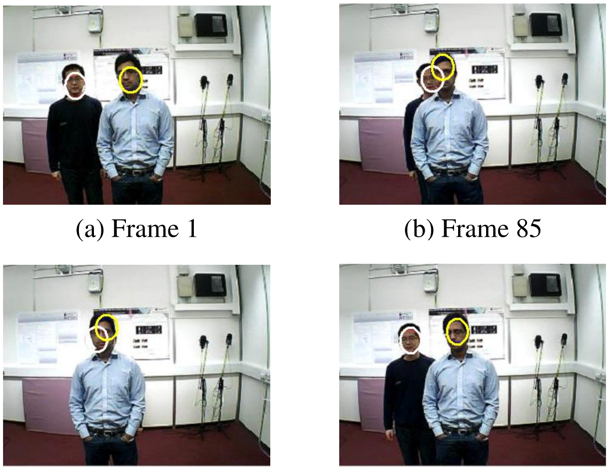
(b) Frame 70



(c) Frame 90

(d) Frame 120

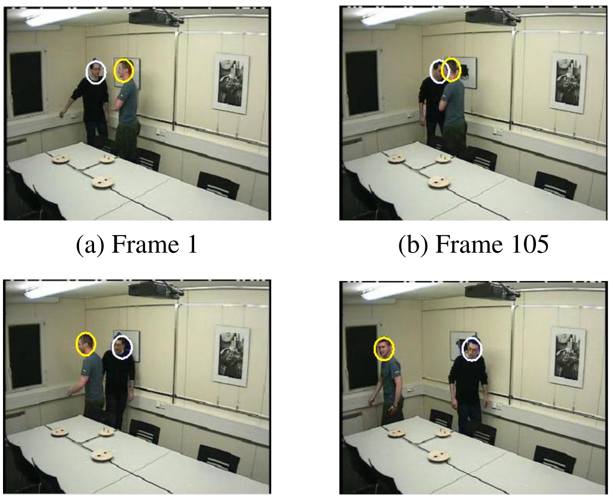
Figure 2: Tracking the heads of moving targets with the MCMC-PF, multiple colour cues and the proposed interaction model. The tracker performs well when targets are apart (a), when they are very close to each other (b) and even when they cross each other (c) and (d).



(a) Frame 1 (b) Frame 85

(c) Frame 135 (d) Frame 190

Figure 3: Tracking the heads of moving targets with the MCMC-PF, multiple colour cues and the proposed interaction model. The tracker performs well when targets are apart (a), when they are very close to each other (b), when one of them is occluded (b) and even when one of them reverses its motion after occlusion (d).



(a) Frame 1 (b) Frame 105

(c) Frame 170 (d) Frame 210

Figure 4: Tracking the heads of moving targets using sequence from AV16.3 corupus [14] with the MCMC-PF, multiple colour cues and the proposed interaction model. The tracker performs well when targets are apart (a), when they are very close to each other (b) and when they cross each other and change their direction (c) and (d).

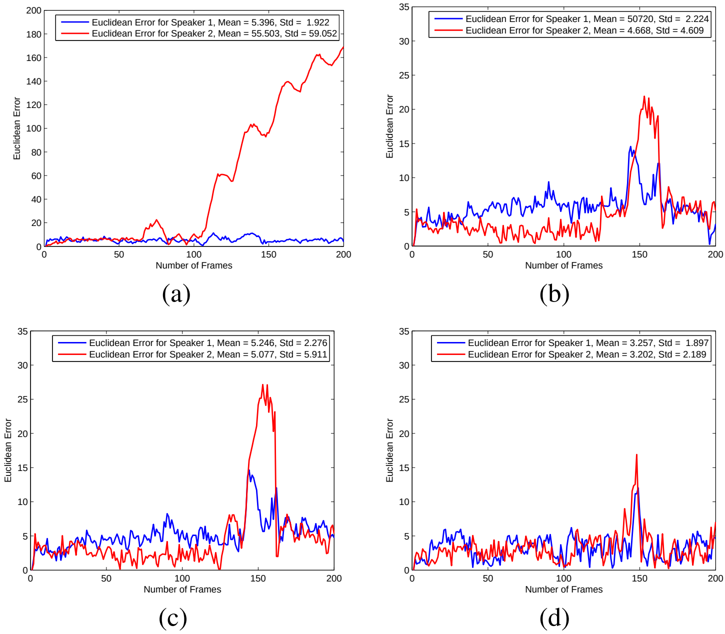


Figure 5: Euclidean Error: Euclidean error is calculated against manually annotated positions of the heads of the targets (a) with a MRF interaction model when targets interact and return back (b), (c) and (d) with the proposed interaction model.

give more weight to the colour cues by setting v equal to 0.7 and $16 \times 16 \times 16$ histogram bins are used for likelihood modeling. Size of the elements of the mixture model R is set to 4. It is assumed that the number of targets does not change and they are visible in the initial frame. Location of the center of the heads and patches defining the head are selected manually in the initial frame.

5.1 Tracking with an MRF Interaction Model

Fig.1 shows the tracking results of the MCMC-PF with multiple video cues and MRF interaction model. It is clear in Fig.1 that the tracker works very well for close interactions but fails when one of the targets reappears after occlusion. This is because the MRF interaction model proposed in [13] does not describe how to reinitialize the tracker when targets come out of occlusion and rather works on the basis of the assumption that two targets don't occupy the same space. As compared to this algorithm our proposed algorithm provides a specific solution for tackling the occlusion problems.

5.2 Tracking with the Proposed Interaction Model

Fig.2 shows that the proposed tracker successfully overcomes the tracking failure when the two targets cross over. The result in Fig.2-d shows that the tracker re-initializes itself very quickly when the target starts appearing again. Fig.3 shows that the tracker keeps tracking the targets even when one target is occluded behind the other and reverses its motion instead of crossing over. Finally, the tracker is tested for a sequence from the AV16.3 corpus and the results are shown in Fig.4. It is shown that the proposed tracker performs well even when the targets cross over and change their direction.

Fig.5-a through 5-d shows the Euclidean error for the sequences shown in Fig.1 through Fig.4 respectively. Error is calculated against the manually annotated positions of the heads of the targets. Fig.5-a shows that the error of the tracker with MRF interaction model [13] is very high and it linearly increases with time after the occlusion. It is clear from the results that the tracker with the proposed interaction model successfully tracks the targets which results in a small error before and after occlusion. During occlusion there is a small increase in error for a very short period of time but the tracker recovers back very quickly.

6 Conclusion

An improved head tracking algorithm using a MCMC-PF, multiple video cues and a new interaction model was implemented to track multiple people in a room environment. The new interaction model helped to overcome the tracking failures. The tracking results of the MCMC-PF with the proposed interaction model were compared with an MCMC-PF with MRF interaction model. It is shown that the MCMC-PF with the proposed interaction model provided a good solution to multiple target tracking when targets occlude and cross each other. In future the work will be extended by including audio localization to make the algorithm more robust.

References

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

[2] Y. Bar-Shalom and X. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, 1993.

[3] A. Bhattacharayya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of Calcutta Mathematical Society*, 35:99–110, 1943.

[4] X. Birchfield. An elliptical head tracker. *31st Asilomar Conference on Signals, Systems, and Computers*, 2:1710–1714, 1997.

[5] P. Brasnett, L. Mihaylova, D. Bull, and N. Canagarajah. Sequential Monte Carlo tracking by fusing multiple cues in video sequences. *Image and Vision Computing*, 25(8): 1217–1227, 2007.

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Online multi-person tracking-by-detection from a single, uncalibrated camera. *IEEE Transactions on pattern analysis and machine intelligence*, 33:1820–1833, 2011.

[7] F. Daum and J. Huang. Curse of dimensionality and particle filters. *IEEE Aerospace Conference*, pages 4.(1979) – 4.(1993), 2003.

[8] D. Gatica-Perez, G. Lathoud, J. Odobez, and I. McCowan. Audiovisual probabilistic tracking of multiple speakers in meetings. *IEEE Transactions on Audio, Speech and Language Processing*, 15(2):601–616, 2007.

[9] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman & Hall/CRC, 1996.

[10] N. Gordon, D. Salmond, and A. F. Smity. Novel approach to non-linear and non-Gaussian Bayesian state estimation. *IEEE Proceedings of Radar and Signal Processing*, 140(2):107–113, 1993.

[11] S. Huang, L. Fu, and P. Hsiao. Region-level motion-based foreground segmentation under a Bayesian network. *IEEE Transactions on Circuits and Systems for Video Technology*, 19(4):522–532, 2009.

[12] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, pages 5–28, 1998.

[13] Z. Khan, T. Balch, and F. Dellaert. MCMC-Based particle filtering for tracking a variable number of interacting targets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1805–1819, 2005.

[14] G. Lathoud, J. M. Odobez, and D. Gatica-Perez. AV16.3: an audio-visual corpus for speaker localization and tracking. *Proceedings of the MLMI'04 Workshop*, 2004.

[15] L. Mihaylova and A. Carmi. Particle algorithms for filtering in high dimensional state spaces: a case study in group object tracking. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5932 – 5935, 2011.

[16] S. M. Naqvi, M. Yu, and J. A. Chambers. A multimodal approach to blind source separation of moving sources. *IEEE Journal of Selected Topics in Signal Processing*, 4 (5):895–910, 2010.

[17] P. Noriega and O. Bernier. Real time illumination invariant background subtraction using local kernel histograms. *British Machine Vision Conference*, 3:979–988, 2006.

[18] K. Nummiaro, E. Koller-Meier, and L. V. Gool. An adaptive color based particle filter. *Image and Vision Computing*, 21(1):99–110, 2003.

[19] S. K. Pang, J. Li, and S. J. Godsill. Detection and tracking of coordinated groups. *IEEE Transactions on Aerospace and Electronic Systems*, 47(1):472–502, 2011.

[20] I. M. Rekleitis. A particle filter tutorial for mobile robot localization. *Technical Report TR-CIM-04-02, Centre for Intelligent Machines, McGill University, 3480 University St., Montreal, Quebec, Canada HčA 2A7*, 2004.

[21] F. Septier, S. K. Pang, A. Carmi, and S. Godsill. On MCMC based particle methods for Bayesian filtering:application to multitarget tracking. *3rd IEEE International Workshop on Computational Advances in Multi-sensor Adaptive Processing*, pages 360–363, 2009.

[22] X. Xu and B. Li. Head tracking using particle filter with intensity gradient and color histogram. *IEEE International Conference on Multimedia and Expo*, pages 888–891, 2005.