Multi-target Tracking by Using Particle Filtering and a Social Force Model

Ata-ur-Rehman*, Syed Mohsen Naqvi**, Lyudmila Mihaylova*, Jonathon A. Chambers**
*Department of Automatic Control and Systems Engineering University of Sheffield UK.
{a.ur-rehman, l.s.mihaylova}@sheffield.ac.uk
**School of Electronic, Electrical and Systems Engineering Loughborough University UK.
{s.m.r.naqvi, j.a.chambers}@lboro.ac.uk

***Abstract*—This paper presents a particle filter for multiple target tracking. The main contribution of this work is in the proposed likelihood function accounting for the interactions between the objects. The filter likelihood function is calculated by combining a social force model for human behaviour with image features such as colour and motion. The added social force model contributes to coping with occlusions between the objects. The performance of the developed algorithm is validated on real video data. The results demonstrate the algorithm accuracy during complex interactions between the objects.**

***Keywords*—multi-target tracking, social force model, particle filter**

I. Introduction

Multi-target tracking generally has many applications, such as surveillance, intelligent transportation, submarine tracking, animal tracking behavioural analysis, human computer interfacing and many more [1], [2]. It is a very challenging problem and enormous efforts have been made in the recent years to solve it [3], [4], [5], [6], [7], [8].

Most of the existing multi-target tracking techniques focus on improving the tracking efficiency with the help of efficient association of available data to the targets. This is achieved either by exploring hard assignment techniques [9], [8], [3], [10], [11] or soft assignments, for instance [12], [13], [14]. Other efforts propose improvements with improved detectors for specific objects [15] or better appearance models.

Many 2-D multiple tracking systems rely on the appearance models of people. For instance [16], [17] use the kernel density-based approach while [9] uses colour histograms, gradients and texture models to track human objects. These techniques use template matching which is not robust to occlusions because occluded parts of the targets can not be matched. They perform an exhaustive search for the desired target in the whole video frame which obviously requires much processing time.

Motion of every target can be influenced by the presence of the other targets around. The social behaviour of targets and how they react when they come close to other target has been exploited for predicting the locations of targets. Different dynamic models have been proposed in the literature for modelling dynamics of targets. These models can be divided in to two different categories, macroscopic and microscopic models [18]. Macroscopic techniques focus on the dynamics of the whole crowd. Microscopic models deal with the dynamics of every individual target by taking into account the behaviour of every single target and how they react to the movement of other targets and static obstacles.

An example of microscopic models is social force model [19]. The social force models can be used to predict the motion of every individual target by keeping into account the repulsive forces from other targets and static obstacles, and driving forces which guides the target towards a certain goal. A modified version of social force model is used in [20] for modelling the movements of targets in the non-observed area. However, it assumes that all the targets maintain a constant speed during the non-observed areas and that there are no interactions between targets.

Another microscopic dynamic model is proposed in [21] which differs from the social force model [19]. Instead of modelling the targets with the help of current locations it predicts the new locations and uses these locations to find the optimal velocity of the targets.

The existing social force model based tracking algorithms incorporate the social behaviour of targets into dynamic models. In the case of Bayesian tracking, these models are used to enhance the prediction stage. However, these social behaviours of targets have not yet been used to improve the data likelihood. This paper proposes a method for calculating the data likelihood function for efficient tracking of multiple targets. The developed technique exploits the social behaviours of targets along with motion and colour cues for evaluating the data likelihood. The improved likelihood model is used in the particle filters for efficiently evaluating the weights of the particles. This helps in overcoming the complex interactions between targets, especially, during partial and full occlusions when the measurement for the occluded targets is not available.

The paper is organised as follows: Section 2 describes the problem formulation while the Section 3 explains the proposed social force model, colour and motion cues based likelihood model. Experimental results are discussed in Section 4 and conclusions and possible future directions of the proposed work are presented in Section 5.

II. Problem Formulation

The goal of the proposed multi-target tracking algorithm is to track the state of N targets. The state of target i at discrete time k is represented as $(\mathbf{x}_k^i)^T$. where $\mathbf{x}_k^i = (\mathbf{p}_k^i, \mathbf{v}_k^i)$ contains the position $\mathbf{p}_k^i$ and velocity $\mathbf{v}_k^i$ information of the i^{th} target at time k. The joint state of all the targets at discrete time k is represented as $\mathbf{X_k} = [(\mathbf{x_k^1})^\mathbf{T}, \dots, (\mathbf{x_k^i})^\mathbf{T}, \dots, (\mathbf{x_k^N})^\mathbf{T}]^\mathbf{T}$. Measurements at time k are represented as $\mathbf{Y_k} = [(\mathbf{y_k^1})^\mathbf{T}, \dots, (\mathbf{y_k^j})^\mathbf{T}, \dots, (\mathbf{y_k^L})^\mathbf{T}]^\mathbf{T}$.

In terms of the Bayesian filtering the tracking problem can be viewed as a two step process. In the first step we estimate the current state of every individual target by using the Chapman-Kolmogorov equation [22].

$$p(\mathbf{X}_k|\mathbf{Y}_{k-1}) = \int p(\mathbf{X}_k|\mathbf{X}_{k-1}) p(\mathbf{X}_{k-1}|\mathbf{Y}_{k-1}) d\mathbf{X}_{k-1} \quad (1)$$

where $p(\mathbf{X}_k|\mathbf{X}_{k-1})$ represents the dynamic model. In the next step we update the prediction by using the measurements $\mathbf{Y}_k$. This update step corresponds to calculating the posterior distribution of the current state $\mathbf{X}_k$ given the measurements $\mathbf{Y}_k$, i.e.

$$p(\mathbf{X}_k|\mathbf{Y}_k) \propto p(\mathbf{Y}_k|\mathbf{X}_k) p(\mathbf{X}_k|\mathbf{Y}_{k-1}) \quad (2)$$

where $p(\mathbf{Y}_k|\mathbf{X}_k)$ represents the likelihood model. In the proposed algorithm we exploit social forces between targets and different video cues such as colour and motion cues for estimating the likelihood of measurements

$$p(\mathbf{Y}_k|\mathbf{X}_k) = \prod_{j=1}^{M} p(\mathbf{y}_k^j|\mathbf{X}_k). \quad (3)$$

Particle filtering [23] is used in this work to estimate the posterior distribution $p(\mathbf{X}_k|\mathbf{Y}_k)$. The particle filter represents the required posterior distribution of a state by a set of random samples taken from the distribution at a discrete time step. These random samples are assigned weights and the estimate of the posterior distribution is computed on the basis of these samples and their associated weights. This can be described by the following equation

$$p(\mathbf{X}_k|\mathbf{Y}_k) \approx \sum_{s=1}^{N_s} \mathbf{w}_k^s \delta(\mathbf{X}_k - \mathbf{X}_k^s) \quad (4)$$

where $\mathbf{X}_k^s$ is the s^{th} sample and $\mathbf{w}_k^s$ is the associate weight, N_s is the total number of samples and $\delta(\cdot)$ is the delta function. As a first step of the particle filter based tracking we predict N_s samples using a motion model. The existing tracking algorithms exploits the social forces between targets to improve this prediction stage. Algorithm proposed by [21] uses the social behaviours of targets to improve this prediction stage. However, a two-dimensional motion of a moving target can also be described by simple constant velocity model [24]. In the second step we assign weights to every particle by using the following equation

$$\mathbf{w}(\mathbf{X}_k^s) = \mathbf{w}^c(\mathbf{X}_k^s) \mathbf{w}^m(\mathbf{X}_k^s) \mathbf{w}^f(\mathbf{X}_k^s) \quad (5)$$

where $\mathbf{w}^c(\mathbf{X}_k^s)$, $\mathbf{w}^m(\mathbf{X}_k^s)$ and $\mathbf{w}^f(\mathbf{X}_k^s)$ represent the weight assigned to s^{th} particle by using, colour and motion cues, and social force model, respectively. Section III provides the detailed description of how these weights have been calculated.

In the final step of the particle filter based tracking we update the prediction by using equation (4).

III. Likelihood Model

In this work, we calculate the weights for every particle by using two video cues: colour and motion, and social force model.

A. Colour Cue

At every time step k, for every target i we create a measurement histogram $\mathbf{H}_k^{i,s}$ for a small patch from the video frame selected according to its predicted state $\mathbf{x}_k^{i,s}$. The Bhattacharya distance [25] is then calculated between the measurement histogram and the reference template histogram $\mathbf{H}_{ref}^i$

$$d_k^{i,s}(\mathbf{H}_{ref}^i, \mathbf{H}_k^{i,s}) = \sqrt{1 - \rho(\mathbf{H}_{ref}^i, \mathbf{H}_k^{i,s})} \quad (6)$$

where, $\rho(\mathbf{H}_{ref}^i, \mathbf{H}_k^{i,s})$ is the Bhattacharyya coefficient

$$\rho(\mathbf{H}_{ref}^i, \mathbf{H}_k^{i,s}) = \sum_{g=1}^{G} \sqrt{\mathbf{H}_{ref}^{i,g}, \mathbf{H}_k^{i,s,g}} \quad (7)$$

where G represents the number of histogram bins and we have used $16 \times 16 \times 16$ colour histograms bins.

The Bhattacharya distance is further used to calculate the measurement likelihood with respect to the colour histograms.

$$w^c(\mathbf{x}_k^{i,s}) \propto exp\left(- \frac{d_k^{i,s}(\mathbf{H}_{ref}^i, \mathbf{H}_k^{i,s})}{2\mathbf{\Sigma}^2} \right) \quad (8)$$

where $\mathbf{\Sigma}$ is the measurement noise.

B. Motion Cue

Colour cues are efficient in detecting targets when targets are prominent in the video, however, they may fail in certain scenarios for instance, when targets are partially occluded or smaller in size. Motion cue can help in measuring other aspects of the video. The basic idea is to detect the moving objects in the video frame. The second weight in equation (5) i.e. $\mathbf{w}^m(\mathbf{x}_k^s)$ is calculated on the basis of the motion cues, which provides more robustness to the data likelihood.

In the most crude form of the motion detection, frame difference between the successive frames is calculated. In this proposed algorithm, for motion cue we calculate the frame difference as a first stage. We denote the frame difference of frames at time index k and $k-1$ as $\hat{\mathbf{Y}}_\mathbf{k}$. At the second stage we calculate a measurement histogram for these measurements with respect to the predicted state. For target i which is represented by the state $\mathbf{x}_k^{i,s}$, the histogram is denoted as $\hat{\mathbf{H}}_\mathbf{k}^\mathbf{i,s}$. We then calculate the Bhattacharya distance between the reference histogram and the measurement histogram calculated according to the predicted state by s^{th} particle. A detailed description of motion cue calculation can be found in [26].

C. Social Behaviour cue

Given the current state at time index $k-1$, the next state at time index k is predicted by using the state transition model. The state model predicts N_s particles to construct the state distribution at time step k. This section explains how we have exploited the social behaviour of humans to assign weights to the particle predicted at the prediction stage of the particle filter.

Due to the missing measurements during partial and full occlusions, the colour and motion cues may fail in such scenarios. The proposed method which is based on the social behaviours of humans enhances the robustness of the overall likelihood model.

The movement of every target i can be influenced by the presence of the set of other targets. In practice, every target tries to avoid collision with every other target. Therefore, there is an unseen repulsive force being applied on every target which forces it to maintain a minimum distance from other targets.

This human behaviour is translated into a weight assignment procedure for predicted particle. The third weight in equation (5) i.e. $\mathbf{w}^f(\mathbf{X}_k^s)$ is calculated by exploiting this social behaviours.

If we assume that at time step k for s^{th} particle the target i has the predicted position $\mathbf{p}_k^{i,s}$ and its neighbouring target j has the predicted position $\mathbf{p}_k^{j,s}$ then the distance between predicted positions of targets i and j can be defined as

$$d_{ij}^2(k) = \parallel \mathbf{p}_k^{i,s} - \mathbf{p}_k^{j,s} \parallel^2 . \tag{9}$$

The unseen force between targets is inversely related to the distance between them. If the distance decreases this force increases. Keeping this fact in mind, the distance between the predicted locations of targets is used to assign weights.

At every time step k, for every particle s we calculate the distance between every target i from every other target j. The weight $w^f(\mathbf{x}_k^{ij,s})$ for target i with respect to target j assigned to the s^{th} particle is calculated as

$$w^f(\mathbf{x}_k^{ij,s}) = exp\left(\frac{-d_{ij}^2(k)}{2\sigma_d^2}\right) \tag{10}$$

where σ_d controls the distance of a target to be avoided. The overall weight $w^f(\mathbf{x}_k^{i,s})$ for i^{th} target is a weighted sum of weights calculated with respect to every other target $r \neq i$

$$w^f(\mathbf{x}_k^{i,s}) = \sum_{r \neq i} g_{ir} w^f(\mathbf{x}_k^{ir,s}) \tag{11}$$

the parameter g_{ir} represents the influence of each target $(r \neq i)$ in the the overall weight for target i. This is defined as

$$g_{ir} = exp\left(\frac{-\parallel \mathbf{p}_{k-1}^i - \mathbf{p}_{k-1}^r \parallel^2}{2\sigma_w^2}\right) \tag{12}$$

where σ_w defines the radius of influence of targets. Weights for every target is calculated in a similar fashion. This weight $w^f(\mathbf{x}_k^{i,s})$ which is calculated with the help of social behaviour cue along with the weight due to motion and colour cues is used in equation (5) to evaluate the overall weight for particle s.

This completes the description of the likelihood model. Algorithm 1 describes the overall proposed tracking algorithm. The next section presents the detailed tracking results and performance comparison of the proposed method with the technique due to [27].

Algorithm 1 Proposed tracking algorithm

$[\mathbf{X_k^s}, \mathbf{w_k^s}]_{\mathbf{s=1}}^{\mathbf{N_s}} = \mathbf{PF}[(\mathbf{X_{k-1}^s}, \mathbf{w_{k-1}^s})_{\mathbf{s=1}}^{\mathbf{N_s}}, \mathbf{Y_k}, \mathbf{H_{ref}}]$

for $k = 1 : T$ **do**
  Calculate the frame difference difference between k and $k-1$
  for $s = 1 : N_s$ **do**
    for $i = 1 : N$ **do**
      Draw $\mathbf{x_k^{i,s}}$ using $p(\mathbf{x_k}|\mathbf{x_{k-1}^{i,s}})$ by using constant velocity model of equation (13)
      Calculate the $\mathbf{H}_k^{i,s}$
      Calculate the $\hat{\mathbf{H}}_k^{i,s}$
      Calculate weights $w^c(\mathbf{x}_k^{i,s})$ according to equation (8)
      Calculate weight $w^m(\mathbf{x}_k^{i,s})$ according to the motion model explained in Section III-B
    end for
    for $i = 1 : N$ **do**
      Evaluate distances according to equation (9)
      Evaluate weights $w^f(\mathbf{x}_k^{i,s})$ according to equation (11)
      Evaluate overall $\mathbf{w}(\mathbf{X}_k^s)$ weight according to equation (5)
    end for
  end for
  Calculate the Total weight: $t = SUM[\mathbf{w}_k^s]_{s=1}^{N_S}$
  for $s = 1 : N_s$ **do**
    Normalize: $\mathbf{w}_k^i = t^{-1}\mathbf{w}_k^s$
  end for
  Update states according to equation (4)
  Resample the particles
end for

The developed social behaviour based likelihood model only considers the repulsive forces between targets. However, this work can easily be extended by considering other behaviours of targets, for instance repulsive forces from static obstacles such as wall, driving forces which force targets to move towards their desired destination and steering mechanisms [21], [18]. Overlapping and oscillation phenomenons [18] can also be included in the proposed likelihood model.

IV. Experimental Results

The algorithm is evaluated by tracking humans in a video recording taken from the AV16.3 corpus [28] available at http://glat.info/ma/av16.3/. To examine the robustness of the algorithm to close interactions and occlusions one of the most complicated sequence from the corpus is selected. To investigate the performance of the proposed technique, results of the proposed social force model based particle filter are compared with [27].

The test sequence is recorded at a resolution of 288×360 pixels at $25Hz$ showing three people moving in a room environment in a close arena.

Particle size $N_s = 60$ is used for the particle filter. Filter requires a manual initialisation of targets when they first enters the tracking region.

For every target in the video, we predict its location by using a dynamic modal. A two-dimensional motion of a moving target is predicted by using the constant velocity model

$$\mathbf{x}_k^{i,s} = \mathbf{T}\mathbf{x}_{k-1}^{i,s} + \boldsymbol{\vartheta}_k \tag{13}$$

where $\boldsymbol{\vartheta}_k$ is the system noise and the matrix $\mathbf{T}$ is defined as

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

and t is the frame sampling interval. Every prediction represents a particle in the prediction.

Figure 1 shows the tracking results for few of the selected frames. It shows that with the proposed technique we can overcome tracking failure which may occur during occlusions.

A technique proposed in [29], [30] is followed in this paper to evaluate the overall performance of the algorithm with the proposed likelihood function.

Let $\mathbf{G}_k = [(\mathbf{g}_k^1)^T, \cdots, (\mathbf{g}_k^i)^T, \cdots, (\mathbf{g}_k^N)^T]^T$ be the ground truth at time $k \in [k_s, k_e]$, where k_s and k_e are respectively the starting and ending points of the observation interval. Each ground truth $\mathbf{g}_k^i = (\hat{\mathbf{x}}_k^i, \hat{I}_i)$ contains the actual position and identity of the target i. Similarly, $\mathbf{O}_k = [(\mathbf{o}_k^1)^T, \cdots, (\mathbf{o}_k^i)^T, \cdots, (\mathbf{o}_k^N)^T]$ represents the output of the tracking algorithm at time k, where each $\mathbf{o}_k^i = (\mathbf{x}_k^i, I_i)$ represents the estimated location and identity of target i. At every time k the error is defined as

- Missed detection: This corresponds to the targets for which there is no filter assigned.
- Wrong identification: This corresponds to the targets which have been given a wrong identity.
- False positive: This corresponds to the filter which is not assigned to any of the targets.

If it is assumed that md_k, wi_k, fp_k and gt_k are respectively total number of missed detections, wrong identifications, false positives and ground truths at time k, the errors are calculated as [30]

$$MD = \sum_k \frac{md_k}{gt_k}, WI = \sum_k \frac{wi_k}{gt_k}, FP = \sum_k \frac{fp_k}{gt_k} \tag{14}$$

The accuracy of the algorithm is then calculated as

$$Accuracy = 1 - MD - WI - FP \tag{15}$$

Table I presents the performance of the over all tracking results which are achieved with particle filter and the proposed likelihood function with colour and motion cues and the social force model. Accuracy of the proposed technique, calculated according to equation (15) is compared with the technique due to [27]. The performance results are compiled by using 300 video frames of the AV16.3 sequence compiled with five runs of the algorithm.

The results show that the proposed algorithm has improved the tracking results. There is a significant reduction in the wrong identifications and missed detections which has improved the overall accuracy of the proposed tracking technique.

TABLE I. COMPARISON OF PERFORMANCE

	Accuracy	MD(%)	WI(%)	FP(%)
Tracking algorithm [27]	73.05	15.11	6.32	5.51
Tracking with proposed likelihood model	80.13	10.85	5.03	3.99

Precision measures are also calculated to evaluate the performance of the algorithm. Precision of the proposed tracking algorithm is calculate as a ratio of the to number of correctly identified pixels in a target patch and the total number of pixels detected

$$Precision = \frac{\text{Total no. of pixels correctly identified}}{\text{Total no. of pixels detected}}. \tag{16}$$

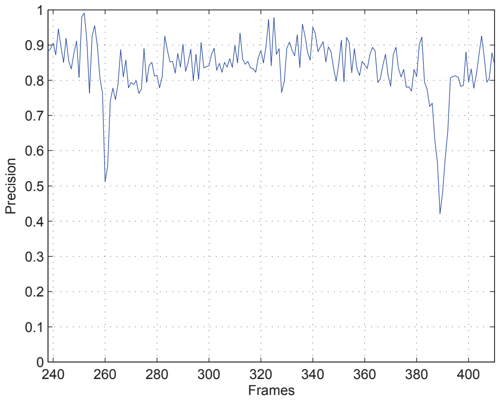


Fig. 2. Precision results plotted against the frames for target 1.

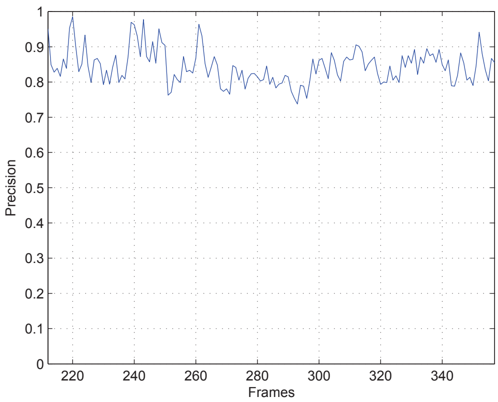
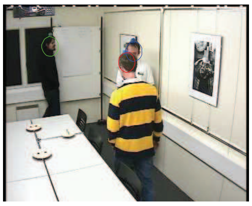


Fig. 3. Precision results plotted against the frames for target 2.

The precision results presented in figures 2, 3 and 4 show the robustness of the proposed algorithm during complex interactions between targets. The performance of the proposed algorithm slightly degrades during the complete occlusion of targets but it recovers well when targets come out of occlusions.

(a) Frame 124



(b) Frame 225

(c) Frame 278

(d) Frame 288

(e) Frame 320

(f) Frame 326

Fig. 1. Tracking results: it is shown that the proposed tracking algorithm can successfully track a number of targets while handling complex occlusions.

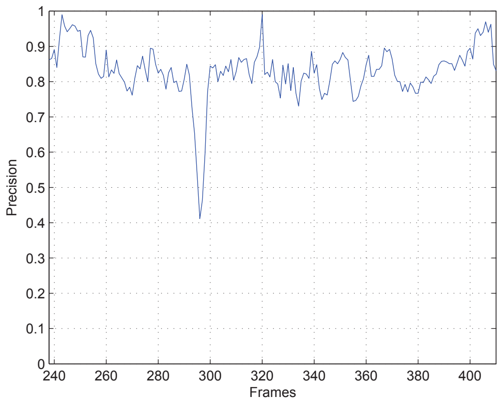


Fig. 4. Precision results plotted against the frames for target 3.

V. Conclusions and Future Work

A particle filter is developed for tracking multiple targets. The likelihood function involves a social force model which helps the algorithm to cope with overlapping objects. The model is used with a particle filter for tracking multiple humans. In the proposed algorithm colour and motion features along with the social forces between human targets are used to design the measurement likelihood. The proposed algorithm is tested on real data sets which demonstrates that the proposed algorithm successfully tracked targets during complex human interactions and an overall performance improvement is achieved.

This work focused on the development of a new likelihood model which is used with a particle filter to track multiple targets. In the current work we have only utilised the behaviour of targets with respect to the the other the targets, in future we intend to exploit the social behaviour of targets due to the static obstacles and the driving forces which force the targets to move towards the desired destination.

VI. Acknowledgments

We are grateful to the support from the UK Engineering and Physical Sciences Research Council (EPSRC) for the support via the Bayesian Tracking and Reasoning over Time (BTaRoT) grant EP/K021516/1.

References

[1] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filter for Tracking Applications*. Boston—London: Artech House Publishers, 2004.

[2] M. S. Khan, S. M. Naqvi, A. ur Rehman, W. Wang, and J. Chambers, "Video-aided model-based source separation in real reverberant rooms," *IEEE Trans. Audio, Speech, and Language Processing,*, vol. 21, no. 9, pp. 1900–1912, Sept 2013.

[3] D. Gatica-Perez, G. Lathoud, J. Odobez, and I. McCowan, "Audiovisual probabilistic tracking of multiple speakers in meetings," *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 2, pp. 601–616, 2007.

[4] Z. Khan, T. Balch, and F. Dellaert, "MCMC-Based particle filtering for tracking a variable number of interacting targets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, 2005.

[5] T. D. Laet, H. Bruyninckx, and J. D. Schutter, "Shape-based online multitarget tracking and detection for targets causing multiple measurements: Variational Bayesian clustering and lossless data association," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2477–2491, 2011.

[6] M. Montemerlo, W. Whittaker, and S. Thrun, "Conditional particle filters for simultaneous mobile robot localization and people-tracking," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 2002, pp. 695–701.

[7] M. Lindstrom and J. Eklundh, “Detecting and tracking moving objects from a mobile platform using a laser range scanner,” in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 1364–1369.

[8] X. Xu and B. Li, “Head tracking using particle filter with intensity gradient and color histogram,” in *Proc. IEEE International Conference on Multimedia and Expo*, 2005, pp. 888–891.

[9] P. Brasnett, L. Mihaylova, D. Bull, and N. Canagarajah, “Sequential Monte Carlo tracking by fusing multiple cues in video sequences,” *Journal of Image and Vision Computing*, vol. 25, no. 8, pp. 1217–1227, 2007.

[10] K. Nummiaro, E. Koller-Meier, and L. V. Gool, “An adaptive color based particle filter,” *Journal of Image and Vision Computing*, vol. 21, no. 1, pp. 99–110, 2003.

[11] A. ur Rehman, S. M. Naqvi, L. Mihaylova, and J. A. Chambers, “Variational Bayesian and belief propagation based data association for multi-target tracking,” *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.

[12] T. Long, L. Zheng, X. Chen, Y. Li, and T. Zeng, “Improved probabilistic multi-hypothesis tracker for multiple target tracking with switching attribute states,” *IEEE Trans. Signal Processing*, vol. 59, pp. 5721–5733, 2011.

[13] I. J. Cox, “A review of statistical data association techniques for motion correspondence,” *International Journal on Computer Vision*, vol. 10, pp. 53–66, 1993.

[14] A. ur Rehman, S. M. Naqvi, L. Mihaylovay, and J. A. Chambers, “Clustering and a joint probabilistic data association filter for dealing with occlusions in multi-target tracking,” in *IEEE International Conference on Information Fusion (FUSION)*, July 2013, pp. 1730–1735.

[15] S. Tang, M. Andriluka, and B. Schiele, “Detection and tracking of occluded people,” in *Proc. of the British Machine Vision Conference*. BMVA Press, 2012, pp. 9.1–9.11.

[16] A. M. Elgammal and L. S. Davis, “Probabilistic framework for segmenting people under occlusion,” in *Proc. IEEE International Conference on Computer Vision*, vol. 2, 2001, pp. 145–152.

[17] A. M. Elgammal, R. Duraiswami, and L. S. Davis, “Efficient non-parametric adaptive color modeling using fast Gauss transform,” in *Proc. IEEE Conference on Computer Vision Pattern Recognition (CVPR)*, vol. 2, 2001, pp. II–563–II–570.

[18] S. Ali, K. Nishino, D. Manocha, and M. Shah, *Modeling, Simulation and Visual Analysis of Crowds*. Springer, 2013.

[19] D. Helbing and P. Molnár, “Social force model for pedestrian dynamics,” *Physical Review E*, vol. 51, pp. 4282–4286, 1995.

[20] R. Mazzon and A. Cavallaro, “Multi-camera tracking using a multi-goal social force model,” *Neurocomputing*, vol. 100, no. 0, pp. 41 – 50, 2013.

[21] S. Pellegrini, A. Ess, K. Schindler, and L. V. Gool, “You’ll never walk alone: Modeling social behavior for multi-target tracking,” in *IEEE International Conference on Computer Vision,*, 2009, pp. 261–268.

[22] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[23] N. Gordon, D. Salmond, and A. F. Smity, “Novel approach to non-linear and non-Gaussian Bayesian state estimation,” in *Proc. IEEE Radar and Signal Processing Conference*, vol. 140, no. 2, 1993, pp. 107–113.

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

[25] A. ur Rehman, S. M. Naqvi, R. Phan, W. Wang, and J. A. Chambers, “MCMC-PF based multiple head tracking in a room environment,” in *Proc. British Machine Vision Workshop*, 2012.

[26] P. Perez, J. Vermaak, and A. Blake, “Data fusion for visual tracking with particles,” *Proceedings of the IEEE*, vol. 92, no. 3, pp. 495–513, 2004.

[27] J. Czyz, B. Ristic, and B. Macq, “A particle filter for joint detection and tracking of color objects,” *Image and Vision Computing*, vol. 25, no. 8, pp. 1271 – 1281, 2007.

[28] G. Lathoud, J. M. Odobez, and D. Gatica-Perez, “AV16.3: An audio-visual corpus for speaker localization and tracking,” in *Proc. International Conference on Machine Learning for Multimodal Interaction (MLMI)*. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 182–195.

[29] K. Bernardin, “Multimodal probabilistic person tracking and identification in smart space,” *PhD Thesis, University of Fridericiana zu Karlsruhe*, 2009.

[30] A. Kumar and C. D. Vleeschouwer, “Prioritizing the propagation of identity beliefs for multi-object tracking,” in *Proc. British Machine Vision Conference*. BMVA Press, 2012, pp. 117.1–117.11.