A Robust PHD Filter With Deep Learning updating For Multiple Human Tracking

Pengming Feng[1], Wenwu Wang[1], Syed Mohsen Naqvi[2], Jonathon A. Chambers[2]
1. Center for Vision Speech and Signal Processing, Department of Electronic Engineering,
University of Surrey, UK
Emails:{p.feng, w.wang}@surrey.ac.uk
2. Advanced Signal Processing Group, Loughborough University, UK
Emails:{s.m.r.naqvi, j.a.chambers}@lboro.ac.uk

***Abstract*—We propose a novel robust probability hypothesis density (PHD) filter for multiple target tracking in an enclosed environment, where a deep learning method is used in the update step for combining different human features to mitigate the effect of measurement noise on the calculation of particle weights. Deep belief networks (DBNs) are trained based on both colour and oriented gradient (HOG) histogram features and then used to mitigate the measurement noise from the particle selection and PHD update step, thereby improving the tracking performance. To evaluate the proposed PHD filter, two sequences with 383 frames from the CAVIAR dataset are employed and both the optimal subpattern assignment (OSPA) and mean of error from each target method are used as objective measures. The results show that the proposed robust PHD filter outperforms the traditional PHD filter.**

***Index Terms*—Multiple human tracking, PHD filter, deep learning, deep belief networks**

I. Introduction

In multiple target tracking (MTT), the number of targets is often unknown and varies with time. In addition, the occlusion problem may occur and this further increases the challenges for reliable target tracking. A particular issue in MTT is that it is not always possible to associate measurements with particular targets and therefore false alarms and missed detection may be generated particularly in the presence of clutter, occlusion and noise [1].

Several methods could be used to address these challenges in MTT. Earlier methods include the Kalman filter and particle filter where the number of targets is assumed to be known and fixed. For a variable number of targets, the random finite set (RFS) [2] based probability hypothesis density (PHD) filter has been recently proposed for the MTT problem. The advantage of the PHD filter is that it can estimate both the number of targets and their locations, and thus avoids the need for data association techniques as part of the multiple target framework [3][4][5]. Moreover, it mitigates the computational complexity issue as often occurs in other multiple target tracking approaches such as multiple hypothesis tracking (MHT) [5]. However, the limitation of the PHD filter is that its performance can be easily affected by estimation errors caused by noise.

In this paper, we propose a novel robust PHD filter for multiple human tracking. Since in an enclosed environment there are limited human features which can be extracted when only using a single camera and accurate measurement of the humans can be difficult to obtain due to illumination and posture changes, we employ deep belief networks (DBNs) to calculate the weights for the particle based PHD filter, which utilizes colour and oriented gradient histogram features due to their accuracy in describing human targets. The DBNs are shown to be robust against background noise in the measurement due to the difference between the human target and noises features, and their ability to extract useful information from limited features in human tracking work. To evaluate the performance of our proposed robust PHD filter, we employ two sequences from the CAVIAR dataset which includes appearance, occlusion, disappearance of humans in the field of view of a camera. It is shown that our proposed PHD filter can obtain more accurate results and perform better when tracking a variable number of humans in an enclosed environment.

II. System Components

The construction of the proposed PHD filter can thus be synthesised as follows: training the DBNs by existing datasets, particle prediction, background subtraction and new-born target selection for the PHD filter, weights calculation with the aid of DBNs, updating the weights by a PHD updating step and resampling, where the main steps are described below in detail.

A. Particle PHD filter for human tracking

To formulate the PHD filter the RFS framework is employed [6]. We denote $\mathbf{D}_{k|k}(\mathbf{x})$ as the PHD at discrete time k associated with the multi-target posterior density $p_{k|k}(\mathbf{X}_k|\mathbf{Z}_{1:k})$, where $\mathbf{X}_k = \{\mathbf{x}_k^m, m = 1, ...M\}$ includes the 2D positions of all the human targets, $\mathbf{x}_k^m$ denotes the state of the m^{th} target at time k, M is the number of targets and $\mathbf{Z}_{1:k}$ denotes the measurements up to time k. The PHD prediction step is defined as:

$$\mathbf{D}_{k|k-1}(\mathbf{x}_k^m) = \int \phi_{k|k-1}(\mathbf{x}_k^m, \xi)\mathbf{D}_{k-1|k-1}(\xi)d(\xi) + \Upsilon_k \quad (1)$$

where Υ_k is the intensity function of the new target birth RFS, $\phi_{k|k-1}(\mathbf{x}_k^m, \xi)$ is the analogue of the state transition probability in the single target case which is calculated from

$$\phi_{k|k-1}(\mathbf{x}_k^m, \xi) = e_{k|k-1}(\xi) f_{k|k-1}(\mathbf{x}_k^m|\xi) + \beta_{k|k-1}(\mathbf{x}_k^m|\xi) \tag{2}$$

in which $f_{k|k-1}$ is the multi-target transition density, $e_{k|k-1}(\xi)$ is the probability that the target still exists at time k and $\beta_{k|k-1}(\mathbf{x}_k^m|\xi)$ is the intensity of the RFS that a target is spawned from the state ξ. The PHD update step is defined as [7]:

$$\mathbf{D}_{k|k}(\mathbf{x}_k^m) = \left[p_M(\mathbf{x}_k^m) + \sum_{z \in \mathbf{Z}_k} \frac{\psi_{k,z}(\mathbf{x}_k^m)}{\kappa_k + \langle \psi_{k,z}, \mathbf{D}_{k|k-1} \rangle} \right] \times \mathbf{D}_{k|k-1}(\mathbf{x}_k^m) \tag{3}$$

where p_M is the missing detection probability, $\psi_{k,z}(\mathbf{x}_k^m) = (1 - p_M) g_k(z|\mathbf{x}_k^m)$ is the single-target likelihood defining the probability that a measurement z is generated by a target with state $\mathbf{x}_k^m$, κ_k is the clutter intensity, and $\langle f, g \rangle = \int f(x) g(x) dx$ [5].

There are numerical solutions [8] for the integrals in (1) and (3), one of which is obtained using a sequential Monte Carlo method that approximates the PHD with a set of weighted random samples, which is called the particle PHD filter and is the focus of this paper. We use this method because it performs well in the non-Gaussian noise and non-linear model framework, besides, it is easy to combine the PHD weight update step with the DBNs classifier. Two fundamental steps in the particle filter are sequential importance sampling and resampling. The basic principle of importance sampling is to represent a PDF $p(\mathbf{X}_k)$ by a set of random particles associated with the weights, where $\mathbf{X}_k = \{\mathbf{x}_k^i, i = 0, ..., N\}$, N is the number of particles we employed in the particle filter. Given a set of particles [9]

$$\{w_{k-1}^i, \mathbf{x}_{k-1}^i\}_{i=1}^N \tag{4}$$

which are independently drawn from importance sampling density $q(\mathbf{X}_k)$ [9], the weight of each particle can be calculated as

$$w_k^i = p(\mathbf{x}_k^i)/q(\mathbf{x}_k^i) \tag{5}$$

thus $p(\mathbf{X}_k)$ can be approximated as

$$p(\mathbf{X}_k) \approx \sum_{i=1}^N w_k^i \delta(\mathbf{X}_k - \mathbf{x}_k^i) \tag{6}$$

where $\delta(\cdot)$ denotes the Dirac delta measure.

Assuming the particles for the PHD filter are independently drawn from the PDF $p(\mathbf{X}_{k-1}|\mathbf{Z}_{1:k-1})$, the particles $\mathbf{x}_k^i, i = 1, ..., N$ are propagated and updated by the Gaussian distribution, which are approximately distributed as $p(\mathbf{X}_k|\mathbf{Z}_k)$ [8]. In this case, the proposed filter is an approximate implementation of the relationship between the prediction and updating step of the filter. The prediction and updating step can be described as follows.

1. Prediction: Draw particle $\mathbf{x}_{k-1}^i$ from $\mathbf{X}_{k-1}$ and feed it into the prediction step to obtain particles at time k. Thus the prediction model can be calculated by

$$p(\mathbf{X}_k|\mathbf{Z}_{k-1}) = \int p(\mathbf{X}_k|\mathbf{X}_{k-1}) p(\mathbf{X}_{k-1}|\mathbf{Z}_{k-1}) d\mathbf{X}_{k-1} \tag{7}$$

2. Measurement update: Upon the receipt of the measurement $\mathbf{Z}_k$, the likelihood of each prior sample $\mathbf{x}_k^i, i = 1, ...N$, can be evaluated and drawn independently from importance sampling density $q(\mathbf{X}_k|\mathbf{Z}_{1:k})$ [8]. The importance weight for each prior sample can be calculated as:

$$w_k^i = \frac{p(\mathbf{Z}_k|\mathbf{x}_k^i) p(\mathbf{X}_k|\mathbf{Z}_{k-1})}{q(\mathbf{x}_k^i|\mathbf{Z}_k)} \tag{8}$$

Equations (8) and (9) form the basis of the proposed robust particle PHD filter.

The above work underpins the traditional particle PHD filter for human tracking; in the next subsection, human features are extracted to train the DBNs in the utilization of particle selection and updating part of the particle PHD filter.

B. Human feature extraction

We detect the target in video using the background subtraction method [10]. The new-born target for the PHD filter can thereby be obtained more accurately than by selecting particles in the whole frame randomly. In this paper, we used a codebook method for background subtraction [11]. There is no parametric assumption on the codebook model and it has several advantages: it has the capability of coping with illumination changes, and the potential to capture structural background motion over a long period of time under limited memory. The detail of the algorithm is described in [11].

The resulting raw background subtraction results generally contain many noise artifacts, which include small 'salt and pepper' [11] and large noises caused by the problem of poor illumination and similar colour between the foreground and background information. This may be regarded as a new born target in the prediction step of the PHD filter and causes the occurrence of a false alarm. Since the noise can be distinguished from the human target by a classifier; in this paper, we used a classifier based on the DBNs with the aid of colour and oriented gradient histogram features of a human target to perform classification and assist the calculation of particle weights.

C. The deep learning networks

Deep learning methods use a hierarchical learning method when training the structure of the system. As described in [12], the hierarchical learning uses natural progression from low level to high level structure as seen in natural complexity, so it is easier to monitor what is being learnt and to guide the machine to better subspaces. For example, when images are input in the system, the first layer of the system represents the ′edges′ of the feature; the second layer represents the ′object parts′ of the feature and the third layer represents the objects in the features [12]. As it is described in [13], composed by restricted Boltzmann machines (RBMs), DBNs are graphical models which learn to extract a deep hierarchical representation of the training data. Given the limitations on the feature extraction in human tracking using a single camera as sensor, DBNs provide the advantage of using the limited information to train a classifier to calculate the weights of particles in the human tracking work. The deep belief network we used is represented in Fig. 1

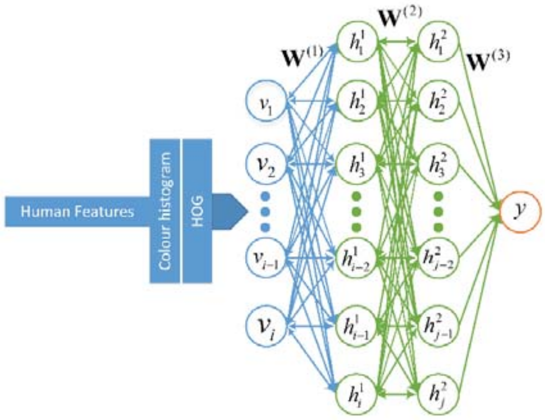

Fig. 1. The architecture for the deep belief networks in our proposed system.

which is composed of one input layer, two RBMs based hidden layers and a one-class output layer. As presented in [13], the RBMs are trained with a contrastive divergence algorithm described in Algorithm 1:

Algorithm 1 RBMs training process [13]

For each training data $\mathbf{x}$ from a training dataset $\mathbf{X}$

1. Input $\mathbf{x}$ to the visible layer to calculate the probability of activating the hidden layer

$p(h_j^0 = 1|v^0) = \sigma(W_j v_0)$

2. Sample h^0 from the probability distribution obtained above $h^0 \sim p(h^0|v^0)$

3. Rebuild the visible layer with h^0

$p(v_i^1 = 1|h^0) = \sigma(W_i^T h^0)$

4. Calculate the probability of activating the hidden layer again by v^1 sampled from $p(v^1 = 1|h^0)$

$p(h_j^1 = 1|v^1) = \sigma(W_j v_1)$

5. Update the weight by

$W \leftarrow W + \lambda(p(h_j^0 = 1|v^0))v^{0T} - (p(h_j^1 = 1|v^1))v^{1T}$

End for

Based on the above process, features extracted for the visible layer can be represented by hidden layers more accurately. Then the DBNs can be obtained by RBMs in Algorithm 2 [13]; the whole structure of the DBNs we used is shown as Fig. 1.

After training the DBNs following the process above, the weight for each particle $\mathbf{x}_k^i$ can be calculated as

$$p(\mathbf{x}_k^i) = e^{(c \cdot \mathbf{1} \cdot \mathbf{W}^{(3)})} \tag{9}$$

where $\mathbf{1}$ is a $(1 \times j)$ all-one vector, c is a constant we set for calculating the weights for the particles. In this way, the likelihood for each particle is obtained and these weights can then be taken as the input to the updating step of the PHD filter for MTT as discussed in the next section.

Algorithm 2 DBNs Process Algorithm [13]

---

1. Train the first layer as an RBM that models the raw input $\mathbf{x} = h^0$ as its visible layer.
2. Use that first layer to obtain a representation of the input that will be used as data for the second layer. Two common solutions exist. This representation can be chosen as being the mean activations $p(h^1 = 1|h^0)$ or samples of $p(h^1|h^0)$
3. Train the second layer as an RBM, taking the transformed data (samples or mean activations) as training examples (for the visible layer of that RBM).
4. Iterate (2 and 3) for the desired number of layers, each time propagating upward either samples or mean values.
5. Fine-tune all the parameters of this deep architecture with respect to a supervised training criterion, after adding extra learning machinery to convert the learned representation into supervised predictions for example, a linear classifier, then we can achieve classification.

D. *Important steps for simulation*

1. Background subtraction. By employing the method described in a Section 2.2, some background subtraction results are shown in Fig. 2.

Fig. 2. Background subtraction results for three frames in the sequence, where the green part in the first figure shows the occlusion of two human targets, which may cause miss detection; the red part in the second figure shows the appearance of another human target and the yellow part in the third figure shows the disappearance of a human target; as shown in the figure, there are noises in the background subtraction results, including the patches of noises and salt and pepper noise, which may cause false alarm in the multiple tracking work.

From the results we found there are many noise patches if we only use background subtraction to provide the human features, which may cause false alarms and miss detection. In order to avoid this, we use an DBNs classifier to classify the noise and the foreground part of the human targets since the features extracted from them are different, and we can also calculate the weight for each particle at the same time.

2. DBNs classifier. To train the DBNs classifier, firstly, we extract the colour and oriented gradient histograms of 34 human beings from the dataset, including different human targets, posture and colours of clothing to train the DBNs classifier. After this, the DBNs classifier can be used to obtain the weights for the predicted particles in the PHD filter.

3. PHD updating. After obtaining the particles and the weights of the surviving targets from Equations (8) and (9), the

particles of the new born targets are drawn from foreground objects using background subtraction, whose weights are given as

$$\tilde{w}_k^i = 1/J_k \tag{10}$$

where J_k is the number of new born targets at time k and '$\tilde{\cdot}$' denotes the value from estimation. Once the new set of observations is available, we can substitute the approximation of $\mathbf{D}_{k|k-1}(\mathbf{x}_k^i)$ into (3) and the weights of each particles are updated as

$$\tilde{w}_k^i = \left[p_M(\tilde{\mathbf{x}}_k^i) + \sum_{\forall z \in \mathbf{Z}_k} \frac{\psi_{k,z}(\tilde{\mathbf{x}}_k^i)}{\kappa_k(z) + C_k(z)} \right] \tilde{w}_{k-1}^i \tag{11}$$

where

$$C_k(z) = \sum_{j=1}^{N+J_k} \psi_{k,z}(\tilde{\mathbf{x}}_k^i) \tilde{w}_{k-1}^i \tag{12}$$

At each iteration k, J_k new particles are added to the old N particles for the new born targets; to limit the growth of the number of particles, a resampling step is performed after the update step, the algorithm for the resampling step is described as Algorithm 3 below [10]:

Algorithm 3 Resampling step of the particle PHD filter [10]

$\{\{\tilde{w}_k^i, \tilde{\mathbf{x}}_k^i\}_{i=1}^{N+J_k}\} \rightarrow \{w_k^i, \mathbf{x}_k^i\}_{i=1}^N$
Compute the target number at time k
$\hat{N}_k = \sum_{i=1}^{N+J_k} \tilde{w}_k^i$
Initialize the cumulative probability $c_1 = 0$
$c_i = c_{i-1} + \frac{\tilde{w}_k^{(i)}}{\hat{N}_k}, i = 2, ..., N + J_k$
Draw a starting point $\mu_1 \sim [0, N^{-1}]$
For $j = 1, ..., N$
$\mu_j = \mu_1 + N^{-1}(j-1)$
while $\mu_j > c_i$, $i = i + 1$. End while
$w_k^{(i)} = \tilde{w}_k^{(i)}$
$\mathbf{x}_k^{(i)} = \tilde{\mathbf{x}}_k^{(i)}$
End for
Rescale the weights by $\hat{N}_k$ to get $\{\mathbf{x}_k^{(i)}, \frac{\hat{N}_k}{N}\}$

III. Simulation Experiments

In order to evaluate the performance of the proposed robust particle PHD filter for multiple target tracking, we employed the dataset of the EC Funded CAVIAR project [14], Video EnterExitCrossingPaths1cor and EnterExitCrossingPaths1front which have 383 frames. There are four human beings appearing, occluding each other, and disappearing in the shopping mall environment. In order to evaluate the performance of our proposed PHD filter, we employ the optimal subpattern assignment (OSPA) [15] and the mean error (ME) for each target metric, the comparison result is shown as Fig. 3 and Table. 1.

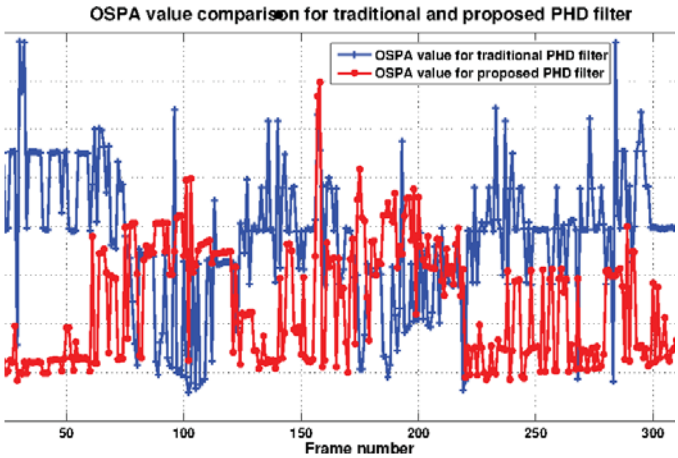

Fig. 3. Comparison of target OSPA value for scenario 1 between the proposed and the traditional PHD filter, where the blue line denotes the OSPA value for the traditional particle PHD filter and the red line denotes the OSPA value for the proposed particle PHD filter.

TABLE I
Comparisons of different tracking results

	Scenario 1		Scenario 2	
	PHD	DBNs-PHD	PHD	DBNs-PHD
OSPA(pixel)	38.25	11.45	34.54	20.62
ME (pixel)	21.68	12.16	19.87	14.61

From the comparison of both OSPA and OSPAMT results, it can be observed that our proposed PHD filter performs much better than the baseline PHD filter since, where in both scenario the OSPA value are much reduced, which means the ability of handling the false alarm and missing detection of our proposed PHD filter is much improved compared to the traditional PHD filter. The error for the PHD filter is mostly from the false alarm and miss detection caused by the noise from the background subtraction step, which can be mitigated by the DBN classifier in our proposed PHD filter; so from the comparison, we can deduce that our proposed PHD filter is more robust and can improve the accuracy of multiple human tracking work.

IV. Conclusion

We have presented a new robust particle PHD filter based on deep belief networks. In order to overcome the problem of false alarms and miss detection caused by the noise from background subtraction, we extracted colour and oriented gradient histogram features of human targets and used them with DBNs to calculate the weights for particles in the prediction step of the PHD filter. From the results we found the proposed method achieved is more accurate than the baseline method in estimating the target number. In future work, to further improve the accuracy of the tracking result, we will employ a variational Bayesian step to predict the parameters in the measurement model [16]; and to make the classifier adaptable to the change of the environment and targets. An online classifier [17] can also be used to improve the robustness of the multiple human tracking system.

REFERENCES

[1] D. E. Clark, "Multiple target tracking with the probability hypothesis density filter," PhD thesis, Heriot-Watt University, 2006.

[2] E. Maggio, M. Taj, and A. Cavallaro, "Efficient multi-target visual tracking using Random Finite Sets," *IEEE Transactions on Circuits and Systems for Video Technology*, 2008, vol. 10, pp. 1–12.

[3] D. E. Clark and J. Bell, "Data association for the PHD filter," *The Second International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, 2005, pp. 217–222.

[4] K. Panta, Ba-Ngu Vo, and S. Singh, "Novel data association schemes for the probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, 2007, vol. 43, pp. 556–570.

[5] E. Maggio and A. Cavallaro, *Video Tracking*, John Wiley and Sons, Ltd, 2011.

[6] R. Mahler, "A theoretical foundation for the Stein-Winter Probability Hypothesis Density (PHD) multitarget tracking approach," *MSS National Symposium on Sensor and Data Fusion, San Antonio, TX*, 2000, vol. 1.

[7] A. Doucet, B.N. Vo, C. Andrieu, and M. Davy, "Particle filtering for multi-target tracking and sensor management," *The International Conference on Informantion Fusion, Annaplis, MD*, 2002, vol. 1, pp. 474–481.

[8] S. Li, H. Wang, and T. Chai, "A t distribution based particle filter for target tracking," *Proceedings of the 2006 American Control Conference*, pp. 2191–2196, 2006.

[9] 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*, 2002, vol. 50, pp. 2454–2467.

[10] Y. Wang, J. Wu, A. A. Kassim, and W. Huang, "Tracking a variable number of human groups in video using probability hypothesis density," *in Proc. IEEE 18th of the International Conference on Pattern Recognition*, 2006.

[11] M. Yu, A. Rhuma, S. M. Naqvi, and J. Chambers, "Fall detection for the elderly in a smart room by using an enhanced one class support vector machine," *in Proc. IEEE 17th of the International Conference on Digital Signal Processing*, pp. 1–6, 2011.

[12] P. Feng, M. Yu, , S. M. Naqvi, and J. Chambers, "Deep learning for posture analysis in fall detection," *in Proc. IEEE 19th of the International Conference on Digital Signal Processing*, pp. 1–6, 2014.

[13] G.E. Hinton and R.R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, pp. 504 – 507, 2006.

[14] CAVIAR, "http://groups.inf.ed.ac.uk/vision/caviar, accessed Oeburary 2001," .

[15] B. Ristic, B. N. Vo, and D. Clark, "Performance evaluation of multi-target tracking using the OSPA metric," *in Proc. IEEE 13th Conference on Information Fusion*, pp. 1–7, 2010.

[16] D. Xu, C. Shen, and F. Shen, "A robust particle filtering algorithm with non-Gaussian measurement noise using Student's-t distribution," *IEEE Signal Processing Letters*, vol. 21, pp. 30–34, 2014.

[17] M. Yu, Y. Yu, A. Rhuma, S. M. Naqvi, L. Wang, and J. A. Chambers, "An online one class support vector machine-based peron-specific fall detection system for monitoring an elderly individual in a room environment," *IEEE Journal of Biomedical and Health Informatic*, vol. 17, pp. 1002–1014, 2013.