Particle PHD Filter Multiple Target Tracking in Sonar Images

Daniel Clark, Ioseba Tena Ruiz, Yvan Petillot and Judith Bell

***Abstract*— Two contrasting approaches for tracking multiple targets in multi-beam forward-looking sonar images are considered. The first approach is based on assigning a Kalman filter to each target and managing the measurements with gating and a measurement-to-track data association technique. The second approach uses the recently developed particle implementation of the multiple-target Probability Hypothesis Density (PHD) filter and a target state estimate-to-track data association technique. The two approaches are implemented and compared on both simulated sonar and real forward-looking sonar data obtained from an Autonomous Underwater Vehicle (AUV) and demonstrate that the PHD filter with data association compares well with traditional approaches for multiple target tracking.**

I. Introduction

Underwater vehicles can be fitted with a range of sensing equipment, including sonar and video. As the vehicles traverse through the water column, the sensing equipment is used to provide sequences of images of the scene. The sequences of data obtained from the underwater vehicles need to be interpreted to gain an understanding of the environment in which the vehicles are deployed.

One of the important tasks is to identify objects on the seabed or in the water column that would need to be avoided in path planning and navigation [1] [2]. In the case where the navigation of the vehicle is determined by its current environment and the path of the vehicle is determined by the incoming data, tracking algorithms are required so that obstacles can be avoided. Two approaches for tracking obstacles in sequences of sonar images are considered, the first of which uses an association technique to assign measurements to single-target filters and the second uses a multiple-target filter and association to enable track continuity. Measurements are found by pre-processing the sonar data to find potential objects based on their size and reflected intensity.

In the first algorithm, the obstacles are tracked by taking the measurements from the pre-processing step and applying a single-target filter to each target [3]. A nearest neighbour algorithm is used to associate new predictions to new obstacles in the scene. In subsequent frames the preprocessing phase is optimised by limiting the amount of processed data to those parts of the images where the obstacles are predicted to appear using gating techniques. New obstacles are also constantly being detected by less cumbersome segmentation methods such as subsampled images, smaller window smoothing, and simple thresholding. Results of the real-time multi-target tracking algorithm using forward-looking multi-beam data sets are presented.

The second approach uses the recently developed particle implementation of the PHD (Probability Hypothesis Density) filter [4] which has the ability to estimate the number of targets and their locations without the requirement of a data association technique or gating. Practical implementations of this technique on simulated data indicate that the PHD filter may perform better than conventional techniques under low SNR conditions [5].

Alternative techniques have been developed to enable track continuity using the particle PHD filter. The method developed by Lin [6], represents the PHD in a resolution cell to differentiate the peaks of the PHD posterior, and validation gating was used to determine the weights of the particles. Panta *et al.* [5], used the PHD filter for pre-filtering the measurements to remove clutter before using a Multiple Hypothesis Tracker.

In another recently developed technique [7], each particle is assigned a label according to its partition determined from the k-means algorithm. The particles are propagated with the prediction and update steps and k-means is used to repartition the particles. Partitions are associated to a target track if the majority of the particles in the new partition correspond to the particles propagated from a partition in the previous time-step.

The Estimate-to-track method was chosen here, since it is simpler, the clutter is low, and it could be applied to PHD filter implementations that do not rely on particle filtering techniques [8].

The tracking output of each of the algorithms is compared on real and simulated sonar data. The accuracy of the two tracking algorithms can be compared directly since the target locations are known in the simulated data. An initial comparison on real sonar data was given in [9].

II. Multiple Target Tracking

The usual approach to multiple-target tracking involves tracking each target independently with a single-target stochastic filter such as a Kalman filter, extended Kalman filter or particle filter. These filters require that the correct measurement is given to them to ensure that they are estimating the correct trajectory. The mechanism for distributing the correct measurement to each filter is called data association or, more specifically, measurement-to-track association. A recent article by Pulford [10] summarises the techniques in widespread use and classifies them into 35 different algorithmic types as well as providing a comprehensive literature survey of this area. All of these techniques rely on using single-target filters with the measurement-to-track association.

The authors are in the Ocean Systems Lab, Electrical and Computer Engineering, Heriot-Watt University, Edinburgh. dec30@cantab.net, i.tena_ruiz@hw.ac.uk, y.r.petillot@hw.ac.uk, j.bell@hw.ac.uk

This work was funded in part by QinetiQ.

An alternative to tracking using single-target filters and measurement-to-track association is to use multiple-target filtering to estimate the set of target states at each time-step and associate the estimates to tracks. One such technique uses the Probability Hypothesis Density (PHD) filter, developed by Mahler [11], to estimate the set of target states. The implementation of this technique used in this paper uses the particle filter implementation developed by Vo [4]. The set of target state estimates at each time-step is determined from the particles using clustering techniques. Techniques to enable continuity of the individual target tracks have been developed specifically for the particle PHD filter [7] [6] [5]. The technique used in this paper uses an Estimate-to-track technique [7], that associates the state estimates from the PHD filter to target tracks.

Tracking techniques previously used on forward-looking sonar include using optical flow [12] and concurrent mapping and localisation [2]. A previous implementation of the PHD filter demonstrated that the PHD filter could successfully estimate multiple targets in sonar [13], but track continuity was not maintained as no methods for data association were developed.

III. Tracking and Data Association

The task here is to estimate the number of targets and their locations at each point in time from a set of noisy measurements which may include false alarms. At each point in time t, we have a set of noisy measurements, $Z_t = \{z_{t,1}, \ldots, z_{t,m_t}\}$, where $z_{t,j}$ represents a single target measurement or false alarm and m_t is the number of observations at time t. From this set of measurements, we must estimate how many targets T_t there are and their set of locations, $X_t = \{x_{t,1}, \ldots, x_{t,T_t}\}$, where $x_{t,i}$ represents the state of an individual target and T_t is the number of targets at time t.

A. Tracking Model

A linear Gaussian dynamic model with the following state space model is used:

$$x_{t+1} = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{pmatrix} x_t + \begin{pmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{pmatrix} v_t, \quad (1)$$

and observation model:

$$z_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} x_t + w_t. \quad (2)$$

v_t and w_t are the process and measurement noises, respectively, and are uncorrelated. The matrices in the dynamic model will be referred to as F and Γ respectively. The observation model matrix will be denoted H.
The state vector is defined as the 2D position and velocity vector of the target:

$$x_t = \begin{pmatrix} \mathbf{x_t} & \dot{\mathbf{x}}_\mathbf{t} & \mathbf{y_t} & \dot{\mathbf{y}}_\mathbf{t} \end{pmatrix}^T. \quad (3)$$

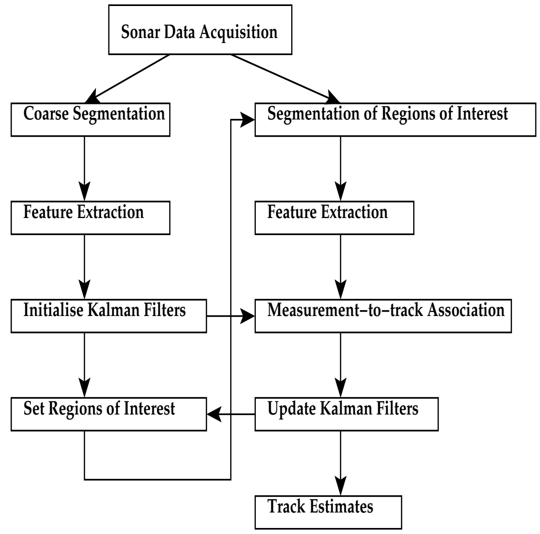

Fig. 1. Kalman Filter Tracking Procedure.

B. Tracking with Kalman filters

The first multiple tracking model assigns one Kalman filter per object and manages the measurements for each filter with a measurement-to-track data association technique. The Kalman filter has been chosen as the single-target filter, since it has been shown to be effective for multiple-target tracking with measurement-to-track association in sonar [3]. Other techniques, such as the Extended Kalman filter or particle filter could also have been used. The procedure used for the tracking algorithm is the Nearest Neighbour Standard Filter [14]. Our implementation is outlined in figure 1. After the sonar data is acquired, it is segmented and features are extracted. These features and the feature extraction process are described in Sections IV-C and IV-D. Depending on whether targets are expected in a region, a Kalman filter is either initialised or measurements are associated. Regions of interest are set to determine which areas to segment more carefully in subsequent iterations.

C. Tracking with the PHD filter

The PHD, or Probability Hypothesis Density, represents the first moment of a multi-target posterior distribution. This is a multi-modal distribution, where the target state estimates are found by determining the peaks which represent high expectation of there being a target. The implementation of the PHD filter we use here is taken from the Sequential Monte Carlo algorithm developed by Vo *et al.* [4].

The procedure for the tracking algorithm is given in figure 2. This can be compared with the procedure for the Kalman filter tracking (figure 1). The main differences are that all the extracted features are used directly as input to the filter and estimates are associated instead of measurements.

The PHD is approximated by a set of discrete samples, or particles, where each particle has an associated weight. These particles are projected into the next time step using the system equation. When the measurements are received, the weights of the particles are calculated based on a likelihood function,

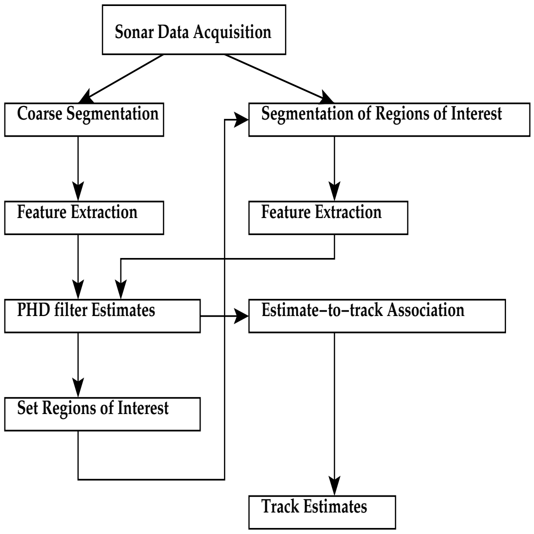

Fig. 2. PHD Filter Tracking Procedure.

which in our case represents proximity to a target. The set of particles and weights represent the PHD distribution and the number of targets is estimated by summing the weights of the particles. The number of particles adaptively changes to be proportional to the number of targets, with $N = 1000$ particles per target. The procedure for the PHD filter tracking algorithm with data association is as follows:

Initialise the algorithm by distributing $L_0 = \hat{T}_0 N$ particles, $\{\xi_0^{(1)}, \ldots, \xi_0^{(L_0)}\}$, across the state space, or field of view, where targets could be located. $\hat{T}_0$ is the expected number of targets at the start of the algorithm. Each particle, $\xi_0^{(i)}$, is assigned a weight, $\omega_{0|-1}^{(i)}$, which is initialised to $\hat{T}_0/N$. The state vectors containing the position and velocity components are randomly assigned within a range of expected values.

When the measurements, $Z_t = \{z_{t,1}, \ldots, z_{t,m_t}\}$, are received at time t, update the weights of the particles using the PHD filter data update equation [4],

$$\omega_{t|t}^{(i)} = \left[(1 - P_D) + \sum_{z_{t,j} \in Z_t} \frac{P_D g_t(z_{t,j}|\xi_t^{(i)})}{\lambda_t c_t(z_t) + P_D \omega_{t|t-1} \cdot g_t} \right] \omega_{t|t-1}^{(i)}, \quad (4)$$

where P_D is the probability of detection, $g_t(z_{t,i}|\xi_t^{(i)}) = \exp(-(1/2)(z_{t,i} - H\xi_t^{(i)})R^{-1}(z_{t,i} - H\xi_t^{(i)}))$ is the likelihood of observing measurement $z_{t,i}$ given particle $\xi_t^{(i)}$ based on the observation covariance matrix R. The dot product of the particle weight and likelihood vectors is given by $\omega_{t|t-1} \cdot g_t$. The expected number of clutter points is $\lambda_t = 1$ and c_t is the uniform distribution of these across the state space.

The particle set along with their weights, $\{(\xi_t^{(1)}, \omega_{t|t}^{(1)}), \ldots, (\xi_t^{(L_t)}, \omega_{t|t}^{(L_t)})\}$, is a discrete weighted approximation of the posterior PHD distribution. The estimated number of targets at time t, $\hat{T}_t$, is calculated by taking the sum of the weights.

Target state estimates are found from the weighted particle set by extracting $\hat{T}_t$ peaks of the PHD distribution using a clustering algorithm. The $k-$means algorithm has been used here as it has proved to be a computationally inexpensive and reasonably accurate means of determining target locations [15] (see subsection 1). An unweighted representation of the posterior distribution is obtained by resampling $N_{t+1} = \hat{T}_t N$ particles according to their weights calculated in the update step. The set of target state estimates are associated with targets in the previous time step using the estimate-to-track data association technique described in subsection 2.

Each particle $\xi_t^{(i)}$ is projected in the prediction step by the state equation,

$$\xi_{t+1}^{(i)} = F\xi_t^{(i)} + \Gamma_t v_t^{(i)}, \quad (5)$$

where $v_t^{(i)}$ is random noise drawn from probability distribution with system covariance matrix Q. In addition, M new-born particles are also introduced from the spontaneous birth model in anticipation of new targets entering the field of view.

Weights for the existing particles are computed,

$$\omega_{t+1|t}^{(i)} = P_S(\xi_t^{(i)})\omega_{t|t}^{(i)}, \quad (6)$$

and for the new particles,

$$\omega_{t+1|t}^{(i)} = P_B/M, \quad (7)$$

where P_S is the probability of survival and P_B is the probability of the birth of a target.

1) Target State Estimation: The k-means clustering algorithm takes a set of points, in this case the particles, and separates them into k partitions, $\{P_{t,1}, ..., P_{t,k}\}$, with means $\{m_{t,1}, \ldots, m_{t,k}\}$, called centres, such that the mean squared distance from each point to its nearest centre is minimised [16] [17]. The value of k is taken to be the nearest integer value to the sum of the PHD weights.

At each stage of the k-means algorithm, every centre point, $m_{t,j}$, is moved to the centroid of its partition $P_{t,j}$. Partition $P_{t,j}$ is updated by recomputing the distance from each point to its nearest centre. These steps are repeated until a convergence criterion is met. The means and covariances of the final partitions determine the state estimate and covariances.

The overall time-complexity of the PHD filter algorithm at each iteration is $O(|\hat{T}_t|Nn)$ [18], where n is the number of iterations in k-means, which is comparable to $\hat{T}_t$ independent particle filters. Although this has a higher complexity than the Kalman filter technique, this can be implemented in real-time.

2) Estimate-to-track Association: The data association technique which we use here is based on associating the target state estimates between frames [7]. The set of observations found by segmentation are given to the PHD filter algorithm and the tracks are updated using the procedure described below.

The particles are partitioned with k-means and, for each partition i, error covariances, $S_{t,i}$, are determined which define the validation gate,

$$V_{t,i}(\gamma) := \{x : [x - \hat{x}_{t,i}]^T (S_{t,i})^{-1} [x - \hat{x}_{t,i}] \leq \gamma\}, \quad (8)$$

where $\gamma = 4$ in our case, representing 2 standard deviations.

It is assumed here that the PHD filter has filtered out the false alarms so that all the estimates are treated as targets and there is only one association per target. The mechanism which we use to obtain the measurements filters out most of the clutter, so this assumption is reasonable. The method will associate estimates from the previous timestep, decide that a target has died or declare a new target. If a false alarm has not been filtered out, a new target track will be declared. The feature extraction process eliminates most of the false alarms in this case so the average number of clutter points λ in the PHD filter is low.

The association method proceeds as follows: The state equation is used to obtain predicted state estimate $\hat{x}_{t|t-1,j} := F\hat{x}_{t-1,j}$ for each estimate in the previous time step. This gives us the set of predicted state estimates, $\{\hat{x}_{t|t-1,1},..,\hat{x}_{t|t-1,\hat{T}_{t-1}}\}$.

The set of validated $1-1$ correspondences, β_t, between $\{\hat{x}_{t|t-1,1},..,\hat{x}_{t|t-1,\hat{T}_{t-1}}\}$ and $\{\hat{x}_{t|,1},..,\hat{x}_{t,\hat{T}_t}\}$ are evaluated using the validation gate. The best association is taken to be $b_t \in \beta_t$ such that $b_t = \arg\max_{b\in\beta_t} \sum_b \exp\{-1/2(\hat{x}_{t,i} - \hat{x}_{t|t-1,j})(S_{t,i})^{-1}(\hat{x}_{t,i} - \hat{x}_{t|t-1,j})\}$. New target tracks are declared for state estimates for which no association has been made.

IV. Implementation on Forward-Looking Sonar

The two multi-target tracking methods have been implemented for tracking obstacles in forward-looking sonar. In this section, the sonar system is presented and the method for obtaining the measurements of the obstacles is explained. Simulated sonar data and real forward-looking sonar data acquired from an underwater vehicle have been used. The segmentation and feature extraction methods for determining the measurements are the same for both the simulated and real sonar data.

A. Simulated Sonar Data

The tracking methods have been run on simulated forward-looking sonar data. The advantage of using simulated data is that it allows various realistic scenarios and trajectories to be created easily. The exact locations of the vehicle and objects are known, and thus the accuracy of the tracker can be determined. This will allow us to directly compare the results of the multi-target tracking algorithms to the ground truth data.

A sequence of forward-looking sonar images has been generated using the Sonar Simulator developed by Bell [19] which has the capability of modelling sonar in complex underwater terrain. An artificial seabed is modelled by a $100 \times 100m^2$ textured image, see figure 3 (middle). Spherical shaped objects of radius $0.5m$ have been placed on the seabed.

The specification of the sonar has been modelled to be as close to the sonar equipment used to provide the real data. The range of the sonar is 40m which scans a sector of 120 degrees, see figure 3 (left) for an example image.

A sinusoidal trajectory with added noise has been simulated for the sonar as though it were fitted onto an Autonomous Underwater Vehicle (AUV), see figure 3 (right) for the simulated trajectory with objects.

B. Real Sonar Data

The sequences of images were obtained from a forward looking multi-beam sonar which was fitted to an Autonomous Underwater Vehicle (AUV). The vehicle was travelling at a

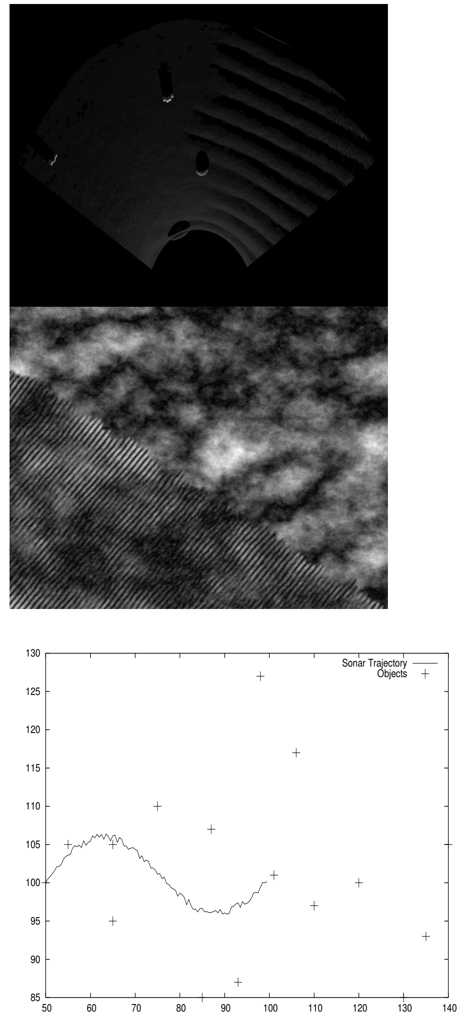

Fig. 3. Simulated Sonar Image (top). Artificial seabed (middle). Sonar Trajectory with Objects (bottom).

rate of approximately 1 knot over a region with stationary targets on the seabed. The sonar was mounted on the front of the AUV scanning forwards for a range of 40m and was angled towards the seabed. The sonar scanned an angular region of 120 degrees, using 120 beams each with a vertical beam width of 1 degree and a horizontal beam width of 40 degrees. The sonar had an operating frequency of 600 kHz.

C. Segmentation

Multi-beam sonar images can be very noisy, due to reverberation from the seabed, surface or water column and so need to be filtered if they are to be of use. The objects which we wish to track have a higher reflectivity property than the surrounding environment, and so the measurements can be determined by thresholding the sonar images on intensity. A two-layer segmentation has been used to identify areas of interest, the first of which uses a fast segmentation algorithm based on the intensity of the returned energy. The second layer more selectively segments regions where objects are expected based on previous knowledge.

To reduce the speckle noise, the images are first filtered. A mean filter was found to be effective at removing the noise and

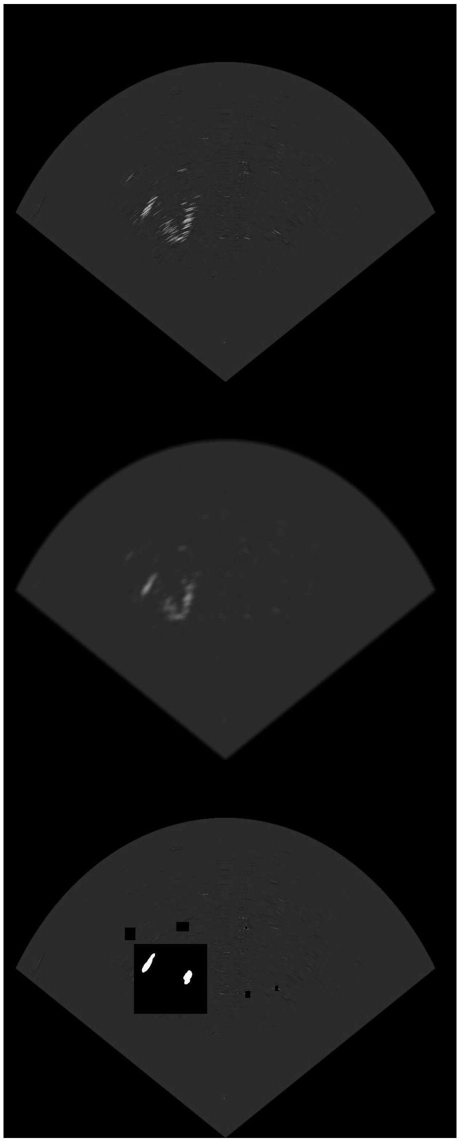

Fig. 4. Original sonar image (left). Image after filtering (middle). Resulting image after segmentation with regions of interests shown as the boxes and potential targets as the white segmented areas (right).

has a relatively cheap computational cost, see figure 4 (middle) for an example of an image after filtering. A threshold is then applied to identify regions with high reflected energy where there are potential objects.

A double threshold is applied as follows: first use an adaptive threshold to identify regions of high reflectivity and, for each region that is bigger than a given area, use a higher threshold to identify only the regions that have the highest returns. Neighbouring pixels are grouped together to form regions, the centroids of these regions are taken as the measurements which will be used as input to the tracking algorithms.

D. Feature Extraction

After the images have been segmented and the regions with high reflectivity have been identified, features of the potential targets can be found, and regions which are too small to be an obstacle are discarded. The features which we use for tracking in our application here are the centroid positions of the segmented regions. Other features have been used in the tracking such as the perimeter and area of the objects [3]; although, for simplicity we restrict ourselves to the positions of the targets. See figure 4 (right) for an example of a segmented image with regions of interest.

V. Results

This section presents the results for both of the tracking algorithms on real and simulated forward-looking sonar data. For the simulated data, the positions of the targets are known which enables us to compare the two methods. A direct comparison of the errors in the set of target state estimates from the true target locations for each of the algorithms is given using the Hausdorff distance [20]. The Hausdorff metric is used for measuring the distance between two sets. If the number of estimated targets is the same as the actual number of targets, the Hausdorff distance gives the error for the worst performing track. There are no false alarms in these examples, so this is what we are measuring.

A. Simulated Data

The simulated sonar data provides us with a ground truth with which we can compare the accuracy of the target estimation from each of the tracking methods. In this example, there is no clutter and the number of estimates is the same as the number of targets in view. Previous studies have demonstrated that the PHD filter can operate successfully in higher levels of clutter [18] [5] [6].

The true positions give the centres of the spherical objects in the image. The trackers, however, estimate the position of the centroid of the highlight of the object from the reflected acoustic energy and, therefore, introduce an inherent bias which is reflected in the results. Figures 6 and 8 show the images with tracking results superimposed.

Let X_t be the set of target states at time t and $\hat{X}_t$ be the set of estimated target states. We compare the performance of the algorithms using the L_2 pixel errors, $d(x_i,\hat{x}_j) = \sqrt{\left((x_i^1-\hat{x}_j^1)^2+(x_i^2-\hat{x}_j^2)^2\right)}$, between the estimates and true positions. For each iteration, the mean and maximum pixel errors have been calculated. The maximum error here is the same as the Hausdorff distance [20], $\max_{x_i\in X_t}\min_{\hat{x}_j\in\hat{X}_t} d(x_i,\hat{x}_j)$, which gives the tracking error in the worst case. The errors have been averaged over the length of the sequence and the table of results is given in figure 5.

The tracking techniques have given comparable performance in their ability to estimate the correct position and in the standard deviation of errors. The average error throughout the sequence was around 30 pixels with a standard deviation of 6 in both cases.

B. Real Data

The two tracking algorithms have been tested on the same sequence of sonar data and in this section a comparison of the different techniques is given.

Tracking Technique	**Kalman Filters**	**PHD filter**
Hausdorff Pixel Error	35.274	35.125
RMS Pixel Error	28.26	28.83
Pixel Standard Deviation	6.2199	6.5523

Fig. 5. Comparison of Errors.

The images in the sequence are 24-bit colour of size 1276×833 which was converted to grayscale. A mean filter of size 11×11 was used to reduce the impulse noise before segmenting the image by thresholding. The measurements obtained by this process are fed into the two tracking algorithms. Selected frames from this sequence are presented in figures 7 and 9.

In the first frame shown, there are three targets being tracked in each image, the two trajectories on the left are fairly similar. The target on the right has been tracked for longer with the PHD filter than the Kalman filter, although the ability to track without measurements has been removed in the case of the Kalman filter [3]. This was to enable a fairer comparison, since this functionality has not been used with the estimate-to-track PHD filter although could be incorporated into future implementations. We notice in the next two images, both techniques have similar target trajectories, although, the Kalman filter tracking is smoother.

VI. Conclusions

Two different approaches for tracking multiple targets in forward-scan sonar images have been implemented. The first approach uses a Kalman filter for each target and assigns measurements to each individual filter with a data association technique. The second approach uses the PHD filter to estimate the number of the targets and their locations at each time step and uses a recently developed association technique, which uses k-means and gating for associating target state estimates between iterations, to enable track continuity.

The first practical implementation of the PHD filter with data association has been demonstrated, showing comparable performance with Kalman filters for tracking multiple targets in real data. The tracking example presented here has a high SNR ratio which may account for the similar performance of the techniques.

The use of multiple target filtering algorithms for tracking applications is in its early stages and some new developments of the PHD filter are yet to be tested on real tracking problems. Future work will assess the ability of the PHD filter to track in environments with higher clutter.

Acknowledgements

The authors wish to thank Florida Atlantic University for providing the forward-looking sonar data.

References

[1] Y. Petillot, I. Tena Ruiz, and D. M. Lane. Underwater vehicle obstacle avoidance and path planning using a multi-beam forward looking sonar. *IEEE Journal of Oceanic Engineering, Vol. 26, No. 2, 240-251, April 2001.*

[2] I. Tena Ruiz, Y. Petillot, D. M. Lane, and C. Salson. Feature Extraction and Data Association for AUV Concurrent Mapping and Localisation. *Proceedings of the 2001 IEEE Conference on Robotics and Automation. Seoul, Korea. May 2001.*

[3] I. Tena Ruiz, Y. Petillot, D. Lane, and J. Bell. Tracking objects in underwater multibeam sonar images. *Motion Analysis and Tracking (Ref. No. 1999/103), IEE Colloquium on , 10 May 1999*, pages 11/1 – 11/7, 1999.

[4] B-N. Vo, S. Singh, and A. Doucet. Sequential Monte Carlo methods for Multi-target Filtering with Random Finite Sets. *IEEE Trans. Aerospace Elec. Systems*, 41, No.4:1224–1245, 2005.

[5] K. Panta, B. Vo, S. Singh, and A. Doucet. Probability hypothesis density filter versus multiple hypothesis tracking. *Proceedings of SPIE – Volume 5429 Signal Processing, Sensor Fusion, and Target Recognition XIII, Ivan Kadar, Editor, August 2004, pp. 284-295.*

[6] L. Lin. Parameter estimation and data association for multitarget tracking. *PhD Thesis, The University of Connecticut*, 2004.

[7] D. E. Clark and J. Bell. Data Association for the PHD Filter. *Proceedings ISSNIP, Melbourne. 5th-8th December 2005.*

[8] B. Vo and W. K. Ma. A closed form solution for the probability hypothesis density filter. *Proc. FUSION 2005.*

[9] D. E. Clark, I. Tena Ruiz, Y. Petillot, and J. Bell. Multiple target tracking and data association in sonar images. *The 2006 IEE Seminar on Target Tracking: Algorithms and Applications. Birmingham, UK. March 2006.*

[10] G. W. Pulford. Taxonomy of multiple target tracking methods. *IEE Radar, Sonar and Navigation, October 2005. Volume 152, Issue 5 , p. 291- 304.*

[11] R. Mahler. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39, No.4:1152–1178, 2003.

[12] D.M. Lane, M. Chantler, D. Y. Dai, and I. Tena Ruiz. Tracking and classification of multiple objects in multibeam sector scan sonar image sequences. *Underwater Technology, 1998. Proceedings of the 1998 International Symposium on , 15-17 April 1998*, pages 269–273.

[13] D.E. Clark and J. Bell. Bayesian Multiple Target Tracking in Forward Scan Sonar Images Using the PHD Filter. *IEE Radar, Sonar and Navigation, October 2005. Volume 152, Issue 5 , p. 327-334.*

[14] Y. Bar-Shalom and T.E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

[15] D. E. Clark, J. Bell, Y. de S.-Pern, and Y. Petillot. PHD Filter Multi-target Tracking in 3D Sonar. *IEEE Oceans Europe Conference, Brest June 2005. Volume 1, June 20-23, 2005 p265 - 270.*

[16] T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu. A Local Search Approximation Algorithm for k-means Clustering. *Proc. of the 18th Annual ACM Symp. on Computational Geometry*, pages 10–18, 2002.

[17] D. E. Clark and J. Bell. Multi-target state estimation and track continuity for the particle phd filter. *IEEE Transactions on Aerospace and Electronic Systems, provisionally accepted.*

[18] C. Haworth, Y. de Saint-Pern, D. Clark, E. Trucco, and Y. Petillot. Detection and tracking of multiple metallic objects in millimetre-wave images. *International Journal of Computer Vision, to appear.*

[19] J.M. Bell. A model for the simulation of side scan sonar. *PhD Thesis. Heriot-Watt University, 1995.*

[20] J. R. Hoffman and R. P. S. Mahler. Multitarget Miss Distance via Optimal Assignment. *IEEE Trans. Systems, Man. and Cybernetics.*, 34, No.3:327–336, 2004.

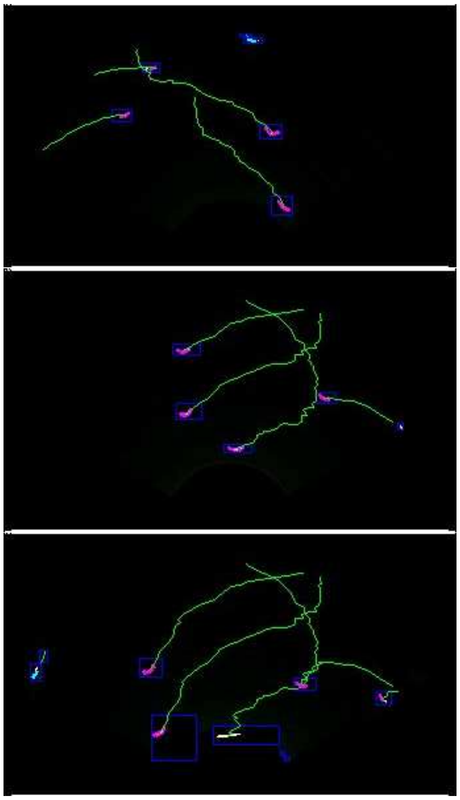

Fig. 6. Kalman filter tracking (simulated data).

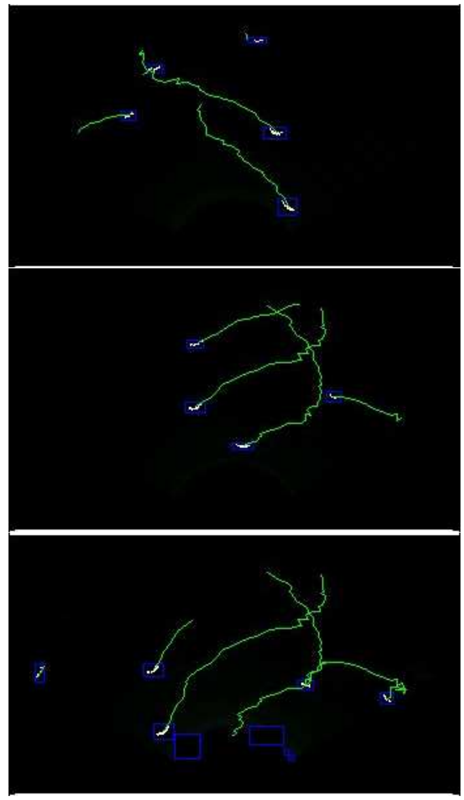

Fig. 8. PHD filter tracking (simulated data).

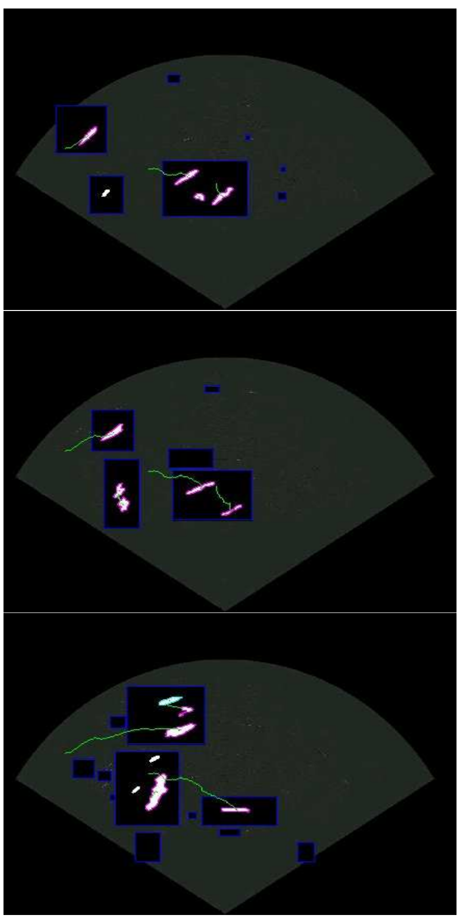

Fig. 7. Kalman filter tracking (real data).

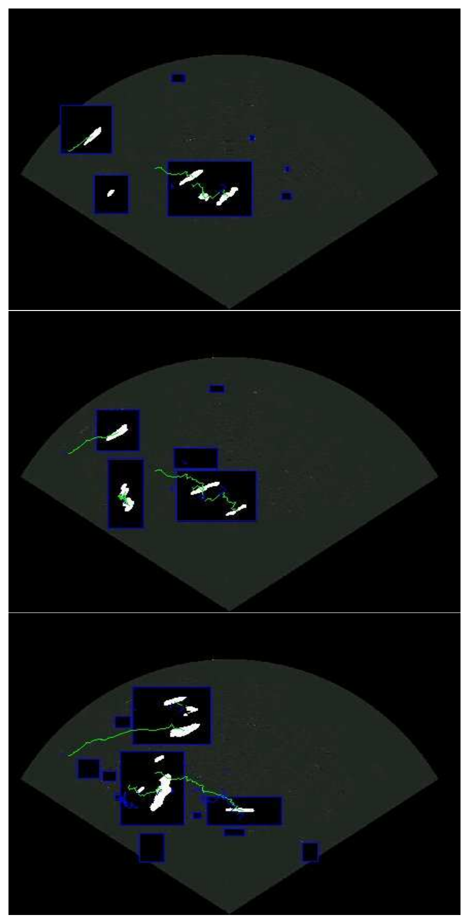

Fig. 9. PHD filter tracking (real data).