

Passive Multi-Object Localization and Tracking Using Bearing Data

Marek Schikora^{1,2}, Daniel Bender¹, Daniel Cremers² and Wolfgang Koch¹

¹Sensor Data and Information Fusion Dept.
Fraunhofer FKIE, Germany

²Computer Science Dept.
TU Munich, Germany

{[marek.schikora](mailto:marek.schikora@fkie.fraunhofer.de),[daniel.bender](mailto:daniel.bender@fkie.fraunhofer.de),[wolfgang.koch](mailto:wolfgang.koch@fkie.fraunhofer.de)}@fkie.fraunhofer.de daniel.cremers@in.tum.de

Abstract – This paper addresses the problem of localization and tracking multiple non-cooperative objects using only passive bearing sensor data. The challenges in this context lie in an unknown number of objects, false alarms and clutter measurements. To avoid the time consuming data association and data storage, an iterative approach, which only considers the sensor data from the actual time-step for an update of every object state, is preferable. Our approach to perform this is a Monte Carlo realization of a probability hypothesis density filter. In this context we use bearing data gained from an antenna or optical camera mounted on an airborne observer. Tests on simulated and real world scenarios show that our approach leads to a stable localization and tracking of multiple targets, even in the presence of clutter and misleading bearing measurements.

Keywords: Tracking, Localization, Bearing Data, Finite Set Statistics, Particle Filter

1 Introduction

Bearing data is produced by a huge number of different sensors, like antenna arrays, cameras or sonar systems. Every sensor of this kind measures 2D bearing data in 3D coordinates. The challenging task is now to invert the measurement step of every sensor in order to generate 3D coordinates from 2D data. In the field of Computer Vision this task is usually done through a set of multiple cameras, which are observing the same scene. This approach leads to stereo systems and 3D-reconstruction algorithms [4, 6, 11]. The drawback of this approach is that it is computationally very expensive to calculate 3D coordinates for the whole scene. In a task where only the 3D information of some objects is essential, like tracking and surveillance, the computed additional information is redundant. Using antenna arrays or sonar systems one traditionally only gets bearing data to process. In [17] the authors use multiple sensors to localize and track multiple objects.

In this work we focus on the task of passive localization and tracking of multiple non-cooperative objects. As sensor data input for our algorithm we use bearing data produced

only by a single sensor. To solve this task, we derive a general sensor model and scenario. The actual tracking and localization is done by sequential Monte Carlo implementation of a probability hypothesis density (PHD) filter. We use this iterative filter strategy to avoid the time consuming data association, as presented in [5], and so achieve better computational performance in comparison to the classical batch algorithms like maximum likelihood estimation [1]. Batch algorithms have been often applied in the case of a single object. In the multi-object case, batch algorithms can also be easily applied, when the objects can be distinguished, e.g. by different received frequencies using an antenna array as sensor. In this work we do not rely on such an assumption.

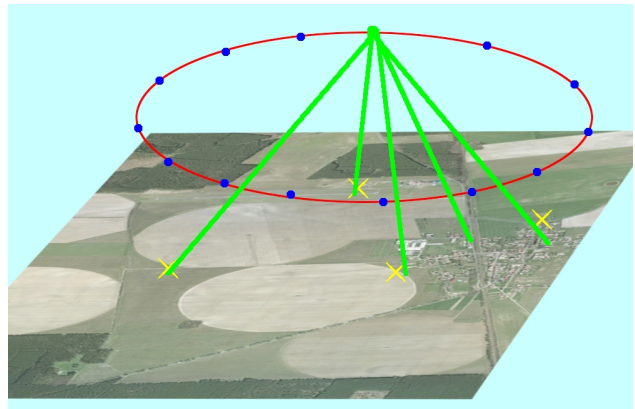


Figure 1: Typical scenario used in our simulations and real world experiments. A possible observer path is illustrated by the red ellipse, whereby the blue dots represent the discrete measurement points. The yellow crosses are objects of interest in this scene, which will be localized and tracked from the algorithm by bearing data. The latter is represented by green rays which point in the direction resulting from processing the sensor output.

2 Sensor and Target Model

The scenario considered in this work is the following: An observer with a single sensor (e.g. camera or antenna array) performs a flight maneuver (non-linear) over a region. Possible objects of interest are detected by some sensor processing. In a camera system this is done by applying an object detection strategy, e.g. [7, 13], to the images. Using an antenna array, a lot of direction finding algorithms exists, e.g. [3, 12, 15]. The result of this process is bearing data from the true objects, false alarms and clutter. This data is used as input for the proposed algorithm to localize and track multiple objects. A typical scenario can be seen in Figure 1. The proposed algorithm is general enough to be easily adapted to a different sensor-object geometry.

We assume that the objects move with respect to a linear constant velocity model, so that the object state is a six dimensional vector

$$\mathbf{x} = (x, y, z, v_x, v_y, v_z) \in \mathbb{R}^6, \quad (1)$$

with three entries for the position and three entries for the velocities in a cartesian coordinate system. A measurement is described by a two dimensional vector

$$\mathbf{z}_{\text{obs}} = (\alpha_{\text{obs}}, \varepsilon_{\text{obs}}) \in \mathbb{R}^2, \quad (2)$$

whereby $\alpha \in [-\pi, +\pi]$ as azimuth and $\varepsilon \in [0, \pi/2]$ as elevation in the local coordinate system of the observer. In order to localize and track a object over the time we need to transform this local measurements in a global coordinate system. This is done by the following three steps:

1. Transform to spherical coordinates:

$$u = \sin(\alpha_{\text{obs}}) \sin(\varepsilon_{\text{obs}}) \quad (3)$$

$$v = \cos(\alpha_{\text{obs}}) \sin(\varepsilon_{\text{obs}}). \quad (4)$$

2. Rotate the bearing ray ρ_{obs} according to the attitude of the observer

$$\rho_{\text{obs}} = (v, u, \sqrt{1 - u^2 - v^2})^T \quad (5)$$

$$\rho_{\text{global}} = (\mathbf{R}(\phi, \theta, \psi))^{-1} \rho_{\text{obs}}, \quad (6)$$

with ϕ, θ and ψ denoting the heading, pitch and roll of the observer and $\mathbf{R}(\phi, \theta, \psi)$ as the corresponding rotation matrix for this angles.

3. Calculate the measurements in a global coordinate system as follows:

$$\alpha_{\text{global}} = \arctan \left(\frac{\rho_{\text{global}}(2)}{\rho_{\text{global}}(1)} \right) \quad (7)$$

$$\varepsilon_{\text{global}} = \arctan \left(\frac{\sqrt{\rho_{\text{global}}^2(1) + \rho_{\text{global}}^2(2)}}{\rho_{\text{global}}(3)} \right), \quad (8)$$

leading to

$$\mathbf{z} = (\alpha_{\text{global}}, \varepsilon_{\text{global}}). \quad (9)$$

3 Localization and Tracking

Tracking multiple objects with bearing data remains a challenging problem. Classical works like [1] concentrate on the task of tracking a single object. Thereby a collection of bearing data is collected and then processed with a batch algorithm, leading to a maximum-likelihood style algorithm. In the case of multiple objects and clutter this approach can only be applied if a correct data association can be established. This is not always the case, especially when two objects emit signals with the same frequency. To avoid the data association problem we propose to localize and track multiple objects with an iterative filter approach, which is able to deal with this easily. Mahler proposed the random set theory as a theoretical framework for multi-sensor multi-object data fusion [10].

3.1 Finite Set Statistics

In a single-object system, the state and measurement at time k are represented as two random vectors of possibly different dimensions. These vectors evolve in time, but maintain their initial dimension. However, this is not the case in a multi-object system. Here the multi-object state and multi-object measurement are two collections of individual objects and measurements. The number of these may change over time and lead to another dimensions of the multi-object state and multi-object measurement. Furthermore, there exist no ordering for the elements of the multi-object state and measurement. Using the theory proposed in [10], the multi-object state and measurement are naturally represented as finite subsets X_k and Z_k defined as follows:

Let $N(k)$ be the number of objects, which are located at $\mathbf{x}_{k,1}, \dots, \mathbf{x}_{k,N(k)}$ in the single-object state space E_S , e.g. \mathbb{R}^d then,

$$X_k = \{\mathbf{x}_{k,1}, \dots, \mathbf{x}_{k,N(k)}\} \in \mathcal{F}(E_S) \quad (10)$$

is the multi-object state, where $\mathcal{F}(E_S)$ denotes the collection of all finite subsets of the space E_S . Analogous to this, we define the multi-object measurement

$$Z_k = \{\mathbf{z}_{k,1}, \dots, \mathbf{z}_{k,M(k)}\} \in \mathcal{F}(E_O), \quad (11)$$

assuming that at the time step k we have $M(k)$ measurements $\mathbf{z}_{k,1}, \dots, \mathbf{z}_{k,M(k)}$ in the single-object space E_O , which correspond to real targets and clutter. The sets X_k and Z_k are also called random finite sets, for which the first moment, or probability hypothesis density, is the analog of the expectation of a random vector. The integral value of the PHD over a given region in state space leads to the expected number of objects within this region. We define $D(\mathbf{x}_k | Z^k)$ as the PHD associated with the multi-object posterior $p(X_k | Z^k)$ at a time step k , with Z^k denoting the accumulated measurement from the time steps 1 to k . The PHD filter consists of two steps: prediction and update.

The prediction can be realized through the following equation:

$$D(\mathbf{x}_k|Z^{k-1}) = b(\mathbf{x}_k) + \int [p_s(\mathbf{x}_{k-1})p(\mathbf{x}_k|\mathbf{x}_{k-1}) + b(\mathbf{x}_k|\mathbf{x}_{k-1})] D(\mathbf{x}_{k-1}|Z^{k-1})d\mathbf{x}_{k-1}, \quad (12)$$

where $b(\mathbf{x}_k)$ denotes the intensity function of spontaneous birth of new objects, $b(\mathbf{x}_k|\mathbf{x}_{k-1})$ describes the intensity function of the random finite set of objects spawned from the previous state \mathbf{x}_{k-1} . $p_s(\mathbf{x}_{k-1})$ is the probability that the object still exists at the time step k given its previous state \mathbf{x}_{k-1} , and $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is the transition probability density of the individual objects. The update equation can be written as

$$D(\mathbf{x}_k|Z^k) \cong F(Z_k|\mathbf{x}_k)D(\mathbf{x}_k|Z^{k-1}), \quad (13)$$

$$F(Z_k|\mathbf{x}_k) = 1 - p_D(\mathbf{x}_k) + \sum_{\mathbf{z} \in Z_k} \frac{p_D(\mathbf{x}_k)p(\mathbf{z}|\mathbf{x}_k)}{\lambda c(\mathbf{z}) + \int p_D(\mathbf{x}_k)p(\mathbf{z}|\mathbf{x}_k)D(\mathbf{x}_k|Z^{k-1})d\mathbf{x}_k}, \quad (14)$$

with $p_D(\mathbf{x}_k)$ denoting the probability of the detection of the state \mathbf{x}_k . Furthermore, $p(\mathbf{z}|\mathbf{x}_k)$ is the measurement likelihood, $c(\mathbf{z})$ the probability distribution for every clutter point and λ is the average number of clutter points per scan.

3.2 Multi-Object Tracking

We implemented the theory described in the last section with a sequential Monte Carlo method, also known as particle filter. The first attempt to implement this technique for a tracking system, using the random finite set theory, was presented in [16]. This was only done for a simulated scenario and not optimized for the special tasks, which exist in visual tracking systems. Following the definition in Equation (1) the state of an individual object will be represented by $\mathbf{x}_k \in \mathbb{R}^6$. Each measurement $\mathbf{z}_k \in \mathbb{R}^2$ is represented analogous, c.f. (9). For the sake of simplicity, we assume that the object motion model of each target is linear with a constant velocity. With this, the object state prediction can be written as:

$$\mathbf{x}_k = \Phi(k, k-1)\mathbf{x}_{k-1} + \mathbf{s}_k, \quad (15)$$

with \mathbf{s}_k a zero mean Gaussian white process noise and

$$\Phi(k, k') = \begin{pmatrix} 1 & 0 & 0 & k-k' & 0 & 0 \\ 0 & 1 & 0 & 0 & k-k' & 0 \\ 0 & 0 & 1 & 0 & 0 & k-k' \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (16)$$

the transition matrix from time step k' to k .

Using the particle filter we can model the birth process $b(\mathbf{x}_k)$ as a uniformly distributed set of new particles with small weights. The likelihood function is given by:

$$p(\mathbf{z}|\mathbf{x}) = \frac{1}{2\pi|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{z} - f(\mathbf{x}))^T \Sigma (\mathbf{z} - f(\mathbf{x}))\right), \quad (17)$$

with Σ the covariance matrix of the measurement noise and

$$f(\mathbf{x}) = \begin{pmatrix} \arctan\left(\frac{\mathbf{x}(1) - \mathbf{x}_{\text{obs}}(1)}{\mathbf{x}(2) - \mathbf{x}_{\text{obs}}(2)}\right) \\ \frac{\pi}{2} + \arctan\left(\frac{\mathbf{x}(3) - \mathbf{x}_{\text{obs}}(3)}{\sqrt{(\mathbf{x}(1) - \mathbf{x}_{\text{obs}}(1))^2 + (\mathbf{x}(2) - \mathbf{x}_{\text{obs}}(2))^2}}\right) \end{pmatrix}. \quad (18)$$

Let L_k denote the number of particles at a time step k , and let J_k denote the number of the newly created particles at this time step. The sequential Monte Carlo (SMC) PHD filter can be summarized as follows:

At a given time step $k \geq 0$, let $\{\mathbf{x}_k^i, w_k^i\}_{i=1}^{L_k}$ represent a particle-based approximation of the PHD.

1. **Predict** the state of each particle pursuant to a given dynamic model. For $i = 1, \dots, L_{k-1}$, transform each particle to $\tilde{\mathbf{x}}_k^i$ according to (15) and keep the weights unchanged. Uniformly sample a new set of particles in the image and set

$$\tilde{w}_k^i = \frac{1}{J_k}, \text{ for } i = L_{k-1} + 1, \dots, L_{k-1} + J_k. \quad (19)$$

2. **Detect** possible objects in the scene. This step depends on the used sensor system. The resulting bearing measurements are formulated analogous to Equation (9).
3. **Update** the state of each particle by a new set of measurements. For each measurement $\mathbf{z} \in Z_k$ compute

$$C_k(\mathbf{z}) = \sum_{i=1}^{L_{k-1} + J_k} p_D(\tilde{\mathbf{x}}_k^i)p(\mathbf{z}|\tilde{\mathbf{x}}_k^i)\tilde{w}_k^i. \quad (20)$$

For all particles $i = 1, \dots, L_{k-1} + J_k$ update their weights

$$\hat{w}_k^i = \left(1 - p_D(\tilde{\mathbf{x}}_k^i) + \sum_{\mathbf{z} \in Z_k} \frac{p_D(\tilde{\mathbf{x}}_k^i)p(\mathbf{z}|\tilde{\mathbf{x}}_k^i)}{\lambda c(\mathbf{z}) + C_k(\mathbf{z})}\right) \tilde{w}_k^i. \quad (21)$$

4. **Resample** the particle set. Firstly the target number has to be estimated as follows:

$$\hat{N}_k = \sum_{i=1}^{L_{k-1} + J_k} \hat{w}_k^i. \quad (22)$$

Initialize the cumulative probability with $c_1 = 0$ and set

$$c_i = c_{i-1} + \frac{\hat{w}_k^i}{\hat{N}_k}, \text{ for } i = 2, \dots, L_{k-1} + J_k. \quad (23)$$

Draw a uniformly distributed starting point a_1 from the interval $[0, L_k^{-1}]$.

For $j = 1, \dots, L_k$,

$$a_j = a_1 + L_k^{-1} \cdot (j - 1) \quad (24)$$

while $a_j > c_i, i = i + 1$. End while.

$$\mathbf{x}_k^j = \tilde{\mathbf{x}}_k^i \quad (25)$$

$$w_k^j = L_k^{-1} \quad (26)$$

Rescale the weights by \hat{N}_k to get a new particle set

$$\left\{ \mathbf{x}_k^i, \frac{\hat{N}_k}{L_k} \right\}_{i=1}^{L_k}.$$

After every time step k we generate a particle cloud $\left\{ \mathbf{x}_k^i, \frac{\hat{N}_k}{L_k} \right\}_{i=1}^{L_k}$, which represents the PHD. To estimate the correct object states from this cloud we have to perform a clustering. The SMC-PHD filter estimates the number of objects for every time step, so it is possible to use a clustering technique, which requires the number of clusters, e.g. k -means clustering [8]. However the estimated object number from the PHD has a high variance [9]. To deal with this problem, we used in our experiments the adaptive resonance theory (ART) clustering [2], which estimates the number of clusters automatically, with a distance parameter as predefined user input. With this kind of clustering we are robust against estimation errors in the number of objects. In fact we only use a subset

$$S \subset \left\{ \mathbf{x}_k^i, w_k^i \right\}_{i=1}^{L_k}, \quad (27)$$

with

$$\left(\mathbf{x}_k^i, w_k^i \right) \in S \text{ if } w_k^i \geq \tau. \quad (28)$$

4 Results

In this section, we present localization and tracking results generated by our algorithm. We will demonstrate the efficiency of the algorithm on simulated data and compare our results to the Cramér-Rao lower bound (CRLB) of this problem. Following this, we will show real-world results achieved with two different sensor systems: a camera system and an antenna array system. For all experiments we used 5000 particles in the SMC-PHD Filter. The other parameters of the proposed strategy are sensor depended and can be estimated, if they are not known.

4.1 Simulation and Cramér-Rao lower bound

To judge a estimation algorithm it is important to know the maximum estimation accuracy, that can be achieved with the measurements. The Cramér-Rao lower bound (CRLB) provides a powerful lower bound for the estimation accuracy. The CRLB is the inverse of the Fisher information for a given scenario. In our case the CRLB is given by

$$\text{CRLB} = \mathbf{J}^{-1}. \quad (29)$$

The Fisher Information can be computed by the following equation:

$$\mathbf{J} = \frac{1}{\sigma_\alpha^2} \sum_{i=1}^{M(k)} \mathbf{v}_\alpha(k_i) \mathbf{v}_\alpha^T(k_i) + \frac{1}{\sigma_\varepsilon^2} \sum_{i=1}^{M(k)} \mathbf{v}_\varepsilon(k_i) \mathbf{v}_\varepsilon^T(k_i), \quad (30)$$

with

$$\mathbf{v}_\alpha(k_i) = \frac{1}{r(k_i) \cos(\varepsilon(k_i))} \Phi^T(k_i, k_r) \begin{pmatrix} \cos(\alpha(k_i)) \\ -\sin(\alpha(k_i)) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (31)$$

and

$$\mathbf{v}_\varepsilon(k_i) = \frac{1}{r(k_i)} \Phi^T(k_i, k_r) \begin{pmatrix} -\sin(\alpha(k_i)) \sin(\varepsilon(k_i)) \\ -\cos(\alpha(k_i)) - \sin(\varepsilon(k_i)) \\ \cos(\varepsilon(k_i)) \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (32)$$

Herein is σ_α and σ_ε the standard derivation for the measurement error in azimuth and elevation, respectively. k_i is the measurement time step and k_r is a reference time step. The vector $r(k_i)$ is the corresponding relative vector between a target and the observer for a time step k_i .

Analogous to the scenario presented in Figure 1 we have processed a Monte Carlo simulation with 1000 runs. Thereby an object was placed at a known position and with a known velocity. From this informations, we computed bearing data in azimuth and elevation, corrupted these true values by a gaussian white process noise and used it as an input for the filtering algorithm. Additionally we added clutter measurements and introduced a detection probability of 95 %. The estimated CRLB for the position information and the estimated end position of the object of each simulation can be seen in Figure 2. For a better legibility we only visualize the results for a single object. However the algorithm does not not how many objects are in the scene and estimates the object number correctly. The plotted CRLB ellipsoid represents the 3- σ range for a single object. With this simulation, we have shown that the variance of the results reaches the CRLB. Figure 2 illustrates that 99.7% of all results lie inside the ellipsoid, so that we can state that the proposed algorithm is asymptotic efficient for this scenario and produces practically no bias. These results can be also achieved for a multi-object scenario.

4.2 Real data

In this subsection we present localization and tracking results achieved with real data. As sensor platform we used a unmanned aerial vehicle (UAV). The UAV was equipped with a Global Positioning System (GPS) and an Inertial Navigation System (INS), so that at every time step we knew the position and the attitude information. In every scenario the UAV was flying at a altitude of circa 1000 meters above ground level with a meandering course. We had two sensor systems, that produce bearing data: a camera system and and a antenna array system. In the following the results achieved with both systems will be illustrated.

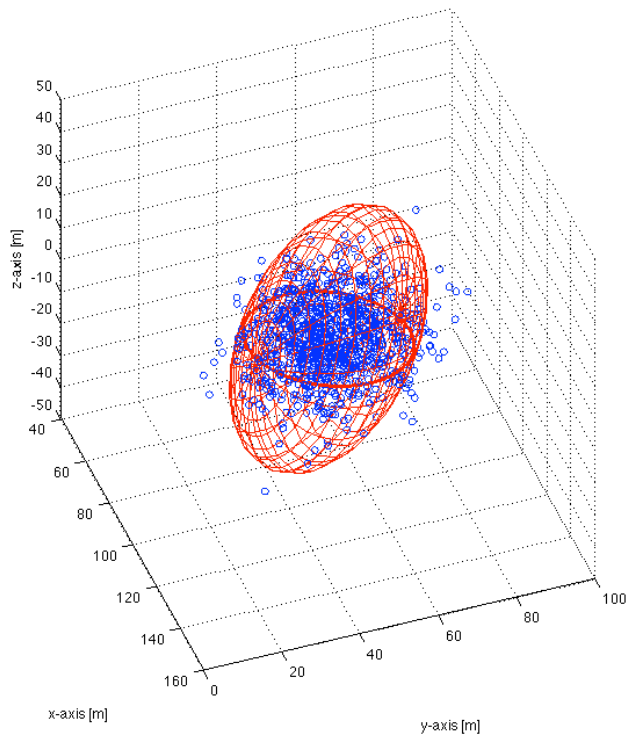


Figure 2: Result of Monte Carlo Simulation. Ellipse in red shows the $3\text{-}\sigma$ range CRLB for the final time step. Blue Points correspond to localization results of the proposed algorithm at the final time step. The true position of the object is $(100, 50, 0)^T$. Within the CRLB-ellipse lie 99.7% of all computed solutions.

4.2.1 Camera System

The camera was mounted beneath the UAV as a fixed down-looking high resolution system. The field of view of the camera used was 114 degree horizontal and 88 vertical. The detection of possible objects was done using the strategy presented in [13]. This detection procedure uses shape and color information to find objects in a image. For this experiment, we limited ourself to airplanes on the ground and cars. Once a object has been detected in the image plane local bearing data can be computed using the position and attitude information of the UAV. Further details on this computation can be found in [14]. Typical input images can be seen in Figure 3. For the car in Figure 3, we knew its exact position as ground truth, so we could compute uncorrupted bearing data. We compared the true bearing data with measured bearing, so we can illustrate in Figure 4 the bearing errors made by our system. These errors were mainly produced by the INS-System. In addition to these bearing errors, the detection scheme produced some clutter. We were able to localize and track all three objects in the scenario. The position estimation error over time for a target with ground truth



Figure 3: Input images produced by our camera system. Top row: scaled image used for processing. Bottom row: cut in original size on a car and two airplanes.

is illustrated in Figure 5.

As it can be easily recognized, the performance of the filter on real data improves over time. By tuning the sampling space even better results can be achieved. A possible way to do this is to limit the sampling area of particles in the z -axis. These results state that the performance we achieved with simulated data also holds for real data. In Figure 6, the evolution of the particle cloud for this scenario is presented. This figure clearly shows that the localization confidence of the filter grows with more measurements.

4.2.2 Antenna Array System

In addition to the optical sensor system we also tested the proposed algorithm with a three-element antenna array. This small antenna array is able to detect and compute bearing

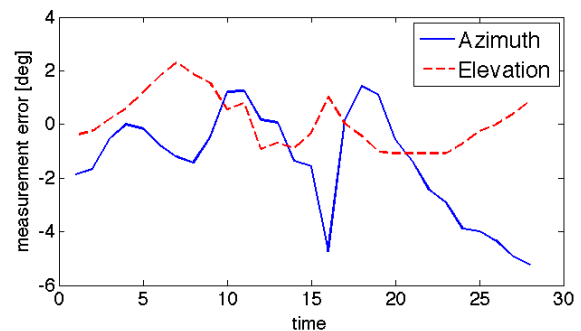


Figure 4: Measurement bearing errors for our camera system over time.

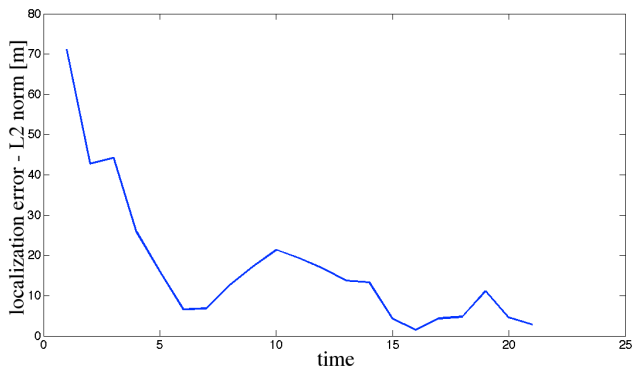


Figure 5: Localization and Tracking position error achieved with our camera system using the proposed SMC-PHD filter.

data for satellite telephone uplink communication. In order to obtain bearing data from the received signals we used the strategy proposed in [15]. The challenge in this data lies in its non-gaussian error distribution and additional grating lobe effects, which lead to high errors in the estimated bearing, c.f Figure 7. As in the previous case we had one object as ground truth to validate the localization and tracking results. The achieved localization accuracy can be seen in Figure 8. This experiment makes clear that even with strongly corrupted input data very good results can be achieved using the proposed non-linear filter technique.

5 Conclusions

This paper presents a novel approach to localize and track an unknown number of objects in a non-cooperative scenario. As input for the proposed algorithm bearing data is used. The sequential Monte Carlo implementation of a probability hypothesis density filter is used to deal with the non-linear measurement equation and the data association problem in the multi-object scenario. We have shown in a Monte Carlo simulation, that this filter is asymptotic efficient for bearing data, because it reaches the Cramér-Rao lower bound. In addition to the simulation results, we have presented results achieved with real data produced by a camera system and a small antenna array. Also in the real world scenario we were able to localize and track multiple objects, even in the case of clutter and high bearing errors.

References

- [1] K. Becker. Target Motion Analysis (TMA). In S. Stergiou, editor, *Advanced Signal Processing Handbook*, chapter 9, pages 284–301. CRC Press, New York, NY, 2001.
- [2] G.A. Carpenter and S. Grossberg. ART 2: Self-organizing stable category recognition codes for analog input patterns. *Applied Optics*, 26(23):4919–4930, 1987.

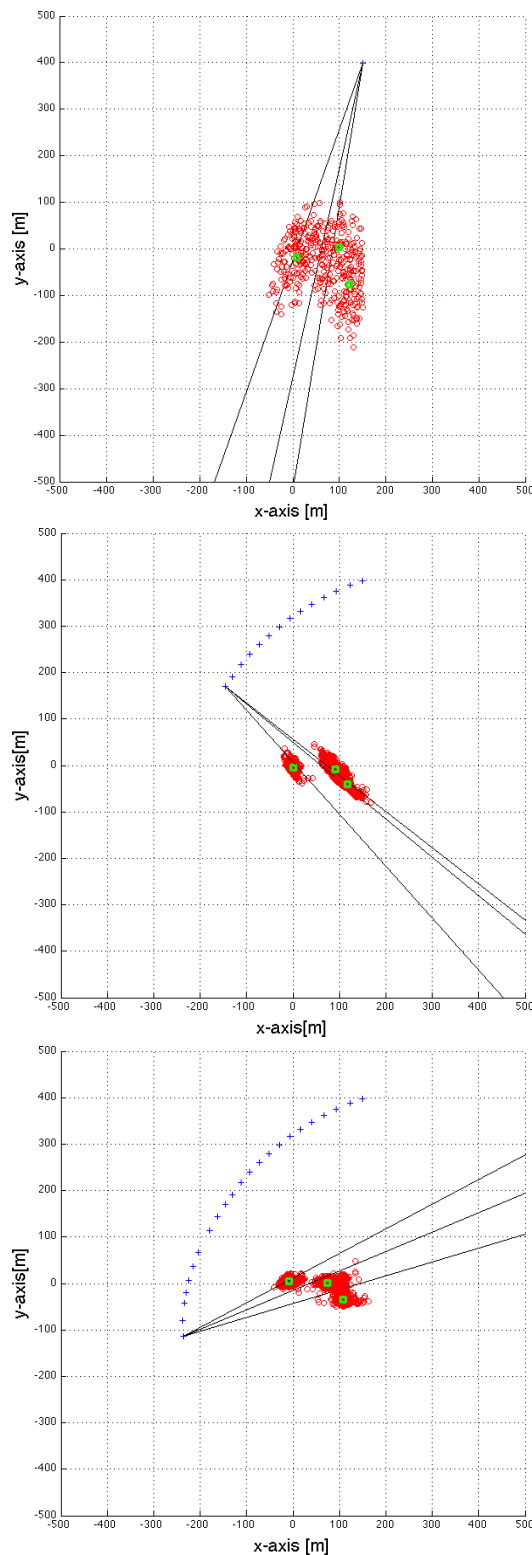


Figure 6: Particle set evolution for a camera system at different time steps. The blue crosses represent the observer position at a bearing measurement. The corresponding bearing is represented by black rays. The individual particles are illustrated as red circles, whereby the localization is displayed as green squares. For a better perspicuity this Figure only shows the top view of the 3D scenario.

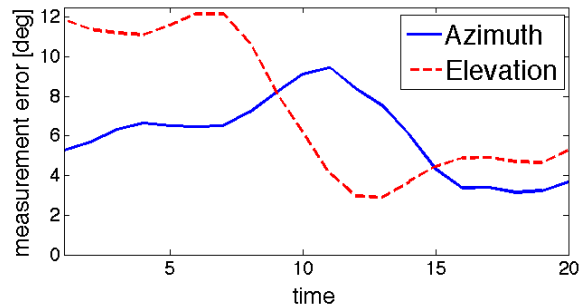


Figure 7: Measurement bearing errors for our antenna array over time.

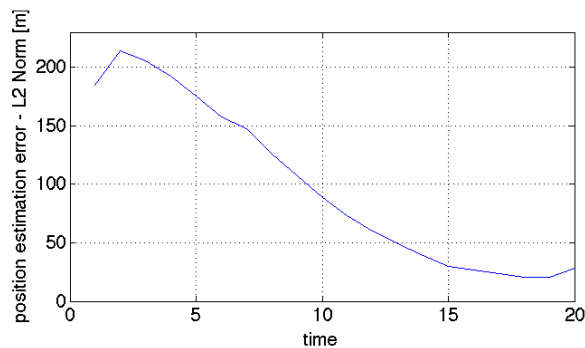


Figure 8: Localization and Tracking position error achieved with our antenna array using the proposed SMC-PHD filter.

- [3] L. C. Godara. Application of antenna arrays to mobile communications, Part II: Beam-forming and direction-of-arrival considerations. *Proc. IEEE*, 85:1195–1245, August 1997.
- [4] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- [5] C. Hue, J.-P. Le Cadre, and P. Pérez. Sequential monte carlo methods for multiple target tracking and data fusion. *IEEE Trans. Signal Processing*, 50(2):309–325, 2002.
- [6] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR*, 2006.
- [7] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision Special Issue on Learning for Recognition and Recognition for Learning*, 77(1–3):259–289, 2008.
- [8] S.P. Lloyd. Least squares quantization in pcm. *IEEE Trans. Inform. Theory*, 28(2):129–137, 1982.
- [9] R. Mahler. PHD filters of higher order in target number. *IEEE Trans. Aerosp. Electron. Syst.*, 43(4):1523–1543, 2007.
- [10] R.P.S. Mahler. Multitarget bayes filtering via first-order multitargets moments. *IEEE Trans. Aerosp. Electron. Syst.*, 39(4):1152–1178, 2003.
- [11] T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers. A convex formulation of continuous multi-label problems. In *European Conference on Computer Vision (ECCV)*, Marseille, France, October 2008.
- [12] S. V. Schell and W. A. Gardner. High-resolution direction finding. In N. K. Bose and C. R. Rao, editors, *Handbook of Statistics, Signal Processing and its Applications*, volume 10, chapter 18, pages 755–817. 1993.
- [13] M. Schikora. Global optimal multiple object detection using the fusion of shape and color information. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, August 2009.
- [14] M. Schikora, M. Haege, E. Ruthotto, and K. Wild. A convex formulation for color image segmentation in the context of passive emitter localization. In *12th International Conference on Information Fusion (FUSION)*, July 2009.
- [15] R. O. Schmidt. Multiple emitter location and signal parameter estimation. *Proc. RADC Spectrum Estimation Workshop, Griffith AFB*, pages 243–258, 1979.
- [16] B.-N. Vo, S. Singh, and A. Doucet. Sequential Monte Carlo methods for multi-target filtering with random finite sets. *IEEE Trans. Aerosp. Electron. Syst.*, 41(4):1224–1245, 2005.
- [17] H. Zhang, Z. Jing, and s. Hu. Localization of multiple emitters based on the sequential phd filter. *IEEE Trans. Signal Processing*, 90(1):34–43, 2010.