GNSS-free navigation

Resilient and accurate vessel positioning with radar and digital elevation models

Today's vessels rely heavily on satellite-based navigation systems for safe operations. These systems, however, are vulnerable to interference, jamming and spoofing. There is another solution: providing absolute positioning of the vessel based on marine radar scans.

JONATAN OLOFSSON
Department of Automatic Control
Linköping University

GUSTAF HENDEBY
Department of Automatic Control
Linköping University

FREDRIK GUSTAFSSON
Department of Automatic Control
Linköping University

DERAN MAAS
Principal Scientist
ABB Corporate Research

STEFANO MARANÒ
Senior Research Scientist
ABB Future Labs
Switzerland

Positioning based on GNSS (Global Navigation Satellite Systems) data has revolutionized the localization of maritime vessels. However, the underlying technology is sensitive to disturbance and spoofing [1–4], and there is growing concern over industry dependence on this single technology and its external transmitter-based approach.

To increase resilience against disturbance and adversarial actions, it is important to consider positioning solutions that do not rely on external technology. This requires mapping of onboard sensor data to a global position, either as prior knowledge or by building a map on the go. The latter is known as Simultaneous Localization And Mapping (SLAM), where a family of algorithms uses sensor data to build a map in real time and position the sensor within the map. Compared to using a Digital Elevation Model (DEM)-based model, SLAM could have the benefit of being less sensitive to the detailedness of underlying data while also adapting to changes more quickly. SLAM has been previously applied to radar data in a maritime setting [5], but as there is no geographical reference to tie the SLAM map to the physical world, a SLAM algorithm would not be able to provide global positioning without additional localization sensors. Even with good initialization, SLAM is prone to drift unless combined with known global reference points. In this paper, we propose the use of DEMs, for the reasons outlined below.

Most importantly, elevation maps can generate more accurate predictions of the radar image. For instance, steep coastlines will give a much more distinct and larger radar return signal than a shallow beach. Moreover, hills and slopes further away from the coastline will also reflect the radar signal, which can explain radar echoes that cannot be predicted from a sea chart. Hence, a DEM model better predicts the multiple responses of each radar beam. Furthermore,

- Sea charts generally only provide a binary map of sea versus not sea. Hence, only one radar response can be modeled for each radar beam, representing the closest shoreline.
- Coastlines are not static objects; they change over time. Elevation maps generated by satellites can reflect these changes faster, and in a more accurate manner than traditional sea charts.
- Elevation maps are global, while sea charts are most accurate in densely trafficked waters.

In this paper we describe a system capable of localization in a known – but not necessarily previously visited – environment through the matching of DEMs with radar scans. The matching is performed using a particle filter utilizing a custom measurement model to model the expected radar response based on the DEM. This gives a method of localization that is resilient to interference, spoofing, and jamming.

The system described in this paper is developed with, and tested against, recorded data from the ABB Marine & Ports sensor suite installed on the Suomenlinna II ferry in Helsinki harbor [6] in Finland. The data includes, but is not limited to, radar, electro-optical (EO) cameras, GNSS, inertial sensors, and a digital compass.

The outline of the paper is as follows. In Section II, the background theory is presented along with an introduction of the data with which the implementation was made and tested. In Section III, the measurement model of the radar is derived, and a likelihood function is proposed based on the matching between modeled and measured radar responses. The implementation used for the evaluation of the proposed method is described in detail in Section IV. This is then used to demonstrate a real tracking scenario in Section V. Section VI concludes the paper.

Background

In this section we review the background theory used in the research presented in this paper.

A. FILTERING THEORY

The basis for modern positioning is Bayesian state-space based filtering, in which the state $\boldsymbol{x}_k$ at time k of the studied system is inferred from measurements $\boldsymbol{z}_{(1:k)} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_k\}$ using the recursion

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k}) = \frac{p(\boldsymbol{z}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1})}{p(\boldsymbol{z}_k|\boldsymbol{z}_{1:k-1})} \tag{1a}$$

$$p(\boldsymbol{x}_k|\boldsymbol{z}_{1:k-1}) = \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{1:k-1})d\boldsymbol{x}_{k-1}\,. \tag{1b}$$

The commonly known Kalman Filter (KF) [7] solves this recursion analytically for the linear Gaussian case. The Extended Kalman Filter (EKF) [8, 9], which linearizes the system around the current estimate, has been successfully used to solve a more general class of problems.

The Particle Filter (PF) [10] is a sequential Monte Carlo method to solve the Bayesian recursion. An overview of the method is presented here; see [11] for a more thorough description in the context of filtering.

The idea behind the PF is to use a number of random samples and weights $\boldsymbol{x}_k^i$ and w_k^i to represent the sought distribution. This yields the approximation

$$p(\boldsymbol{x}_k|\,\cdot) = \sum_{i=1}^{N_p} w_k^i\, \delta_{\boldsymbol{x}_k^i}(\boldsymbol{x}_k), \tag{2}$$

where N_p is the number of samples, or particles, used in the approximation, all weights are positive $w_k^i > 0$, $\sum_i w_k^i = 1$ and $\delta_{\boldsymbol{x}_k^i}(\boldsymbol{x}_k)$ is the Dirac delta function.

The particle filter then updates the particles and weights over time to reflect the current information. In its basic form, also called the Bootstrap filter, the filter boils down to the following steps all performed for $i = 1, \ldots, N_p$:

- Initialization: Draw N_p samples from the initial distribution, all with the same weight:

$$\boldsymbol{x}_0^i \sim p(\boldsymbol{x}_0), \qquad w_0^i = \frac{1}{N_p}. \tag{3}$$

- Prediction: Based on the expected motion, draw N_p new particles to represent the state in the next time instance:

$$\boldsymbol{x}_k^i \sim p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}), \qquad w_k^i = w_{k-1}^i \tag{4}$$

- Measurement correction: Reweigh the particles based on how well they fit the measurements:

$$w_k^i \propto p\big(\boldsymbol{z}_k|\boldsymbol{x}_k^i\big)w_{k-1}^i \tag{5}$$

such that $\sum_{i=1}^{N_p} w_k^i = 1$

- Resample: To avoid particle degeneration i.e. all but one particle lose all their weight and importance, new particles are drawn from the old set with probability w_k^i, with replacement.

There are two differences between the particle filter and the Kalman filter that makes the particle filter suitable for the studied application as outlined below.

Whereas the Kalman filter is limited to represent the state with a single mean and a covariance, the particle representation is more flexible allowing for several possible hypothesis to be considered simultaneously. Hence, resolving ambiguities in the estimated location can be delayed until more information is made available.

The particle filter does not require an explicit measurement equation where the measurement is a function of the current state and noise, as the Kalman filter does. This will prove essential when incorporating the DEM in the measurement.

B. MARINE RADAR

Marine radar is today an indispensable tool for situational awareness on commercial ships, in particular in low-visibility environments. The marine radar instrument measures electromagnetic (EM) reflectivity of its surroundings by active illumination with EM waves. Rotating around an axis, it yields samples at given angles. The raw radar returns have to undergo significant signal processing to make the relevant information accessible as imagery or individual detections [12]. The result is the sum of all reflections from the illuminated areas, as well as unwanted noise and interference phenomena – known as speckle.

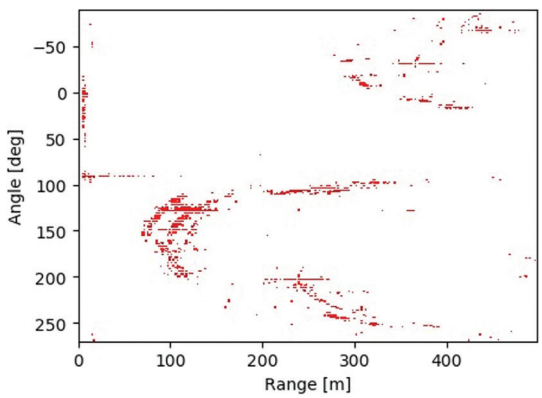


Figure 1: Radar response is sampled in polar coordinates centered at the radar

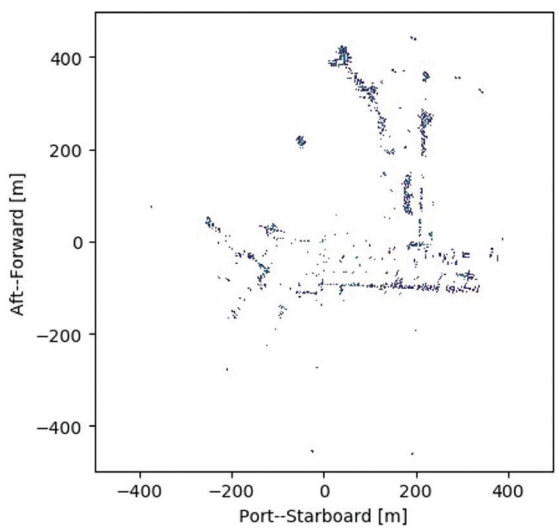


Figure 2: When the radar response is converted to Cartesian coordinates, the shape of the harbor becomes apparent to the human eye

The returned signals strongly depend on the properties of the object causing the reflection, such as its reflectivity and its orientation in relation to the radar source. These properties are summarized in the Radar Cross-Section (RCS) [12], which is a measure of the object's detectability. The return signal power is also proportional to the inverse fourth power of the distance to the target, so limiting the range of the radar.

Returns from an entire radar revolution yields a scan. In implementations, this is often stored as a matrix with rows and columns in accordance with angular and range resolution. However, when thresholded, this representation can be sparsified to yield individual measurements – reports – as 2D-points $\boldsymbol{z}^c$ and their intensities, $\imath$,

$$\boldsymbol{z} = (\boldsymbol{z}^c, \imath) \in \mathbb{R}^2 \times \mathbb{R}. \tag{6}$$

In this paper, $\boldsymbol{z}^c$ is defined in Cartesian coordinates centered around the ship radar. In each scan $\mathbb{Z}$, $N_{\mathbb{Z}}$ reports are detected.

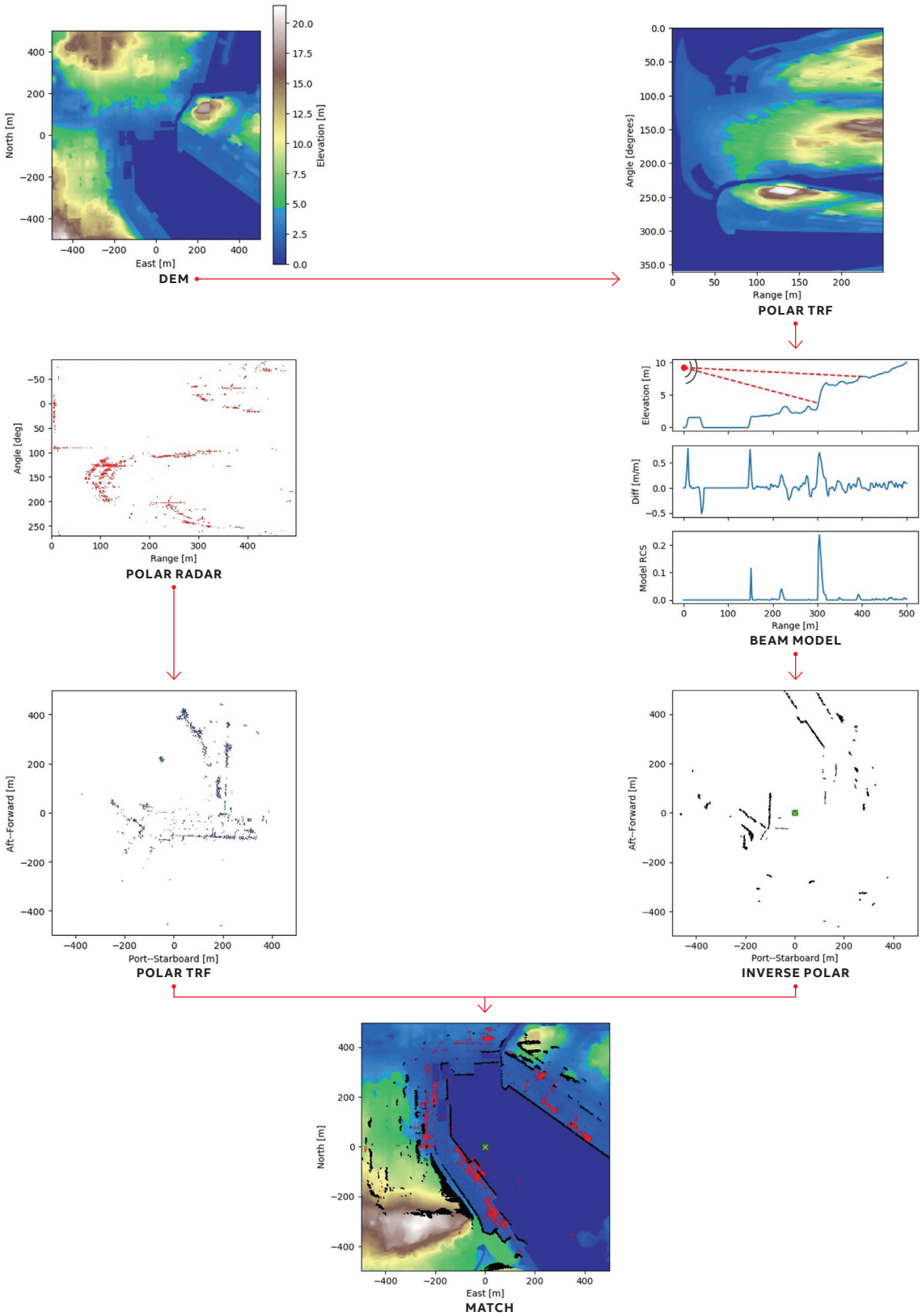
The radar's polar response is exemplified in Figure 1, with its Cartesian counterpart in Figure 2.

Measurement model

This section outlines how the radar response is modeled from the DEM data, as needed to use the PF. For simplicity, only the position of the radar detections is considered, not the intensity. The model is separated in two parts; the first part models what radar detections

$$\hat{\mathbb{Z}}(\boldsymbol{x}) = \mathrm{h}(\boldsymbol{x}). \tag{7}$$

to expect given specific position and orientation, using a radar response model $h(\boldsymbol{x})$. The second part in which the likelihood $p(\mathbb{Z}|\boldsymbol{x}) = p\left(\mathbb{Z}\middle|\hat{\mathbb{Z}}(\boldsymbol{x})\right)$ is derived. The former step generates potential radar responses, whereas the latter in essence



—
Figure 3: Measurement model pipeline

defines a probabilistic distance between two sets. In the paper, the orientation of the radar is assumed as known, as measured by e.g. the digital compass. If the orientation is not known, the state $\boldsymbol{x}$ could easily be extended to estimate the orientation.

The full pipeline of the resulting algorithm is illustrated in Figure 3 for a single particle position. The left column of the pipeline and the matching is then repeated for each particle in the particle filter.

A. RADAR RESPONSE MODEL

The radar response model $h(\boldsymbol{x})$ proposed in this paper bases its reports on the elevation surrounding the vessel, as given by a DEM. As radar measurements are taken in a radial fashion, we start by discussing how reports can be extracted for each individual angle, based on the elevation along a virtual beam radiating at an angle ψ from the radar, $e_\psi(r)$. Based on this elevation, we model the RCS at a given range. This single-beam model is then applied in parallel to all angles reported by the radar.

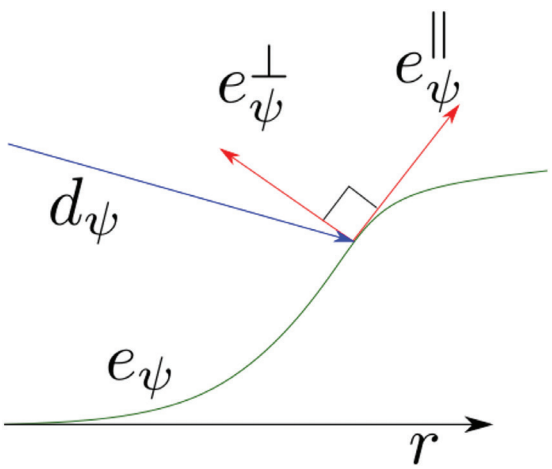


Figure 4: Here, the relevant quantities of the radar reflection model are illustrated, with $d_\psi(r)$ being the vector originating in the radar ending in the reflecting point on the surface, and $e_\psi^\perp(r)$ the surface normal in the reflecting point

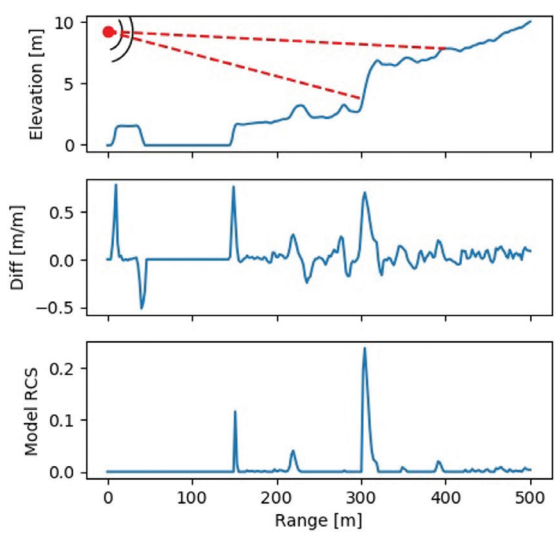


Figure 5: The model of the radar response in a single given direction is primarily a function of the incidence angle between the radar and the ground at range r. Here, σ_{thres} =0

The RCS in a given direction ψ and range r is approximated to be proportional to the area of the reflecting surface projected onto the plane perpendicular to the radar beam. This approximation makes it straightforward to compute $\hat{\sigma}'_\psi$. Let $\boldsymbol{d}_\psi(r)$ be the vector originating in the radar ending in the reflecting point on the surface, and $\boldsymbol{e}_\psi^\perp(r)$ the surface normal in the reflecting point. Then, as illustrated in Figure 4, the RCS follows from linear algebra as

$$\hat{\sigma}'_\psi = \frac{-\langle \boldsymbol{d}_\psi(r), \boldsymbol{e}_\psi^\perp(r) \rangle}{\|\boldsymbol{d}_\psi(r)\|}, \tag{8}$$

where $\langle \cdot, \cdot \rangle$ represents the scalar product and $\|\cdot\|$ the two-norm.

To further model the radar's received response, the shadowing effect of the landscape is considered. To simplify the line-of-sight computations, all points along the direction of the ray are assumed to be visible if they are located higher than any preceding point, otherwise not. Hence, the final

RCS becomes

$$\hat{\sigma}_\psi(r) = \begin{cases} \hat{\sigma}'_\psi(r) & \text{if } e_\psi(r) < \max_{r' \in [0,r)} e_\psi(r') \\ 0 & \text{otherwise} \end{cases}. \tag{9}$$

This process is illustrated in Figure 5.

In the final step all points with sufficiently large RCS are identified as part of the report set. A threshold σ_{thres} is applied to $\hat{\sigma}_\psi(r)$ to obtain the set of modeled polar reports,

$$\hat{\mathbb{Z}}^p(\boldsymbol{x}) = \bigcup_\psi \{(r, \psi) | \hat{\sigma}_\psi(r) \geq \sigma_{\text{thres}}, r \leq r_{\max}\}, \tag{10}$$

where $r_{\max}$ is the maximum range of the radar. This polar representation is then, through inverse polar transform, converted to Cartesian coordinates to give $\hat{\mathbb{Z}}(\boldsymbol{x})$ i.e. $h(\boldsymbol{x})$.

B. POINT CLOUD MATCHING

With the measured and modeled radar response available, $\mathbb{Z}$ and $\hat{\mathbb{Z}}(\boldsymbol{x})$, respectively, a measure of their correspondence can be created to provide the measurement likelihood needed in the PF measurement update (5).

Defining the difference between two point clouds is nontrivial. From the literature of point-cloud registration, there are a number of error measures as summarized in e.g. [13]. One of the most basic measures is the mean (sometimes squared) distance between associated points. With unknown point-to-point association, a hypothetical association needs to be established to calculate the measure. An intuitive and, importantly, computationally efficient method of association is the selection of the nearest neighbor of each point, from the other point set. Below, $\mathrm{NN}\left(\boldsymbol{z}_{k,j}^{c}, \hat{\mathbb{Z}}(\boldsymbol{x})\right)$ is the nearest neighbor of $\boldsymbol{z}_{k,j}^{c}$ in $\hat{\mathbb{Z}}(\boldsymbol{x})$, by Euclidean norm.

Based on the mean error, the following likelihood function is proposed:

$$p_{\text{radar}}\left(\mathbb{Z}_k \middle| \hat{\mathbb{Z}}(\boldsymbol{x}_k)\right) = \mathcal{N}\left(\boldsymbol{\mu}(\boldsymbol{x}_k); 0, \sigma_{ptp}^2\right) \tag{11}$$

$$\boldsymbol{\mu}(\boldsymbol{x}_k) = \frac{1}{N_{\mathbb{Z}}} \sum_{j=1}^{N_{\mathbb{Z}}} \left| \boldsymbol{z}_{k,j}^{c} - \mathrm{NN}\left(\boldsymbol{z}_{k,j}^{c}, \hat{\mathbb{Z}}(\boldsymbol{x})\right) \right|, \tag{12}$$

i.e. the mean distance of measured points to their nearest modeled point is considered Gaussian, with zero mean and covariance σ_{ptp}^2.

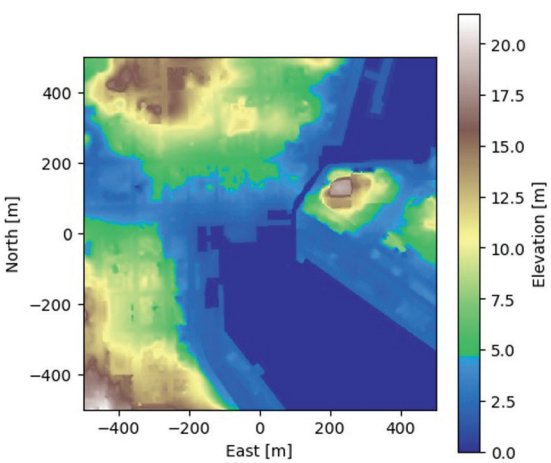


Figure 6: Elevation maps provide further information compared to sea-charts – information which can be exploited to model the response from a radar sensor

Note that when comparing two point-cloud sets, the association generally associates points from one set to points in the other, potentially leaving points in the second set without corresponding points. The question is hence whether to compare the measurements to the predicted points, vice versa or both. We found that due to the dominating cardinality of the modeled response, the one-sided matching of sensor data to the modelled points yielded a more stable tracking performance in this particular application – the map is in a sense more complete than the incoming radar scans.

C. ADDITIONAL MODELS

Further to modeling the radar response from the DEM, there are several other useful sources of data available on a ship, and more information to be gained from a DEM.

For example, radar returns not only come from natural sources – as modeled by the DEM – but also from other vessels in the vicinity. Commonly, nearby vessels report both position and size through the standardized Automatic Identification System (AIS) protocol. There are primarily two ways that ships can be included in the match between sensor data and the single-beam return model; either the returns can be excluded from the sensor data, or the returns can be included in the measurement model. For this, each ship could, for example, be modeled as an ellipsoid described by the reported AIS parameters, with an added approximate height. By adding this to the DEM prior to Algorithm 2, it would automatically be included in the beam model and detection matching. It should be noted, however, that AIS data is not fully reliable and could potentially be subject to spoofing and jamming.

Furthermore, from the DEM it is trivial to extract a virtual measurement of “being on water” contra being on land. Being a maritime vessel, it is natural to assume that it is on water i.e. not being on an elevation above water level as calibrated for tidal movements. Thus, given a tidal model, it is trivial to extract this data from a DEM and to implement this as a particle-filter measurement step, or as part of the prediction model.

Implementation

In this section, we further describe the algorithm and discuss practical aspects of the implementation that was made for this paper. The implementation of the particle filter and the likelihood function presented in Section III was made in the PYTHON programming language along with necessary supporting functions. For example, in the implementation the tracking was performed with latitude–longitude state variables, but for each time-update all particles were transformed into a local Cartesian north–east system, updated, then transformed back to latitude–longitude.

A. RADAR RESPONSE MODEL

The DEM is given as a discrete map, expressed in this paper as the matrix $\boldsymbol{D}$. The measurement model is most efficiently computed in the natural frame of the radar, using polar coordinates. Hence, this data is transformed to polar coordinates to form the polar elevation matrix $\boldsymbol{E}$, with one row per radar angle. The Cartesian map is illustrated in Figure 6.

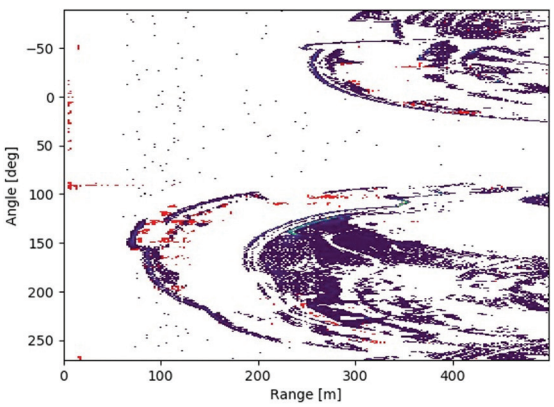


Figure 7: Measurement model with the corresponding polar radar response overlaid in red

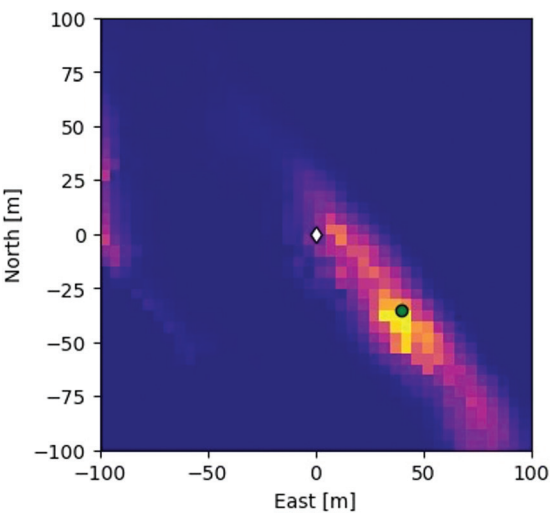


Figure 8: The likelihood function from (11) evaluated around the reference GNSS position, in a 5m resolution grid In the plot, the GNSS reference position (white) as well as a local maxima (green) are marked

The elevation for each angle ψ can then be defined as a lookup function of the distance (range r) from the radar,

$$e_\psi(r) = \boldsymbol{E}\left[\text{round}\left(\frac{\psi}{\Delta_\psi}\right), \text{round}\left(\frac{r}{\Delta_r}\right)\right]. \tag{13}$$

Here, Δ_ψ and Δ_r is the angular and range resolution, respectively.

Pseudocode for the single-beam model from Section III-A is given in Algorithm 1. Note that the projection of (8) is performed with the trigonometric identity of $\tan(\alpha-\beta) = \frac{\tan(\alpha)-\tan(\beta)}{1-\tan(\alpha)\tan(\beta)}$..

Algorithm 1 – Single beam model pseudocode

Input: Matrix row i, RCS threshold, σ_{thres}.

$\mathbb{Z}_\psi^p \leftarrow \emptyset$
$e_{\max} \leftarrow 0$
for $j \in [1, N_{\text{range}})$ **do**
  $\boldsymbol{r} \leftarrow (j \cdot \Delta_r)$
  $\Delta_{e_\psi} \leftarrow (E[i,j] - E[i,j-1])$
  $m_1 \leftarrow \Delta_{e_\psi}/\Delta_r$
  $m_2 \leftarrow (E[i,j] - e_{\text{radar}})/r$
  $d \leftarrow \sin\left(\arctan\left(\left|\frac{m_2-m_1}{1+m_1m_2}\right|\right)\right)$
  $\hat{\sigma}_\psi \leftarrow \Delta_{e_\psi} \cdot d$
  if $E[i,j] > e_{\max}$ **then**
    $e_{\max} \leftarrow E[i,j]$
    if $\hat{\sigma}_\psi > \sigma_{\text{thres}}$ **then**
      $\mathbb{Z}_\psi^p \leftarrow \mathbb{Z}_\psi^p \cup \{(r,\psi)\}$
    end if
  end if
end for

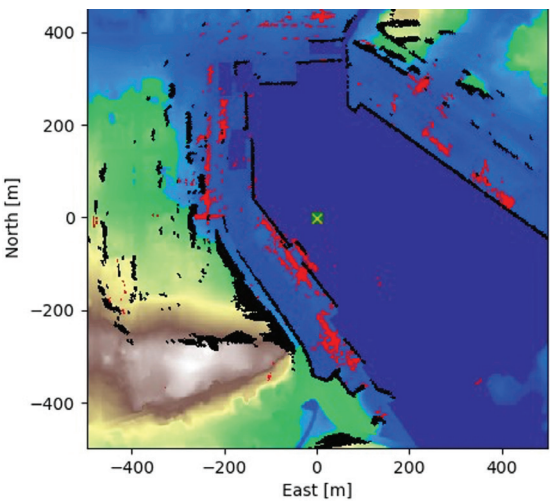
Output: $\mathbb{Z}_\psi^p$

The result from the algorithm applied to all angles yields the polar model of Figure 7, which is subsequently converted to Cartesian coordinates as in Figure 9a to match the format reported from the sensor.

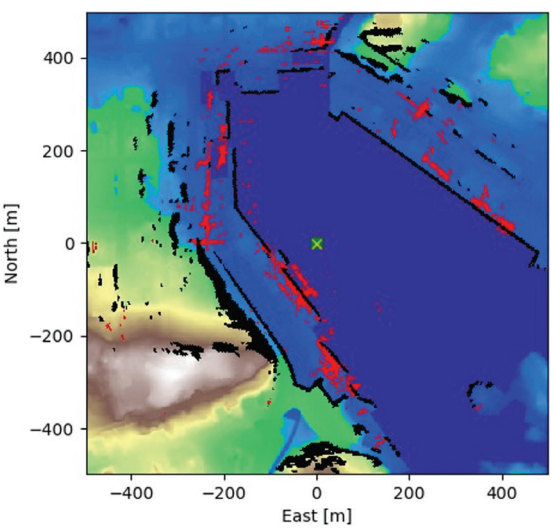
In the figures, the model is overlaid with the measured radar response to reveal that the model, based on the available elevation data, significantly overmodels the amount of responses. This overmodeling, however, is counteracted in the choice of error measure in the point-cloud matching between the sensed and modeled points.

B. POINT CLOUD MATCHING

The likelihood function of (11) is illustrated in Figure 8 for the example scan in Figure 9a over an ±100m north/east area with a gridded 5m resolution. It is clear that in this particular scan and DEM, a more or less nearby position provides an even better match for the received radar data. The corresponding match is illustrated in Figure 9b. A likely reason for this is the fact that the available elevation data does not include the houses along the shore that dominate the radar response within this particular harbor. This illustrates a weakness of the DEM approach compared to SLAM solutions, which are inherently more robust to discrepancies by not relying on external maps.



—
Figure 9a: Match at GNSS reference. In the likelihood plot in Figure 8, the marked positions correspond to the matches illustrated to the right. The modeled response is plotted in black and the measured radar response in red



—
Figure 9b: GNSS, −35m north, +40m east

The pseudocode implementation of the point-cloud measure used in this paper is presented in Algorithm 2, where the measurement likelihood of (11) is applied to the joint set of expected measurements.

Algorithm 2 – Measurement model pseudocode

Input: Latitude (lat), longitude (lon), $r_{\max}$, heading θ.

$N_{\text{range}} = [\text{max\_range}/\Delta_r]$

$N_{\text{angles}} = \lceil 360/\Delta_\psi \rceil$

$x_{\min}, x_{\max} = x(\text{lat}) \pm r_{\max}$

$y_{\min}, y_{\max} = y(lon) \pm r_{\max}$

$\boldsymbol{D} \leftarrow \text{DEM}(x_{\min}, x_{\max}, y_{\min}, y_{\max})$

$\boldsymbol{E} \leftarrow warp\_polar(\boldsymbol{D}), \boldsymbol{E} \in \mathbb{R}_{N_{\text{angles}} \times N_{\text{range}}}$

$\boldsymbol{E} \leftarrow shift\_rows\left(\boldsymbol{E}, \text{round}\left(\frac{\theta}{2\pi}\right)\right)$

$\boldsymbol{M} \leftarrow \emptyset_{N_{\text{angles}} \times N_{\text{range}}}$

for $i \in [0, N_{\text{angles}}) \subset \mathbb{N}$ do

$M[i,:] \leftarrow \text{single\_beam\_model}(i, \sigma_{\text{thres}})$

end for

$\hat{\mathbf{Z}} \leftarrow \text{extract\_points}(\boldsymbol{M})$

$\boldsymbol{\mu} \leftarrow \frac{1}{N_{\mathbf{z}}} \sum_{j=0}^{N_{\mathbf{z}}} \left| \boldsymbol{z}_{k,j}^{c} - \text{NN}(\boldsymbol{z}_{k,j}^{c}, \hat{\mathbf{Z}}) \right|$

$p \leftarrow \mathcal{N}(\boldsymbol{\mu}; 0, \sigma_{ptp}^2)$

Output: p

Experiments

The Suomenlinna II is a ferry that runs between Helsinki harbor and the Suomenlinna fortress in the Helsinki archipelago. The ferry is equipped with Double Acting Technology, meaning its forward/aft direction can be fully reversed, which leads to some additional considerations in algorithm implementations.

The ship has also been retrofitted with a sensor suite including radars, lidars, cameras, and GNSS. These are connected through a backbone network to an onboard server for algorithm development. For the development presented in this paper, multiple passes were recorded and post-processed using a real-time communication framework. This framework enables the same implementation that is developed offline to be used online without modification. Online tests are planned in the future. The marine radar in use performs pre-processing of the radar response. The radar data is sparsified by extracting individual points from a thresholded radar response, with a threshold set at approximately 25 percent of the radar decibel range.

DEMs were obtained from the National Land Survey of Finland [14] who provide DEM data under a Creative Commons Attribution 4.0 International (CC BY 4.0) Licence [15].

The particle filter was run with 100 particles initiated at first around a known GNSS position, with an added Gaussian noise having a standard deviation of 10 m in the north $7.2\Delta_t m^2$ Δ_t time lapsed since the last measurement update. The RCS threshold was set at $\sigma_{\text{thres}} = 0.05\sigma_{ptp}^2 = 5$ m.

Figure 10 shows an example run of the proposed algorithm. We considered 120 s of navigation, with the ferry cruising in a south-southeast direction. In the figure, the reference GNSS track and the estimated position are shown on the map in red and yellow, respectively. The estimation error and the particles' standard deviation of spread are shown in the lower plots. The estimation error, post measurement updates, is kept below approximately 10m to 15m throughout the majority of the evaluation.

In this case, the tracking was performed on every 20th 3Hz radar measurement, resulting in a runtime of approximately real time on a consumer computer. The runtime is dominated by the point-cloud matching, indicating where future optimizations could be focused.

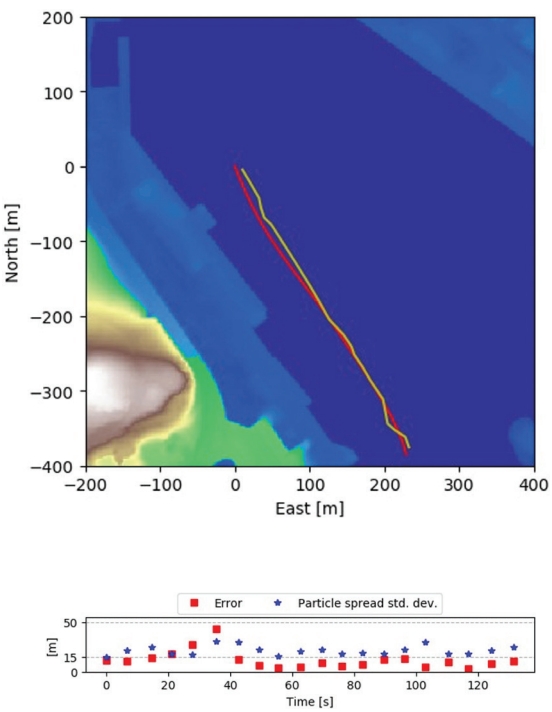


Figure 10: Track and error of the particle filter tracker, as compared with the GNSS reference. The reference GNSS track and the estimated position are shown on the map in red and yellow, respectively

Conclusion

In this work, we propose a method for maritime navigation independent of GNSS that is able to provide absolute positioning of a vessel based on marine radar scans. We describe the method and demonstrate the feasibility of using Digital Elevation Model (DEM) to model the response of a ship-mounted marine radar. Tracking of a vessel is demonstrated using real-world data from a ferry transiting Helsinki harbor. The quality of the tracking is, without surprise, observed to be dependent on the detailedness of the DEM. In particular, the exclusion of man-made structures in the DEM – in this case buildings along the harbor front – sometimes skews the results by not providing a major source of detections present in the sensor data, instead comparing it to the more subtle coastline. Nevertheless, the model is capable of tracking the vessel in the harbor with a tracking error comparable to that of GNSS, maintaining an error in the harbor of less than around 10m to 15m throughout the majority of the test-case using only 100 particles.

At this point, the model does not factor in ground properties such as varying reflectivity, as only the position of expected returns are considered.

The implemented particle filter using the proposed likelihood was shown to be capable of real-time tracking of the vessel. Real-time tracking enables robust localization. This provides a redundant system to complement GNSS navigation. It can also be used to detect GNSS malfunctions or attacks, providing immunity to interference, jamming, and spoofing. The tracking performance could, with optimizations and increase in computing power, further be increased by including more of the radar scans at a higher rate or increasing the number of particles in the particle filter.

Acknowledgements

We wish to thank ABB Marine & Ports and associates for the dataset used in this article, as well as for the assistance provided in reading and interpreting the data.

—

References

[1] J. Bhatti and T. E. Humphreys, “Hostile Control of Ships via False GPS Signals: Demonstration and Detection,” Navigation, Journal of the Institute of Navigation, vol. 64, no. 1, pp. 51–66, 2017.

[2] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, “Unmanned Aircraft Capture and Control Via GPS Spoofing,” Journal of Field Robotics, vol. 31, no. 4, pp. 617–636, 2014.

[3] M. Harris, “Ghost ships, crop circles, and soft gold: A GPS mystery in Shanghai,” 2019. [Online]. Available: https://www.technologyreview.com/s/614689/ ghost-ships-crop-circles-and-soft-gold-a-gps-mystery-in-shanghai/

[4] M. L. Psiaki and T. E. Humphreys, “GNSS Spoofing and Detection,” Proceedings of the IEEE, vol. 104, no. 6, pp. 1258–1270, 2016.

[5] J. Han, Y. Cho, and J. Kim, “Coastal SLAM With Marine Radar for USV Operation in GPS-Restricted Situations,” IEEE Journal of Oceanic Engineering, vol. 44, no. 2, pp. 300–309, 2019.

[6] https://new.abb.com/marine/marine-references/suomenlinna-ii

[7] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” Journal of Basic Engineering, vol. 82, no. 1, p. 35, 1960.

[8] A. H. Jazwinski, Stochastic Processes and Filtering Theory, ser. Mathematics in Science and Engineering. Elsevier Science, 1970.

[9] G. L. Smith, S. F. Schmidt, and L. A. McGee, Application of Statistical Filter Theory to the Optimal Estimation of Position and Velocity on Board a Circumlunar Vehicle, ser. NASA technical report. National Aeronautics and Space Administration, 1962.

[10] N. Gordon, D. Salmond, and A. Smith, “Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation,” IEE Proceedings F - Radar and Signal Processing, vol. 140, no. 2, p. 107, 1993.

[11] F. Gustafsson, “Particle Filter Theory and Practice with Positioning Applications,” Aerospace and Electronic Systems Magazine, IEEE, pp. 1–40, 2010.

[12] J. A. Richards, Remote Sensing with Imaging Radar, ser. Signals and Communication Technology. Berlin, Heidelberg: Springer, 2009.

[13] S. Rusinkiewicz and M. Levoy, “Efficient variants of the ICP algorithm,” Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM, pp. 145–152, 2001.

[14] https://tiedostopalvelu.maanmittauslaitos.fi/tp/kartta?lang=en

[15] https://www.maanmittauslaitos.fi/en/opendata-licence-cc40