A Radar Dataset from the Trondheim City Canal

Petter Hangerhagen
Dept. of Marine Technology
NTNU
Trondheim
petter@hangerhagen.com

Edmund Førland Brekke
Dept. of Engineering Cybernetics
NTNU
Trondheim
edmund.brekke@ntnu.no

Egil Eide
Dept. of Electronic Systems
NTNU
Trondheim
egil.eide@ntnu.no

Roger Skjetne
Dept. of Marine Technology
NTNU
Trondheim
roger.skjetne@ntnu.no

***Abstract*—In the automotive community, methods for tracking, localization, and situational awareness are routinely tested on well-known open-source datasets from the real world. In many other applications of target tracking, such as maritime radar tracking, there is a lack of such data. In this paper, we present a large dataset consisting of data recorded by a frequency-modulated continuous wave radar overlooking the Trondheim City Canal over several weeks during the summer of 2023. The dataset includes a rich variety of boat traffic, ranging from large ferries to formations of kayaks. All the data have been analyzed by means of classical joint integrated probabilistic data association-based multiple target tracking. We point out several challenges that arise in this dataset, such as merged measurements and multipath. We also demonstrate that the data are sufficient to generate statistical information about traffic patterns in the City Canal.**

***Index Terms*—Radar tracking, marine radar, maritime dataset, autonomous surface vehicle (ASV), marine technology.**

I. Introduction

Methods for detection, tracking, and localization play a central role in several safety critical applications, such as traffic control and autonomous vehicle systems. The performance of such methods can be verified through three different approaches: Implementation on simulated data, implementation on real data, and real-time implementation in the intended application. In the literature on radar tracking, which is a cornerstone in the sensor fusion literature, verification by simulation is the dominating approach. In contrast, verification on real data is the dominating approach in the computer vision literature. In the robotics literature, where lidar and camera are popular sensors, several benchmark datasets are routinely used to assess the performance of new methods. However, no established benchmark datasets exist for radar tracking in the maritime domain.

In this paper we focus on maritime radar tracking, largely motivated by research on autonomous surface vessels (ASVs). In particular, the autonomous passenger ferry milliAmpere 2 was developed to bring passengers across the City Canal in Trondheim, Norway. It uses radar and lidar as its main exteroceptive sensors. Consequently, it is of importance to establish a comprehensive understanding of both radar tracking in the canal, and of the patterns that can be found in the boat traffic in the canal. For these reasons, a comprehensive radar dataset was recorded over several days during the summer 2023.

The main contribution of this paper is to present this dataset, which has been made publicly available for the benefit of the engineering community. It can be found free to download at http://doi.org/10.5281/zenodo.10706215. We analyze the data by means of a multi-target tracker in the Joint Probabilistic Data Association (JPDA) family. This analysis brings attention to several challenges which may be of interest, such as multiple closely spaced targets, multi-path effects, and merged measurements. Furthermore, we sketch how a simple counting technique can provide information about traffic patterns in the canal based on the tracking results.

This paper is organized as follows. In Section II we provide a brief survey of related work. In Section III we present the dataset, and in Section IV an analysis on dataset is presented including a traffic analysis. Finally, in Section V we conclude the paper.

II. Related work

A. Target Tracking of Maritime Vessels

Tracking of vessels using shore-mounted or ship-mounted radars is a well studied problem, and an ingredient in many commercial products. Several papers, [1], [2], [3], [4], considers the use of various versions and extensions of JPDA for multi-target tracking on maritime radar data. Less established approaches, such as extended object tracking [5] and track-before-detect [6], have also been proposed. Systems for multi hypothesis tracking (MHT) on maritime radar data have been reported in [7] and [8] in the scientific literature, although without detailed descriptions of the algorithms.

B. Vessel Traffic Analysis

Several papers have proposed techniques for using historical data to analyse and predict ship traffic. To the best of our knowledge, all of these papers have used historical Automatic Identification Systems (AIS) data. This has been achieved through various methods, including nearest neighbor techniques [9], [10], recurrent neural networks [11], and clustering of trajectories [12]. Long-term prediction of vessel motion has also been studied by taking a hand-crafted traffic pattern as a prior input [13].

The absence of radar data in the input to these methods is problematic because not all vessels carry AIS transponders. While AIS data may provide a comprehensive understanding of how large vessels navigate in a fjord or across an ocean, it

fails to reveal the traffic patterns of smaller vessels, such as recreational boats and kayaks.

C. Maritime Radar Datasets

In the automotive and robotics communities, benchmark datasets are routinely used for verification of sensor fusion methods. The most famous is the KITTI dataset [14], which involves annotated data from GPS, lidar, and stereo cameras mounted on a car driving around the mid-size city of Karlsruhe. Over the previous decades, a lot of different datasets for the automotive industry have been posted, where each dataset has its own strengths and weaknesses. Many datasets for computer vision have been posted, including CamVid ([15]), Cityscapes ([16]), and Mapillary Vistas ([17]), to mention some.

In more recent times, extended datasets have been published, containing complete sensor suites including radars. Marulan dataset ([18]) was one of the first, followed by some others, such as nuScenes ([19]), Oxford Radar Robotcar ([20]), Mulran ([21]), and Boreas ([22]).

In the maritime context, the most similar dataset is the Pohang Canal Dataset [23], which includes data from radar, lidar, omnidirectional cameras, and stereo cameras mounted on a vessel traversing a 7.5km long canal. The dataset does not come with annotations, and the paper does not provide any analysis of the boat traffic present during the recording.

Other datasets of a similar nature have been recorded from Unmanned Surface Vehicles (USVs). The dataset USVInland [24] contains data from multiple sensors, including a millimeter-wave radar (with a range of 18.08 meters), lidar, stereo cameras, GPS, and IMUs. It was collected during the summer of 2020 in three different cities in China, covering a trajectory of over 26km. The dataset FloW [25] includes data from two cameras and a millimeter-wave radar. It can be divided into two subsets: FloW-IMG, consisting of videos and images, and FloW-RI, consisting of radar data and image data. Both datasets can be requested on their homepage.

Several other large maritime datasets exist. However, the majority of these datasets comprise camera images. The well-known Singapore Marine Dataset (SMD), presented in [26], consists of 81 videos. Another dataset, the Marine Object Detection Dataset (MODD), presented in [27], consists of 12 videos containing 4454 annotated frames. One more, MODD2, presented in [28], consists of 28 videos containing 11675 annotated frames. All three of these datasets are combined into a unified dataset known as Marine Object Detection and Segmentation (MODS), as described in [29]. They all serve as benchmark datasets for computer vision.

To the best of our knowledge, the scientific literature contains only two datasets in which multi-target tracking has been implemented on maritime radar data. Three scenarios recorded outside of the Strait of Kadet are introduced in [4]. This datasets are not publicly available but can be obtained upon request to the authors pending clearance considerations such as nationality. Two scenarios from an autonomy experiment conducted in Trondheimsfjorden, Norway, are presented in [1].

Fig. 1: The sensor station, with a Simrad 4G FMCW solid-state radar and a Sony-Ipela HD SNC-HM662 IP camera.

The second of these scenarios, the Autosea Final Demonstration, included both a large research vessel and three high-speed Rigid Inflatable Boats (RIBs). Thus, both [4] and [1] presented only a handful of scenarios. In contrast, the present paper reports a dataset containing a much larger number of scenarios

Other types of maritime radar datasets have also been recorded to enable and validate detection methods based on machine learning. An example of this is the dataset reported in [30], which is constructed based on Plan Position Indicator (PPI) images displayed on the terminal of the maritime radar. However, this dataset has not been made publicly available.

III. Dataset

A. Experimental Setup

The onshore sensor station used for collecting data is located at *Fosenkaia* in Trondheim. The station consists of two sensors: a Simrad 4G Frequency Modulated Continuous Wave (FMCW) solid-state radar and a Sony-Ipela HD SNC-HM662 IP camera, as illustrated in Fig. 1. The exact position of the sensor station in WGS-84 coordinates is 63°26'06.6"N and 10°23'34.9"E. The radar is mounted roughly 3.3 meters above ground level and approximately 4-7 meters above sea level, depending on the tide. The position of the radar in relation to its environment can be seen in Fig. 2.

B. Land Filter

Due to the sensor station being located onshore in an urban environment, it frequently received returns from surrounding buildings, terrain, and other onshore objects. These returns were effectively filtered out using information from a map supplied by the Norwegian Mapping Authorities. The setup of the land filter follows the same description provided in [32]. The map indicates the locations of land within the radar's field of view, and all detections overlapping with land are filtered away. By also incorporating a 10-meter error margin in this land filter, returns from stationary objects and docks located close to land could also be filtered away, leaving only the

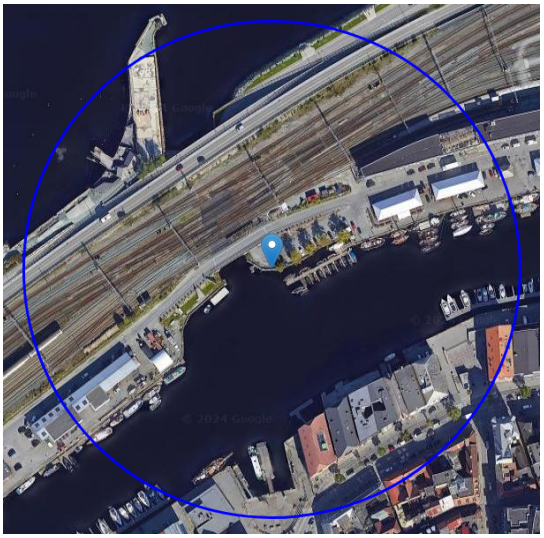

Fig. 2: Radar position and range displayed on Google Earth [31].

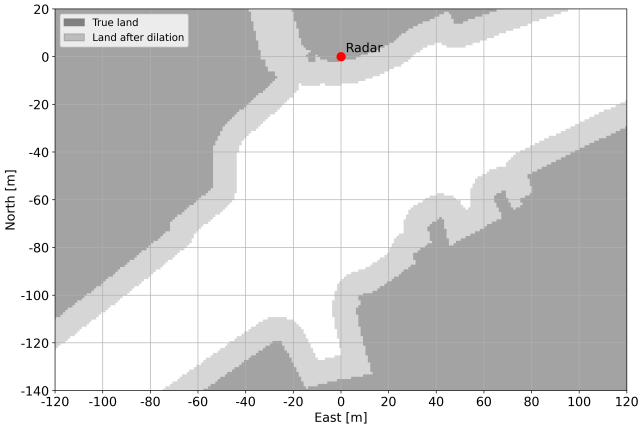

Fig. 3: The white areas are water, the dark grey areas are true land, and the light grey areas are dilated land given by the land filter.

activity in the canal available for recording. Fig. 3 shows the dilated map compared with actual land.

C. The Trondheim City Canal

The canal runs from south-west to north-east, from area C to area A in Fig. 4. Both ways are leading out into the fjord, but the shortest way is travelling passed area C. Area B leads into a tunnel, which is a shortcut out to the fjord. Area D is a small harbor, Ravnkloa, where a passenger ferry docks. Area E is a location where medium sized vessels dock at the quay. Area F is a marina for small wooden boats. The region between area D and area A are private property, and targets rarely move to this area.

D. Radar Tuning

The radar was tuned using the parameters listed in TABLE I. Due to the limited size of the canal, a range of 150 meters

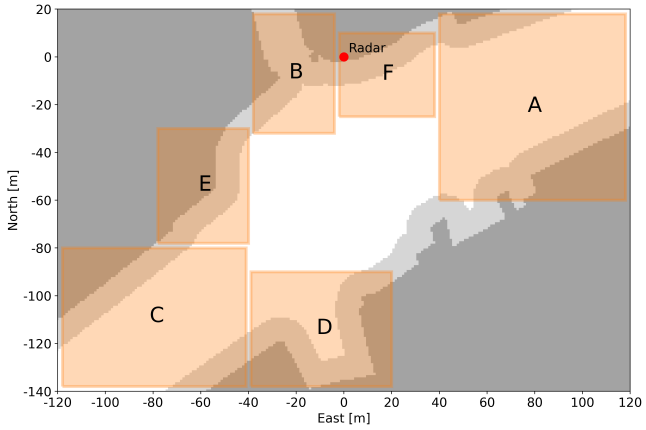

Fig. 4: The six orange areas are regions where most vessels enters and leaves the surveillance region.

TABLE I: Radar parameters during dataset recording.

Radar parameters	
state	3: Transmit
range	150 [m]
autogain	0: Off
gain	75
seaClutterType	1: Harbour
rpm	48 [rpm]
beamSharpening	0: 2.6°

was sufficient to cover its interesting parts, as illustrated in Fig. 2. For this specific radar, the gain was manually set to 75. A larger gain could have been used; however, a gain of 75 proved sufficient for detecting all objects within the canal, with the benefit of reducing multipath effects. Sea clutter was automatically tuned using the radars harbour functionality. The radar's rotation speed was set to 48 rpm. Finally, the beam sharpening parameter was set to zero, giving in a beam width of 2.6°.

E. Clustering Algorithm

The collected dataset has poor angular resolution and is not suitable for extended object tracking. However, by clustering the measurements and using the centroid as the true detection, the standard assumption model for multi-target tracking can be considered valid. The clustering algorithm used on the data is also based on the pipeline described in [32]. The algorithm assumes that detections that are close to each other come from the same object. Each cluster of multiple detections represents a single measurement. We refer to detections that are close to each other as neighbors, and the neighbors are defined by having an Euclidean distance lower than a threshold, that is,

$$\|p_i - p_j\| \leq R, \tag{1}$$

where p_i and p_j are points, and R is the threshold distance. The clustering algorithm is using single-link hierarchical clustering with k-d trees for optimizing the algorithm's complexity.

F. Dataset Description

The dataset was recorded during the summer of 2023, from August 15th to September 24th. In total, the dataset includes data from 27 different days, with a total of 1027 scenarios with various environmental conditions. However, most of the data is recorded during excellent weather conditions. Some recordings are also done by night. The published dataset only consists of data collected from the radar. Some images from the camera were captured and used as ground truth to aid in the data analysis in an early phase. However, these are not synchronized with the radar data and is hence not included in the dataset.

1) Dataset structure: Due to memory considerations, the published data consists of radar detections after clustering and land filtering, rather than the raw radar spokes. The dataset is stored in JavaScript Object Notation (JSON) files, formatted as JSON arrays. Each array element corresponds to a single time instance and is saved using a custom-defined Robot Operating System (ROS) message, containing all the information generated by the clustering algorithm. This message includes the Unix timestamp, the position of the clustered measurements center, the center of the polygon enclosing the cluster, a list of points outlining the polygon, and the area of the polygon. All the data are given relative to the radar in cartesian coordinates.

The data is structured in a total of eight directories, and follows the structure below, where the numbers in the directories names are the specific dates of the recordings.

```
/
└── radar_data.zip/
    ├── data_aug_15-18/
    │   ├── rosbag_2023-08-15-13-31-51.json
    │   └── ...
    ├── data_aug_18-19/
    │   ├── rosbag_2023-08-18-13-30-58.json
    │   └── ...
    ├── ...
    └── data_sep_17-18-19-24/
        ├── rosbag_2023-09-17-12-12-38.json
        └── ...
```

G. The Recording Process

During recording, the sensor station was not connected online. So, to initiate data recording, manual intervention was necessary to connect to the sensor station and activate the ROS driver. While one could have simply initiated the ROS driver and allowed it to continuously record, this approach posed several drawbacks, including the rapid consumption of the station's limited storage capacity and the storage of vast amounts of irrelevant data.

To avoid the manual intervention when recording, an automated recording process was developed. This process is operated by monitoring the output from the algorithm that clusters any measurement present after the data has passed through the land filter. When the clustering algorithm detects an object, a new recording is initiated. The recording continues for 10 seconds after the clustering algorithm ceases to produce output. A drawback with this procedure was that returns from stationary objects present in the surveillance region made the clustering algorithm continuously give output. To avoid this, some manual adjustments to the land filters were made on a daily basis, depending on the location of different vessels moored in the area. However, these manual adjustments were removed from the data before uploading it, ensuring that all the activity in the canal is present in the dataset, including stationary targets.

A scenario in the dataset is defined as a single continuous recording. This means that a scenario starts when the clustering algorithm starts giving output and continues until 10 seconds after it ceases giving output. It may include anything from one target to multiple targets, depending on whether new objects enter the surveillance region before the recording should stop.

H. Key Features in the Dataset

The dataset contains a wide variety of scenarios, capturing every activity in the canal during an active recording period. This includes everything from a single vessel passing to multiple kayaks arriving in groups, and the passenger ferry departing and arriving.

Since the dataset consists of real world radar data, common limitations with a radar system are reflected. Merged measurements situations and multipath effects are phenomena that occur.

1) Multi-target scenarios: One aim behind the construction of this dataset has been to obtain bona-fide multi-target scenarios from the real world, where two or more vessels are so close that significant measurement contention arises. Out of the 1027 scenarios, 69 have been identified as multi-target scenarios. The process behind identifying them is quite simple. For every time step, the distances between the current positions of the tracks is compared. If some of the tracks are in close proximity to each other, the Mahalanobis distance of the posterior estimates is checked. If this distance is below 3, the scenario is assumed to be multi-target. In Fig. 5 and Fig. 6, two examples of multi-target scenarios can be seen.

2) Merged measurements scenarios: A common limitation of a radar systems is resolution. If targets move too close to each other, they can typically merge into one cluster. In the dataset, 112 scenarios of this kind have been identified. The idea behind the identification process is quite simple and contains two main steps.

The first step is to identify close measurement pairs. At every time step, the distances between all the measurements that appear is calculated. If the distance between two measurements is below some threshold, they are saved as a close measurement pair. There are also some other conditions for the approval of a close measurement pair: the cluster area is larger than a threshold, and the close measurement pair lies in the middle of the canal and not close to land. These conditions are included to reduce the number of false positive scenarios due to many cluttered scenarios.

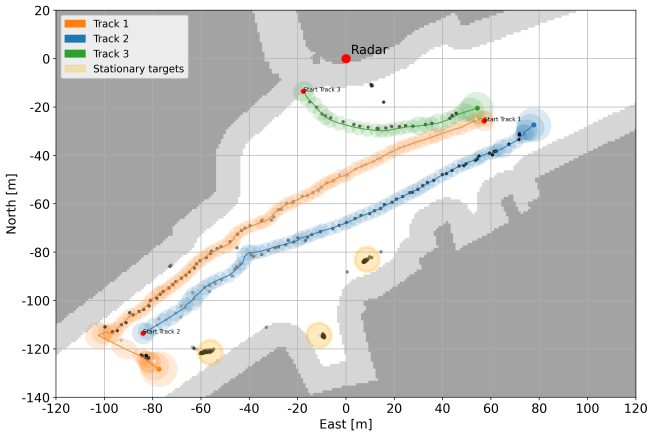

Fig. 5: An example of a multi-target scenario with 3 tracks. Stationary targets are also displayed. The measurements are plotted as points ranging from light gray to black. The color scaling of the points represents when measurements were taken, with the lightest gray indicating the first measurement and fully black indicating the last measurement. The tracks starts with a red dot, and the states of the tracks follows the lines corresponding to the track's color. The semi-transparent circles at every step represent the posterior covariance ellipsoid of the track at the current step. Scenario captured on 18.09.2023 at 17:42:41.

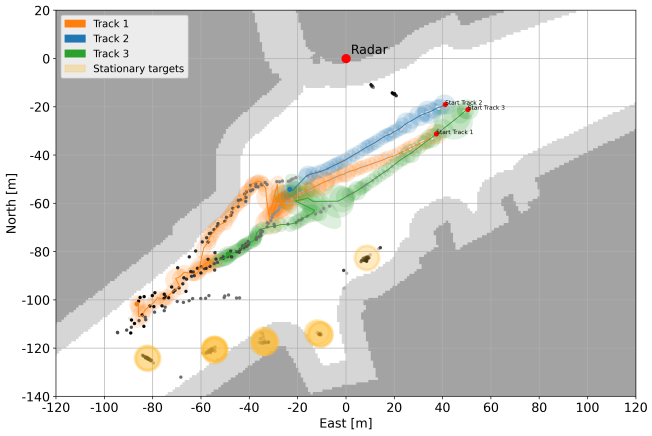

Fig. 6: Another example of a multi-target scenario with 3 tracks. This scenario has many of the same stationary targets as in Fig. 5. Scenario captured on 18.08.2023 at 15:14:30.

The second step is to check if the close measurement pairs merge into a single measurement with a cluster area larger than the combination of cluster areas of the close measurement pair in the next time step. In Fig. 7 and Fig. 8, two examples of merged measurement situations can be seen.

3) Multipath scenarios: Multipath effects are something that occur frequently in the dataset, typically happening when large vessels are moving close to the sensor station. A total of 117 scenarios have been identified with multipath effects. The process of identifying them can be divided into several steps.

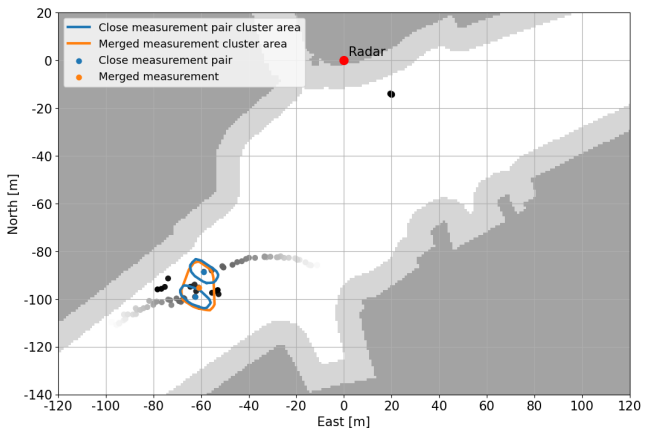

Fig. 7: A scenario with merging measurements. The actual scenario contains more measurements than displayed. The figures shows 30 time steps before merging happens, and 10 time steps after. It displays two objects passing close to each other. Scenario captured on 19.08.2023 at 14:22:41.

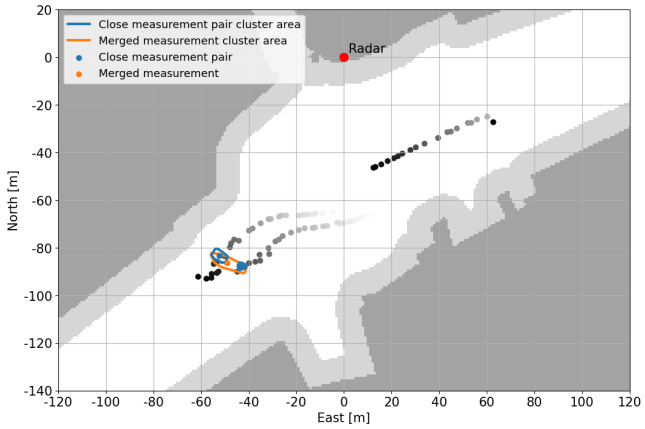

Fig. 8: Another scenario with merging measurements. This scenario also contains more measurements than displayed. It displays two objects coming from the same direction and moving close to each other. A third object is also coming from the same direction but is a little way behind. Scenario captured on 29.08.2023 at 18:41:39.

The first step is to check if measurements are close to the sensor station, that is, not farther than $50m$ away. The second step is to check the size of the cluster area. When multipath occurs in this dataset, the objects are quite large. So, the cluster area have to be a minimum of $150m^2$. If a measurement satisfies these two conditions, it is defined as a multipath parent.

Moreover, measurements at the same timestamp are checked. If they are located in the sector behind the multipath parent, they are defined as multipath children. One multipath parent must have at least one multipath child for it to be approved as a valid parent. The sector for the multipath children are limited to 6° on each side of the multipath

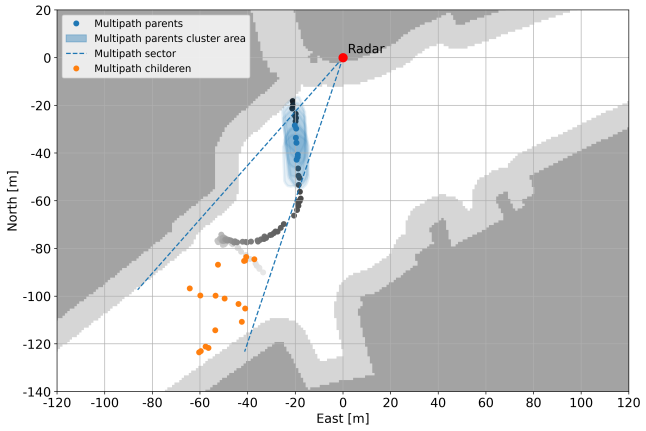

Fig. 9: Example of a multipath scenario. The actual scenario contains more measurements than displayed. The figure shows only the measurements relevant for defining it as a multipath scenario. The dashed lines for the sectors represent 6° to each side of the first and last multipath parent. Number of parents: 7, number of children: 17. Scenario captured on 17.09.2023 at 12:12:38.

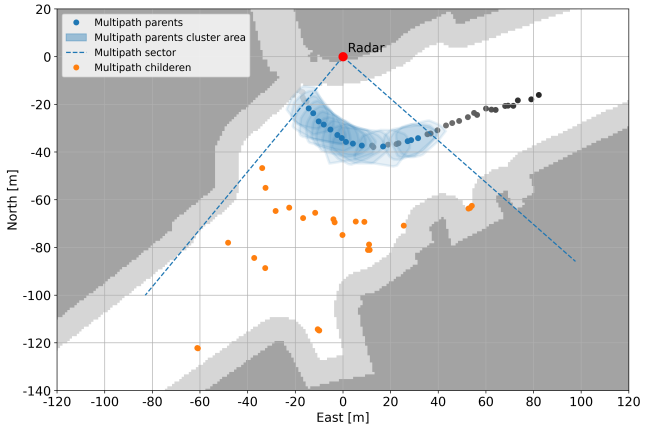

Fig. 10: Another example of a multipath scenario. This scenario also contains more measurements than displayed. Number of parents: 14, number of children: 26. Scenario captured on 18.08.2023 at 18:32:57.

parent. By visual inspection, multiple valid multipath parents occur in the same scenario if it is truly a multipath scenario. Therefore, a minimum number of 3 valid multipath parents must be present in a scenario for it to be defined as a multipath scenario. In Fig. 9 and Fig. 10, two scenarios containing multipath effects can be seen.

An overview of the different types of classified scenarios can be seen in TABLE II. It is worth noting that a single scenario can be classified as all three types.

IV. Traffic Analysis

A. Multi-Target Tracking

The multi-target tracker developed in [1], a generalized version of the JIPDA, was used to establish tracks for the traffic analysis. These tracks were also used to identify the different kind of multi-target scenarios described earlier. This specific JIPDA includes functionality for multiple models. However, only a single constant velocity model was used, as the vessels inside the canal have relatively low speeds. The tuning parameters for the multi-target tracker can be seen in TABLE III. The notations are the same as in [1].

TABLE II: List of the number of time the different scenarios occur in the dataset.

Type of scenario	Number of scenarios
Multi-target	69
Merged measurement	112
Multipath	117

TABLE III: Parameters for the multi-target tracker.

Quantity	Symbol Unit	Value
Initial velocity std.	σ_v[m/s]	3
Detection probability	$P_{\mathrm{D}}[\%]$	92
Clutter intensity	$\lambda\,[1/\mathrm{m}^2]$	1×10^{-5}
Gate size	$g[-]$	3.0
Survival probability	$P_{\mathrm{S}}[\%]$	90
Existence confirmation threshold	$T_c[\%]$	99
Existence termination threshold	$T_d[\%]$	10
Visibility Markov probability	$\omega^{11}[-]$	0.90
Visibility Markov probability	$\omega^{10}[-]$	0.10
Visibility Markov probability	$\omega^{01}[-]$	0.52
Visibility Markov probability	$\omega^{00}[-]$	0.48
Cartesian noise std.	$\sigma_c[\ \mathrm{m}]$	4.0
Polar range std.	$\sigma_r[\ \mathrm{m}]$	2.0
Polar bearing std.	$\sigma_\theta\,[^\circ]$	1.0
Model process noise	$\sigma_{a,1}\ [\ \mathrm{m/s}^2]$	0.5

B. Track filtering

A tracking method applied to real data will typically generate tracks on both stationary objects as well as moving targets. Spurious tracks caused by birds, multi-path, and clutter in general, are inevitable. Several methods for filtering away nonphysical tracks have been tested, resulting in these four filtering methods.

1) Coherence Factor: We can use the concept of track coherence that was proposed in [33] to distinguish tracks on moving targets with an intentional motion, from other targets. Let $\boldsymbol{u}_k = \boldsymbol{\rho}_k - \boldsymbol{\rho}_{k-1}$, where $\boldsymbol{\rho}_k$ contains the x- and y-positions of the k-th state, $\boldsymbol{v}_k$ represents the velocity of the current state, and γ_l denotes the angle between $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$. The track coherence c_k can then be calculated using,

$$c_k = \frac{1}{k}\sum_{l=1}^{k} \cos\gamma_l = \frac{1}{k}\sum_{l=1}^{k} \frac{\mathbf{v}_l^\top \boldsymbol{u}_l}{\|\mathbf{v}_l\|\,\|\boldsymbol{u}_l\|}. \tag{2}$$

Track coherence measures how straight an object is moving by examining the angle between the position vector and the velocity vector. When this angle is small, the cosine of the angle approaches one. By applying this criterion to the state

history of a track and checking if the value is below a certain threshold, nonphysical tracks can be filtered out.

2) Minimum number of track states: We assume that true targets within the canal are in motion. Therefore, established tracks must have a minimum of 20 states to be deemed valid. Tracks with less than 20 states are therefore assumed invalid and are filtered out.

3) Average track speed: The canal has a speed restriction of 5 knots. Occasionally, some targets may surpass this limit slightly. However, no targets should exceed a speed of 5m/s (9.7 knots). Consequently, any tracks exhibiting an average speed beyond this threshold are filtered out.

4) Jumping tracks: The established tracks are supposed to represent the movements of the objects in the canal, and in general these objects exhibit low accelerations. Thus, a nonphysical movement involves a track jumping a significant distance from one state to another. However, nearby measurements might disturb some tracks, causing them to be drawn towards these measurements and swiftly return to the 'correct' ones. By permitting tracks to exhibit this behavior only once, tracks with nonphysical movements can be filtered out.

C. Traffic Matrix

It is of interest to investigate whether there are correlations between where a vessel enters the surveillance region and where it exits the surveillance region. To conduct such an analysis, the regions where targets may appear or disappear are defined by the six areas drawn out in Fig. 4.

After running all 1027 scenarios through the multi-target tracker, and filtering out invalid tracks, a total of 1657 tracks were established. Out of these, 412 were not accounted for, as they either did not start or stop at valid locations. Therefore, the resulting traffic matrix, in TABLE IV, consists of 1245 tracks. The left column describes where a track starts, and the top row describes where it ends. For example, for tracks starting in area B, 113 of them end at area A. The numbers on the diagonal should be close to zero, since a vessel would normally travel from one place to another. When comparing the traffic matrix with the average length of the tracks in TABLE V, it can be seen that the average length is generally much lower on the diagonal. This implies that most of the tracks on the diagonal are stationary tracks, or nonphysical tracks that passed through the filtering process.

If we simplify the traffic matrix by assuming there are no tracks on the diagonal, a new matrix can be set up, as in TABLE VI. As seen, area A is now where most tracks start. This coincides with the arbitrary observation made from the sensor station. The parts of the canal farther east serve as an alternative marina for recreational boats, so it is logical that many vessels come from this area, as recreational boats typically heads out into the fjord. Additionally, kayak formations originate from area A, as a company dealing with kayak rentals has a popular route originating there.

TABLE IV: Resulting traffic analysis matrix.

	A	B	C	D	E	F	
A	14	98	214	8	8	2	344
B	113	4	51	90	9	14	281
C	123	61	159	13	5	2	363
D	13	44	16	27	2	0	102
E	1	7	10	0	3	0	21
F	4	9	22	3	8	88	134
	268	223	472	141	35	106	1245

TABLE V: Average length [m] of the tracks at each cell in the traffic matrix.

	A	B	C	D	E	F
A	22	76	158	98	106	39
B	84	8	109	92	47	41
C	162	103	11	52	37	135
D	118	96	46	4	57	0
E	128	49	51	0	4	0
F	46	38	134	91	85	1

TABLE VI: Simplified traffic analysis matrix.

	A	B	C	D	E	F	
A	0	98	214	8	8	2	330
B	113	0	51	90	9	14	277
C	123	61	0	13	5	2	204
D	13	44	16	0	2	0	75
E	1	7	10	0	0	0	18
F	4	9	22	3	8	0	46
	254	219	313	114	32	18	950

D. Histogram of Tracks Duration

The graph in Fig. 11 illustrates the distribution of durations for the 1657 tracks analyzed, where the green portions of the bars represent stationary tracks, and the blue portions represent tracks in motion. A track is defined as stationary if it does not move outside a circle drawn at the start position with a radius of 30m. This definition is also based on the assumption that objects are consistently in motion through the canal. As seen there are no tracks with a duration lower than 20 seconds, this is because of the second filtering method, described in Section IV-B3. The number of stationary track coincides almost perfectly with the number of tracks on the diagonal in TABLE IV, with 295 tracks.

V. Conclusion

In this paper, we have introduced a new benchmark maritime radar-based dataset. The dataset consists of 1027 scenarios, of which 69 scenarios are defined as multi-target, 112 scenarios involve merged measurement situations, and 117 scenarios include multipath effects. A multi-target tracker was used to establish tracks, which were then used to set up a traffic matrix that provides insights into the traffic patterns in the canal.

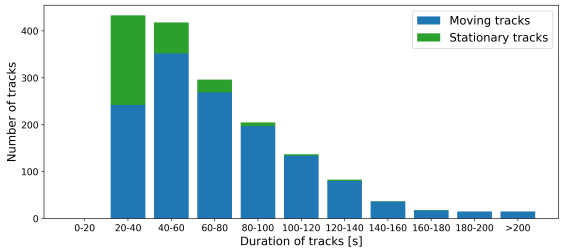

Fig. 11: Histogram of the tracks duration. Classified into either moving or stationary, with 1357 moving tracks and 300 stationary tracks.

In further research, we aim to test out the dataset with respect to vessels prediction. We also want to investigate the targets compliance with COLREG, which can be useful information for the autonomous ferry operating in the City Canal.

ACKNOWLEDGMENT

The authors would like to thank Øystein Kaarstad Helgesen and Erik Wilthil for helping with technical details during the recording process. Work partially funded by the Reasearch Council of Norway through SFI AutoShip (RCN project 309230). ChaptGPT have been used for correcting grammar.

REFERENCES

[1] E. F. Brekke, A. G. Hem, and L.-C. N. Tokle, "Multitarget tracking with multiple models and visibility: Derivation and verification on maritime radar data," *IEEE Journal of Oceanic Engineering*, vol. 46, no. 4, pp. 1272–1287, 2021.

[2] P. Braca, S. Maresca, R. Grasso, K. Bryan, and J. Horstmann, "Maritime surveillance with multiple over-the-horizon HFSW radars: An overview of recent experimentation," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 12, pp. 4–18, 2015.

[3] G. Vivone, P. Braca, and J. Horstmann, "Knowledge-based multitarget ship tracking for HF surface wave radar systems," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 7, pp. 3931–3949, 2015.

[4] J. S. Fowdur, M. Baum, and F. Heymann, "Tracking targets with known spatial extent using experimental marine radar data," in *Proceedings of Fusion*, 2019, pp. 1–8.

[5] G. Vivone and P. Braca, "Joint probabilistic data association tracker for extended target tracking applied to X-band marine radar data," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 4, pp. 1007–1019, Oct 2016.

[6] M. McDonald and B. Balaji, "Track-before-detect using Swerling 0, 1, and 3 target models for small manoeuvring maritime targets," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, pp. 1–9, 2008.

[7] C. Carthel, S. Coraluppi, and P. Grignan, "Multisensor tracking and fusion for maritime surveillance," in *Proceedings of Fusion*, 2007, pp. 1–6.

[8] L. Elkins, D. Sellers, and W. R. Monach, "The autonomous maritime navigation (AMN) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *Journal of Field Robotics*, vol. 27, no. 6, pp. 790 – 818, 2010.

[9] S. Hexeberg, A. L. Flåten, B.-O. H. Eriksen, and E. F. Brekke, "AIS-based vessel trajectory prediction," in *Proceedings of Fusion*, Jul. 2017.

[10] B. R. Dalsnes, S. Hexeberg, A. L. Flåten, B.-O. H. Eriksen, and E. F. Brekke, "The neighbor course distribution method with gaussian mixture models for AIS-based vessel trajectory prediction," in *Proceedings of Fusion*, Cambridge, UK, Jul. 2018.

[11] N. Forti, L. M. Millefiori, P. Braca, and P. Willett, "Prediction of vessel trajectories from AIS data via sequence-to-sequence recurrent neural networks," in *Proceedings of ICASSP*, 2020, pp. 8936–8940.

[12] R. P. Jain, E. F. Brekke, and A. Rasheed, "Unsupervised clustering of marine vessel trajectories in historical AIS database," in *Proceedings of Fusion*, 2022, pp. 1–6.

[13] T. Tengesdal, L. Millefiori, P. Braca, and E. Brekke, "Joint stochastic prediction of vessel kinematics and destination based on a maritime traffic graph," in *Proceedings of ICEMCCME*, Maldives, 11 2022.

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231 – 1237, 2013.

[15] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[16] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," arXiv preprint arXiv:1604.01685, 2016.

[17] G. Neuhold, T. Ollmann, S. R. Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of Computer Vision*, 2017, pp. 5000–5009.

[18] T. Peynot, S. Scheding, and S. Terho, "The marulan data sets: Multi-sensor perception in a natural environment with challenging conditions," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1602–1607, 2010.

[19] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of Computer Vision and Pattern Recognition*, 2020, pp. 11 618–11 628.

[20] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset," in *Proceedings of Robotics and Automation*, 2020, pp. 6433–6438.

[21] G. Kim, Y. S. Park, Y. Cho, J. Jeong, and A. Kim, "Mulran: Multimodal range dataset for urban place recognition," in *Proceedings of Robotics and Automation*, 2020, pp. 6246–6253.

[22] K. Burnett, D. J. Yoon, Y. Wu, A. Z. Li, H. Zhang, S. Lu, J. Qian, W.-K. Tseng, A. Lambert, K. Y. Leung, A. P. Schoellig, and T. D. Barfoot, "Boreas: A multi-season autonomous driving dataset," *The International Journal of Robotics Research*, vol. 42, no. 1-2, pp. 33–42, 2023.

[23] D. Chung, J. Kim, C. Lee, and J. Kim, "Pohang canal dataset: A multimodal maritime dataset for autonomous navigation in restricted waters," 2023.

[24] Y. Cheng, M. Jiang, J. Zhu, and Y. Liu, "Are we ready for unmanned surface vehicles in inland waterways? The USVInland multisensor dataset and benchmark," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3964–3970, 2021.

[25] Y. Cheng, J. Zhu, M. Jiang, J. Fu, C. Pang, P. Wang, K. Sankaran, O. Onabola, Y. Liu, D. Liu, and Y. Bengio, "FloW: A dataset and benchmark for floating waste detection in inland waters," in *Proceedings of ICCV*, 2021, pp. 10 933–10 942.

[26] D. K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 1993–2016, 2017.

[27] M. Kristan, V. Sulić Kenk, S. Kovačič, and J. Perš, "Fast image-based obstacle detection from unmanned surface vehicles," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 641–654, 2016.

[28] B. Bovcon, R. Mandeljc, J. Perš, and M. Kristan, "Stereo obstacle detection for unmanned surface vehicles by IMU-assisted semantic segmentation," *Robotics and Autonomous Systems*, vol. 104, pp. 1–13, 2018.

[29] B. Bovcon, J. Muhovič, D. Vranac, D. Mozetič, J. Perš, and M. Kristan, "MODS—A USV-oriented object detection and obstacle segmentation benchmark," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 13 403–13 418, 2022.

[30] X. Chen, J. Guan, Z. Wang, H. Zhang, and G. Wang, "Marine targets detection for scanning radar images based on radar- YOLONet," in *Proceedings of Radar*, 2021, pp. 1256–1260.

[31] Google. Google maps. Figure draw thorugh a python script taken on February 27, 2024. [Online]. Available: https://www.google.com/maps/@63.4352719,10.3930753,360m/data=!3m1!1e3?entry=ttu

[32] Øystein Kaarstad Helgesen, K. Vasstein, E. F. Brekke, and A. Stahl, "Heterogeneous multi-sensor tracking for an autonomous surface vehicle in a littoral environment," *Ocean Engineering*, vol. 252, p. 111168, 2022.

[33] E. Brekke, O. Hallingstad, and J. Glattetre, "The signal-to-noise ratio of human divers," in *Proceedings of Ocean*, 2010, pp. 1–10.