Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Radar Target Classification using Recursive Knowledge-Based Methods

Jochumsen, Lars Wurtz

DOI (link to publication from Publisher):
10.5278/vbn.phd.engsci.00096

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Jochumsen, L. W. (2016). *Radar Target Classification using Recursive Knowledge-Based Methods*. Aalborg Universitetsforlag. https://doi.org/10.5278/vbn.phd.engsci.00096

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: April 18, 2024

RADAR TARGET CLASSIFICATION USING RECURSIVE KNOWLEDGE-BASED METHODS

BY
LARS WÜRTZ JOCHUMSEN

DISSERTATION SUBMITTED 2016

AALBORG UNIVERSITY
DENMARK

Radar Target Classification using Recursive Knowledge-Based Methods

PhD Dissertation

Lars Würtz Jochumsen

Dissertation submitted April 4, 2016

Dissertation submitted: April 4, 2016

PhD supervisor: Assoc. Prof. Jan Østergaard
Aalborg University

PhD Co-Supervisor: Prof. Søren Holdt Jensen
Aalborg University

PhD Company supervisor: Morten Østergaard Pedersen
Terma A/S

PhD committee: Lektor Thomas Arildsen (Formand)
Aalborg University

Professor Lars-Kai Hansen
Technical University of Denmark

Professor Mats Viberg
Chalmers University

PhD Series: Faculty of Engineering and Science, Aalborg University

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-545-0

Published by:
Aalborg University Press
Skjernvej 4A, 2nd floor
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

© Copyright: Lars Würtz Jochumsen, except where otherwise stated

Printed in Denmark by Rosendahls, 2016

This work is supported by Terma A/S and the Danish Ministry of Higher Education and Science under grant no 13-135424

Abstract

The topic of this thesis is target classification of radar tracks from a 2D mechanically scanning coastal surveillance radar. The measurements provided by the radar are position data and therefore the classification is mainly based on kinematic data, which is deduced from the position. The target classes used in this work are classes, which are normal for coastal surveillance e.g. ships, helicopters, birds etc. The classifier must be recursive as all data of a track is not present at any given moment. If all data were available, it would be too late to classify the track, as the track would have been terminated. Therefore, an update of the classification results must be made for each measurement of the target. The data for this work are collected throughout the PhD and are both collected from radars and other sensors such as GPS. The thesis has tree main contributions.

The first contribution of the thesis focus on using the kinematic and temporal information of the radar target. This is compared to only using the kinematics. The classifier uses geographical information such as if a target is over land or sea. The strength of the signal returned from the target is also used, referred to as intensity below. It is shown that by using the temporal information in the kinematics a better classification result can be achieved compared to not using the temporal information. We use Gaussian mixtures models (GMM) and recursive Bayes classifier as the classifier.

The second contribution of the thesis shows how to utilize the uncertainty in the position measurement. As radars have high uncertainty in the measured position compared to e.g. GPS, the deduced kinematics will also have high uncertainty. If this uncertainty is not handled in some way, misclassification or rapid changes in the classification results may occur. It is shown that by using the position uncertainty, a more robust classification can be made i.e. the probability for a target does not change as rapidly compared to where the uncertainty is not used. However, the computational load is very high when using the uncertainty and in general classification accuracy is not as good as when the uncertainty is not used. This work also uses GMM and recursive Bayes classifier.

The third contribution utilizes the work from the first and second con-

tribution described above by using the temporal information and implicitly including the uncertainty in the kinematic data. This is done by using a number of position measurements to create a feature vector. The feature vector consists of deduced kinematics from the position measurements. By using a Random Forest as the classifier, the features which do not contribute much to the classification rate will be weighted lower than features which contribute more. This implicitly includes the uncertainty in the deduced kinematic as each feature is not weighted equally in the classification results. Because the position measurements of stationary targets fluctuate, the targets can have high estimated speed. Therefore, an alpha beta filter has been used to classify into two classes which are stationary or moving, while the Random Forest classifies the moving targets from each other.

Resumé

Emnet for denne afhandling er målklassifikation af radarmål fra en 2D mekanisk scannende kystovervågningsradar. Målingerne fra radaren er positions data, og derfor er klassifikationen baseret på kinematisk data, som er udledt fra positionerne. Klasserne som målene vil blive klassificeret som, er typiske klasser for kystovervågning af for eksempel skibe, helikoptere, fugle og så videre. Endvidere skal klassifikationen foregå rekursivt, da en klassifikation af et mål er ligegyldigt, hvis først målet er forsvundet fra radarens søgeområde. Derfor skal sandsynligheden opdateres for hver måling. Data for dette arbejde er opsamlet igennem PhD studiet og er fra vidt forskellige sensorer såsom GPS og radar. Afhandlingen har tre hovedbidrag.

Det første bidrag for afhandlingen fokuserer på udnyttelsen af den temporale information af kinematisk data. Resultaterne er sammenholdt med kun at benytte den kinematiske data til klassifikationen. Geografisk data blev også brugt til at klassificere målene. De geografiske data blev brugt til at sige om målet er over land eller vand. Yderligere blev styrken af det modtagne signal brugt. Dette bliver senere omtalt som "intensitet". Vi viser at ved at bruge den temporale information i det kinematiske data, kan en bedre klassifikation af mål fortages, i forhold til hvis vi ikke bruger den temporale information. Vi bruger Gaussisk mixturemodeller samt en rekursiv Bayes klassifikationsmodel til at fortage klassifikationen.

Det andet bidrag omhandler hvordan positionsusikkerheden i positions målinger kan udnyttes. Da positionsmålingerne fra en radar har meget større usikkerhed i forhold til for eksempel GPS, har de afledte kinematiske data også stor usikkerhed. Hvis der ikke tages højde for denne usikkerheed kan det lede til misklassifikation eller meget hurtige ændringer i sandsynlighederne for de forskellige klasser. I det andet bidrag er det vist, at indarbejdning af usikkerheden i klassifikationsmodellen giver en mere robust klassifikation. Det vil sige at sandsynligheden ændrer sig ikke så hurtigt, som når man ikke indarbejder usikkerheden. Problemet er, at klassifikationen bliver meget beregningstung når usikkerheden indarbejdes, samt at den generelle klassifikationsrate bliver en del lavere i forhold til når usikkerheden ikke bruges. Lige som i det første bidrag, bliver der brugt Gaussisk mixturemodeller samt

en rekursiv Bayes klassifikationsmodel. Det skal bemærkes, at dette bidrag ikke bruger den temporale information i kinematikken.

Det tredje bidrag bruger erfaringerne fra de to første bidrag ved at bruge den temporale information samt indirekte bruger usikkerheden i positionsmålingerne. Metoden til dette er at bruge en række positionsmålinger til at lave en elementvektor, som består af afledte kinematik fra disse positionsmålinger. Ved at benytte klassifikationsmetoden Random Forest vil features med en høj usikkerhed, ikke blive brugt. Dette indarbejder implicit usikkerhed i målingerne, sådan at hver feature ikke får lige stor vægt i klassifikationen. Da stationære mål kan have høj estimeret hastighed på grund af fluktuerende positionsmålinger, beskrevet senere, bliver et alpha beta tracking filter brugt, sådan at det klassificerer stillestående/bevægende mål, hvorefter Random Forest så klassificerer de ikke stillestående mål i de forskellige klasser.

Contents

Thesis Details

Thesis Title:	Radar Target Classification using Recursive Knowledge-Based Methods
PhD Student:	Lars Würtz Jochumsen
PhD Supervisor:	Assoc. Prof. Jan Østergaard, Aalborg University
PhD Co-supervisor	Prof. Søren Holdt Jensen, Aalborg University
PhD Company supervisor	Morten Østergaard Pedersen, Terma A/S

The main body of this thesis consist of the following papers.

[A] L. W. Jochumsen, J. Østergaard, S. H. Jensen, M. Ø. Pedersen, "Modelling Temporal Variations by Polynomial Regression for Classification of Radar Tracks," *EURASIP, European Signal Processing Conference (EUSIPCO)*, pp. 1412–1416, 2014.

[B] L. W. Jochumsen, M. Ø. Pedersen, K. Hansen, S. H. Jensen, J. Østergaard, "Recursive Bayesian Classification of Surveillance Radar Tracks based on Kinematic with Temporal Dynamics and Static Features," *IET, International Radar Conference*, pp. 1–6, 2014.

[C] L. W. Jochumsen, E. Nielsen, J. Østergaard, S. H. Jensen, Morten Ø. Pedersen, "Using Position Uncertainty in Recursive Automatic Target Classification of Radar Tracks," *IEEE, International Radar Conference*, pp. 168–173, 2015.

[D] L. W. Jochumsen, M. Ø. Pedersen, S. H. Jensen, J. Østergaard, "Classify with Confidence: Exploiting Position Uncertainty in Recursive Radar Track Classification," *EURASIP, Journal on Advances in Signal Processing*, Submitted undergoing review.

[E] L. W. Jochumsen, J. Østergaard, S. H. Jensen, C. Clemente, M. Ø. Pedersen, "A Recursive Kinematic Random Forest and Alpha beta Filter classifier for 2D radar tracks," *EURASIP, Journal on Advances in Signal Processing*, Submitted undergoing review.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured.

Preface

This thesis concludes the work done from the 1st of April 2013 to the 31st of March 2016. The work has been carried out through a collaboration between Terma A/S and the Doctoral School of Engineering and Science at Aalborg University and is conducted as an industrial PhD. The thesis partially satisfies the requirements of the degree of Doctor of Philosophy stated by the Danish government in "ph.d.-bekendtgørelsen" [1]. The student has been employed at Terma A/S throughout this work.

The topic of the thesis is target classification of radar tracks. The classification is based on kinematic data derived from the position estimate of the targets. The radars used in this work are mechanically scanning 2D coastal surveillance radars which only delivers the position of targets. In addition to the kinematic data, geographical and radar specific data has also been used to assist in the classification. The first part of the thesis consist of an introduction to the radars and the data used in this work followed by the state-of-the-art within radar based target classification. Lastly a conclusion is made based in the hypothesis also introduced in this part. The second part of the thesis consist of the papers upon which the thesis is based. At the time of submission three papers were accepted while two papers were undergoing peer-review.

I would like to thank the Innovation Fund Denmark and Terma A/S for funding this work. I would also like to thank Aalborg University for accepting me as a PhD student. I would like to thank my company supervisor Morten Østergaard Pedersen for supporting me through this work, and always having time for discussions regarding the ongoing work, radars, questions and reviewing of the papers. Without Morten I would not have the required knowledge of radars which enabled this work. I would like to thank my main supervisor at Aalborg University, Jan Østergaard for in-depth discussion which ensured that I had the understanding for scientific communication and methodology which facilitated the scientific contribution of this

[1] Bekendtgørelse om ph.d.-uddannelsen ved universiteterne og visse kunstneriske uddannelsesinstitutioner (ph.d.-bekendtgørelsen). Accessed 8 March 2016. Available: `https://www.retsinformation.dk/pdfPrint.aspx?id=152430exp=1`

work. Jan has been essential for the quality of the work and papers written during this work. I would like to thank Søren H. Jensen for reviewing the papers and for being positive throughout my study. Thanks to Preben Schmidt Nielsen at Terma for believing in me throughout the project and after. Further I would like to thank Esben Nielsen, Kim Hansen, Christian Glinsvad, Rasmus Engholm, Sune Wolff and David Malmgren-Hansen for fruitful discussions of radar technologies, machine learning approaches and reviews of the papers. I would like to thanks Carmine Clemente from University of Strathclyde for hosting my trip abroad and discussions of my work. I would like to give my wife a special thanks for enduring me during this work. I would like to quote Ben King - stand by me "If the sky that we look upon, Should tumble and fall, And the mountains should crumble to the sea, I won't cry, I won't cry, no I won't shed a tear, Just as long as you stand, stand by me" I would also like to thank my family for always believe in me. You have taught me to be who I am.

Lars Würtz Jochumsen
Terma A/S and Aalborg University, April 4, 2016

Part I

Introduction

Introduction

In this thesis introduction an overview of the work and contributions are presented. The introduction starts with a short description of the overall problem statement, the sensors and data used in this work. This is done in section 1. In section 2, the hypotheses for this work is stated. State-of-the-art techniques and methods are described in section 3. The contribution of the work is stated in section 4 and finally in section 5, the conclusion and future work is described. The collection of the papers are shown in Part II.

1 Problem statement

In this section, the outline of the thesis is described. The problem statement that motivated this work is that the amount of information coming from a modern 2D coastal surveillance radar is very high and it is difficult to get a surveillance overview of a given area without a great deal of information overload. We define 2D radar as a radar which only measures the range and azimuth to the target. The focus of this work is on automatic classification of radar tracks such that a better surveillance overview can be achieved. In section 1.1, the definition and motivation of the work is described. In section 1.2, a description of simple radar theory is given. The radar used in this work is described in section 1.3. In section 1.4, a description of the other sensors used for this work is presented. Finally in section 1.5, the data and the classes are described.

1.1 Problem description

Radars have typically been used as the primary sensors for coastal surveillance. The radars are used for collision detection and prevention, intrusion detection, and general safety in harbors and near harbor areas [1]. This requires that even small targets can be detected in rough weather conditions like rain, fog, darkness and at a great distance. Under these conditions, sensors such as cameras are of little use. Surveillance radars which can detect

small targets are currently in a high demand.[2] However, with the increase of detectability of small targets comes an increase in false alarm rates, as small target such as birds, wakes from ferries etc. are detected. An image of a surveillance area, as seen from the radar observers perspective, is shown in Fig. 1. The scenario shows a rigid inflatable boat (RIB) which is marked with the red arrow. Most of the other tracks are birds. The circles are tracks and the angles of the lines from the circles indicates headings, and the lengths are proportional to the speeds of the tracks. As it can be seen from this image, it is nearly impossible to spot the RIB. This makes threat assessment difficult, as it is difficult to get a surveillance overview. There can be several approaches to reducing the number of tracks observed. One approach is to reduce the sensitivity of the radar. However, this approach also reduces the probability of detecting small but important tracks such as the RIB in Fig. 1. This reduces the situational awareness. Another approach is to make very simple classification based on the geographical location and direction of the targets. This approach has some disadvantages such as the direction and geographical location of targets are unique to each radar site. Additional, the weather condition can also make it difficult as wind speed and direction can change e.g. sailing boats typically sailing paths. By this approach, suspicious behavior can be disguised by simply moving in these geographical locations. This work takes the approach that classification of each track is necessary to limit the number of tracks, such that only the most important tracks are presented to the observer. This requires that the classifier must classify tracks on-line in a recursive manner. In this work, on-line is defined as incrementally updating the current probabilities of the classes upon receiving new information. It is therefore necessary to have a recursive classification approach. The classification must use a probabilistic approach so that an observer can choose a sensitivity of publishing tracks. Instead of only using position, which is unique for every site a radar can be placed, we use kinematic information derived from the position measurement i.e. speed, acceleration etc. which is independent of the geographical radar site.

To clarify what classification means in this work, we use the definition in [2] to describe the order from detection to fully identified target. By this definition, five stages exist in uniquely identifying a target. First a target must be detected, then it must be discriminated from other targets before the target can be classified, then recognized and lastly identified. We use the definition from [3] to describe the classification, recognition and identification. The definitions are:

Classification: "Recognition that the echo on a radar display is that of an aircraft, ship, motor vehicle, bird, person, rain, chaff, clear-air turbulence, land clutter, sea clutter, bare mountains, forested areas, meteors, aurora, ionized

[2]This is a statement from Terma A/S.

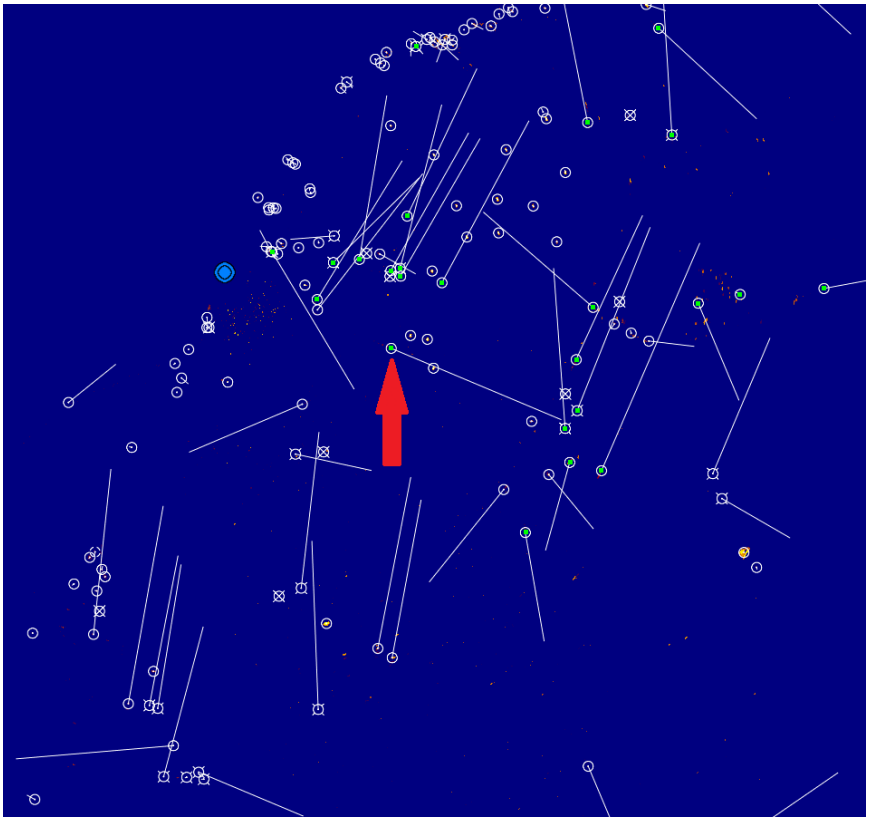


Fig. 1: The figure shows a typically screen dump from a radar scenario. The circles are tracks and the line from each circle is the track heading and estimated speed. The red arrow pointing on a rigid inflatable boat (RIB). Many of the moving tracks are birds. As it can been seen it is difficult to get a surveillance overview, as there are many bird tracks. The blue circle is the location of the radar. The yellow and red colors are the backscatter from the environments (later known as video) and the green dots are the video which is above a threshold and symbolize a measurement (later known as a plot)

media, or other natural phenomena. A trained and experienced radar operator with the right type of radar should be able to sort these broad classes of target echoes from one other." [3, p. 370] [3]
Recognition: "This includes recognizing a fighter aircraft from a multi-engine bomber aircraft, a cargo ship from a tanker, a tracked military vehicle from a truck, chaff rather than a ship, a buried rock instead of a mine; or a surface-to-air missile site from a dump site." [3, p. 370]
Identification: "This involves determining the particular class to which a target belongs among the many possible classes. For example, if the radar believes it is detecting an aircraft, is it an F-18, F-22, MIG-31, B-2, A-6, Rafale-2000, or something else? If the target is a ship, does it belong to the Aegis destroyer Class DDG-51, Aegis cruiser CG-41 Kara, Sovermeney, or so forth, or is the echo that of a chaff decoy? If it is a bird, is it a starling, mallard, or what else?" [3, p. 370]

The purpose of this work is to do classification of targets e.g. does the target belong to a ship, a bird etc.

1.2 Radar theory

In this section, a short description of the general radar theory will be presented. The section starts with the most simple radar systems and then continues with a short description of the most common radar types.

Simple radar systems

This section is primary based on [3]. The term radar is an acronym of radio detection and ranging. The transceiver of the radar emitters an electromagnetic pulse (EM pulse) through an antenna. This EM pulse travels through the air until it meets an object, here some of the pulse will be reflected from the surface of the object and some will be absorbed. The reflected EM pulse will then travel back to the antenna and will be received by the radar system. The duration the EM pulse take from leaving the radar until it returns Δt_r will give the distance r to the object by $r = \frac{\Delta t_r c}{2}$ where c is the speed of light and $c \approx 3 \times 10^8 m/s$, see Fig. 2. For a radar with the same antenna used for transmit and receive, the received power p_r can be calculated as

$$p_r = \frac{p_t A_e^2 \sigma_{\text{rcs}}}{4\pi\lambda^2 r^4}, \tag{1}$$

where p_t is the transmitted power from the radar, A_e is the receive effective area, σ_{rcs} is the radar cross section (RCS), which is the "size" of the target and λ is the wavelength of the transmitted signal. The RCS depends on the angle

[3] The name clutter is a common name used for unwanted backscatter from targets. This can be from rain, sea, buildings, birds, etc.

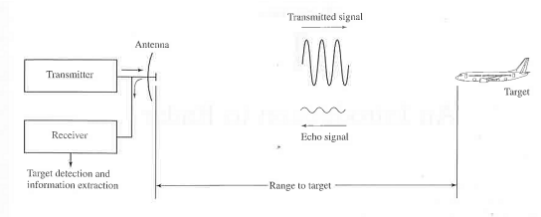


Fig. 2: A figure showing a radar transmitting a signal and receiving the backscatter signal. The signal is transmitted from the transmitter and into the air. The reflection from a target is reflected back to the receiver. From [3]

Table 1: Typically values of RCS for different targets. The values are taken from eihter [3] or [4]

Target type	RCS $[m^2]$
Small, single engine aircraft	1
Helicopter	3
Jumbo jet	100
Small open boat	0.5-5
Small metal ships	10-100
Birds	up to 1

to the target, the frequency used and many other parameters. It is in general difficult to calculate the RCS of a real target. However general examples are given in Table 1. As it can be seen from (1) the range to the target is a very important factor for the received power. By increasing the detection range by a factor 2 requires increase of transmitted power by a factor of 16. The receiver cannot detect echoes from the backscatter before the transmitted pulse is finished. If the length of the pulse is increased an integration of the pulse length can be made, which increase the effective energy such that the radar can detect the target. However, a longer pulse decreases the area of surveillance close to the radar. Therefore, a compromise must be made between how far the radar can see and how close to the radar a target can be seen. The pulses are repeated and the repetition time is called t_{pr}. The faster the repetition time, the faster updates from the radar. However, for fast repetition rates, a smaller amount of time is available to transmit the power i.e. less energy. For fast repetition rates combined with high transmit power, a range ambiguity problem can occur. This can be seen in Fig. 3. The range ambiguity is a problem if the range to a target is larger than $\frac{ct_{pr}}{2}$ as the transmitted pulses cannot be distinguished from each other.

The resolution in range for the radar is given by (2)

$$\Delta R = \frac{c\Delta t_c}{2}. \tag{2}$$

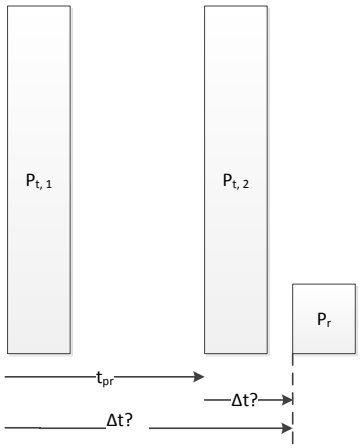


Fig. 3: The range ambiguous problem can be seen here. Is he received backscatter pulse P_r from the $P_t, 1$ or from $P_t, 2$?

As it can be seen the smaller pulse length the better resolution. Again a compromise must be made between the transmitted power i.e. the length of the pulse, and the resolution. To compensate for this problem, pulse compression can be used. That is, we change the pulse to a known sequence e.g. a chirp. By knowing the transmitted chirp, a cross correlation can be made of the received signal such that the chirp length Δt_c can be large while maintaining a high range resolution. By using the pulse compression the range resolution is now given by

$$\Delta R = \frac{c}{2B_t}, \tag{3}$$

where B_t is the sampling bandwidth assuming that the bandwidth of the chirp uses all the sampling bandwidth. To overcome the range ambiguity problem, different carrier frequencies in the chirp can be used, such that it is possible to know which successive chirp the backscatter originated from.
The main types of radars are:

- Passive
- Monostatic
- Bistatic
- MIMO
- SAR

The passive radar does not have a transmitter but instead utilize other transmitting units such as digital video broadcasting - terrestrial (DVB-T) transmitter [5, 6]. The monostatic radar has a transceiver, which transmits and

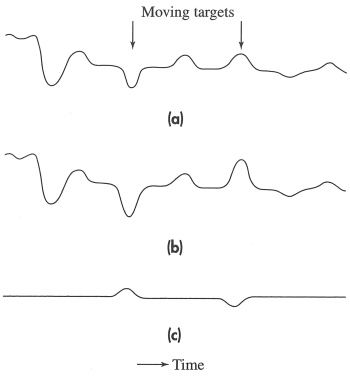


Fig. 4: Figure of a first order MTI filter. The two backscatter from two successive pulses are shown in (a) and (b). In (c) the subtraction is shown. The image is from [3, p. 110].

receives the EM pulse. The bistatic radar is where the transmitter and the receiver are separated physically from each other by a fixed distance. All of these radars have an azimuth resolution given by the antenna beam. The azimuth resolution is given by the antenna beam, and the antenna beam is described by the effective size of the antenna and the carrier frequency given by

$$\Phi_B \approx 65\frac{\lambda}{L_a}, \tag{4}$$

where L_a is the length of the antenna. To separate two small targets at a great distance an antenna with a significant size is required[4]. Multiple input multiple output (MIMO) radars are an expansion of bistatic i.e. a combination of more than two static radars are used. The azimuth resolution is given by the distance between the different radars. As an alternative to the above mention radars, the synthetic aperture radar (SAR) has been invented. The SAR radar simulates a bigger antenna by moving the antenna along the target. The SAR radar requires that the targets are stationary or a Doppler compensation of the target much be made. The resolution of a very common dataset, MSTAR [7], used for SAR recognition is 0.3 meters (1 foot) [8].

As much clutter is stationary, moving target identification (MTI) filter can be used to suppress stationary targets. An MTI filter exploits the fact that more pulses hit the target. A first order MTI filter subtracts two successive complex pulses. This can be seen in Fig. 4. The transfer function of an MTI filter can be seen in Fig. 5 As it can be seen, a first order filter has two main problems. The first is that the frequency response is not a brick wall filter i.e. the change in gain per frequency is low. This can be changed by using

[4]The antenna must also be very large to give an estimate of the size of the target. For a distance of 20 km and an antenna size of 21 foot and 10 GHz the resolution is about 106 meters.

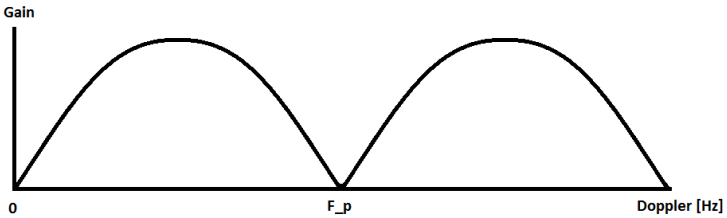


Fig. 5: A sketch of the transfer function for a first order MTI filter.

Table 2: The typically bands for radars used for surveillance purposes [3]

Band	Frequency [GHz]
UHF	0.3 - 1.0
L	l - 2
S	2 - 4
X	8 - 12
Ku	12 -18

a higher order MTI filter. This requires longer dwell time i.e. for scanning radars a longer scanning time. A first order MTI filter has a zero in the transfer function for every multiple $n \times f_p$ where $f_p = \frac{1}{t_{pr}}$ and $n \in \mathbb{N}$. A solution to this can be using staggered pulse repetition frequencies[5], which change the t_{pr} for each pulse. Instead of using an MTI filter the Doppler information from the target can also be used. The Doppler frequencies f_d is given as

$$f_d = \frac{2V_r}{\lambda}, \tag{5}$$

where V_r is the radial velocity.

There are at least two ways of making a scanning radar: a mechanically scanning antenna and an active electronic scanning antenna. The azimuth resolution is given by how exact it is possible to measure or calculate where the main antenna beam is. In this work two different terms for resolution is used. The cell resolution, which corresponds to the pixels resolution in a normal image and the actual resolution which is the radars capability to discriminate targets.

The typically carrier radio bands for radars can be seen in Table 2. With a low carrier frequency, a smaller bandwidth of the chirp is possible. This means lower range and azimuth resolution. However, the advantages of using a lower carrier frequency is that the EM chirp can curve around the horizon and it is easier to produce power to transmit i.e. it is cheaper to see things

[5]Staggered is a common word, in radar, to describe a sequence of different pulse repetition frequencies.

at larger distance. This holds even though the wavelength is squared in (1).

1.3 The radars used in this work

This section is based on [9–11] and internal knowledge at Terma A/S. The radars used for this work are SCANTER radars from Terma A/S. The SCANTER 5000 and 6000 are two series in the Terma radar portfolio. The radars are primarily used as coastal surveillance (CS), vessel traffic service (VTS), shipborne radar system (SRS) and surface movement radars (SMR). The SCANTER 5000 and 6000 uses pulse compression and has a peak power of either 50 W or 200 W. The amplifier is a solid-state power amplifier (SSPA) with duty cycle of maximum 20 %. The SCANTER 4000 radars have also been used in this work. This radar is used in wind farm area surveillance (WFAS), as primary surveillance radar (PSR) and SRS. The radar is a pulse compression radar with a travelling wave tube (TWT) amplifier which delivers 6 kW or 12 kW peak power with a duty cycle of 5 %. All of the radars are X-Band radars with the possibility to use multiple carrier frequencies. The range resolution are 3 meters after the pulse compression. However, to remove time sidelobes from the pulse compression, a window function is used which makes the actual range resolution 12 meters while the radar cell resolution remains at 3 meters. All of the radars deliver normal processed video (NR)[6]. Further, some of the radars also have MTI video available. MTI makes it possible to reduce the number of stationary target. This is especially useful if a radar observer only wants to see moving target. The MTI filter is a seven order filter with staggered pulse repetition frequencies. The radars have an embedded tracker [12]. The tracker informs the radar observer with speed, position and the plot of the target. A plot consists of a number of different metadata about the associated measurement. The plot includes as a minimum: measured position, position uncertainty and a time stamp of the observation. The plot can also contain the intensity in the backscatter from NR and MTI channel. The intensity is related to the RCS of the target σ_{rcs} used in (1). In Fig. 6 the position and uncertainty of a target is shown . The target consist of backscatter which expands multiple radars cells i.e. the blue cells. Each cell has an intensity, like a grayscale BMP image, and the position estimate is a weighted average of the intensity and the position of the cell. The calculation of the position is done as

$$\boldsymbol{\mu} = \frac{1}{\sum_p e(\hat{r}_p)} \sum_p \hat{r}_p e(\hat{r}_p), \tag{6}$$

where $\hat{r}_p = [r_p, \phi_p]^T$ is the pth range, azimuth cell respectively, $e(\hat{r}_p)$ is the intensity in that radar cell, and $\boldsymbol{\mu} = [\mu_r, \mu_\phi]^T$, which is the estimated mean in

[6]Video is a common word for describing the processed backscatter from the transmitted pulse [3]

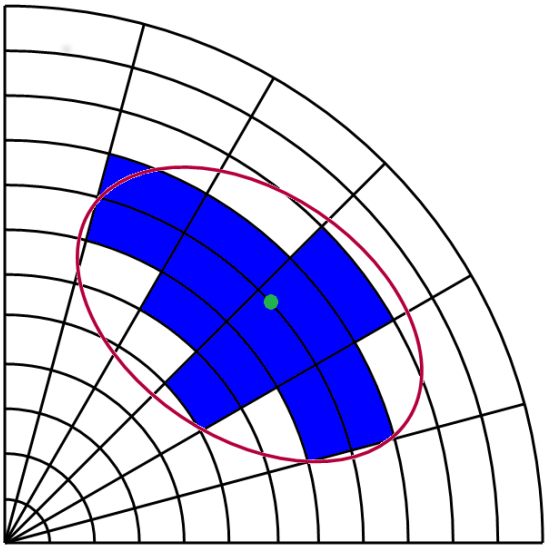


Fig. 6: The blue color cells are the backscatter from the target which is above a threshold i.e. there must be a target. The green dot is the calculated center of mass i.e. the measured position of the target, the red ring around the cells is the estimated ellipsoid, which corresponds to an uncertainty of the position.

range and azimuth respectively. The variance of the position i.e. the uncertainty, is given by

$$\hat{\sigma}^2 = \frac{1}{\sum_p e(\hat{r}_p)} \sum_p (\hat{r}_p - \mu_r)^2 e(\hat{r}_p), \tag{7}$$

where $\hat{\sigma}^2 = [\sigma_r^2, \sigma_\phi^2]^T$ are the variance of the range and the variance in azimuth respectively. The covariance of the range and azimuth is

$$\sigma_{r,\phi}^2 = \frac{1}{\sum_p e(\hat{r}_p)} \sum_p (r_p - \mu_r)(\phi_p - \mu_\phi) e(\hat{r}_p), \tag{8}$$

and the full covariance matrix for the position

$$\Sigma^P = \begin{bmatrix} \sigma_r^2 & \sigma_{r,\phi}^2 \\ \sigma_{r,\phi}^2 & \sigma_\phi^2 \end{bmatrix}. \tag{9}$$

The intensity is defined as the highest value of all the cells in a target

$$I = \underset{p}{\operatorname{argmax}} \left(e(\hat{r}_p) \right). \tag{10}$$

Because of adaptive preprocessing in the radar, the radar cells intensity can change from scan to scan as the gain through the radar system was unavailable.

The radar has an adaptive preprocessing step consisting of a constant false alarm rate (CFAR) regulation and a sea clutter discriminator (SCD). The CFAR process is adjusting video level such that only a specific number of video cells are allowed every second i.e. a high number of returns will likely decrease the video level. The SCD also adjust the video level. However, the SCD includes the history of the area of up to two scans back in time.

The antennas used are either horizontal polarized, which reduces sea clutter or circular polarized which reduces the clutter from rain [3]. The length of the antennas are from 12' to 21'. These antennas are mechanically rotating and uses a slotted waveguide as the radiating element. The antennas rotation rate is given by the desired detection range. In general, the rotation speeds are from 20 to 60 RPM i.e. the update rates are from 0.33 to 1 Hz. The rotational speed of the antenna is set such that approximately eight chirps will hit a given area at each scan.

The limitations of the radars used in this work are that the radars only have 2D information namely slant range[7] and azimuth to the target i.e. we do not have any height information of the target. Further, because of the relative few hits on the target of the chirps sent from the radar at each scan it is not possible to get informative Doppler information of the target. As mentioned above the intensity is related to the RCS. However, because of the adaptive processing the intensity may vary from scan to scan and from radar to radar. The intensity also depends on the antenna used. In general, for the radars used in this work, the intensity has high deviation from scan to scan and radar to radar. As this work is for coastal surveillance purpose, the range to a target can be up to 20 km as the radar horizon is typically longer than 20 km, which means that nearly every target will look as a point target in azimuth because of the antenna beam size. At best case, i.e. by using a 21' antenna, the azimuth resolution will be $\approx 100m$. This means that only the largest ships will be larger then the azimuth resolution. The azimuthal extent can therefore not be used to estimate a targets size as nearly every target will be point targets. It is possible to use the range resolution however, the range resolution is not high enough to use as high resolution range profiles (HRRP) described in section 3. This is because a target smaller than 12 meters will be seen as a point target in range. The video the radar delivers is optimized for human viewing and the preprocessing in the radar is optimized for target separation and not for target classification. As targets typically accelerate or decelerate and the antenna speed depends on the mode of the radar i.e. if the radar is set to view objects at great distance, the RPM is lower than if it is set to view targets at a short distance. Therefore the sampling rate of the targets is not deterministic and equidistant. In general, very few features are

[7] The slant range is the actual distance from the radar to the target and can be different from the ground range.

available for target classification. Because of this we limit the features to:

- Kinematic data, such as speed, acceleration, normal acceleration etc. These features are derived from the estimated position and as the position estimate has a relative high uncertainty the kinematic data will also have a high degree of uncertainty.
- Radar specific features, such as intensity for NR and MTI. The intensity is different from radar to radar and can vary with range and weather condition. Therefore intensity should be used carefully.
- Geographic features such as whether the target is over land, sea, road etc. The feature for land and sea is estimated from maps, which can have errors or missing areas. As an example, some of the harbor in Aarhus, Denmark is not in the map material we use. The estimate of a road also has high uncertainty, as there also can be errors in the geographical position of the roads.

For most of the work done in this thesis, the main feature is speed. This is because, as it can be seen latter in Fig. 8, the speed appears to be the most distinct feature for the different targets.

1.4 Other sensor used in this work

Besides the radar data, other sources and sensors have been used to collect the data. The sensors are GPS loggers, automatic dependent surveillance – broadcast (ADS-B) [13] and automatic identification system (AIS) [14]. ADS-B is a broadcast system used in commercial and private aircraft. The ADS-B system broadcast the position of the aircraft with regular intervals together with a unique number which identifies the aircraft. The AIS system is a broadcast system for ships above 300 tons [15]. The AIS sends at non regular intervals vessel position and a unique ID called Maritime Mobile Service Identity (MMSI). The update times depend on how fast and large the ships are. It has been shown in [16] that it is possible to tamper/make mistakes with the AIS signal, which can make the use of AIS tracks difficult. This is also possible for ADS-B [17]. However, in this work we assume that the dataset of AIS and ADS-B are without errors. Because of the different sensors used to gather the training data, a forward and backward Kalman filtering, known as fixed interval smoothing [18], was used to derive the best estimate of the position data. In [19] a compression of a normal forward Kalman filter and fixed interval smoothing is done on a spring damped system. It is shown that the fixed interval smoothing has a better performance than the forward Kalman filter. The fixed interval smoothing was done to make the position uncertainty in the training data negligible and thereby make it possible to use the data from the different sensors. However, it is not possible to do the

fixed interval smoothing on the test data as this requires knowledge of both past and future data.

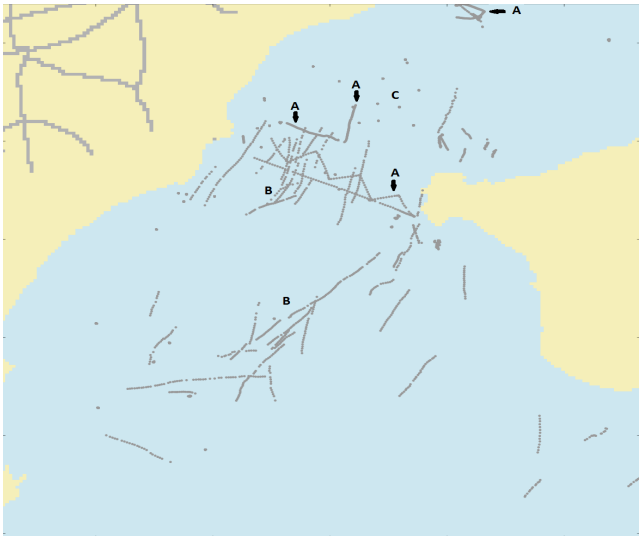
1.5 Target classes and data

Two data collection periods were used in this work, one at the first part of this work i.e. 2013 and a second at the summer 2015. The first collection of data was made by viewing a lot of prerecorded video from Terma A/S radars. All of the tracks were manually classified and the tracks and plots were extracted. The data format was amended such that all of the necessary metadata about the tracks was included. The manually classified data was split into two sets, a training dataset, and a validation dataset [20]. For the test set, some real world scenarios are used [20] [8]. Multiple test sets are used through this work. The scenarios primarily contain unlabeled data, however some controlled tracks are present in the scenarios. For some of the tracks, we have a qualified guess of the target class. The sizes of the training and validation set can be seen in Table 3. The validation set is used to get a confusion matrix and adjust the hyper-parameters in the classifiers. The test is used only once to show the real performance of the classifier. By only using the confusion matrix it can be difficult to show the real world performance, where tracks will be suppressed depending on the probability i.e. the confidence of the classification. This is because the confusion matrix does not show the probability of the classified tracks but only the class with the highest probability. We therefore show some scenarios where the tracks in these scenarios are colored depending on the classifiers output of the class. If the probability of the class is below a certain threshold, the tracks are colored gray. Different classes have been used throughout this work however, some common classes have been used. These are: Birds, large ship, rigid inflatable boat (RIB), stationary sea targets (wind turbines and sea buoys). In addition to these classes, commercial aircraft and helicopters are also used in most of the work.

There are two different scenarios. The first scenario is the Egaa scenario. This scenario has a RIB sailing back and forth at the Aarhus bay area, Denmark. Additionally a large number of birds as well as stationary sea targets are present. The second scenario, Horns Rev, is a wind farm scenario where two wind farms are present and a vast amount of birds. Additional, a commercial aircraft, a general aviation aircraft and some vessels are also present in the scenario. The scenarios can be seen in Fig. 7

The second data collection was in the summer 2015 and was from a radar at the geographical location of Terma A/S headquarters. The surveillance area was Aarhus bay area. The data was recorded over two weekends. More

[8] A scenario is defined as an image of all the tracks in a given time period from a real radar recording.



(a) The Egaa scenario, this scenario includes the following clases: birds (B), sea vessels (A) and stationary targets (C).

(b) The Hornsrev scenario, this scenario includes targets like birds (B), aircrafts (D), sea vessels (A) and stationary targets (C).

Fig. 7: Here an overview of the two scenarios is shown. The best guess on the targets are shown with an arrow or without. If the letter is followed by an arrow this means that it is a specific track and if no arrow is following the letter is means that multiple tracks of this class exists in the area. The best guess is limited to; sea vessels (A), birds (B), stationary targets (C), aircraft (D).. In Fig. 7a the Egaa scenario is shown, in Fig. 7b the Hornsrev scenario is shown.

Table 3: The sizes of the training and validation set used for all papers except E. The size is the number of updates (plots).

Classes	Trainings size	Test size
large boat	23130	2462
stationary sea	13662	7492
RIB	23470	720
helicopter	19143	205
commercial aircraft	6725	8840
birds	3289	262

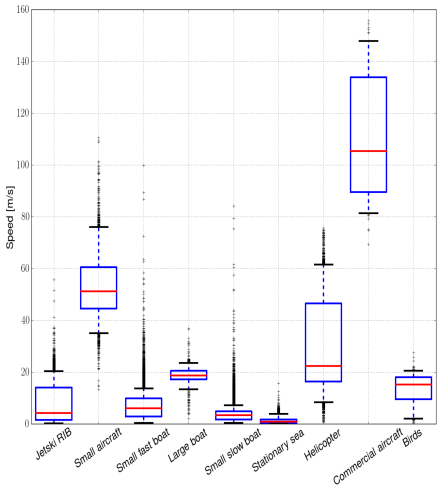
Table 4: The table shows the number of training size which is used in paper E. The size of the training set is the number of updates (plots).

class	size
Birds	1441
Stationary sea	3402
Jetski/RIB	14720
Helicopter	24089
Commercial aircraft	7179
Large ship	3491
Small fast boat	61628
Small slow boat	77099
Small aircraft	2917

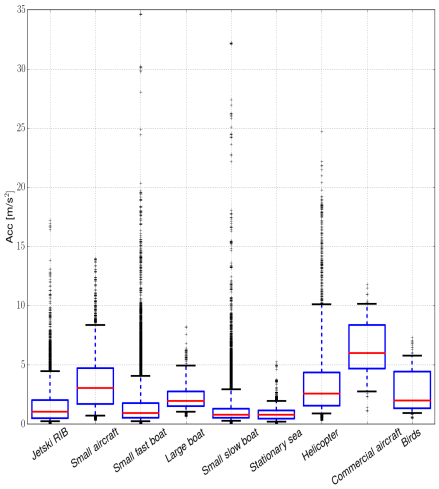
than 10,000 tracks were collected and over 2000 tracks were manually classified into the following categories: Commercial aircraft, jetski/RIB, small fast boat, small slow boat, small aircraft, helicopter and high-speed ship. As no ground truth was available, misclassification can occur. Some of the classes have very similar properties such as helicopter and small aircraft. It is therefore possible that the training database for small aircraft contains helicopters. This is due to the fact that helicopters only were classified as helicopters if they at some point in the track was hovering. The size of the training set used for paper E can be seen in Table 4.
The definition of the classes used in this work:
Birds: The class is primarily birds which are normally located at the sea e.g. seagulls. The training data is manually classified without ground truth and knowledge about the tracks real class.
Stationary sea: The class contains wind turbines and sea buoys. The training data is manually labelled without ground truth. A map has been used to label the tracks from the radar.
Jetski/RIB: The class contains sea vessels, which are fast and small. These vessels are characterized by having high acceleration and low turn radius.

Some of the tracks are from a GPS others are manually labelled without any ground truth.
Helicopter: The class contains targets, which can move over both sea and land. The target can hover and fly up to 300 km/h. Some of the tracks are from GPS logs; others are manually classified without any ground truth.
Commercial aircraft: This class contains commercial aircrafts, typically turbofan engine, i.e. high speed. The data are from ADS-B sensors however, some are also from radars. The data from the radars are manually classified without ground truth.
Large ship: This class contains sea vessels such as cargo ships, and high-speed ferries, the common features of these vessels are that the speed and heading are constant and they have high intensity. However, the class has high variance because of the differences in the speed range for the different vessels. Some of the data are from AIS others are from radars. The radar data are manually classified, some of it with assistance of AIS tracks, some without ground truth.
Small fast boat: This class contains fast boats without the high acceleration and turn rate as jetski/RIB. The data are manually classified without ground truth.
Small slow boat: This class contains small slow moving boats. These are typically sailing boats, slow moving motor boats, fishing ships and towing ships that do not have AIS transponder on-board. The training data are manually labelled without ground truth.
Small aircraft: This class consists of small aircrafts. The speeds are much slower than commercial turbofan aircraft. The training data are manually labelled without ground truth. Some confusion between helicopters and this class can occur, as it is possible that some helicopters, which do not hover, are classified as this class.

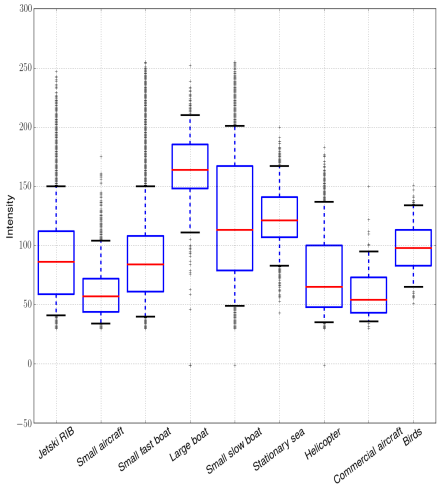
The classes and their typical properties can be seen in Fig. 8. It is clear that some outliers exist in the data. For example it is not possible for a small fast boat to sail with a speed of 100 m/s. The data are shown with a boxplot, where the 5 %, 25 %, 50 %, 75 % and 95 % percentile is shown. It can also be seen that some classes are very similar in speed and acceleration. For example large ships and helicopters overlap in speed. Radar intensity may assist in classification, however this will require keeping track of system gain i.e. the adaptive processing, to allow an RCS estimate. This was not implemented at the time of this work. It can also be seen that stationary targets have a fairly high speed. The reason for this is that wind turbine blades can move from scan to scan, another reason is because of the uncertainty in the position measurement since the target position will likely fluctuate from scan to scan. The classes from the first and second collection periods which match each other are merged to one class. The second collection of data is only used for paper E.



(a) Speed for the different classes



(b) Acceleration for the different classes



(c) Intensity for the different classes

Class	Sea	Land
Jetski/RIB	X	
Small aircraft	X	X
Small fast boat	X	
Large boat	X	
Small slow boat	X	
Stationary sea	X	
Helicopter	X	X
Commercial aircraft	X	X
Bird	X	

(d) Possible terrain for the different classes

Fig. 8: An overview of the kinematic and terrain features for the different classes. The blue box represents 25 % and 75 % percentile and the whiskers shows the 5 % and 95 % percentiles. The red line represents the median of the data.

2 Hypotheses

In this section the hypotheses for this work will be stated. We base our work on two hypotheses:

- Based on a temporal behavioral analysis of radar track evolution, is it possible to achieve high-accuracy radar track classification. The temporal behavior analysis can be based on e.g. kinematic parameters of the targets such as speed or acceleration, or on the target appearance in the radar image (size, extent, structure, aspect ratio) or geographical such as location (land, water, road etc).
- Based on track classification, is it possible to remove or suppress unwanted tracks without affecting the main performance criteria such as small target detection capability.

3 State-of-the-art in Radar Classification

In this section the state-of-the-art is described. The section describes in brief different approaches to classifying radar targets. The approaches differ in the amount of data and features available from the radar e.g. a SAR radar delivers images with a spatial equidistant sampling and can best be compared to aerial image performance whereas a 2D scanning radar only deliver range and azimuth to the target.

The most promising classifier for images recognition is deep convolutional neural network (Deep CNN). It has for example shown consistently good results on imageNet [21]. As synthetic aperture radar (SAR) recordings are nearly similar to optical images, a new research area of using Deep CNN in SAR images has emerged. The following papers [22–24] all use Deep CNN to classify SAR image targets. One of the major drawbacks of using Deep CNN is the huge amount of training data that is required. Therefore, it is also difficult to use Deep CNN in SAR, as the data are very limited. Other SAR recognition approaches are using feature extraction and then different classifiers like for example k-nearest neighbors (KNN) [25]. Another approach is to use high-resolution range profiles (HRR-P) extracted from SAR images. In [26], support vector machine (SVM) is used to classify different fighter aircrafts. The features are from HRR-P. It is shown that the SVM is better than maximum likelihood (ML) and Fisher linear likelihood (FLL) classifiers. In [27], they use information from HRR-P, inverse SAR (ISAR) and SAR to make target classification. The classification is used to assist the tracking of the targets as a better estimate of the kinematic performance when the target type is known. In [28], HHR-P is used to estimate pose angles and orientation for a target, which can help a kinematic classifier that uses an

interacting multiple model (IMM) filter. In [29], feature extraction is done by using Fourier-transformation. Feature reduction is done by using a linear discriminant analysis (LDA). Bayes classifier is used as the classification algorithm.

When using radars that does not have high range resolution or nearly optical imaging performance, such as SAR and ISAR, the Doppler information can be used. In [30], they extract bispectrum features from a Doppler surveillance radar. They use a Gaussian mixture model (GMM) as their classifier. The purpose of the classifier is to classify humans walk in a vegetation cluttered environment. The results use real radar data to evaluate the performance. In [31], a Doppler radar is used to classify human walk. The classification is done by GMM and SVM. The feature is from the short time Fourier transform (STFT) i.e. the spectrogram. Further, there are eight classes where most of them are humans walking, however trucks and clutter are also present as classes. The best performance is achieved with SVM with a prediction accuracy of 95.97 % however, nearly the same performance is achieved by the GMM at 95.69 % accuracy. In [32], micro-Doppler is used to classify simulated radar data. The micro-Doppler according to [33] is defined as the Doppler shift from a part of a target. This can be the Doppler shift from a bird (target) wings (part). They use Gabor filtering as feature extraction and compare SVM, KNN and Bayes linear classifier. The best results are achieved with SVM. In [34–36], micro-Doppler is used to classify small rotary wings like quad copters and radio controlled helicopters. In [37], they analyse the micro-Doppler signature of different animals.

Almost all of the work described above does not utilize recursive classifiers however, as we do not have all necessary information at the start of a track, a recursive classifier is necessary. One of the most common approaches is the hidden markov model (HMM), which is mostly used in speech recognition [38, 39]. The problem of using HMM for target classification of radar tracks is that the HMM requires a finite number of states possible i.e. human speech has only a finite number of possible sounds. This is not applicable as the speed is a continuous variable, which requires infinite many stages. The speed can be discretized into a finite number of bins, which makes it possible to use HMM. In [40], a histogram of the Doppler information is used to classify vehicles with HMM. Deep recursive neural networks (RNN) are also emerging for example for speech recognition [41, 42]. Similar to the problem with using deep CNN in SAR image, the required amount of training data, which must be used in the deep RNN, is too high for the data collected in this work. In [43], an introduction to Bayes classifier and the recursive version is made. The recursive Bayes classifier is used through this work and is explained in 4.1.

All of the work mentioned above have in common that more features are available for the target classification compared to this works. When only the

position data is available, the amount of related work done is very limited. This can be where they use the position of tracks to detect anomaly behavior. This has been done in [44, 45]. In [44], an estimated model of the main shipping lanes is made by adaptive kernel density estimator (adaptive KDE) and a GMM. It is shown that the adaptive KDE is better in modelling the shipping lanes than GMM. In [45], KDE and particle filters are used to check for anomalies in the path and kinematic of incoming AIS tracks. In [46], GPS tracks are used to classify when a person is walking, biking, taking a bus etc. Speed is not used as this will be too dependent on traffic and traffic lights. Instead, the use of acceleration, direction, turning, are used as the features because these are more independent on traffic and traffic lights. In [47], they use a classifier to predict if a human is travelling by bus or car. The features used are distance between stops, the number of stops etc.

The most common approach for target classification of radar tracks where the primary data are position dependent features, is the joint tracking and classification algorithms. The joint tracking and classification is used in e.g. [48–52]. In [48], two approaches are used. One, where a multiple model particle filter is used and one where a multiple model Kalman filter is used. The features are only kinematic and it is a two class problem i.e. commercial aircraft and military aircraft. The method show good performance. In [49, 51], a particle filter is used. In [50], a comparison is performed of Kalman filter used in either a probabilistic or transferable belief model (TBF) framework. It is argued that TBF is best. In [52], an ESM sensor [53] and speed is used to classify tracks with a multi model particle filter.

One of the problems of using kinematic filters like Kalman and IMM filters is that it is difficult to get a state-dependent filter i.e. a car driving 130 km/h on a highway with speed limit at 130 km/h, the likelihood for changing the speed becomes biased towards deceleration. Particles filters can solve this. However, the computational load for a particle filter makes it inappropriate to use in an on-line and recursive classifier.

Very limited work has been done where the only information is position i.e. kinematic and where machine learning techniques have been used i.e. feature extraction and different kinds of classifiers. In [54], they use GPS logs to classify trucks from cars. Only kinematic data are used with SVM as the classifier. The classifier is not recursive. In [55], a combination of HMM and fuzzy logic is used to classify radar tracks. However, other features such as RCS and Doppler information is also utilized. In [56], a decision tree is used. In addition to kinematic data, Doppler and height information is also used in the features. In common for most of the related work is that they solely use simulated data. The exception is [54] but here other features are utilized than the kinematic information.

4 The Contributions for this work

In this section we describe the contribution of this work. However, as the same fundamental theory is used in most of the work, i.e. recursive Bayes classifier and GMM, a short description is given here. All the code in this work is written in Python 2.7.5. Multiple libraries have been used in this work however, the primary libraries are Numpy 1.8.2, Matplotlib (Pylab) 1.5.1 and Scipy 0.16.0. For Paper E, the library Scikit-learn 0.16.1 has also been used. Most of the python parsers and the geographic information system (GIS) were written by the author of this thesis. All of the classifier algorithms are written by the author using the libraries mentioned above. The work has been carried out on a Lenovo Thinkpad W520 with a Fedora 20 environment with 8GB ram, core i7 2670QM 2.2GHz and Nvidia Quadro 1000M.

4.1 Recursive Bayes Classifier

The Bayes classifier is simply defined as

$$P(c_i|X_n) = \frac{P(X_n|c_i)P(c_i)}{P(X_n)}, \tag{11}$$

where c_i is the ith class and X_n is the nth measurement. If X_n consists of multiple features and these are mutually independent, it is possible to write $P(X_n) = \prod_f P(X_n(f))$ where f is the fth component of the feature vector. If X_n is mutually independent of all previous and future measurements, i.e. $P(X_n|X_{\hat{n}}) = P(X_n), \forall n, \hat{n}, \hat{n} \neq n$, for the recursive Bayes classifier, we can write

$$P(c_i|\{X_n\}) = \frac{P(X_n|c_i, \{X_{n-1}\})P(c_i|\{X_{n-1}\})}{P(\{X_n\})}, \tag{12}$$

where $\{X_n\} = \{X_n, X_{n-1} \cdots X_0\}$. If X_n is governed by a first order Markov chain, i.e. $X_n \leftrightarrow X_{n-1} \leftrightarrow \{X_{n-2}\} \forall n$[9], we can modify (12) to

$$P(c_i|\{X_n\}) = \frac{P(X_n|c_i, X_{n-1})P(c_i|\{X_{n-1}\})}{P(X_n|X_{n-1})}, \tag{13}$$

where $P(X_n|X_{n-1}) = \sum_i P(X_n|c_i, X_{n-1})P(c_i|\{X_{n-1}\})$. Equation (13) has been used throughout this work. The advantage of using the recursive Bayes classifier is that the probabilities are updated at each measurement of the targets.

4.2 Gaussian Mixture Model

As GMM [57] has been used extensively in this work for modelling the probability density function (PDF) of the features, a short introduction to GMM,

[9]The notation $a \leftrightarrow b \leftrightarrow c$ denotes a is statically independent of c if b is known.

and the algorithms to fit the GMM i.e. Kmeans and EM-algorithm, is given. GMM is a mixture model, which means that the GMM consist of multiple Gaussian distributions with a weight on each Gaussian

$$P(X;\mu,\Sigma) = \sum_{j}^{J} \omega_j \mathcal{N}(X;\mu_j,\Sigma_j), \tag{14}$$

where $\mu = \{\mu_j\}$, $\Sigma = \{\Sigma_j\}$, j is the jth mixture $\mathcal{N}$ is the Gaussian function, ω_j is the weight, μ_j the mean, Σ_j is the covariance of the jth mixture and J is the number of mixtures in the GMM. Since a GMM is a PDF, it holds that $\sum_j \omega_j = 1$ and $\omega_j \in \mathbb{R}, 0 \leq \omega_j \leq 1$. The number of mixtures in the GMM is the hyper parameter of the GMM. This must be chosen either by experiments or expert knowledge. Some algorithms exist to choose the number of mixtures e.g. Gmeans [58]. The Kmeans algorithm [57] is an algorithm to find the mean μ_j for each of the mixtures. Kmeans is a specific case of the more generalized expectation-maximization algorithm (EM) [57]. EM finds both the mean and the covariance of each of the mixtures however, the EM algorithm incurs a high computational load. Therefore, Kmeans is customarily used to find the means, which are used to initialize the EM algorithm. The problem of using Kmeans and EM algorithm is that it cannot be guaranteed to be the global best fit, because the problem is not convex [59]. Multiple runs of the algorithms with different starting points can help to find the best fit.

Throughout the work, partial integration of a GMM (i.e. marginalization) has been done. For a multivariate normal distribution, the derivation is done in [60]. For GMMs, the result is shown here. If we have the following inverse covariance matrix

$$\Sigma_j^{-1} = \begin{bmatrix} A_j & B_j \\ B_j & C_j \end{bmatrix}, \tag{15}$$

and we want to marginalize the first feature out, the new covariance can be calculated as

$$U_j^{-1} = C_j - B_j A_j^{-1} B_j^T. \tag{16}$$

From the above equation, we can condition a joint GMM as:

$$P(X(1)|X(2)) = \frac{P(X(1),X(2))}{\int P(X(1),X(2))dX(1)}. \tag{17}$$

However, this can introduce a problem caused by the infinitely long tails of Gaussian distributions. An observation far outside the largest value present in the training data, will still be assigned a probability greater then zero. This could be a problem because the underlying training data does not provide support for the classification, which can lead to misclassification. In order to truncate the tails of the Gaussians, we follow an approach similar to that

of [61], where the Gaussians are truncated during the training of the GMMs. However, in our case, we propose to only truncate during classification. Particularly, during the training of the GMMs, the support of the training data is found, i.e. the minimum and maximum values of each feature component of the feature vector X_n^t is obtained, where superscript t is to emphasize that this is the training data. Thus, for each feature f:

$$\xi_f^{\min} = \min\{X_n^t(f)\}, \quad \xi_f^{\max} = \max\{X_n^t(f)\}. \tag{18}$$

We assign zero probability if X_n is outside a confidence interval given by the training data. That is, if there exist an f such that $X_n(f) > \xi_f^{\max} + N_s \operatorname{std}\{(X_n^t(f)\}$ or $X_n(f) < \xi_f^{\min} - N_s \operatorname{std}\{(X_n^t(f)\}$, then $P\left(X_n|c_i, X_{n-1}\right) = 0$. Here we use $\operatorname{std}\{\cdot\}$ to denote the standard deviation of the feature set $\{\cdot\}$ and N_s is a constant describing the distance that the features in the test data is allowed to be away from the support of the training data.

A further development of the approach from paper A and B could be an interpolation between different GMMs which are trained at different timesteps. The idea is that we have a set of PDFs where each PDF in the set is corresponding to a specific time since last update. We denote the speed V_n and Δt is a fixed set of times e.g. $\{1, 2, \cdots 30\}$.

$$P(V_n|\Delta V_n, c_i, \Delta t) = \{P(V_n|\Delta V_n, c_i)\}_{\Delta t}, \tag{19}$$

where $\Delta V_n = V_n - V_{n-1}$. As our training data are filtered with a fixed interval smoother, we can chose the sampling time and thereby achieve the desired Δt. We use the difference in speed $V_n - V_{n-1}$ as feature space. From the training data we obtain a discrete set of Δt's. Then, by use of linear interpolation, we can obtain a PDF for any $\Delta t \in \mathbb{R}$. This means we have a set of predefined Δt. We define $\Delta t_{max} = \max(\{x \in \Delta t | \Delta t < \Delta t_n\})$ and $\Delta t_{min} = \min(\{x \in \Delta t | \Delta t > \Delta t_n\})$. We also define $P_{min} = P(V_n|\Delta V_n, c_i, \Delta t_{min})$ and $P_{max} = P(V_n|\Delta V_n, c_i, \Delta t_{max})$ we can now estimate the probability for a measurement given the time since last update and the class by

$$P_k(V_n|\Delta V_n, c_i, \Delta t_n) = (1-k)P_{min} + kP_{max}, \tag{20}$$

where

$$k = \frac{\Delta t_n}{\Delta t_{max} - \Delta t_{min}} - \frac{\Delta t_{min}}{\Delta t_{max} - \Delta t_{min}} \in [0; 1]. \tag{21}$$

The time step between the GMMs may be chosen to be logarithmic such that we have more GMMs at the small time steps, where we except most lookups to occur and large distances between the time steps where few lookups are expected, and where the changes are smaller. We can thereby reduce the training time of the algorithm. Additionally, it is easy to change the linear interpolation to a more advanced form such as polynomial interpolation. The

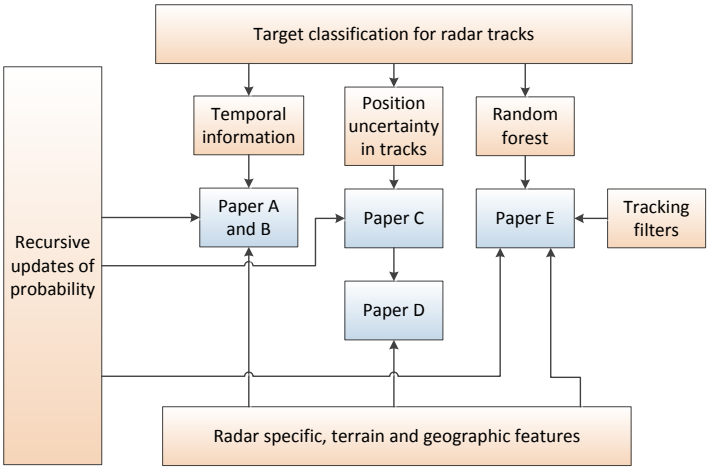


Fig. 9: This figure shows an overview of the contribution of this work. The figure also shows how the papers are linked, which features and theories that are used

proposed method does not exploit correlation between the time steps. This has shown better performance than the proposed methods in paper A but not as good as the deltaGMM proposed in paper B.

4.3 Paper contribution

The contribution of this work is based on five papers. In Fig. 9 the papers are shown with the features, theory and the linkage between them. The first two papers main contribution is utilization of the temporal information in tracks [10]. The main contribution of the next two papers, is how to utilize the position uncertainty in the measurements.

Paper A - Modelling Temporal Variations by Polynomial Regression for Classification of Radar Tracks

This paper is published at EURASIP, European Signal Processing Conference (EUSIPCO), 2014.

In this paper, we introduce a polynomial to describe the temporal information. We fit a polynomial such that the variable in the polynomial is the speed and the output is the typically acceleration given the speed. This is

[10]The temporal information is the information which comes from knowing something about the past i.e. a car driving 130 km/h on a highway is not likely to accelerate much, but more likely to decelerate. The temporal information is also information loss due to the time between updates i.e. if a car has a speed of 130 km/h an hour ago it is nearly impossible to estimate the current speed compared to if it was two seconds ago.

used to estimate a variance of a Gaussian PDF. i.e. a large acceleration will give a large variance. The output from the classifier is a weighted sum of the Gaussian described above, and a GMM which has the prior knowledge of the speed[11]. We also utilize other features such as geographical and radar specific features in both of the classifiers. We compare the results of the classifier with a naive recursive Bayesian classifier (RGMM) which does not utilize the temporal information. The results from this work shows that by exploiting the temporal information a better estimate of the target class can be achieved.

Paper B - Recursive Bayesian Classification of Surveillance Radar Tracks based on Kinematic with Temporal Dynamics and Static Features

This paper is published at IET, International Radar Conference, 2014.
We introduce a GMM only approach to utilize the temporal information called the deltaGMM. The feature space [57] of the GMM is given by the set $\{V_n, V_{n-1}, a_n, a_{n-1}, \Delta t_n\}$, where V is the speed, a the acceleration, n is the update number and Δt_n is the time between n and $n-1$. By introduction the time in the feature space the temporal information will be used as in paper A. By using a GMM for the temporal information we use the conditional statistics which paper A does not use. In this paper we also utilize the other features as in paper A. We compare the results with RGMM and show an even better performance compared to the RGMM and polynomial approach from paper A.

Paper C - Using Position Uncertainty in Recursive Automatic Target Classification of Radar Tracks

This paper is published at IEEE, International Radar Conference, 2015.
In the papers A and B the uncertainty in the position was not used. This resulted in a very fluctuating probability from the classifier over the full length of the track. Therefore, in this paper we introduce a probabilistic approach to handle the uncertainty. We limit the work such that we do not use the temporal information i.e. we take the basis in the RGMM classifier and we do not utilize the geographically and radar specific features. From the position and position uncertainty, we estimated the speed and the uncertainty in the speed. We then choose a number of speeds that lie within the uncertainty of the speed. We then use these speeds as an input in a GMM and weight the output of the GMM with the probability for the speed given the uncertainty. We show with both simulated and real data that by using the uncertainty, a more robust classification can be achieved. The downside of this approach is the high computational load and the disability to classifying correctly.

[11] In this context the prior knowledge is refereed to as the probability for a given speed and class i.e. without the temporal information.

Paper D - Exploiting Position Uncertainty in Recursive Radar Track Classification

This paper is submitted to EURASIP, Journal on Advances in Signal Processing. From the paper C, it was clear that while the probability of the classifier does not fluctuate, the performance of the classifier was not so good. Therefore, in this paper we introduce other features also used in papers A and B. We also introduce two new features, which are the estimated minimum height of the target to be within line of sight, and a feature, which estimates if the target is moving or stationary. We combined these features with the classifier from the paper C. In this paper, we also show a more thorough review of the mathematics of the classifier in paper C. The result show better performance of the classifier, however the results are still not as good as when using temporal information.

Paper E - A Recursive kinematic Random forest and alpha beta filter classifier for 2D radar tracks

This paper is submitted to EURASIP, Journal on Advances in Signal Processing. All of the above mentioned papers utilize a GMM PDF and a probabilistic approach. We wanted to try another approach by using a random forest[12]. We use the pseudo probability from the random forest[13] with a weight function where the weight is the amount of new information in the extracted feature vector. We also use an alpha beta filter to determine if the target is stationary or moving. The classifier shows better performance in the real world scenario than all of the above mentioned classifiers.

4.4 Comparison of some of the classifiers

In this section, a small comparison of some of the proposed classifiers is made. The data sizes used for training and testing are shown in Table 3. It must be noted that the confusion matrix uses a small test data size. The comparison is made with a confusion matrix [57] where each update of the tracks is a sample in the confusion matrix. The confusion matrices are shown in Table 5. As it can be seen, the deltaGMM has the best performance in general. The random forest with alpha beta filter is the second best and worst is the uncertainty classifier. It can be seen that the random forest is nearly as good as the deltaGMM and it is believed that the random forest is actually better in real world performance, see the papers B and E.

[12]We have chosen the random forest over e.g. SVM as we believe that the random forest has better performance. Further the SVM is a native binary classifier and has a high training time compared to the random forest.

[13]The pseudo probability is a number of trees which classify the target as the same class divided with the total number of trees in the random forest.

Table 5: The confusion matrices for some of the classifier proposed in this work. The confusion matrix for deltaGMM is shown in Table 5a, the confusion matrix for the uncertainty with other features is shown in Table 5b and the confusion matrix for the random forest with alpha beta filter is shown in Table 5c.

(a) Confusion matrix of the best performing temporal classifier from the paper B

	Predicted:					
Actual:	Birds	RIBs	Stationary sea targets	Large ships	Helicopters	Commercial aircrafts
Birds	**98.9**	0.0	0.0	0.0	1.1	0.0
RIBs	0.0	39.9	41.8	18.3	0.0	0.0
Stationary sea targets	0.0	0.0	**99.9**	0.0	0.1	0.0
Large ships	6.6	36.4	0.1	56.9	0.0	0.0
Helicopters	0.0	0.0	0.0	0.0	**100.0**	0.0
Commercial aircrafts	0.0	0.0	0.1	0.0	0.0	**99.9**
Overall performance	82.6					

(b) The confusion matrix of the classifier, which utilize the uncertainty and other features.

	Predicted:					
Actual:	Birds	RIBs	Stationary sea targets	Large ships	Helicopters	Commercial aircrafts
Birds	85.5	0.0	3.8	8.4	2.3	0.0
RIBs	17.9	1.7	19.2	61.3	0.0	0.0
Stationary sea targets	2.2	14.6	58.6	4.0	20.6	0.0
Large ships	3.4	0.2	0.4	**91.2**	0.8	3.9
Helicopters	6.8	0.0	2.0	62.0	5.9	23.4
Commercial aircrafts	1.0	0.1	3.5	1.7	0.7	93.0
Overall performance	56.0					

(c) The confusion matrix of the random forest combined with the alpha beta filter.

	Predicted:					
Actual:	Birds	RIBs	Stationary sea targets	Large ships	Helicopters	Commercial aircrafts
Birds	67.9	9.2	0.0	21.0	1.9	0.0
RIBs	6.4	**62.4**	0.0	31.2	0.0	0.0
Stationary sea targets	0.5	0.0	99.5	0.0	0.0	0.0
Large ships	21.4	5.1	0.3	61.5	11.6	0.0
Helicopters	12.2	0.0	0.0	0.0	87.8	0.0
Commercial aircrafts	0.8	0.0	0.0	0.0	0.0	99.2
Overall performance	79.7					

5 Conclusion and Future Work

In this work, we use the position estimates from a target delivered by a 2D scanning coastal surveillance radar. We use this position measurements to estimate the class each track belongs to. The work started with two hypotheses. The first hypothesis is, that if it is possible to achieve high-accuracy radar track classification based on temporal behavioral analysis. The work in the papers A and B shows that it is possible to classify radar tracks based on temporal behavior. However, it can be discussed if the classification can achieve a high accuracy. It is clearly possible to separate targets which have very different kinematic constraints e.g. stationary sea targets and commercial aircrafts, but it is more difficult to separate targets which have more common kinematic constraints e.g. RIBs and birds. By using more information such as geographical features e.g. land, sea, line of sight, an improvement of the classification result can be achieved. However, this can have the unfortunately influence that the probability of the classes will increase faster than the true confidence of the classification, which can result in rapid changes in probability for the different classes. This is believed to be because of the uncertainty in the features. In the papers C and D we try to deal with the rapidly changing probabilities by using the position uncertainty to achieve a more robust estimate of the speed. In paper C, we show that it is possible to achieve a more robust classification. We then, in paper D, combine more features and show that the probability and the confidence of the classification still match. However, as it can be seen in Table 5b, the classification results are not good even though we use additional features. In the paper E, we try to use a random forest classifier as random forest by itself sorts bad features[14] out. We then assume that the uncertainties in the features are minimized. The reason for this is that the features which in general have high uncertainty with a great overlap between the classes will not be weighted as much as other features. Further, the random forest also includes the temporal information. We show that the classification results are promising.

The first hypothesis is proven true. It is possible to use temporal behavioral analysis to achieve radar target classification. However, high accuracy classification can only be achieved when the targets has very distinct temporal behavioral i.e. stationary sea targets and commercial aircraft in this work.

The second hypothesis is, that it is possible to remove or suppress unwanted tracks without affecting the performance of the small target detection capability. In this work, we defined unwanted targets to be birds and stationary sea targets, as these targets only confuse a surveillance images for a radar observer. As it can be seen from the work it is possible without much perfor-

[14]Bad features are defined as features which do not improve the classification results or have very little new information.

mance degradation to suppress stationary targets. However, birds cannot be suppressed without also suppress some other targets.

The second hypothesis can be true if the unwanted targets have distinct temporal behavioral but, as the unwanted targets also includes birds in this work it is not possible to suppress unwanted target without also suppressing wanted targets.

As classification for 2D surveillance radars tracks are an unexplored area for machine learning techniques, there is a lot of research still to be done. From this work, a set of recommendations for future work will be listed. However, this list is not exhaustive:

- As explained in the section 3, tracking classification is the most used approach to do classification of radar tracks. The limitation of using tracking filters such as IMM filters is that the parameters are not state-dependent and therefore cannot reflect the temporal information. Some research must be done to determine if it is possible to make the parameters in the IMM dependent on the state in the IMM e.g. by making a linear interpolation of a set of parameters which are specified for a specific state in IMM[15]. A fast optimizing algorithm must also be found, such that the different parameters for the IMM, including the state-dependent ones, can be found for a given class.

- Another approach could be used where an IMM followed by a random forest, such that the IMM is fitted to the class and the likelihood is the input to the random forest. In this approach the IMM will hopefully deal with the uncertainty in the measurement while the random forest can do the actual classification.

- Another approach for the random forest is to use multiple random forests where the number of measurements used to extract the feature vector is different for each of the random forests. By using this method, there is no need for a recursive update of the probability as the random forest takes the complete set of available data at the given time step. Of course, this is very memory-intensive, as tracks with different length must be classified in a single scan, which will require that all of the random forests are in the computer memory.

- A deep neural network could also be tested where the plots are used as the input to the network. There are possible problems with this approach, as this may lead to the measured positions are directly used for the classification, and hence the classifier is not immune of different radar positions. Therefore a feature vector may be required as in the case of the random forest.

[15]When describing the state it is primarily the speed which must be dependent.

The recommendations for the next step in this work is to make some feature engineering for the random forest such that the features will be immune to uncertainty in the position. Additionally, features such as local wind direction and local wind speed could be used to normalize the speed of the target. However, this requires the knowledge of whether the target is airborne or a surface target. That is, a bird will fly much faster in tailwind than headwind, whereas a ship will not be affected of the wind. Information on ship lanes, and speed limitations in the lanes can also be used to improve the classification. For the specific radars in this work, a gain through the adaptive preprocessing could help to get an absolute intensity for the target, which makes the intensity less uncertain.

References

[1] J. N. Briggs, *Target detection by marine radar*. IET, 2004, vol. 16.

[2] S. J. Anderson, "Target classification, recognition and identification with HF radar," in *Proceedings of the NATO Research and Technology Agency. Sensors and Electronics Technology Panel Symposium SET–080/RSY17/RFT:"Target identification and recognition using RF systems", RTO–MP–SET*. Citeseer, 2004, p. 18.

[3] M. I. Skolnik, *Introduction to Radar Systems 3nd Edition*, 3rd ed. McGraw Hill Book Co., 2001.

[4] *Guideline No. 1111 on Preparation of Operational and Technical Performance Requirements for VTS Systems*, IALA Std., Rev. 1, 2015.

[5] D. Langellotti, F. Colone, P. Lombardo, M. Sedehi, and E. Tilli, "DVB-T based passive bistatic radar for maritime surveillance," in *Radar Conference, 2014 IEEE*. IEEE, 2014, pp. 1197–1202.

[6] T. Peto, L. Dudas, and R. Seller, "DVB-T based passive radar," in *Radioelektronika (RADIOELEKTRONIKA), 2014 24th International Conference*, April 2014, pp. 1–4.

[7] T. D. Ross, S. W. Worrell, V. J. Velten, J. C. Mossing, and M. L. Bryant, "Standard SAR ATR evaluation experiments using the MSTAR public release data set," in *Aerospace/Defense Sensing and Controls*. International Society for Optics and Photonics, 1998, pp. 566–573.

[8] A. R. Wise, D. Fitzgerald, and T. D. Ross, "Adaptive SAR ATR problem set (adaptsaps)," in *Defense and Security*. International Society for Optics and Photonics, 2004, pp. 366–375.

[9] Terma. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/211400/scanter_5000_series__w.pdf

[10] ——. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/291389/sc6k_-_asia.pdf

[11] ——. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/211428/scanter_4100_-_naval_air___surface_surveillance_2d_radar.pdf

[12] K. Hansen, A. Thomsen, M. Riis, O. Marqversen, M. Ø. Pedersen, and E. Nielsen, "Detection and tracking of aircraft over wind farms using SCANTER 4002 with embedded tracker 2," *IET Radar*, pp. 1–6, 2012.

[13] D. Hicok and D. Lee, "Application of ADS-B for airport surface surveillance," in *Digital Avionics Systems Conference, 1998. Proceedings., 17th DASC. The AIAA/IEEE/SAE*, vol. 2, Oct 1998, pp. F34/1–F34/8 vol.2.

[14] Z. Ou and J. Zhu, "AIS database powered by GIS technology for maritime safety and security," *Journal of Navigation*, vol. 61, no. 04, pp. 655–665, 2008.

[15] T. S. No, "International convention for the safety of life at sea, 1974," vol. 1/7/02, 1974.

[16] A. Harati-Mokhtari, A. Wall, P. Brooks, and J. Wang, "Automatic identification system (AIS): data reliability and human error implications," *Journal of navigation*, vol. 60, no. 03, pp. 373–389, 2007.

[17] A. Costin and A. Francillon, "Ghost in the air (traffic): On insecurity of ADS-B protocol and practical attacks on ADS-B devices," *Black Hat USA*, 2012.

[18] D. E. Catlin, "Fixed interval smoothing," in *Estimation, Control, and the Discrete Kalman Filter*. Springer, 1989, pp. 188–199.

[19] S. Storve, "Kalman smoothing techniques in medical image segmentation," 2012.

[20] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001, vol. 1.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[22] D. A. Morgan, "Deep convolutional neural networks for ATR from SAR imagery," in *SPIE Defense+ Security*. International Society for Optics and Photonics, 2015, pp. 94 750F–94 750F.

[23] S. Chen and H. Wang, "SAR target recognition based on deep learning," in *Data Science and Advanced Analytics (DSAA), 2014 International Conference on*. IEEE, 2014, pp. 541–547.

[24] C. Bentes, D. Velotto, and S. Lehner, "Target classification in oceanographic SAR images with deep neural networks: Architecture and initial results," in *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. IEEE, 2015, pp. 3703–3706.

[25] M. Vespe, C. Baker, H. Griffiths, P. Lombardo, and D. Pastina, "Feature extraction for SAR target classification," in *London Communications Symposium*, 2005.

[26] S. Kent, N. Kasapoglu, and M. Kartal, "Radar target classification based on support vector machines and high resolution range profiles," in *Radar Conference, 2008. RADAR'08. IEEE*. IEEE, 2008, pp. 1–6.

[27] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, "Classification and tracking of moving ground vehicles," *Lincoln Laboratory Journal*, vol. 13, no. 2, pp. 275–308, 2002.

[28] C. Yang, M. Bakich, and E. Blasch, "Pose angular-aiding for maneuvering target tracking," in *Information Fusion, 2005 8th International Conference on*, vol. 1. IEEE, 2005, pp. 8–pp.

[29] A. Zyweck and R. E. Bogner, "Radar target classification of commercial aircraft," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 32, no. 2, pp. 598–606, 1996.

[30] P. O. Molchanov, J. T. Astola, K. O. Egiazarian, and A. V. Totsky, "Target classification by using pattern features extracted from bispectrum-based radar doppler signatures," in *Radar Symposium (IRS), 2011 Proceedings International*. IEEE, 2011, pp. 791–796.

[31] P. Molchanov, J. Astola, K. Egiazarian, and A. Totsky, "Classification of ground moving radar targets by using joint time-frequency analysis," in *2012 IEEE Radar Conference (RADAR)*, 2012, pp. 0366–0371.

[32] J. Lei and C. Lu, "Target classification based on micro-doppler signatures," in *Radar Conference, 2005 IEEE International*. IEEE, 2005, pp. 179–183.

[33] Y. Luo, Q. Zhang, C.-w. Qiu, X.-j. Liang, and K.-m. Li, "Micro-doppler effect analysis and feature extraction in ISAR imaging with stepped-frequency chirp signals," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 48, no. 4, pp. 2087–2098, 2010.

[34] R. Harmanny, J. de Wit, and G. P. Cabic, "Radar micro-doppler feature extraction using the spectrogram and the cepstrogram," in *European Radar Conference (EuRAD), 2014 11th*. IEEE, 2014, pp. 165–168.

[35] F. Fioranelli, M. Ritchie, H. Griffiths, and H. Borrion, "Classification of loaded/unloaded micro-drones using multistatic radar," *Electronics Letters*, vol. 51, no. 22, pp. 1813–1815, 2015.

[36] P. Molchanov, R. I. Harmanny, J. J. de Wit, K. Egiazarian, and J. Astola, "Classification of small UAVs and birds by micro-doppler signatures," *International Journal of Microwave and Wireless Technologies*, vol. 6, no. 3-4, pp. 435–444, 2014.

[37] V. Chen, *The micro-Doppler effect in radar*. Artech House, 2011.

[38] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[39] L. Rabiner and B.-H. Juang, "Fundamentals of speech recognition," 1993.

[40] G. Kouemou and F. Opitz, "Hidden markov models in radar target classification," in *Radar Systems, 2007 IET International Conference on*. IET, 2007, pp. 1–5.

[41] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

[42] O. Irsoy and C. Cardie, "Deep recursive neural networks for compositionality in language," in *Advances in Neural Information Processing Systems*, 2014, pp. 2096–2104.

[43] P. Langley, "Induction of recursive bayesian classifiers," in *Machine Learning: ECML-93*. Springer, 1993, pp. 153–164.

[44] R. Laxhammar, G. Falkman, and E. Sviestins, "Anomaly detection in sea traffic - a comparison of the gaussian mixture model and the kernel density estimator," in *Information Fusion, 2009. FUSION'09. 12th International Conference on*. IEEE, 2009, pp. 756–763.

[45] B. Ristic, B. L. Scala, M. Morelande, and N. Gordon, "Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction," in *Information Fusion, 2008 11th International Conference on*. IEEE, 2008, pp. 1–7.

[46] Y. Zheng, Y. Chen, Q. Li, X. Xie, and W.-Y. Ma, "Understanding transportation modes based on GPS data for web applications," *ACM Transactions on the Web (TWEB)*, vol. 4, no. 1, p. 1, 2010.

[47] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson, "Cooperative transit tracking using smart-phones," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2010, pp. 85–98.

[48] D. Angelova and L. Mihaylova, "Joint target tracking and classification with particle filtering and mixture kalman filtering using kinematic radar information," *Digital Signal Processing*, vol. 16, no. 2, pp. 180–204, 2006.

[49] N. J. Gordon, S. Maskell, and T. Kirubarajan, "Efficient particle filters for joint tracking and classification," in *AeroSense 2002*. International Society for Optics and Photonics, 2002, pp. 439–449.

[50] P. Smets and B. Ristic, "Kalman filter and joint tracking and classification based on belief functions in the TBM framework," *Information fusion*, vol. 8, no. 1, pp. 16–27, 2007.

[51] S. Maskell, "Joint tracking of manoeuvring targets and classification of their manoeuvrability," *EURASIP Journal on Applied Signal Processing*, vol. 2004, pp. 2339–2350, 2004.

[52] H. Jiang, K. Zhan, and L. Xu, "Joint tracking and classification with constraints and reassignment by radar and ESM," *Digital Signal Processing*, vol. 40, pp. 213–223, 2015.

[53] I. Moir, A. G. Seabridge, and M. Jukes, *Military avionics systems*. John Wiley & Sons, 2006.

[54] Z. Sun and X. J. Ban, "Vehicle classification using GPS data," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 102–117, 2013.

[55] G. Kouemou and F. Opitz, "Radar target classification in littoral environment with HMMs combined with a track based classifier," in *Radar, 2008 International Conference on*. IEEE, 2008, pp. 604–609.

[56] M. Garg and U. Singh, "C & R tree based air target classification using kinematics," in *National Conference on Research Trends in Computer Science and Technology (NCRTCST), IJCCT_Vol3Iss1/IJCCT_Paper_3*, 2012.

[57] C. M. Bishop, *Pattern recognition and machine learning*. springer New York, 2006, vol. 1.

[58] G. Hamerly and C. Elkan, "Learning the k in k-means," in *In Neural Information Processing Systems*. MIT Press, 2003, p. 2003.

[59] J. J. Verbeek, N. Vlassis, and B. Kröse, "Efficient greedy learning of gaussian mixture models," *Neural computation*, vol. 15, no. 2, pp. 469–485, 2003.

[60] T. B. Schön and F. Lindsten, "Manipulating the multivariate gaussian density," Technical report, Linkoeping University, 2011. http://www. rt. isy. liu. se/schon/Publications/SchonL2011. pdf, Tech. Rep., 2011.

[61] J. Lindblom and J. Samuelsson, "Bounded support gaussian mixture modeling of speech spectra," *Speech and Audio Processing, IEEE Transactions on*, vol. 11, no. 1, pp. 88–99, 2003.

Part II

Papers

Paper A

Modelling Temporal Variations by Polynomial Regression for Classification of Radar Tracks

Lars W. Jochumsen, Jan Østergaard, Søren H. Jensen, Morten Ø. Pedersen

The paper has been published in the
EURASIP, European Signal Processing Conference (EUSIPCO) pp. 1412–1416, 2014.

©2014 EURASIP
The layout has been revised.

Abstract

The sampling rate of a radar is often too low to reliably capture the acceleration of moving targets such as birds. Moreover, the sampling rate depends upon the target's speed and heading and will therefore generally be time varying. When classifying radar tracks using temporal features, too low or highly varying sampling rates therefore deteriorates the classifier's performance. In this work, we propose to model the temporal variations of the target's speed by low-order polynomial regression and use this to obtain the conditional statistics of the target's speed at some future time given its speed at the current time. When used in a classifier based on Gaussian mixture models and with real radar data, it is shown that the inclusions of conditional statistics describing the targets temporal variations, leads to a substantial improvement in the overall classification performance.

1 Introduction

The aim of this work is to provide a better overview for a radar operator and thereby enhanced situation awareness in mission critical environments. This is done by real–time classification of radar tracks. A commercial state of the art surveillance radar provides a huge amount of information and it can therefore be difficult for a radar operator to keep up with the information, see Fig. A.1. Integrated tracking in radars are becoming standard and in coastal surveillance small targets are of great importance. Consequently the radar and tracker must be sensitive enough to track these small targets. It will therefore be likely that unwanted tracks, like birds, will be tracked aswell. Suppression of such tracks requires real–time classification. Compared to synthetic aperture radar (SAR) a 2D surveillance radar does not have height, Doppler or radar imagery available in the classification process. Therefore a method must be developed which uses other attributes like for example kinematic and geographic attributes. By exploiting the temporal development of the kinematic data more information can be extracted from the data. Two of the main problems with using the temporal feature is the low sampling rate for a radar compared with the typically acceleration of target like birds etc. The second problem is that the sampling rate is inconsistent because of the targets movements and scan period is different for short and long range profiles in the radar.

There are two main reasons for target classification. The first is improving situation awareness for a radar operator by filtering or color coding tracks according to their classes. The second reason is with the knowledge of the target class the tracker parameters can be optimized and thereby resulting in the joint classification and tracking approach [1].

An advanced knowledge–based radar tracker [2] converts the measured

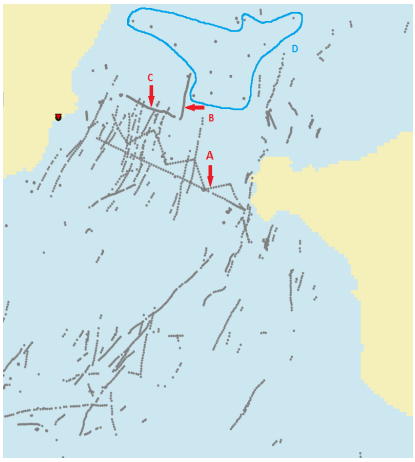


Fig. A.1: Radar scenario from Egaa Marina in Denmark showing a rigid inflatable boat (RIB) sailing out and zigzagging back. A large quantity of bird tracks is observed. A is the RIB, B and C are unknown vessels, D is an area with a number of sea buoys, and the rest of the tracks are birds.

backscatter from the radar sweeps into a number of observation called plots. These plots are then used for the actual tracking algorithm and the classification algorithms gets the kinematic information from the tracker.

In the following the term classification is used to describe the broad identification of a track belonging to a given class of targets such as "large ship", "rigid inflatable boats (RIBs)", "bird" etc.

A lot of work has been carried out for classification in SAR systems ([3] and [4]) but only very little has been done for 2D surveillance radars [5]. In [6] the authors are using a tree–based approach with kinematic features from a 3D radar. In [1], the authors are using joint tracking and classification where they have multiple tracking algorithms, one for each target classes and in [7] kinematic and radar cross section (RCS) are used for joint classification and tracking. In [8] the authors are using high range resolution (HRR) profiles to classify ground moving targets.

In this work, we consider the situation where a radar and its tracker provide information about the target's speed as well as the back scatter intensity. We then propose to model changes in the target's speed over time by polynomial regression. In particular, a low order polynomial is fitted in a least squares sense to training data acquired by commercial radars in realistic scenarios. Based on this model, we provide a closed-form expression to an approximation of the conditional probability density function (pdf) of the target's speed at time $t + \Delta t$ given its speed at time t. This pdf consists of a weighted sum between the target's prior and a Gaussian kernel whose standard deviation characterizes the uncertainty of the target's speed. The

optimal weight depends upon the target class and is numerically obtained by solving a maximum likelihood estimation problem. We then use the naive Bayesian classifier proposed in [5] for online classification of real radar data. It is shown that a substantial improvement is possible when including the statistics of the speed's temporal variations. For comparison, we also propose to simply model these variations by a Gaussian mixture model (GMM) using the framework of [5]. In this case, the proposed polynomial modelling of velocity GMM (PMVGMM) based on polynomial regression shows a slight improvement over the purely GMM based scheme.

2 Method

In this section, we first briefly introduce the naive Bayesian framework proposed in [5], which we will base our classifier upon. For more details about this framework, we refer the reader to [5]. Then, we present our main contributions, i.e., a model of the conditional probability of a target's speed, which is given as a weighted sum of the target's prior and a Gaussian kernel that introduces uncertainty into the model and whose standard deviation depends upon the time lag and target class.

2.1 Recursive naive Bayesian

The framework take the basis from [5] where a recursive update algorithm is used. From [5] (B.2) is the update and smoothing equation which will prevent the probability for a given class to reach zero.

$$P_s(c_p|X_n, X_{n-1}) = \frac{P(c_p|X_n, X_{n-1}) + \epsilon}{\sum_{y=1}^{N_c}(P(c_y|X_n, X_{n-1}) + \epsilon)}, \tag{A.1}$$

where X_n is the newest update from the radar, N_c is the number of classes. c_p is the given class and ϵ is some constant.

The recursive Bayesian update rule can be extended with more features, using a naive approach by making the assumption that the features are mutually independent.

2.2 Framework

The radar information is: normal radar intensity I_{nr} and moving target indication (MTI) intensity I_{mti}. The information used for kinematic update are: Speed over ground V_n and temporal dynamic e.g. how the target speed evolve.

The Probability for a given class will then be provided by

$$p(c_p|X_n, X_{n-1}) = \frac{P(X_n|c_p, X_{n-1})P(c_p|X_{n-1}, X_{n-2})}{\sum_{i=1}^{N_c} P(X_n|c_i, X_{n-1}))P(c_i|X_{n-1}, X_{n-2})}, \tag{A.2}$$

where $X_n = [V_n, I_{nr}, I_{mti}, \Delta t]^T$. In (A.2),

$$P(X_n|c_p, X_{n-1}) = P(V_n|V_{n-1}, \Delta t, c_p)P(I_{nr}, I_{mti}|c_p), \tag{A.3}$$

where $P(V_n|V_{n-1}, \Delta t, c_p)$ denotes the kinematic PDF and $P(I_{nr}, I_{mti}|c_p) = P(I_{nr}|c_p)P(I_{mti}|c_p)$ is the intensity PDF and they are both modelled as a GMMs. It is assumed that the radar intensity features are mutually independent.

2.3 The proposed PMVGMM method

The main problem with classifying with a temporal feature is that the sampling rate is low in compare to moving targets e.g. between 10 to 40 scans per minute, secondary is the radar does not have a constant sampling rate. Therefore a method must be developed which can handle the slow and varying sampling rate.

Let us assume that the measurements are noise free. A measurement of a target's speed is obtained at time t, which will give a probability of one for that particular speed (Fig. A.2, blue). After some time, say Δt, the target may have accelerated or de-accelerated and we are therefore less certain about its speed. Thus, the PDF of the target's speed, which was initially a delta function at time t should reflect the uncertainty in the speed at time $t + \Delta t$ (red and green). Indeed, as more time passes the less is known about the speed, and the PDF should tend to the targets prior PDF (black). We model this uncertainty by convolving the probability of V_n (that is a delta function) with a Gaussian kernel (A.4). We let the mean of the kernel be the previous measurement V_{n-1}, and let the standard deviation depend upon $V_n, V_{n-1}, \Delta t$. By using the Gaussian kernel, it is assumed that the acceleration and deceleration is equally distributed and therefore no skewness is present. For example, in the case of the speed feature we propose the following PDF P_u for modelling the uncertainty:

$$P_u(V_n|V_{n-1}, \Delta t) = \frac{1}{\sqrt{2\pi}\sigma(V_n, V_{n-1}, \Delta t)} \exp\left(\frac{-(V_n - V_{n-1})^2}{2\sigma(V_n, V_{n-1}, \Delta t)^2}\right). \tag{A.4}$$

We require that $\tau \gg 1$ implies $\sigma \gg 1$. With this, we therefore propose the

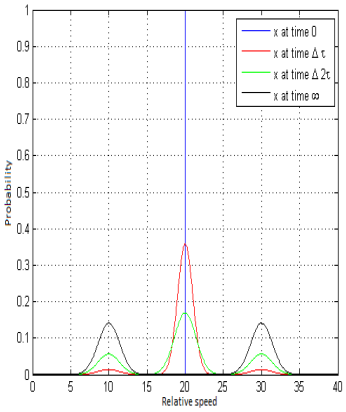


Fig. A.2: A simple example, with synthetic data, $K(X_n|C_p)$ is the conditional PDF that will be model in the following.

following formula for the standard deviation:

$$\sigma(V_n, V_{n-1}, \tau) = \left| \frac{\tau}{V_{n-1} - V_n} \int_{V_{n-1}}^{V_n} T_{acc}(q) dq \right|, \tag{A.5}$$

where $T_{acc}(q)$ is a fit of the acceleration given the speed described in section 2.3.

The standard deviation $\sigma(V_n, V_{n-1}, \tau)$ of the Gaussian kernel is then the average acceleration from the last measurement to the new measurement given the speed multiplied with the time since the last measurement. As time increases from the last measurement the variance will also increase because less is known about the speed. For $\tau \to \infty$ one cannot expect to have much knowledge about the speed except for the prior information given by the speed without temporal dynamics $P_{prior}(V_n)$. The transition from P_u to P_{prior} depends upon the class. We describe this dependency by an exponential weighting function with time constant C. The resulting PDF is the weighted sum of P_u and P_{prior}:

$$\begin{aligned} P(V_n|V_{n-1}, \tau) = &\exp(-C\tau) P_u(V_n|V_{n-1}, \Delta t) \\ &+ (1 - \exp(-C\tau)) P_{prior}(V_n), \end{aligned} \tag{A.6}$$

where

$$P_{prior} = \sum_{i=0}^{N} \pi_i \mathcal{N}(V_n; \phi_i) \tag{A.7}$$

and π_i is the weighing factor of the i^{th} Gaussian distribution. The kinematic model (A.6) must be modelled for each target class and hence it depends on the given class, that is $P(V_n|V_{n-1}, \tau, c_p)$.

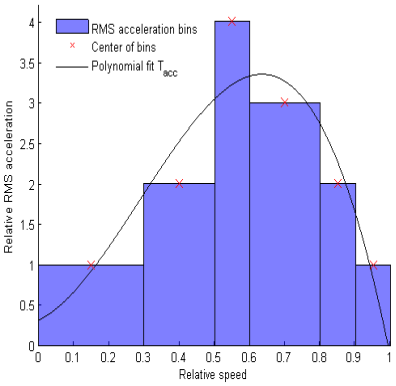


Fig. A.3: An example of the polynomial fit using synthetic data

Finding the $T_{acc}(q)$

The function T_{acc} from (A.5) is found from training data based on real radar measurements. Since radar data are noisy we have applied fixed interval smoothing [9] in order to reduce the noise.[1] A dataset $\{(V_n, \Delta V_n)\}$ which consist of pairs of speeds and associated accelerations $\Delta V_n = V_n - V_{n-1}$ are divided into bins based on the first coordinate, i.e. V_n. Each bin contains an equal amount of pairs see Fig. A.3 for an example. From this it is now possible to calculate the acceleration RMS value for each speed bin.

$$Acc_{RMS}(i) = \sqrt{\frac{1}{N}\sum_{n=1}^{N} \Delta V_n^2(i)}, \tag{A.8}$$

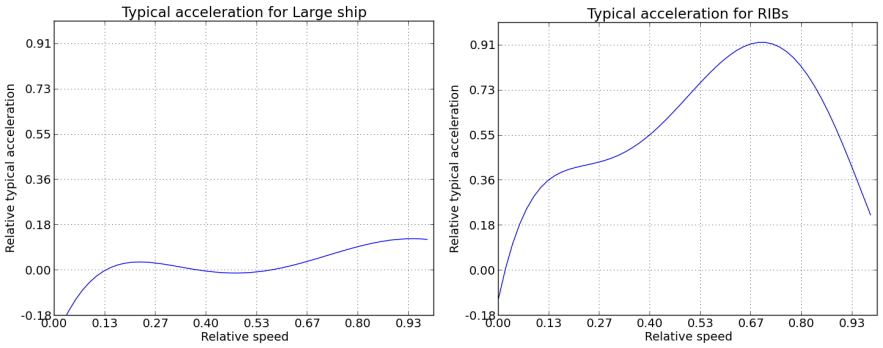
where i is the speed bin number, N is the number of data points in each bin and $\Delta V_n(i)$ is the n^{th}acceleration in the i^{th} speed bin. We fit a k^{th} order polynomial $T_{acc}(q)$ to the bins using the center of each bin. The polynomial thus provides the average acceleration given any speed V_n. See Fig. A.3.

In Fig. A.4 a subset of the class's acceleration fit is shown. It is clearly visible that the large ships does not accelerate much and the RIBs are a lot more agile then the large ships.

Finding the time constant C

The time constant C that is used in the weighting which combines P_u and P_{prior} can be obtained off-line for each class by maximizing the following

[1] Due to the filtering the training are less noisy than the test data and we are therefore able to deduce the target's acceleration based on the filtered speed training data. Due to the on-line classification we cannot apply fixed interval smoothing on the test data in a similar manner



(a) Large ship (b) RIBs

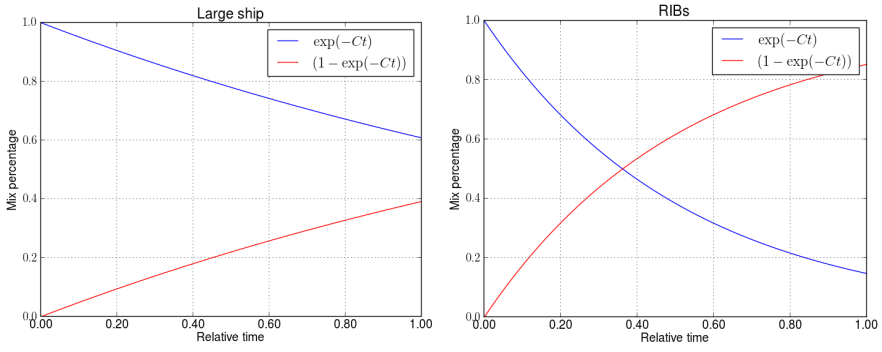
Fig. A.4: A sixth order acceleration fit for (a) Large ship (b) RIB

likelihood function:

$$L(C) = \prod_{n=1}^{N} \prod_{\tau} \exp(-C\tau) P_u(V_n|V_{n-1}, \Delta t) + (1 - \exp(-C\tau))\, P_{prior}(V_n), \tag{A.9}$$

where σ is described in (A.5).

In Fig. A.5 shows the amount of information the algorithm uses from the uncertainty and the prior given the time since last measurement. It is cleat that is more difficult to predict the RIB as the prior information is used quickly compared to the large ship. This is as expected because a large ship will normally not change speed as often as RIBs will.



(a) Large ship (b) RIBs

Fig. A.5: A plot over the relative weight of the the uncertainty and the prior. The weight is given by $exp(-C\tau)$. (a) large ship, (b) RIB

3 Results of experiments

In this section we present the performance of the algorithm. For comparison the results for RGMM and DeltaGMM [5] are also shown. The database is the same as described in [5] however in the Egaa marina scenario unwanted tracks originating from returns associated with static land features are removed. In table A.1 the matrix is shown for the PMVGMM. In table A.2 the confusion matrix from the RGMM method is shown. This algorithm does not use the temporal information. In table B.3 the DeltaGMM algorihm is shown. This algorithm exploit the temporal feature by making a GMM of the entire feature space. The bold font is the best performing algorithm for that class. The matrices are shown in percentage. In Fig. A.7, A.6 and A.8 a scenario is shown where a RIB is sailing out from Egaa marina in Aarhus, Denmark and zigzagging back. It is clear from both the confusion matrices and the radar scenario that the PMVGMM algorithm is the best performing. There is an improvement in the RIB track and the PMVGMM is keeping the performance for the birds.

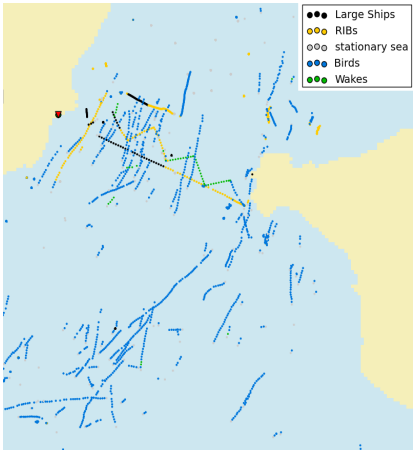


Fig. A.6: Egaa Marina scenario with colored tracks from the classification using the PMVGMM method.

4 Discussion

From Table A.2 and A.1 the PMVGMM is classifying with an overall accuracy of 12.9% better than the RGMM method and 1.3% better then DeltaGMM. The large ships and wakes are classified better with the PMVGMM algorithm. We believe this is because that the PMVGMM is requiring less training data then the DeltaGMM and RGMM as large ships are sailing with a constant speed.

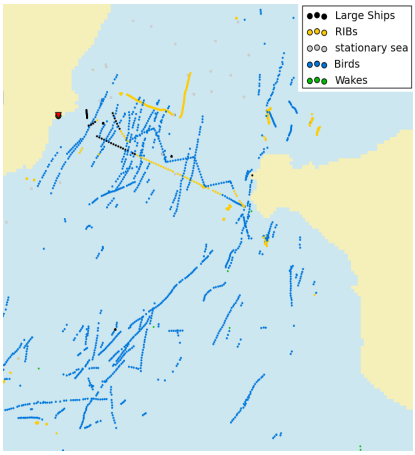


Fig. A.7: Egaa Marina scenario with colored tracks from the classification using the RGMM method.

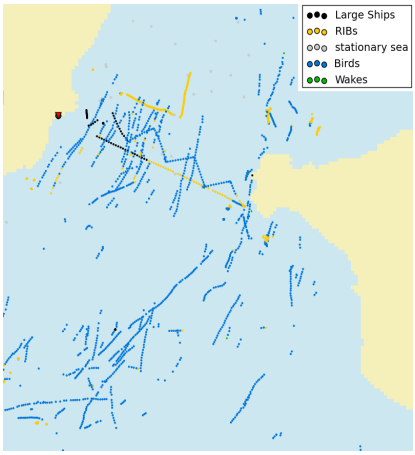


Fig. A.8: Egaa Marina scenario with colored tracks from the classification using the DeltaGMM method.

Therefore a large amount of different large ships must then be added to the database to get the right PDF. Because PMVGMM does not depend as much on the speed feature and more on the typical acceleration fewer types of large ships can be used to train the class. For more agile targets like birds and RIBs it is more difficult to predict the targets movements and therefore RGMM and DeltaGMM is better at these targets. The radar scenario is shown in Fig. A.6, Fig. A.7 and A.8. All of the classifiers has difficulty with classifying the RIB. The RGMM and DeltaGMM is classifying a large part of the RIB track as birds, however the PMVGMM is only classifying a small part as birds. False

	Predicted:				
Actual:	Large ships	Birds	Wakes	RIBs	Stationary sea targets
Large ships	**85.4**	3.9	0.8	9.6	0.2
Birds	16.8	79.0	0.8	0.8	2.7
Wakes	0.0	1.7	**97.5**	0.0	0.8
RIBs	51.4	2.6	0.4	36.7	8.9
Stationary sea targets	0.0	4.6	0.0	0.0	95.4
Overall performance	78.8				

Table A.1: PMVGMM confusion matrix

	Predicted:				
Actual:	Large ships	Birds	Wakes	RIBs	Stationary sea targets
Large ships	67.5	5.5	0.0	27.1	0.0
Birds	12.6	**87.4**	0.0	0.0	0.0
Wakes	0.0	72.3	27.7	0.0	0.0
RIBs	52.2	0.3	0.0	47.5	0.0
Stationary sea targets	0.0	0.0	0.0	0.6	**99.4**
Overall performance	65.9				

Table A.2: RGMM confusion matrix

	Predicted:				
Actual:	Large ships	Birds	Wakes	RIBs	Stationary sea targets
Large ships	58.9	7.3	0.3	33.5	0.0
Birds	0.0	85.5	14.5	0.0	0.0
Wakes	0.0	3.4	96.6	0.0	0.0
RIBs	51.5	1.2	0.0	47.2	0.0
Stationary sea targets	0.0	0.0	0.0	0.9	99.1
Overall performance	77.5				

Table A.3: DeltAGMM confusion matrix

negative classification of e.g. RIBs as birds is not desirable, as this may cause targets to be removed from the situations display when applying class filters such as birds. All of the algorithms is classifying most of the birds correct.

5 Conclusion

In this paper we present an algorithm which uses a polynomial to predict the acceleration given the speed for a target class. This is combined with a prior GMM as time passes and less is known of the targets speed and position. The results shows clearly that by exploiting the temporal information it is possible to give a better estimate of which target class a tracks belongs to.

References

[1] S. Challa and G. W. Pulford, "Joint target tracking and classification using radar and ESM sensors," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 37, no. 3, pp. 1039–1055, 2001.

[2] K. Hansen, A. Thomsen, M. Riis, O. Marqversen, M. Ø. Pedersen, and E. Nielsen, "Detection and tracking of aircraft over wind farms using SCANTER 4002 with embedded tracker 2," *IET Radar*, pp. 1–6, 2012.

[3] T. Porges and G. Favier, "Automatic target classification in SAR images using MPCA," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1225–1228.

[4] X. Ye, W. Gao, Y. Wang, and X. Hu, "Research on SAR images recognition based on ART2 neural network."

[5] L. W. Jochumsen, M. Pedersen, K. Hansen, S. H. Jensen, and J. Østergaard, "Recursive bayesian classification of surveillance radar tracks based on kinematic with temporal dynamics and static features," in *Radar Conference (Radar), 2014 International*. IEEE, 2014, pp. 1–6.

[6] M. Garg and U. Singh, "C & R tree based air target classification using kinematics," in *National Conference on Research Trends in Computer Science and Technology (NCRTCST), IJCCT_Vol3Iss1/IJCCT_Paper_3*, 2012.

[7] S. Sutharsan, R. Tharmarasa, T. Lang, and T. Kirubarajan, "Tracking and classification using aspect-dependent RCS and kinematic data," in *SPIE Defense and Security Symposium*. International Society for Optics and Photonics, 2008, pp. 696 913–696 913.

[8] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, "Classification and tracking of moving ground vehicles," *Lincoln Laboratory Journal*, vol. 13, no. 2, pp. 275–308, 2002.

[9] D. E. Catlin, "Fixed interval smoothing," in *Estimation, Control, and the Discrete Kalman Filter*. Springer, 1989, pp. 188–199.

Paper B

Recursive Bayesian Classification of Surveillance Radar Tracks based on Kinematic with Temporal Dynamics and Static Features

Lars W. Jochumsen, Morten Ø. Pedersen, Kim Hansen, Søren H. Jensen, Jan Østergaard

The paper has been published in the
IET, International Radar Conference, pp. 1–6, 2014.

©2014 IET
The layout has been revised.

Abstract

In this paper, it is shown that kinematic and static features are very useful in on-line classification of surveillance radar tracks based on real radar data. A simple classifier called recursive Gaussian mixture model (RGMM) is constructed using a recursive naive Bayesian approach combined with a multivariate GMM. The kinematic features used in the RGMM classifier are speed and normal acceleration, the geographic features are road, sea, land and the sensor features are intensities. It is then shown that if the feature vector is augmented with information about the temporal dynamics of the kinematic parameters, a substantial improvement in target classification is achieved. The classifiers are tested with several target classes relevant for coastal surveillance and different data sources such as radar and GPS. The proposed algorithms are classifying with 86% accuracy with 10 target classes versus 78% for the RGMM classifier.

1 Introduction

The overall aim of this work is to provide radar users with an enhanced situational awareness by performing real-time target classification. Modern surveillance radars provide an increasing amount of information and it can be difficult for an operator to keep up with the amount of data coming from the radar, see Fig. B.1. Integrated tracking is becoming standard for most surveillance radars and in coastal surveillance small targets is of great importance. Consequently, the radar and tracker must be sensitive to small targets, and unwanted targets like birds will therefore be tracked, which is not desirable. Suppression of unwanted targets requires classification.

In contrast to highly complex military radars, such as active electronically scanned arrays (AESA) or synthetic aperture radars (SAR), radar imagery is not available in 2D modern surveillance radars, and methods must be developed based on other attributes. The data available from SAR are most often based on space-borne or air-borne radars, which are very different from surface based surveillance radars in terms of performance and attributes. Classification of targets is of importance both directly and indirectly. Directly, target classification provides a user with tracks augmented with a target class for aid in setting automated alarms etc. Indirectly, a target class assists tracker performance through the ability to adaptively change tracking parameters for a given target class, the so-called joint target tracking and classification.

An advanced knowledge–based radar tracker [1] converts the radar return signals to a number of observations (plots) and detects and tracks targets from the plots by combining association logic with kinematic such as speed, acceleration etc. By adding further logic and knowledge to the radar sensor, the track data can be augmented to supply additional information to the track

data user, thereby improving the situational awareness. In Fig. B.1, is shown a radar scenario of a rigid inflatable boat (RIB) sailing in a straight line out from Egaa Marina in Denmark and zigzagging back again. All other tracks are birds with exception of two other vessels. As can be seen in the scenario it can be difficult for an operator to focus on a small size vessel like a RIB with so many bird tracks.

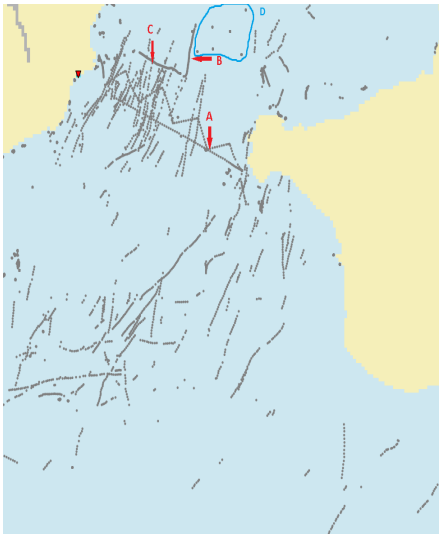


Fig. B.1: Radar scenario from Egaa Marina in Denmark showing a RIB sailing out and zigzagging back. A large quantity of bird tracks is observed. A is the RIB, B and C are unknown vessels, D is an area with a number of sea buoys, and the rest of the tracks are birds.

In the following the term classification is used to describe the broad identification of a track belonging to a given class of targets such as "RIB", "helicopter", "fixed-wing aircraft" etc.

A lot of work has been carried out for classification in SAR systems ([2] and [3]) but only very little has been done for 2D surveillance radars. In [4] the authors are using a tree–based approach with kinematic features from a 3D radar. In [5], the authors are using joint tracking and classification where they have multiple tracking algorithms, one for each target classes and in [6] kinematic and radar cross section (RCS) are used for joint classification and tracking. In [7] the authors are using high range resolution (HRR) profiles to classify ground moving targets.

The contribution of this work is based on a 2D radar without Doppler, RCS, HRR profile or height information. This work will rely on kinematic information together with location specific features such as land, sea, and road. In addition, normal radar and moving target indication (MTI) intensity

from the sensor are used. Because of the requirement for on-line classification a recursive Bayesian update approach is used. Two methods are presented, a simple algorithm called RGMM and an advanced algorithm called DeltaGMM.

In 2.1 the update method for on-line classification will be explained together with the naive Bayesian approach. In 2.2 the Gmeans algorithm will be briefly described. In 2.3 the features used for the classification are described and in 2.4 a general setup for both algorithms is described together with the missing feature problem. In 2.5 the RGMM algorithm will be explained together with the features used for the algorithm. Further in 2.6 the DeltaGMM method will be described. The algorithm uses the same features as the RGMM algorithm but also include a way to predict a the variations over time of the targets kinematic behaviour, which we will refer to as the targets temporal dynamic features.

2 Method

The algorithms use a naive Bayesian approach to combine multiple features and a recursive update for on-line classification. The joint probability density functions (PDFs) of the features are modelled by Gaussian mixture models (GMMs). To make the training of the GMM more robust for varying data sizes and classes a Gmeans approach has been used to select the number of mixtures for each class. The framework consists of both kinematic and geographic attributes. The two classification algorithms only differ in how their kinematic models are constructed.

2.1 Recursive naive Bayesian

For off-line target classification it is common to use a Bayesian rule to convert the probability $P(X_n|C_p)$ of observation measurement X_n given the target class C_p to the probability $P(C_p|X_n)$ for the class given the measurement. For an on-line classifier a recursive approach to the Bayesian rule can be used [8]:

$$P(C_p|X_n, X_{n-1}) = \frac{P(X_n|C_p)P(C_p|X_{n-1})}{\sum_{p=1}^{N_c-1} P(X_n|C'_p)P(C'_p|X_{n-1})}, \tag{B.1}$$

where X_n is the newest update from the radar, N_c is the number of classes and C_p is the given class. If a class is unlikely for a long time the probability for the given class will decrease and move towards zero. To avoid very small target class probabilities a smoothing factor ϵ is introduced to ensure that the probability never become zero i.e.,

$$P_s(C_p|X_n, X_{n-1}) = \frac{P(C_p|X_n, X_{n-1}) + \epsilon}{\sum_{y=1}^{N_c}(P(C_y|X_n, X_{n-1}) + \epsilon)}. \tag{B.2}$$

In the naive Bayesian approach, it assume that $x_1 \dots x_t$ are mutually independent and it is therefore straight forward to extend the Bayesian update with more features

$$P(C|x_1 \dots x_q) = P(C|x_1) \dots P(C|x_q). \tag{B.3}$$

2.2 Gmeans

The Gaussian Mixture Model (GMM) is trained with Kmeans and the expectation-–maximization (EM) algorithm [9]. The problem with these algorithms is that the number of mixtures in the GMM is not known and therefore must be found by an expert or by empirical trials. A way to get around this is to use the Gmeans algorithm [10]. It is an iterative process, which splits the data into two clusters, then tests the hypothesis that the original data set, projected to a one dimensional space, is Gaussian distributed. If the hypothesis fail then keep the two clusters for next iteration. If the hypothesis is true then keep the original data set. The statistical test is given by Anderson-–Darling test [11]:

$$A^2 = -\frac{1}{N}\sum_{i=1}^{N}\left[(2i-1)\left(\log(z_i) + \log(1 - z_{N+1-i})\right)\right] - N \tag{B.4}$$

and then corrected with [12]

$$A_*^2 = A^2\left(1 + \frac{4}{N} - \frac{25}{N^2}\right) \tag{B.5}$$

where z_i is the data projected to the one dimensional line expanded by the means of the two clusters and then converted to a univariate normal distribution with zero mean and a variance of one. N is the number of data points. The hypothesis is rejected if $A_*^2 > \phi$ where ϕ is some constant.

2.3 Features

In this section the general setup will be described. This is the framework which both the algorithms will use. There are three types of information used; geographical, radar specific and a kinematic information. The algorithms differ by the way in which they exploit the kinematic information..

The features used for the geographic and radar specific are

- Area (land/sea)

- Route (road/not road)
- Radar intensity (normal radar (nr))
- Radar intensity (MTI radar (mti))

The features used for the kinematic are

- Speed (SOG)
- Normal acceleration (na)
- Temporal dynamics (Only used in DeltaGMM)

X_n will then be defined by

$$X_n = \begin{bmatrix} \bar{g}_n \\ \bar{k}_n \end{bmatrix} \tag{B.6}$$

where $\overrightarrow{g}_n$ is the geographic and radar specific features and $\overrightarrow{k}_n$ is the kinematic features.

2.4 Framework

In this section the framework for the two algorithms are shown. The update equation for the new measurement is

$$P(C_p|X_n, X_{n-1}) = \frac{K(\overrightarrow{k}_n|C_p)G(\overrightarrow{g}_n|C_p)P(C_p|X_{n-1})}{\sum_{p=1}^{N_c} K(\overrightarrow{k}_n|C'_p)G(\overrightarrow{g}_n|C'_p)P(C'_p|x_{n-1})} \tag{B.7}$$

where $K(\overrightarrow{k}_n|C_p)$ is the kinematic PDF of the kinematic feature and this will differ in the two methods and $G(\overrightarrow{g}_n|C_p)$ is the geographic and radar specific PDFs and are defined by:

$$G(\overrightarrow{g}_n|C_p) = P(r|C_p)P(l|C_p)P(nr|C_p)P(mti|C_p), \tag{B.8}$$

where r is road/not road, l is land/sea, nr is the intensity for normal radar and mti is the intensity for MTI radar.

The classification is done by taking argmax of the probability given in (B.7)

$$C = \underset{C_p}{\operatorname{argmax}} \left(P_s(C_p|X_n, X_{n-1}) \right) \tag{B.9}$$

The land/sea and road/not road have binary sample spaces and their probability mass functions (PMFs) is modelled with Bernoulli distributions.

The intensity PDF is modelled as a GMM. The training database consists of multiple data sources and some of them do not have the intensity features.

By using the intensity feature only for some of the classes, the classes which do not have this feature will be favored because the probability will generally be less than one. Therefore a global intensity PDF is trained using all the available intensities from all classes. This global PDF will then be used for those classes that do not have intensity.

In classification from a radar, missing detections may occur, therefore a measurement does not necessarily have all the features like for example intensity. If the measurement does not have a feature, the feature will not be used in this update. This can be done be setting the probability for the missing feature equal to one.

2.5 RGMM method

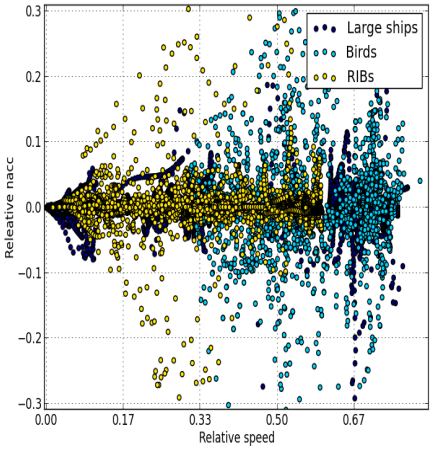


Fig. B.2: The relative feature space for birds, large ships and RIBs.

In the RGMM method given by (B.7) the PDF $K(k_n|C_p)$ models the kinematic feature space spanned by $\overrightarrow{k}_n$, i.e. speed over ground (sog) and the normal acceleration (nacc), and is given by

$$K(\overrightarrow{k}_n|C_p) = \sum_{i=1}^{N_m} \pi_i \mathcal{N}\left(\overrightarrow{k}_n; \Sigma_i^{C_p}, \mu_i^{C_p}\right), \tag{B.10}$$

where N_m is the number of mixtures in the GMM and π_i is the weight on the given Gaussian distribution.

The feature space for birds, large ships and RIBs is presented in Fig. B.2. As it can be seen there is significant overlap between the target classes.

2.6 DeltaGMM method

A problem with a radar is that it does not have a consistent sampling rate. In order to deal with this problem, we propose to model how a given targets speed and normal acceleration will evolve over time, see Fig. B.4 for an example. A bird will for example change its speed more frequently than a large ship. In order to model such temporal dynamics we introduce the following set of features:

$$\{v_i, \Delta v_i^\tau, na_i, \Delta na_i^\tau, \Delta t_i^\tau\}, \tag{B.11}$$

where i is the index of the measurements and τ is the measurement indexed by $i + \tau$, where $\tau \in \mathbb{N}_0$ and $\Delta t_i \in \mathbb{R}_+$,

$$\begin{aligned} \Delta v_i^\tau &= v_{i+\tau} - v_i \\ \Delta na_i\tau &= na_{i+\tau} - na_i \\ \Delta t_i^\tau &= t_{i+\tau} - t_i, \end{aligned} \tag{B.12}$$

where t_i denotes the time when measurement index by i was received. See Fig. B.3 for an example.

The aim is to model the following PDF:

$$K(k_n|\Delta t, C_p) = \frac{P(v,\Delta v,na,\Delta na,\Delta t|C_p)}{P(\Delta t|C_p)} \tag{B.13}$$

which gives the likelihood of a measurement given the class and the time since the last measurement. We find $P(\Delta t|C_p)$ as follows

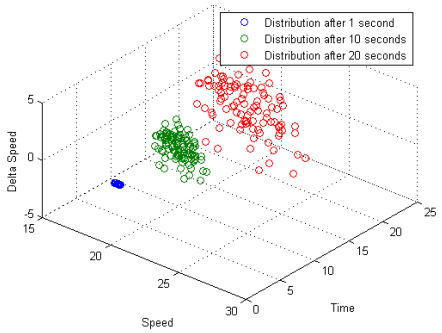


Fig. B.3: A visual interpretation of temporal dynamics of with synthetic data

$$P(\Delta t|C_p) = \iiiint P(v, \Delta v, na, \Delta na, \Delta t|C_p), dv, d\Delta v, dna, d\Delta na \tag{B.14}$$

Finally, the PDF $K(k_n|\Delta t, C_p)$ (B.13) of the kinematics is modelled by a GMM, that is:

$$K(\overrightarrow{k}_n|C_p) = \frac{\sum_{i=1}^{N_m} \pi_i \mathcal{N}(\overrightarrow{k}_n; \Sigma_i^{C_p}, \mu_i^{C_p})}{P(\Delta t|C_p)}. \tag{B.15}$$

For illustrational purposes Figs. B.4 and B.5 show contour plots of trained GMMs. It is clear from Fig. B.4 that it is more likely that a RIB will be changing speed then a large ship. In Fig. B.5 a plot is shown with typical

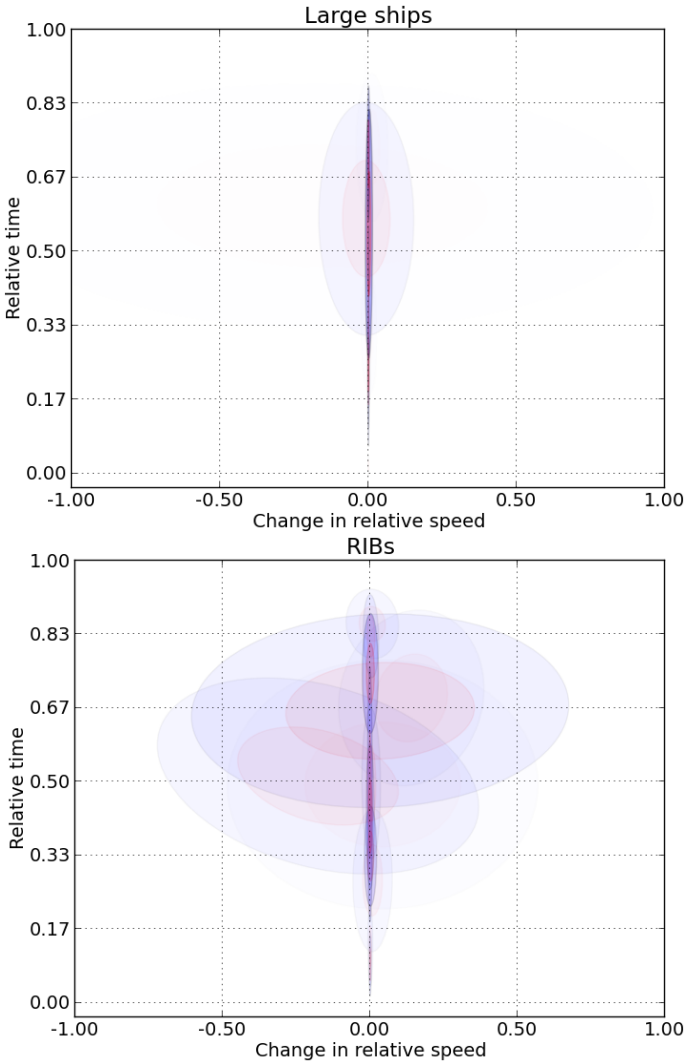


Fig. B.4: A plot of the trained GMM with only change in speed and time. Red ellipse is a half standard deviation and the blue is one standard deviation. The color transparency is how much weight on the given Gaussian in the GMM. Top: Large ships. Bottom: RIBs

acceleration given the speed. It is also clear that the RIB is more agile then large ships The speed PDF can be seen in Fig. B.6. It is clear that large ships

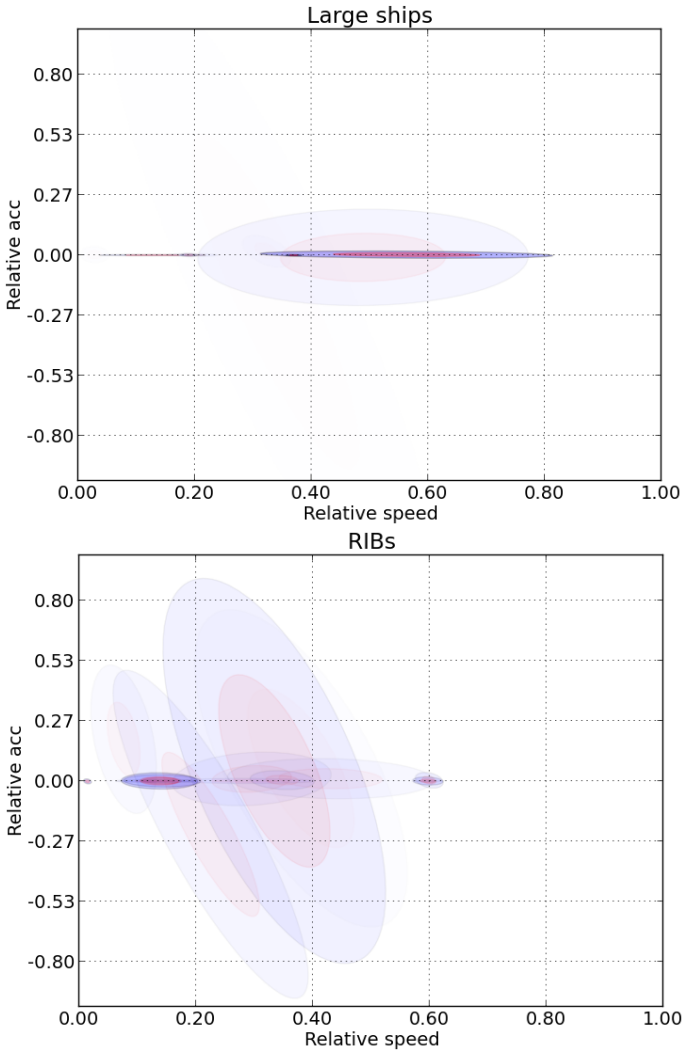


Fig. B.5: A plot of the trained GMM with acceleration given speed. Red ellipse is a half standard deviation and the blue is one standard deviation. The color transparency is how much weight on the given Gaussian in the GMM. Top: Large ships. Bottom: RIBs

and RIBs can reach the same speed. However there are clearly differences in the probability of the different speeds. Especially the large ships have some significant peaks. This is partly due to a limited number of training data and partly because large ships frequently will sail with a constant speed.

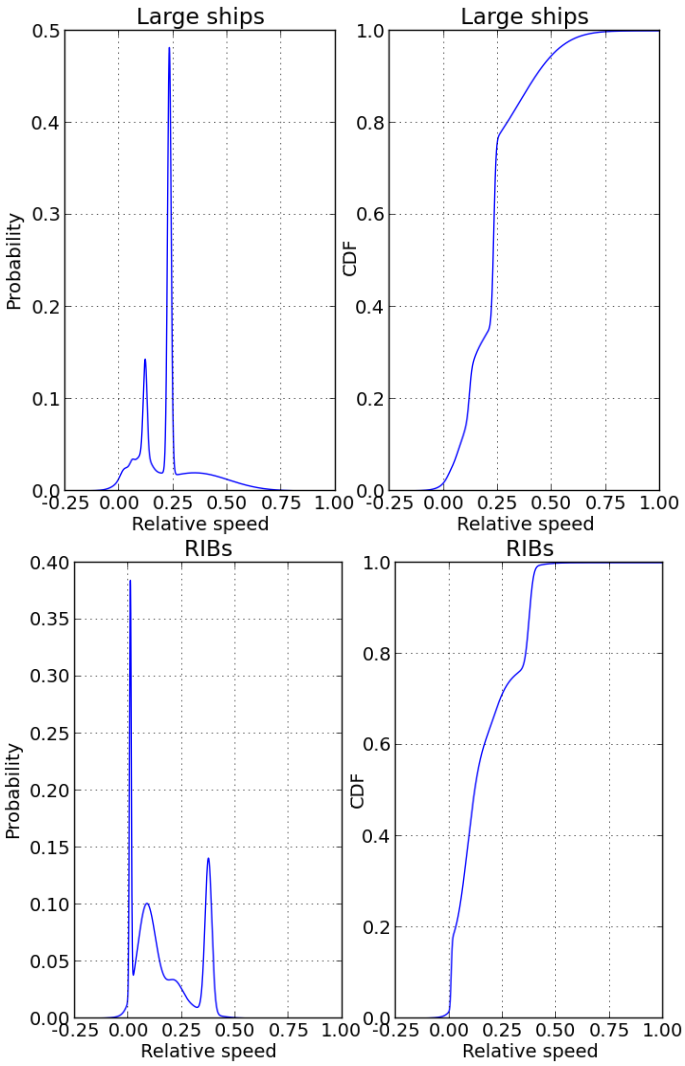


Fig. B.6: The probability for relative speed for RIBs and large ships.

3 Results of experiments

In this section we present the performance of the two proposed classifiers in a setup using real radar data.

OpenStreetMap [13] is used to determine if an object is over road or not. To determine if an object is over sea or land the surface water body data (SWBD) [14] is used. The target database for this work is built up from GPS logs, recorded radar tracks, Automatic Identification System (AIS) reported

from ships and Automatic Dependent Surveillance-Broadcast (ADS–B) used for commercial aircraft. By using many different sources for data it is shown that the algorithm is robust to variations in the sampling rate, i.e. time between measurements and precision in data. No simulated data have been used. The database is split into two, one for training the GMMs and one for testing. There is no overlap between the testing and training databases. The sizes of the databases can be seen in table B.1.

Classes	Trainings size	Test size
Sea buoys	3407	5138
Wakes	1861	354
Cars	10305	2613
Large ships	23130	2462
person	6137	12909
RIBs	23470	720
Helicopters	19143	205
Wind turbines	10255	2354
Commercial aircrafts	6725	12229
Birds	3289	262

Table B.1: Size of the trainings database used for this work. The number is the total number of seconds for each target type

The results from the RGMM method is shown as confusion matrices in table B.2 and the results for the DeltaGMM method is shown in table B.3. The matrix elements are probabilities. The bold font are best performing algorithm for the classes

	Predicted:									
Actual:	Large ships	Commercial aircrafts	Persons	Sea buoys	Birds	Helicopters	Wakes	Wind turbines	RIBs	Cars
Large ships	**0.632**	0.000	0.000	0.000	0.023	0.000	0.000	0.000	0.344	0.000
Commercial aircrafts	0.000	**1.000**	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Persons	0.000	0.000	**0.995**	0.000	0.000	0.003	0.000	0.001	0.000	0.002
Sea buoys	0.000	0.000	0.000	**0.982**	0.000	0.000	0.000	0.000	0.018	0.000
Birds	0.191	0.000	0.000	0.000	0.481	0.000	0.000	0.000	0.328	0.000
Helicopters	0.000	0.000	0.000	0.000	0.000	**1.000**	0.000	0.000	0.000	0.000
Wakes	0.000	0.000	0.000	0.000	0.540	0.006	0.455	0.000	0.000	0.000
Wind turbines	0.000	0.000	0.000	0.355	0.000	0.000	0.000	0.527	0.119	0.000
RIBs	0.086	0.000	0.000	0.000	0.042	0.000	0.000	0.000	0.872	0.000
Cars	0.000	0.000	0.163	0.000	0.000	0.005	0.000	0.000	0.000	0.831
Overall performance:	0.778									

Table B.2: Confusion matrix for the RGMM method

In Fig. B.7 a example of the Egaa Marina scenario is shown with colored tracks which are from the classification using the RGMM method. In Fig. B.8 the same scenario is shown with the DeltaGMM method.

It is clear from Fig. B.8 that the DeltaGMM is better at classifying birds than the RGMM algorithm. Nearly all of the birds are classified correct with only a small penalty in the RIB classification

	Predicted:									
Actual:	Large ships	Commercial aircrafts	Persons	Sea buoys	Birds	Helicopters	Wakes	Wind turbines	RIBs	Cars
Large ships	0.544	0.000	0.000	0.000	0.009	0.000	0.024	0.000	0.423	0.000
Commercial aircrafts	0.000	0.998	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.000
Persons	0.005	0.000	0.987	0.000	0.000	0.000	0.000	0.002	0.000	0.006
Sea buoys	0.000	0.000	0.000	0.962	0.000	0.000	0.000	0.037	0.002	0.000
Birds	0.000	0.000	0.000	0.000	**0.576**	0.008	0.111	0.000	0.305	0.000
Helicopters	0.000	0.000	0.000	0.000	0.000	0.985	0.000	0.000	0.000	0.015
Wakes	0.000	0.000	0.000	0.000	0.008	0.011	**0.980**	0.000	0.000	0.000
Wind turbines	0.000	0.000	0.000	0.247	0.000	0.000	0.000	**0.737**	0.016	0.000
RIBs	0.006	0.000	0.000	0.000	0.015	0.006	0.000	0.017	**0.957**	0.000
Cars	0.002	0.000	0.124	0.000	0.000	0.000	0.000	0.003	0.000	**0.870**
Overall performance:	**0.860**									

Table B.3: Confusion matrix for the DeltaGMM method

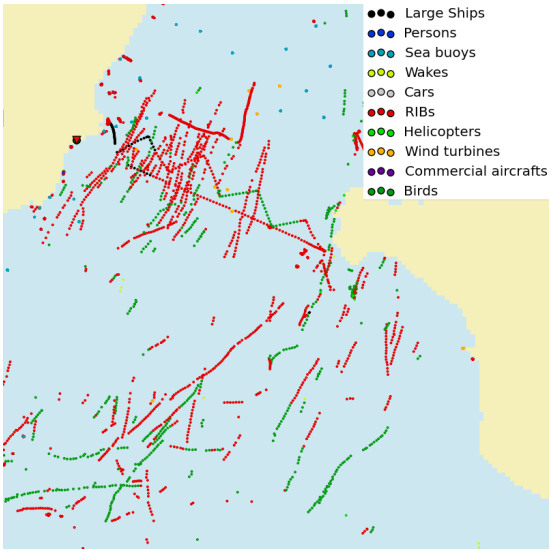


Fig. B.7: The radar image from Egaa Marina colored with the colors from the classification using the RGMM method

4 Discussion

As it can be seen from table B.2 and table B.3 The DeltaGMM method is classifying with 8.2% better accuracy than the RGMM method. As can be seen from the tables the DeltaGMM is better at classifying birds, wakes, wind turbines, RIBs and cars target classes. However the RGMM method is classifying large ships, commercial aircrafts, persons, sea buoys and helicopters better than the DeltaGMM algorithm. We suspect that this is because of a limited training data set as a larger training set is required for DeltaGMM. We are currently in the process of acquiring larger databases.

From table B.2 and B.3 it is interesting to observe that both algorithms can separate targets that are very similar in speed and placement like sea buoys and wind turbines. Both target types are placed in the sea and are

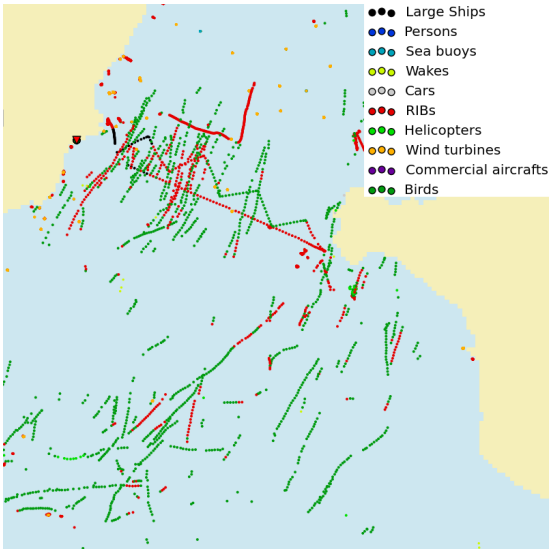


Fig. B.8: The radar image from Egaa Marina colored with the colors from the classification using the DeltaGMM method

stationary targets. For a Doppler radar it will be easy to separate the targets because of the rotating blades on the wind turbine. However in this work only MTI intensity is available. Another possibility is that the blades of the wind turbines are rotating and the primary reflection will therefore change its position in the radar plots, which will lead to a changing target speed.

Birds, large ships, wakes and RIBs are within the same speed range and it is therefore helpful to exploit the temporal dynamics. This is incorporated in the DeltaGMM algorithm. As can be seen in table B.2 and Fig. B.7, birds and RIBs are often misclassified. This is because they have the same speed range and both change directions very frequently. An improved classification is obtained by the DeltaGMM method as can be seen in table B.3 and Fig. B.8. In Fig. B.7 and Fig. B.8 the RGMM and DeltaGMM algorithm are shown on a real world scenario with a RIB sailing from left to right in a straight line and then zigzagging back. The scenario is the same as described in the introduction. In Fig. B.7 the RGMM method is presented is clear that the RIB are classified as a RIBs most of the time. However nearly all the bird tracks (tracks with about the direction going from the lower left corn to the upper right corner) are also classified as RIBs. In Fig. B.8 the DeltaGMM method is shown. It is clear that the RIB is classified a little worse compared to the RGMM method because more of the zigzag back is classified as a bird. Nevertheless a lot more birds are now classified as birds and not RIBs.

5 Conclusions

In this paper we presented two novel on-line classifying algorithms which uses kinematic and location data for classifying real world coastal surveillance radar data. The DeltaGMM algorithm are able to classify with 86% accuracy and the simple RGMM with an accuracy of 78%.

Acknowledgment

The authors would like to thank Esben Nielsen and Christian Glinsvad for fruitful discussions.

References

[1] K. Hansen, A. Thomsen, M. Riis, O. Marqversen, M. Ø. Pedersen, and E. Nielsen, "Detection and tracking of aircraft over wind farms using SCANTER 4002 with embedded tracker 2," *IET Radar*, pp. 1–6, 2012.

[2] T. Porges and G. Favier, "Automatic target classification in SAR images using MPCA," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1225–1228.

[3] X. Ye, W. Gao, Y. Wang, and X. Hu, "Research on SAR images recognition based on ART2 neural network."

[4] M. Garg and U. Singh, "C & R tree based air target classification using kinematics," in *National Conference on Research Trends in Computer Science and Technology (NCRTCST), IJCCT_Vol3Iss1/IJCCT_Paper_3*, 2012.

[5] S. Challa and G. W. Pulford, "Joint target tracking and classification using radar and ESM sensors," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 37, no. 3, pp. 1039–1055, 2001.

[6] S. Sutharsan, R. Tharmarasa, T. Lang, and T. Kirubarajan, "Tracking and classification using aspect-dependent RCS and kinematic data," in *SPIE Defense and Security Symposium*. International Society for Optics and Photonics, 2008, pp. 696 913–696 913.

[7] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, "Classification and tracking of moving ground vehicles," *Lincoln Laboratory Journal*, vol. 13, no. 2, pp. 275–308, 2002.

[8] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *Signal Processing, IEEE Transactions on*, vol. 50, no. 2, pp. 174–188, 2002.

[9] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.

[10] G. Hamerly and C. Elkan, "Learning the k in k-means," in *In Neural Information Processing Systems*. MIT Press, 2003, p. 2003.

[11] T. W. Anderson and D. A. Darling, "Asymptotic theory of certain" goodness of fit" criteria based on stochastic processes," *The annals of mathematical statistics*, pp. 193–212, 1952.

[12] G. R. Shorack and J. A. Wellner, *Empirical processes with applications to statistics*. Siam, 2009, vol. 59.

[13] Openstreetmap. [accessed 14 january 2014]. [Online]. Available: http://www.openstreetmap.org/copyright/en

[14] SWBD. [accessed 4 december 2014]. [Online]. Available: http://dds.cr.usgs.gov/srtm/version2_1/SWBD/

Paper C

Using Position Uncertainty in Recursive Automatic Target Classification of Radar Tracks

Lars W. Jochumsen, Esben Nielsen, Jan Østergaard, Søren H. Jensen, Morten Ø. Pedersen

The paper has been published in the
IEEE, International Radar Conference, pp. 168–173, 2015.

© 2015 IEEE
The layout has been revised.

Abstract

In this paper, we show how radar plot uncertainty can be included leading to a more robust classification of targets observed by a rotating 2D radar. Targets far from the radar will have a greater uncertainty in the position and therefore the estimated speed of the targets will be more uncertain. The uncertainty is sensor dependent and will therefore need to be taken into account when classifying based on training data from multiple different sources. Including the uncertainty in the radar plot positions, leads to an improved estimate of the probability of a target belonging to any given class in a list of possible classes. We show results for two synthetically generated cases, where we include the uncertainty and from a real world radar scenario.

1 Introduction

The aim of this work is to provide a more robust classification of radar tracks and thereby increase the situational awareness for coastal surveillance radars. In recent years, trackers are becoming standard part of radars used for coastal surveillance, Vessel Traffic Services, surface movement radars at airports etc. These radars must have a very high sensitivity such that small targets such as e.g. rigid inflatable boats (RIBs) can be detected, leading to detection and tracking of undesired targets such as birds, see Fig. C.1. Removal of such tracks requires classification, with automated decision of which targets to display. However it is important that such classification does not degrade the situation awareness by misclassifying tracks or show overconfident probability for a target class.

Radar measurement from a 2D rotating surveillance radar has a high uncertainty in the position, especially in azimuth, therefore derived measurements i.e. speed V_n in this work, has a high uncertainty. These uncertainties must be included when using speed as a feature in target classification. Normally in classification these uncertainties are ignored because the sensor is the same in both training and classification. If the sensor is not the same, a small training set is used to adapt the training model to a new model which fit the sensor ([1], [2]). Some exceptions could be the k-nearest neighbor (KNN) algorithm, where a weighting factor for the k-nearest could be given from the uncertain [3]. The downside of the KNN approach is to estimate the real PDF k must approach N where N is the training sample number. Further KNN must hold every training sample in memory. We therefore choose to make a parametric approach.

In our case, the model is trained from different sources, such as GPS, automatic identification system (AIS), automatic dependent surveillance-broadcast (ADS-B) and different radars. A fixed interval smoother [4] has been applied to the training data before the data are used to train the classifier, such that

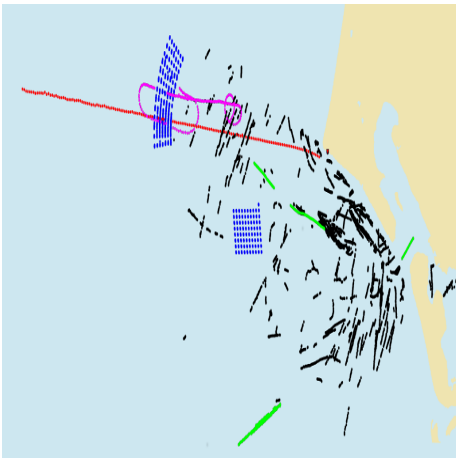


Fig. C.1: The plot shows a real world scenario of a surveillance radar at the east coast of Denmark. The red colored track is a helicopter. The pink colored track is a small general aviation aircraft of the type Grumman GA-7. The blue colored tracks are wind turbines. Some of the wind turbines in the area furthest away from the radar are smeared together because of the azimuth resolution. The green colored tracks are unknown ships. The rest of the tracks (black) are birds.

the uncertainty in these data is reduced. As for real-time classification, it is not possible to apply fixed interval smoothing as the data is streamed in real-time to the classifier. This creates a problem where the real-time data does not fit the trained model. We could use the approach where we has a small set of training data for each radar, antennea and target. However, it will be a huge task to collect training data for all target classes for all possible surveillance radars with different sensor characteristics. We therefore seek a general way of making the classifier sensor independent. Furthermore, the uncertainty can change from target to target, as an example, a large wind turbine with huge blades can have different main scatter points at each scan of the radar, whereas a medium ship will often only have one main scatter point. Additionally, the range to the target and weather conditions can have an impact on the uncertainty, as a large range to the target will result in large uncertainty in azimuth. The weather in coastal surveillance can have an impact on sea clutter, which results in different amplitude characteristics because of the radars preprocessing. We therefore propose to use a method, which includes the uncertainty in the measurement for each probability update for the target.

In this work we use the term classification as a term which describe a broad identification of a target, such as birds, helicopters, large ships.

2 Method

One possible way to include uncertainty is given in [5] where a tracking filter is applied. This tracking filter handles the uncertainty by mapping the uncertainty from the feature space to the class space. In this work we take a different approach where the uncertainty can be included to an on-line Bayesian classifiers such as ([6], [7]). We proposed to use the uncertainty in the position to estimate the uncertainty in the speed, which we use as a feature in the classifier. The uncertainty of the position is estimated from the plot extend in range and azimuth. We include the uncertainty in the position to find the probability distribution of all possible speeds, followed by an estimation of the probability for jumping from one of the speeds to another speed at the next scan of the radar. The plot association is done by the radar tracker [8].

2.1 Calculation of the Speed Uncertainty

From the radar we get plots which have a position $\mathbf{r}_n = [x_n, y_n]^T$ and a covariance $\mathbf{\Sigma}_n^p$ of the position where the subscript n is the update number and the superscript p is to emphasize that it is the position covariance. We assume that the position is a Gaussian distributed and the position is independent of earlier and future uncertainties, where the latter assumption is valid when the time between between the position is not used. We are interested in the speed V_n estimate and the uncertainty of the speed. As a simple approximation, we can calculate the velocity as a two point estimate from

$$\mathbf{v}_n = \frac{\mathbf{r}_n - \mathbf{r}_{n-1}}{\tau} \tag{C.1}$$

where τ is the time between the measurements i.e. $\tau = t_n - t_{n-1}$. The resulting covariance of the velocity is then

$$\mathbf{\Sigma}_n^v = \frac{\mathbf{\Sigma}_n^p + \mathbf{\Sigma}_{n-1}^p}{\tau^2}. \tag{C.2}$$

The speed can be calculated as the magnitude of $\mathbf{v}_n$, that is $V_0 = \|\mathbf{v}_n\|_2$. It is computational expressive to calculate the variance of the magnitude of a vector with non-zero mean and different variance for the components. Instead we propose to use the average of the eigenvalues of the speed covariance as an estimate of a non-zero mean vector with dissimilar variance in both dimensions. For a two-component vector, the average of the eigenvalues can be found as the trace of the matrix

$$\sigma_n^v \approxeq \frac{Tr\left(\mathbf{\Sigma}_n^v\right)}{2}. \tag{C.3}$$

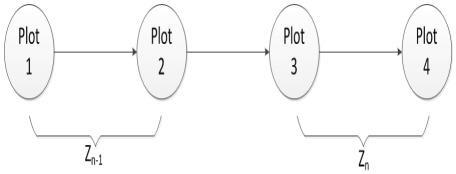


Fig. C.2: An overview of how the measurement from the radar is used. Plot is the combined information from the radar i.e. position and uncertainty. z is described in 2.1.

We propose to use a Rice distribution to model the probability distribution of the speed. The distribution converges to a Rayleigh distribution when the speed is low and converges to a Gaussian distribution when the speed is high. The Rice distribution is given as

$$P_m(V_n|z_n) = \frac{V_0}{\sigma_n^v} exp\left(\frac{-(V_n^2 - V_0)}{2\sigma_n^v}\right) I_0\left(\frac{V_n V_0}{\sigma_n^{n^2}}\right), \tag{C.4}$$

where I_0 is the modified Bessel function of the first kind with zero order [9] and $z_n = \{x_n, x_{n-1}, y_n, y_{n-1}, \Sigma_n^p, \Sigma_{n-1}^p\}$ is the set of two measurements (plots) needed to deduce the speed and its covariance. The subscript m in P_m is to emphasize that this is the measurement uncertainty PDF.

2.2 Classifying with Uncertainty

In the following we assume that the latest set of measurement z_n is statistically independent of the previous measurements. As the derived speed depends on a set of two measurements, a simple way of ensuring statistical independence is to group the plots pairwise, and only update the classification for every other measurement. This will decouple the dependencies, see Fig. C.2. Let us assume that we have received n-1 updates we can now predict the probability of a given class conditioned on the latest speed as

$$\begin{aligned} P(c|V_n, \{z_{n-1}\}) = \int_{V_{n-1}} & P(c|V_n, V_{n-1}, \{z_{n-2}\}) \\ & P_m(V_{n-1}|z_{n-1}) dV_{n-1}, \end{aligned} \tag{C.5}$$

where $\{z_{n-1}\}$ is the set of paired measurement up until n-1 updates, z_{n-1} is the current paired measurement and c is the class. We can use Bayes' theorem on the class dependent probability in order to get the probability of the class given the speeds

$$\begin{aligned} & P(c|V_n, V_{n-1}, \{z_{n-2}\})) = \\ & \frac{P(V_n|c, V_{n-1}) P(c|V_{n-1}, \{z_{n-2}\})}{\sum_{c'} P(V_n|c', V_{n-1}) P(c'|V_{n-1}, \{z_{n-2}\})}. \end{aligned} \tag{C.6}$$

We only have $P(V_n, V_{n-1}|c)$ from the training of the classifier, and we need $P(V_n|c, V_{n-1})$. This can be calculated as

$$P(V_n|c, V_{n-1}) = \frac{P(V_n, V_{n-1}|c)}{P(V_{n-1}|c)}, \tag{C.7}$$

by exploiting the first order Markov chain assumption. We use a Gaussian mixture model (GMM) to model $P(V_n, V_{n-1}|c)$. The GMM is fitted to the training data by the EM algorithm [10]. After receiving the nth update, we can calculate the probability for the class as

$$P(c|z_n) = \int_{V_n} P(c|V_n, \{z_{n-1}\}) P_m(V_n|z_n) dV_n. \tag{C.8}$$

For each prediction step we have to store the latest $P(c|V_n, \{z_{n-1}\})$ distribution as this must be used in the next prediction.

As $P(V_n, V_{n-1}|c)$ is a GMM, the analytical expression for the distribution $P(c|V_n, \{z_{n-1}\})$ will contain an ever increasing number of mixtures for every update. This is impractical from an implementation point of view, and in principle possible to solve by fitting a new GMM with a specified number of mixtures at each update. As this is very computationally demanding, we have chosen to solve the equations numerically. We change the integral in (C.6) and (C.8) to a sum. We normalize P_m such that $\sum_{\{\hat{V}\}} P_m(\hat{V}|z_{n-1}) = 1$, where $\{\hat{V}\}$ is the set of speeds we sum over, this will ensure that the probability for class given by (C.8) will sum to 1. $P(c|V_n, \{z_{n-1}\})$ will be stored as a set of data points from the distribution. In Fig. C.3 we have tried to illustrate this process. In Fig. C.4 a typical example of the P_m and the GMM is shown. From this it can be seen that P_m typically has a smaller variance then the GMM, it is therefore natural to use the P_m to find the points to include in the sum.

3 Simulations

In this section, we will present the simulation results with synthetically generated data. The classes are a low-speed, a medium-speed and a high-speed class. We use the feature space $[V_n, V_{n-1}, t]^T$ where V_n is the newest speed derived from the plots, V_{n-1} is the previous speed and t is the time between the two speeds. We use a modified DeltaGMM approach explain in [6]. We use the above mentioned features and using a Markov chain assumption, that is $P(V_n|V_{n-1}, c)$. The speed classes are trained on a large amount of noisy signal with a constant speed and fixed interval smoothing [4] has been applied. The three classes trained GMMs is shown in table C.1. As it can be seen from the table the GMMs has only one mixture for each class. It can also be seen that the average speed from the low-speed class is around 4 m/s, the medium-speed class is 7 m/s and the high-speed class has an average speed

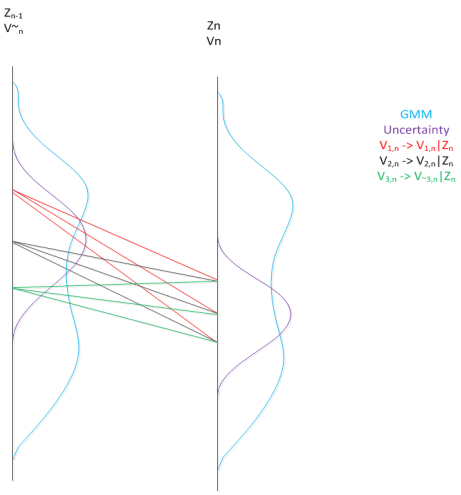


Fig. C.3: Here is shown an example of the probability calculation. The figure is only for the calculation of the probability for one class i.e. a Bayesian update must be conducted after this calculation. P_m is purple and the GMM is blue. We estimate three possible starting points in speed (indicated by red, black and green) from the predict step Z_{n-1}. When we get a new measurement,we can now calculate the probability from going from the predicted speed to the actual speed Z_n. By multiplying this with the GMM and summing over the three speeds we can calculate the probability of the jump between the two measurement given that specific class.

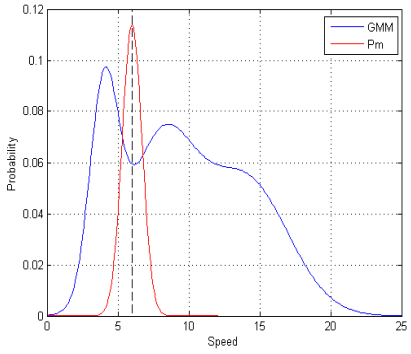


Fig. C.4: A synthetic example of trained GMM PDF and the measurement PDF, the dashed line is the mean of the measurement PDF

Table C.1

class	Mean vector	Covariance
low-speed	$\begin{bmatrix}4.03 & 4.03 & 5.5\end{bmatrix}^T$	$\begin{bmatrix}1.62 & 0.16 & 0.0\\ 0.16 & 1.63 & 0.0\\ 0.0 & 0.0 & 8.25\end{bmatrix}$
medium-speed	$\begin{bmatrix}7.02 & 7.02 & 5.5\end{bmatrix}^T$	$\begin{bmatrix}1.66 & 0.16 & 0.0\\ 0.16 & 1.67 & 0.0\\ 0.0 & 0.0 & 8.25\end{bmatrix}$
high-speed	$\begin{bmatrix}8.02 & 8.02 & 5.5\end{bmatrix}^T$	$\begin{bmatrix}1.66 & 0.16 & 0.0\\ 0.16 & 1.67 & 0.0\\ 0.0 & 0.0 & 8.25\end{bmatrix}$

around 8 m/s. Further it can be seen that the covariance is nearly the same for all three classes. We have two cases in the simulation. In 3.1 the purpose is to show how the classifier handles a target that switches from one class to another class. The second case is in 3.2 where we show how the classifiers handle a target with constant speed, but different uncertainty in the position measurement.

3.1 Linearly Increase in Speed

The two Figs. C.5 and C.6 show the results of a linearly increasing speed. This behaviour simulates that a target accelerates and thereby changes from one class to another. The plot uncertainty is $\Sigma_n^p = \begin{bmatrix}1.0 & 0.1\\ 0.1 & 1.0\end{bmatrix}$.

3.2 Steady Speed with High Uncertainty

In Figs.C.7 and C.8 the speed is constant with a very high uncertainty in the plots i.e. $\Sigma_n^p = \begin{bmatrix}30.0 & 3.0\\ 3.0 & 30.0\end{bmatrix}$. This simulates that we are very unsure about the measured position. In Fig. C.9 the uncertainty in the plots are $\Sigma_n^p = \begin{bmatrix}5 & 0.5\\ 0.5 & 5\end{bmatrix}$. This simulates that we are more confident in the position of the plots. The speed is the same as in Figs. C.7 and C.8.

4 Real world scenario results

The goal for this work is not only to classify the targets correctly, but also to provide a probability of classification which reflects our confidence in the

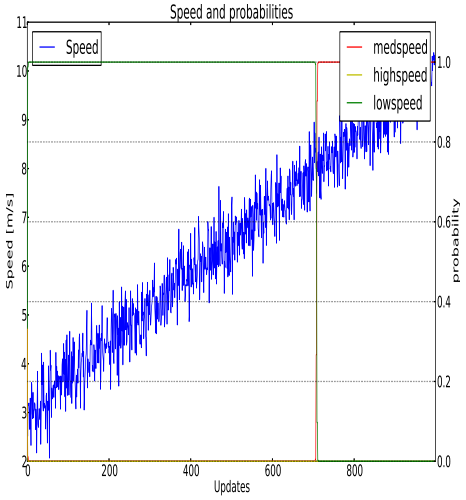


Fig. C.5: Plot where the algorithm ignores the plot uncertainty. The blue color is the speed measurement.

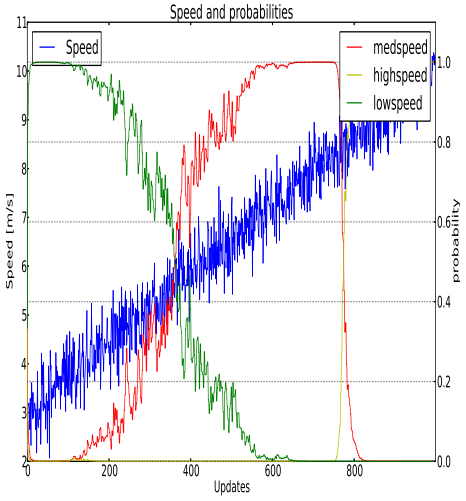


Fig. C.6: Plot where the algorithm includes the plot uncertainty. The blue color is the speed measurement.

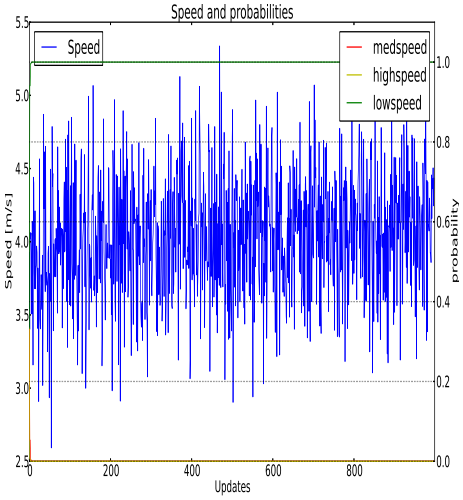


Fig. C.7: Figure where a target has a steady speed. The measurement has high uncertainty however, in this figure, the algorithm ignores the uncertainty.

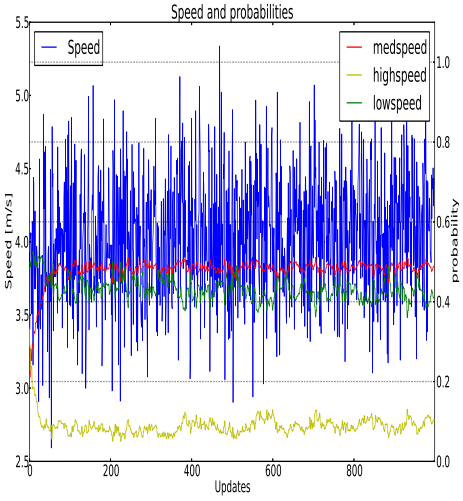


Fig. C.8: Plot showing the same as Fig C.7, but in this figure the algorithm includes the uncertainty in the plots.

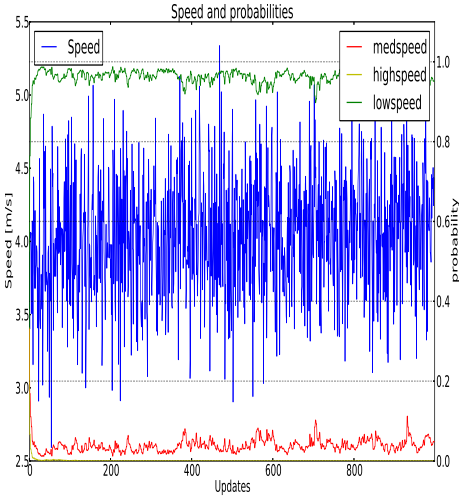
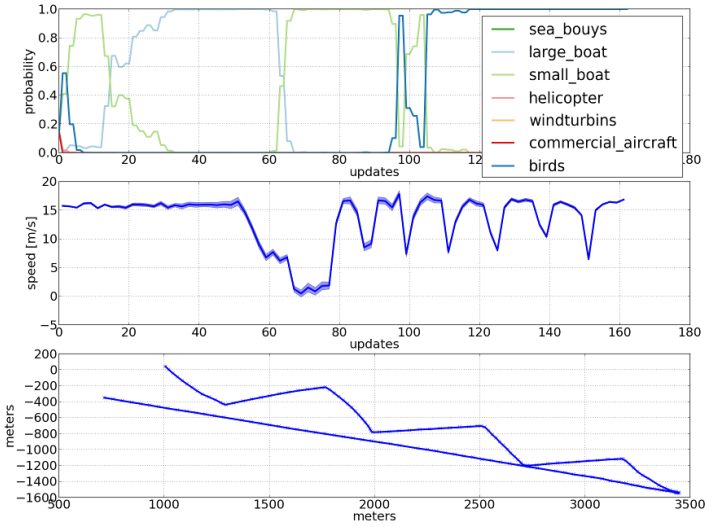


Fig. C.9: Figure showing the same speed progress as Figs. C.7 and C.8 but with a lower uncertainty in the plots. The algorithm includes the uncertainty

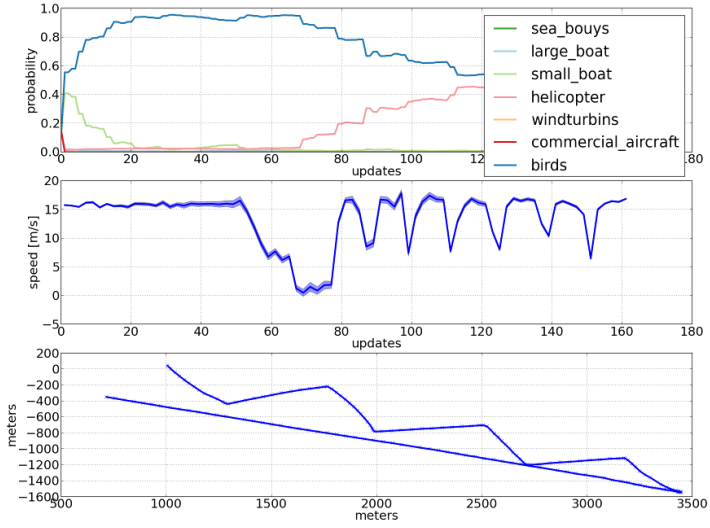
target class i.e. the confidence of the classification must be match in the probability. That is if we get a probability of one we must be completely sure that it is the real class of the target. In Fig. C.10 a RIB's track is shown where C.10a is without using the uncertainty in the measurement and C.10b is with the uncertainty. In Fig. C.11 a bird track is shown from the scenario shown in the introduction, Fig. C.1. Figure is also with and without the uncertainty. Further the whole scenario from C.1 is shown in C.12 where C.11a is without using the uncertainty and C.11b is with using the uncertainty.

5 Discussion

From Fig. C.5 we can see that the when uncertainty is not included, there is a long lag in changing classification result for an target that accelerates and changes class. This is due the the fact, that the probability score of the initial class has continued to increase monotonically, leading to an overconfidence in the classification result. i.e. very high confidence on the class. In automatic target classification, it is important that if the classifier has high confidence the target must belong to the class. It can clearly be seen this is not the case in Fig. C.5. A way to solve this is to use the uncertainty in the measurement as described in 2. This can be seen in Fig. C.6, by using the uncertainty we reduce the confidence in the derived speed and the classifier will not be as sure of the class. In the figure, it is clear that the classifier is more correct when comparing to table C.1. The classifier is changing class at around 5.6

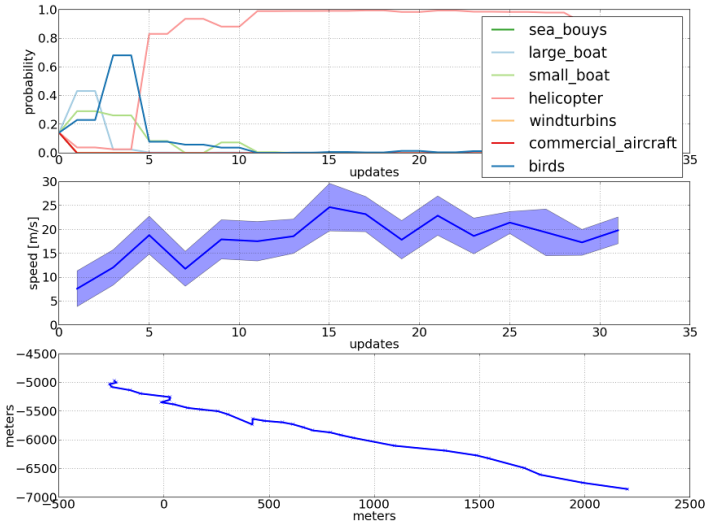


(a) Without use of uncertainty

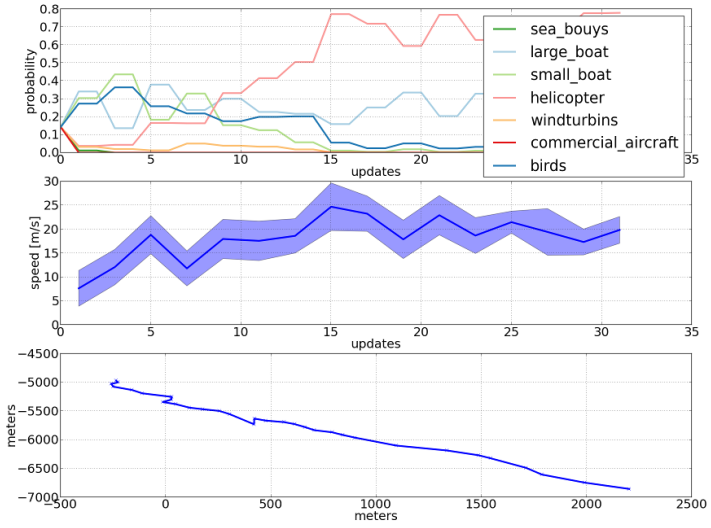


(b) With use of uncertainty

Fig. C.10: A Rib sailing outwards from a harbour and then zigzagging back. C.10a is without uncertainty and C.10b is with uncertainty

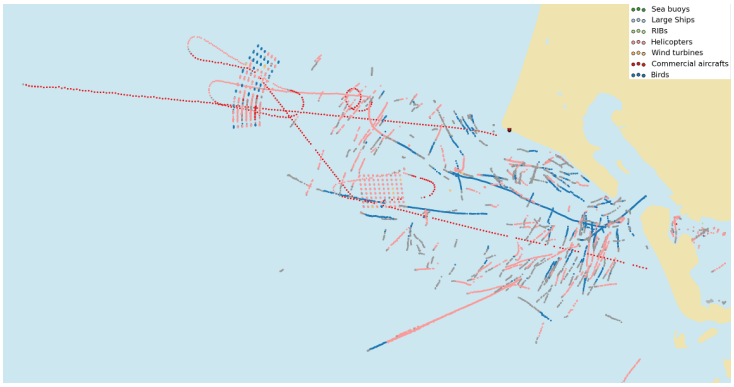


(a) Typically sea bird without using the uncertainty

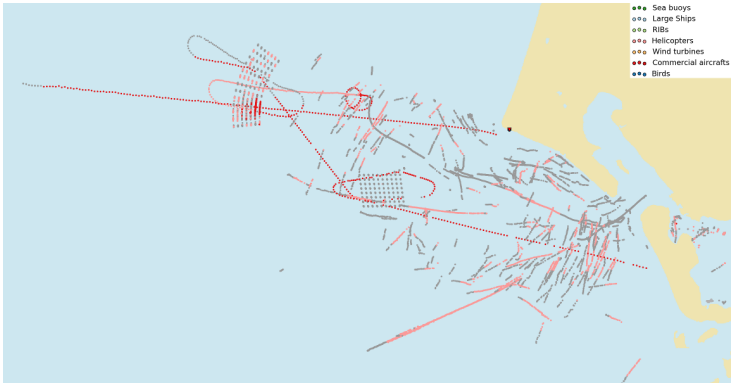


(b) Typically sea bird with using the uncertainty

Fig. C.11: A typically bird track from a sea bird. This is a bird from the scenario in the introduction Fig. C.1. Fig. C.11a is without using uncertainty and Fig. C.11b is with using the uncertainty



(a) The scenario in the introduction classified without using the uncertainty



(b) The scenario in the introduction classified using the uncertainty

Fig. C.12: The radar scenario from the introduction rev classified with and without uncertainty. Gray is unknown (if the probability is is below 97 % for a given class)

m/s, which is nearly half way between low and medium speed class. Further the classifier also change class around 8 m/s which also matches table C.1.

In Fig. C.7 a steady speed scenario is shown with large uncertainty in plots from the radar. This figure is without using the uncertainty. From the figure it is shown that the classifier which does not use the uncertainty classifies this as low speed class, which the speed also indicates. However, the uncertainty in the plots is big and we cannot be sure on the speed. This can be seen in Fig. C.8 where the uncertainty is used. It is clear from the figure that the classifier cannot choose a unique class that the measurement belongs. The classifier finds a steady state estimate. In Fig. C.9 the same scenario as Figs. C.7 and C.8 but with lower uncertainty in the radar plots it can be seen that we can exclude the high-speed class, however the classifier can still not choose between the low and medium speed classes.

As mention in the section 4 we are not interested in classifying correct but more in the probability should match the confidence of the result. As it can be seen from Fig. C.10 the result without using the uncertainty shifting between classes and further the probability is sometimes 1.0 which means the classifier must be 100 % certain for the class, but later the classifier change to other class. If we see Fig. C.10b where we use the uncertainty the classifier does not rapidly change from one class to other, and the probability never reaches 1.0 which mean the classifier is not to confident of the class. From Fig. C.12 it is clear that the uncertainty algorithm does not get a better classification results than without uncertainty, however the confidence of the classification match better with the probability as many of the tracks is coloured gray, which is unsure on the class.

6 Conclusion

In a surveillance scenario the classifier output probability vector for the target belonging to one amongst different classes must correctly reflect how confident the classification algorithm is in the target class assignment when used to filter unwanted targets. By introducing the uncertainty in the classifier algorithm, we have shown that it is possible to address the problem of having high uncertainty in the test data for an algorithm while training on low uncertainty data. We have also shown that by introducing the uncertainty we can make the classifier more robust for misclassification as the probability reflects the confidence of the classification.

References

[1] M. J. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer speech & language*, vol. 12, no. 2,

pp. 75–98, 1998.

[2] Y.-H. Chao, W.-H. Tsai, and H.-M. Wang, "Discriminative feedback adaptation for GMM-UBM speaker verification," in *Chinese Spoken Language Processing, 2008. ISCSLP'08. 6th International Symposium on*. IEEE, 2008, pp. 1–4.

[3] T. Denoeux, "A k-nearest neighbor classification rule based on dempster-shafer theory," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 25, no. 5, pp. 804–813, 1995.

[4] D. E. Catlin, "Fixed interval smoothing," in *Estimation, Control, and the Discrete Kalman Filter*. Springer, 1989, pp. 188–199.

[5] W. Mei, G.-l. Shan, and Y.-f. Wang, "A second-order uncertainty model for target classification using kinematic data," *Information Fusion*, vol. 12, no. 2, pp. 105–110, 2011.

[6] L. W. Jochumsen, M. Pedersen, K. Hansen, S. H. Jensen, and J. Østergaard, "Recursive bayesian classification of surveillance radar tracks based on kinematic with temporal dynamics and static features," in *Radar Conference (Radar), 2014 International*. IEEE, 2014, pp. 1–6.

[7] L. W. Jochumsen, J. Ostergaard, S. H. Jensen, and M. O. Pedersen, "Modelling temporal variations by polynomial regression for classification of radar tracks," in *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European*. IEEE, 2014, pp. 1412–1416.

[8] K. Hansen, A. Thomsen, M. Riis, O. Marqversen, M. Ø. Pedersen, and E. Nielsen, "Detection and tracking of aircraft over wind farms using SCANTER 4002 with embedded tracker 2," *IET Radar*, pp. 1–6, 2012.

[9] K. D. Ward, R. J. A. Tough, and S. Watts, *Sea clutter: Scattering, the K Distribution and Radar Performance*. IET, 2006.

[10] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.

Paper D

Exploiting Position Uncertainty in Recursive Radar Track Classification

Lars W. Jochumsen, Morten Ø. Pedersen, Søren H. Jensen, Jan Østergaard

The paper has been published in the
EURASIP, Journal on Advances in Signal Processing, Submitted undergoing review.

©2016 EURASIP
The layout has been revised.

Abstract

We present a Bayesian recursive classifier that includes sensor uncertainty about the position measurements provided by a 2D scanning radar when classifying radar tracks. Knowledge of the measurement uncertainty makes it possible to use different sensors for training and testing. Thus, one can increase the training set by including data from a variety of sensors (not only radars), and during tests it is possible to use different radars and radars with different range to targets. We test our classifier in real world coastal surveillance radar scenarios and show that the probability matches better with the confidence of the classification.

1 Introduction

With an ever increasing need to monitor coastlines and shipping to avoid collisions and detect illegal activities, the number of high performance radar sensors installed grow continuously. Modern, high sensitivity 2D scanning radars deliver large amounts of information, and there is a growing need to classify detected targets in order to lower the burden placed on system operators. For example, radar systems monitoring the littoral zone could classify targets as birds, large vessels, static surface target etc., allowing operators to filter targets of interest.

As an example, a typically scenario of a coastal surveillance area is shown in Fig. D.1, where a huge number of birds and wind turbines tracks are observed, making observations of particular targets of interest difficult. By filtering away tracks originating from e.g. birds, a reduced number of potential targets are required to be evaluated by an operator. For example it has been shown in [1–5] that it is possible to classify tracks and thereby remove unwanted tracks. As it was also shown in [1, 2] the probability for the targets was highly fluctuating as seen in Fig. D.2 These large fluctuations in the probability of a target belonging to a given class, makes classification and filtering difficult. As will be described in this paper, such fluctuations in classification may occur, when a classifier is trained on training data with little position uncertainty and subsequently used to classify data with significant measurement position uncertainty. Fluctuations in data due to this uncertainty will be interpreted as e.g. large variations in parameters of interest and will lead to rapid changes in classification output. As the radar used for this work only delivers position it is not possible to use the most normal approach for target classification like e.g. Doppler and high resolution techniques [6]. In [6] it is also proposed to use speed and acceleration. We believe that because of the noise in the deduced acceleration it can not be used. We therefore only use speed as this most be the most important features for classification, and as speed is derived from positional measurements, an uncertainty in measuring

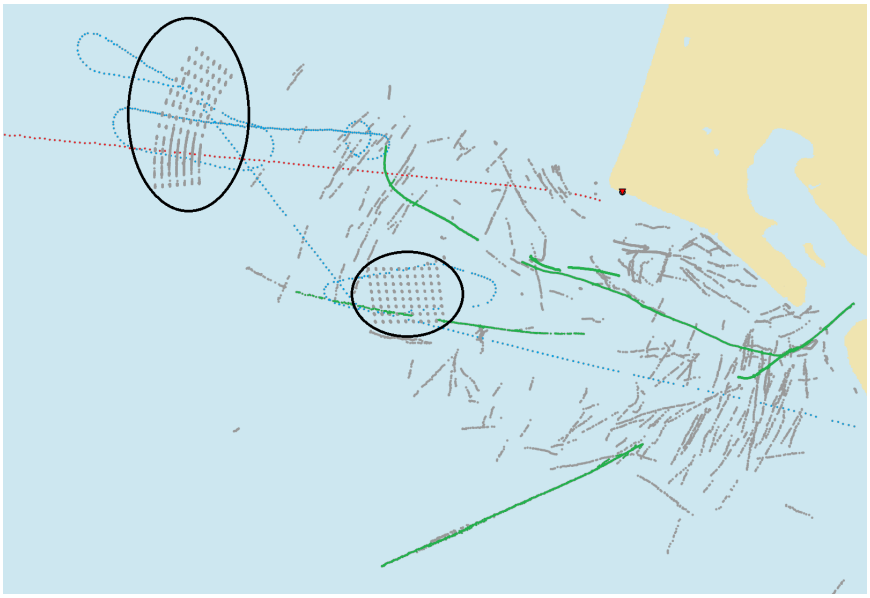


Fig. D.1: A scenario of two wind farms in Denmark. The windfarms a encircled with a black color. A commercial aircraft comming from left to right, coloured with red. A commercial aircraft circling the two windfarms coloured with blue and some unknown sea vessels coloured with green. The rest of the tracks originates from birds

a targets position will translate into an uncertainty in the speed feature used for classification.

One solution is to train only on data acquired from the same sensor as used subsequently to observe the targets of interest. This is typically done in speech recognition [7, 8]. This method requires a huge amount of training data, as there must be enough training data for each class with different radar sensors, in changing weather conditions etc.

To make a balanced training data set, it is desired to use data from a variety of sensors to ensure that all targets of interest are represented in the training database. A classifier may subsequently be used to classify data from different radar sensors, where different antennas, range to targets etc, will result in different sizes of measurement uncertainty for the same target.

To overcome the problem of having a different sizes of measurement uncertainty for the same target one could use parallel tracking filters such as Kalman filters [9] or Interacting Multiple Model (IMM) [10] filters. The IMM filter has be used in [11] to classify three different classes of simulated data. However, training and running the parallel filters on real data is computationally challenging.

In this paper, we prose a method to classify radar tracks from a 2D scan-

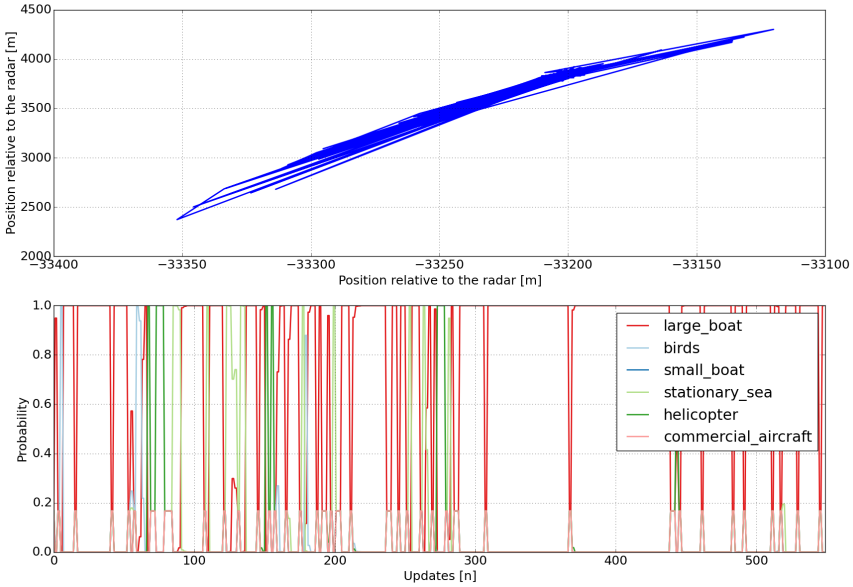


Fig. D.2: The top graph shows the measurement position of a stationary target and the bottom graph shows the probability curve for the different target classes, this is classified using a recursive Bayesian classifier.

ning radar without the use of Doppler information. The proposed classifier use a probabilistic approach to incorporate the uncertainty in the position estimate while the training can be done on ground truth data. By this approach we can utilize different sensors e.g. GPS to train the classifier and making it possible for the classifier to handle different sensors when testing the classifier. The classifier uses Gaussian mixture models (GMM) to model the classes probability distribution functions (PDFs). We use the Bayesian classifier such that the classifier is a recursive classifier i.e. the probability will change as more knowledge of the target is collected over time. We use real world radar data to show the performance of the classifier.

1.1 Preliminaries

The radars used are 2D scanning radars with either solid-state power amplifier (SSPA) or travelling-wave tube (TWT) amplifiers. The radars are fully coherent radars with pulse compression and inbuilt tracker. The radar delivers both normal radar video and moving target video (MTI) [12]. The radars observe targets in a polar coordinate system, with an azimuth resolution between 0.36° to 0.6° and a range resolution of 12 meters [13–15].

Apart from kinematic features derived from radar observations, in this

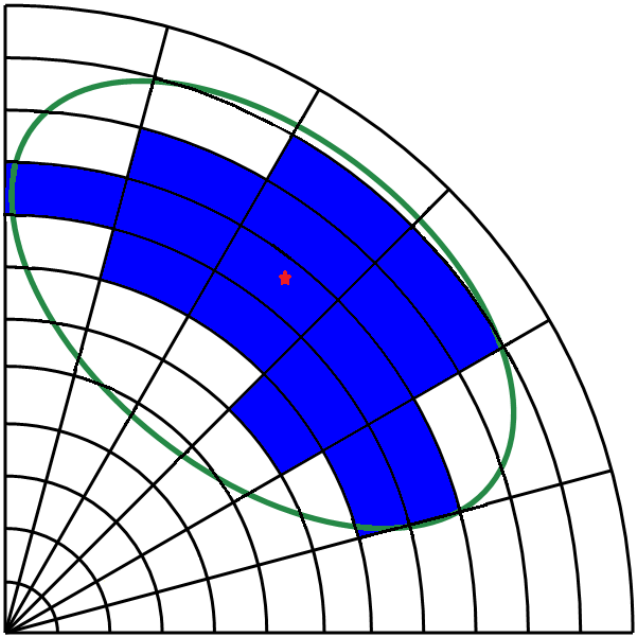


Fig. D.3: An image of radar cells. We model the cell to be an ellipse in the Cartesian coordinate system. The blue colored cells could be a target which cover these cells. Further the red star in the middle is the estimated position r_j and the green ellipsoid is the estimated position uncertainty Σ_j^p is the estimated position and uncertainty described in (D.17), (D.18) and (D.19). Each radar cell has a unique index i.

work we also use features such as backscatter intensity from the radar as well as geodetic information such as whether a target is observed over land or sea, presence of roads and whether or not a clear line of sight exist between the radar and a target at a given height. The measurement from a radar consist of a number of radar cells which will be wider at far range. Each of the cells is a measurement of the backscatter and if the backscatter is above a certain threshold a target may be present. A typical target will normally occupy multiple radar cells, see Fig. D.3. The uncertainty originates from the number of connected radar cells as it is hard to predict where the correct center is. The number of cells and the intensity in each cell can differ from scan to scan as a result of noise in the hardware, adaptive preprocessing and movements of the target with respect to the radar [12].

2 Method

In this section, we present our classifier, which is an extension of our conference contribution given in [16]. The overall goal is to obtain the class

probability $P(c_i|Z_n)$ for all the classes c_i, $i = 1, \cdots, L$, given the set of current and past radar measurements $\{Z_n\} = Z_n, Z_{n-1}, \cdots, Z_1$. Measurements from a practical radar are generally contaminated by noise, e.g., due to being based upon noisy radar return signals. The measurement do therefore not contain accurate estimates of the true target position or true target speed. Indeed, the 2D scanning radar that we are considering in this work, outputs the position of the target and the uncertainty of the position. The uncertainty is typically range dependent and much larger than for example a GPS position. Furthermore, a time stamp of the measurement time is provided. When deriving the class probabilities, the inherent uncertainty within the radar measurements should be taken into account. Moreover, in order to use the classifier on-line, we are interested in a recursive approach, which exploits previously estimated class probabilities. We begin in Section 2.1 by deriving the classifier, when we have only a single feature, i.e., the target's speed. Then, in Section 2.2, we show how to model the uncertainty within this speed feature. In section 3, we then include multiple features and in Section 4, we show how to take the uncertainty into account in the classifier.

2.1 Recursive estimates of the conditional class probabilities

Let $\hat{V}_n$ denote the true speed of the target at the measurement index n, and let V_n denote the estimated speed. Moreover, let $V_n = \hat{V}_n + \xi_n$, where ξ_n is the estimation error. Before we derive the classifier, some assumptions are needed.

Assumptions:

(i) The estimation errors $\{\xi_i\}$ are mutually independent i.e. $P(\xi_i|\xi_j) = P(\xi_i), \forall j, i, i \neq j$.

(ii) $c_i \leftrightarrow V_n \leftrightarrow Z_n, \forall i, n$.

(iii) $V_n \leftrightarrow Z_n \leftrightarrow Z_k, \forall n, k, n \neq k$.[1]

(iv) $V_n \leftrightarrow (c_i, V_{n-1}) \leftrightarrow \{V_{n-2}\}, \forall i, n$.

(v) $V_n \leftrightarrow V_{n-1} \leftrightarrow \{V_{n-2}\}, \forall n$.

Assumption (i) implies that the measurement noise are mutually independent and hence the uncertainties in the position estimates are also mutually independent. This assumption can be justified by thinking of the backscatter as being a function of orientation, distance and angle to the target combined with the antenna beam and the adaptive preprocessing in the radar. Assumption (ii) implies that by knowing the speed of the target, the class is

[1] We use double sided arrows as in $x \leftrightarrow y \leftrightarrow z$ to denote a Markov chain, which means that x and z are conditionally independent given y.

independent of all measurement. The reasoning for this is upon knowledge of the speed, the remaining information in the measurement is the uncertainty about the class then that already provided by the speed. For coastal surveillance radars this can be justified as all targets will be point targets because of the typical distance to the target and the large antenna beam. Assumption (iii) implies that by knowing the measurement Z_n the remaining information in speed V_n is the estimation error, which can be assumed independent of the other measurement Z_k. This assumption is a requirement to make the classifier recursive. Assumption (iv) implies that the speed can be modelled as a first order Markov chain. This is justified because of knowing a targets speed and the class of the target it is possible to do a very accurate predication of the next speed. Of course more information is present by also using speed V_{n-2} i.e. acceleration. However, by this assumption, less computational load is needed. Assumption (v) implies that we can model the speed without knowing the class as a first order Markov chain. This assumption can be justified by implicit knowing that we have physical targets, which cannot accelerate infinitely fast. It is therefore possible to obtain an accurate prediction of the speed given the last speed.

We will now start by describing the probability for c_i:

$$P(c_i|\{Z_n\}) = \int \cdots \int P(c_i|\{V_n\},\{Z_n\})P(\{V_n\}|\{Z_n\})d\{V_n\}. \tag{D.1}$$

By using assumption (iii), we can rewrite (D.1) as

$$P(c_i|\{Z_n\}) = \int \cdots \int P(c_i|\{V_n\})P(\{V_n\}|\{Z_n\})d\{V_n\}. \tag{D.2}$$

By using assumption (i), we can write

$$P(\{V_n\}|\{Z_n\}) = \prod_{j=0}^{n} P(V_j|Z_j). \tag{D.3}$$

We can use Bayes theorem on $P(c_i|\{V_n\})$ to get

$$P(c_i|\{V_n\}) = \frac{P(\{V_n\}|c_i)P(c_i)}{P(\{V_n\})}. \tag{D.4}$$

By using assumption (iv) we can rewrite $P(\{V_n\}|c_i)$ to

$$P(\{V_n\}|c_i) = \prod_{j=1}^{n} P(V_j|V_{j-1},c_i)P(V_0|c_i). \tag{D.5}$$

Finally, we can write $P(c_i|\{Z_n\})$ as

$$\begin{aligned} P(c_i|\{Z_n\}) = \int \cdots \int & \frac{\prod_{j=1}^{n} P(V_j|V_{j-1},c_i)P(V_0|c_i)P(c_i)}{P(\{V_n\})} \\ & \times \prod_{j=0}^{n} P(V_j|Z_j)dV_n \cdots dV_0, \end{aligned} \tag{D.6}$$

and by using the assumption (v), we get

$$P(\{V_n\}) = \prod_{j=1}^{n} P(V_j|V_{j-1})P(V_0), \tag{D.7}$$

and

$$\begin{aligned} P(c_i|\{Z_n\}) = \int \cdots \int & \frac{\prod_{j=1}^{n} P(V_j|V_{j-1}, c_i)P(V_0|c_i)P(c_i)}{\prod_{j=1}^{n} P(V_j|V_{j-1})P(V_0)} \\ & \times \prod_{j=0}^{n} P(V_j|Z_j) dV_n \cdots dV_0. \end{aligned} \tag{D.8}$$

As an example, assuming that we have two measurements, we show how the calculation is done in a recursive way. When the first measurement arrives, we calculate

$$P(c_i, V_0|Z_0) = \frac{P(V_0|c_i)}{P(V_0)} P(V_0|Z_0)P(c_i), \tag{D.9}$$

and we then marginalize V_0 to get the probability for the class, i.e.,

$$P(c_i|Z_0) = \int P(c_i, V_0|Z_0) dV_0. \tag{D.10}$$

Using the second measurement, we can calculate the probability for c_i, where we use the probability distribution from (D.9)

$$\begin{aligned} P(c_i, V_1|\{Z_1\}) = \int & \frac{P(V_1|V_0, c_i)P(c_i, V_0|Z_0)}{P(V_1|V_0)P(V_0)} \\ & \times P(V_1|Z_1) dV_0, \end{aligned} \tag{D.11}$$

and we can now marginalize V_1:

$$P(c_i|\{Z_1\}) = \int P(c_i, V_1|\{Z_1\}) dV_1. \tag{D.12}$$

In general, when we receive a measurement, we calculate

$$\begin{aligned} P(c_i, V_n|\{Z_n\}) = \int & \frac{P(V_n|V_{n-1}, c_i)P(c_i, V_{n-1}|\{Z_{n-1}\})}{P(V_n|V_{n-1})P(V_{n-1})} \\ & \times P(V_n|Z_n) dV_{n-1}, \end{aligned} \tag{D.13}$$

where $P(c_i, V_{n-1}|\{Z_{n-1}\})$ is the probability distribution from the previous update of the probability. In the next step, we marginalize the speed to get the probability for the classes i.e.,

$$P(c_i|\{Z_n\}) = \int P(c_i, V_n|\{Z_n\}) dV_n, \tag{D.14}$$

where $P(V_n|V_{n-1}, c_i)$ can be found from

$$P(V_n|V_{n-1}, c_i) = \frac{P(V_n, V_{n-1}|c_i)}{P(V_{n-1}|c_i)}, \tag{D.15}$$

where $P(V_{n-1}|c_i) = \int P(V_n, V_{n-1}|c_i)dV_n$ and $P(V_n|V_{n-1})$ can be found as

$$P(V_n|V_{n-1}) = \frac{P(V_n, V_{n-1})}{\int P(V_n, V_{n-1})dV_n}, \tag{D.16}$$

and where $P(V_n, V_{n-1}) = \sum_i P(V_n, V_{n-1}|c_i)P(c_i)$.

In the next section, we will explain how we deduce the speed feature from the measurement and estimate the uncertainty in the speed from the uncertainty in the position.

2.2 Modelling the uncertainty in the feature

The measurement for this work is from a 2D rotating surveillance radar and consists of the range and azimuth to the target, the uncertainty of the position and a time stamp. The range and azimuth is mapped internally in the radar to latitude and longitude, which can be mapped to a Cartesian coordinate system [17], see Fig. D.3. In a similar manner we can map the uncertainty to Cartesian coordinates. The uncertainty in the measurements primarily depends on the range and azimuth resolution together with the size of the target, the orientation of the target, the adaptive preprocessing and thermal noise. The position of the target is estimated as the center of mass from the backscatter, see Fig. D.3, which can be calculated as

$$\boldsymbol{\mu} = \frac{1}{\sum_i e(\hat{\mathbf{r}}_i)} \sum_i \hat{\mathbf{r}}_i e(\hat{\mathbf{r}}_i), \tag{D.17}$$

where $\boldsymbol{\mu} = [\mu_r, \mu_\phi]^T$ is the center of mass for range r and azimuth ϕ respectively, $\hat{\mathbf{r}}_i$ is defined as the ith range-azimuth cell. Further $e(\hat{\mathbf{r}}_i)$ is the backscatter intensity in the radar cell. The uncertainty is the weighted variance of the backscatter. The uncertainty is given by the covariance matrix. The variance for range and azimuth can be found as a weighted sum of the backscatter, that is

$$\boldsymbol{\sigma}^2 = \frac{1}{\sum_i e(\hat{\mathbf{r}}_i)} \sum_i \hat{\mathbf{r}}_i^2 e(\hat{\mathbf{r}}_i) - \boldsymbol{\mu}^2 \tag{D.18}$$

and the covariance of range and azimuth can be found as

$$\sigma_{r\phi}^2 = \frac{1}{\sum_i e(\hat{\mathbf{r}}_i)} \sum_i r_i \phi_i e(\hat{\mathbf{r}}_i) - \mu_r \mu_\phi. \tag{D.19}$$

The diagonal of the covariance matrix is given by the vector (D.18) and the off-diagonal by the scalar (D.19)

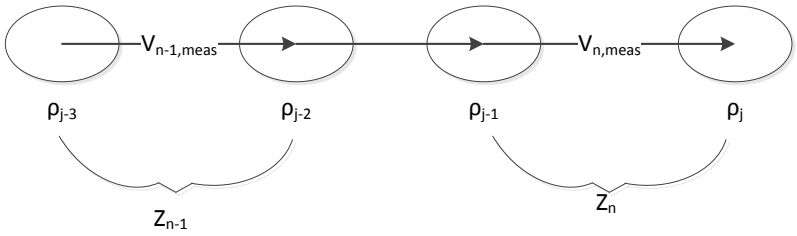


Fig. D.4: A figure showing how the measurements (plots) are used from the radar. Here $\rho_{j-3} \cdots \rho_j$ is the plots from the radar.

We let $\mathbf{r}_j$ be the Cartesian coordinate converted from the range azimuth i.e. $\mathbf{r}_j = [x_j, y_j]^T$. We let $\mathbf{\Sigma}_j^p$ be the uncertainty converted from range azimuth to Cartesian coordinates, t_j is the time stamp, and j is the measurement number. These three variables together are called a plot and are denoted as $\rho_j = \{r_j, \Sigma_j^p, t_j\}$. To estimate the speed we use a two point estimate. To satisfy the assumption $P(\xi_i|\xi_j) = P(\xi_i), \forall j, i, i \neq j$ used to get (D.3) we define Z_n as shown in Fig. D.4, that is $Z_n = \{\rho_j, \rho_{j-1}\}$, where $j = 2n$. The two point estimate of the velocity is calculated as

$$\mathbf{V}_{n,meas} = \frac{\mathbf{r}_{2n} - \mathbf{r}_{2n-1}}{\Delta t_n}, \tag{D.20}$$

where $\Delta t_n = t_{2n} - t_{2n-1}$. By assuming that the uncertainty in the position is independent from measurement to measurement, the uncertainty of the velocity can be represented by the two-dimensional covariance matrix

$$\mathbf{\Sigma}_n^V = \frac{\mathbf{\Sigma}_{2n}^p + \mathbf{\Sigma}_{2n-1}^p}{(\Delta t_n)^2} \tag{D.21}$$

We can calculate the speed of the target as $V_{n,meas} = \|\mathbf{V}_{n,meas}\|_2$. To calculate the uncertainty (standard deviation) of the speed $V_{n,meas}$ we average the minor and major axes of the ellipsoid representing the covariance matrix $\mathbf{\Sigma}_n^V$, given by the average of the eigenvalues of $\mathbf{\Sigma}_n^V$. As the $\mathbf{\Sigma}_n^V$ is a 2x2 matrix the average of the eigenvalues can be calculated as the average of the trace of $\mathbf{\Sigma}_n^V$. Thus, the uncertainty of the speed is

$$(\sigma_n^V)^2 = \frac{Tr(\mathbf{\Sigma}_n^V)}{2}, \tag{D.22}$$

which for this case just is the average of the diagonal in the matrix.

We will now shown that Σ_n^p is approximately Gaussian distribution. In order to do that we use real radar position measurements of sea buoys. We use sea buoys since they are stationary targets, and we are therefore able to

get a good estimate of the true position by simply taking the average of all the measurements. With knowledge of the true position, we can calculate the Mahalanobis distance [18] between each measurement and the true position of the target. The distribution of these Mahalanobis distances of the stationary target, will follow a chi-squared distribution with two degrees of freedom if the uncertainty of the position is Gaussian distributed [19]. We have used 10 radar tracks from different sea buoys and calculated the true position by averaging over around 600 measurements for each sea buoy. The probability mass function is obtained empirically by using the normalized frequency count (histogram) and is illustrated in Fig. D.5. Also shown in Fig. D.5 is the exact chi-squared distribution with two degrees of freedom. As it can been seen the match is relatively good for these radar measurements.

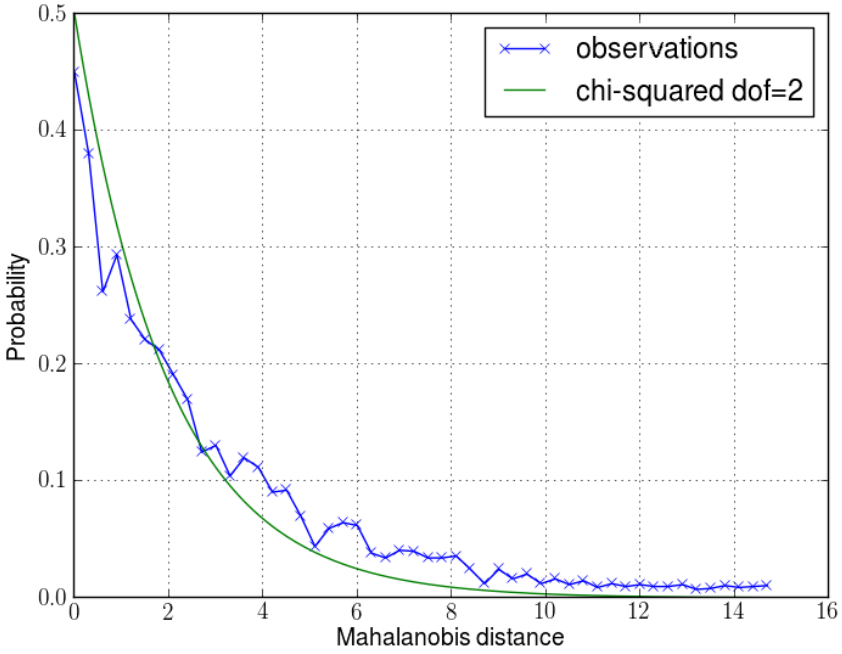


Fig. D.5: The distribution of the Mahalanobis distance for 10 different sea buoys. Both the true chi-squared and the measured distribution is shown

The Rice distribution is the probability distribution of the magnitude of a bivariate normal random variable, which fits well with our work. We therefore propose to use a Rice distribution [20] to model the speed distribution. The Rice distribution asymptotically approach a Rayleigh distribution for low speeds i.e. $V_{n,meas} \rightarrow 0$, and a Gaussian distribution for large speeds. The Rice

distribution is defined as:

$$P(V_n|Z_n) = \frac{V_{n,meas}}{\sigma_n^V} \exp\left(\frac{-(V_n^2 + V_{n,meas}^2)}{2(\sigma_n^V)^2}\right) I_0\left(\frac{V_n V_{n,meas}}{(\sigma_n^V)^2}\right), \tag{D.23}$$

where I_0 is the modified Bessel function of the first kind [20]. For the case without uncertainty, $P(V_n|Z_n)$ is describes by delta function centred at V_n.

In the next section, we will describe how to include additional features. These feature have negligible amount of the uncertainty and will therefore be used as the ground truth.

3 Multiple features

As it was shown in [16] the kinematic features $V_n, V_{n-1}, \Delta t$ are not solely capable of separating the different classes. Therefore we introduce the same features as used in [1, 2], specifically whether a given target is observed over land or sea as well as whether the target is observed near a road or not. In addition to the position features, we also propose to use a feature related to line-of-sight between the radar and the target and a static target feature derived from an average position of the target. The radar specific features are delivered from the radar sensor and the geographic features are derived from the position of the measurement.

We denote by $x_n \triangleq \{l_n, \Gamma_n, I_n^N, I_n^M, H_n, \rho_n\}$ the set additional features, where $l_n \in \{0, 1\}$ indicates sea/land respectively, $\Gamma_n \in \{0, 1\}$ indicates no road/road, $I_n^N \in \{0, 1, \ldots, 255\}$ indicates the normal intensity, $I_n^M \in \{0, 1, \ldots, 255\}$ describes the moving target indication intensity, $H_n \in \mathbf{R}_0^+$ is the line of sight minimum height, and $\rho_n \in \mathbf{R}_0^+$ is a indicator of stationarity. The features are described in further details below. To simplify the notation we denote the jth feature in the set x_n by $x_n(j)$ and the set $\{x_n(j)\}$ denotes the sequence of the jth feature from time $0, \ldots, n$. We assume that the features in x_n are mutually independent and $\{x_n\}$ satisfy the first-order Markov chain $x_n \leftrightarrow x_{n-1} \leftrightarrow x_{n-2}$. With this we can apply Bayes rule to obtain

$$P(c_i|\{x_n(j)\}) = \frac{P(x_n(j)|c_i, x_{n-1}(j))P(c_i|\{x_{n-1}(j)\})}{P(x_n(j)|x_{n-1}(j))}, \tag{D.24}$$

where

$$P(x_n(j)|c_i, x_{n-1}(j)) = \frac{P(x_n(j), x_{n-1}(j)|c_i)}{P(x_{n-1}(j)|c_i)}. \tag{D.25}$$

It follows

$$P(c_i|\{x_n\}) = \prod_j P(c_i|\{x_n(j)\}), \forall n, i. \tag{D.26}$$

We can now modify (D.14) such that this equation includes the features x_n. We assume that $\{x_n\}$ is independent of $\{Z_n\}$, i.e. $\{x_n\} \perp\!\!\!\perp \{Z_n\}$, which allow us to write

$$P(c_i|\{Z_n\}, \{x_n\}) = P(c_i|\{Z_n\})P(c_i|\{x_n\}) \tag{D.27}$$

In the following sections, the individual features, $x_n(j)$ will be described

3.1 Surface features

The surface features are two binary feature, l_n and Γ_n, where l_n takes the value 1 when a target is observed on land and 0 when observed over sea. Similarly, the feature Γ_n is 1 for targets when observed near a known road, and 0 elsewhere.

3.2 Radar plot intensity

From the radar sensor we have radar specific features such as the plot intensity from normal radar processing I_n^N and moving target identification (MTI) processing I_n^M. With the radar for this work the intensity is the amplitude of the backscattered signal from the target and it is related to the radar cross section size and the distance to the target.

3.3 Line of sight

The line of sight feature is a feature based on the required minimum height for a given target to be within the radar line-of-sight. We use this estimate as the radar in this work does not have height information of the target i.e. as the radar is a 2D radar. We therefore try to estimate the minimum height the target must be in to be seen from the radars position. An illustration of the problem can be seen in Fig. D.6. Due to the curvature of the Earth, a target beyond the radar horizon (D) must have a minimum height above the terrain to be visible. For the ideal spherical Earth, the minimum height H_n, for the n'th update, can be estimated as

$$H_n = R_h - \sqrt{D^2 + R_e^2}, \tag{D.28}$$

for definitions see D.6. The radar used for this work is an X-band radar [12] and to compensate for standard atmospheric refraction the local radius of the Earth is multiplied with a factor of $4/3$ [12, p. 503]. In practice, a digital terrain elevation model [21] is used to evaluate the minimum height required at the target location for a clear line-of-sight to exist between the radar antenna and the target. We show an example in Fig. D.7 of a surface target obstructed by hilly terrain between the radar antenna and the target.

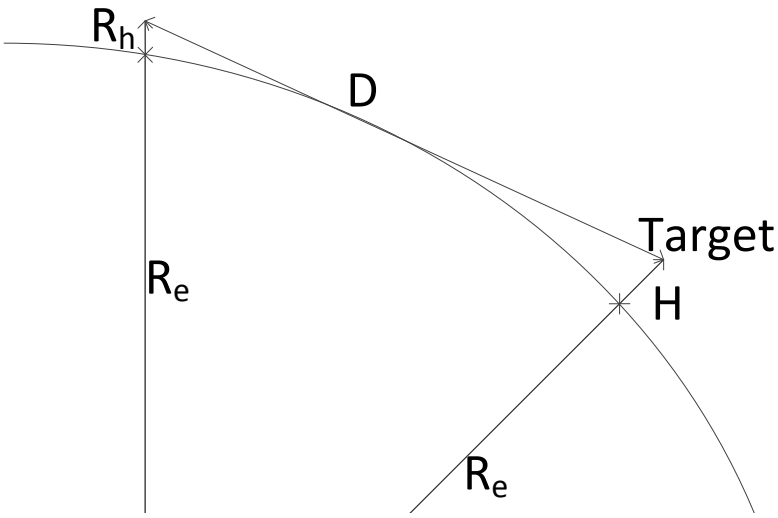


Fig. D.6: Schematic drawing of the earth. R_h is the radar height from the earth ellipsoid, R_e is the radius of the earth taking at the geographic location of the radar placement, D is the measured distance to the target and H is the relative height we want to estimate.

It is clear this minimum height must be used cautiously, as the framework does not support a full electromagnetic propagation model including e.g. multipath effects and ducting [12, pp. 502–518] [12, pp. 239–240].

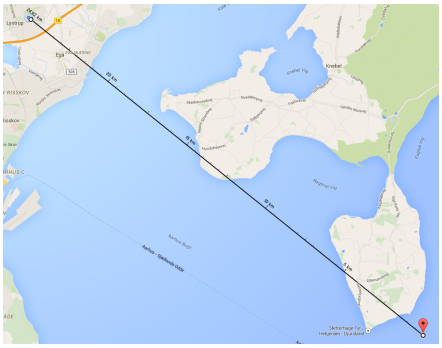
To use the estimated minimum height H_n, for the nth update, we define a set of predefined heights for each target class, $\hat{H}_i$, where the subscript i is the ith class. If the predefined maximum height $\hat{H}_i$ of the target is lower than the estimated minimum height for a clear line-of-sight, H_n the probability for that class will be multiplied with ϵ, where $\epsilon < 1$ [2].

$$P(c_i|H_n;\hat{H}_i) = \begin{cases} \epsilon, & \text{if } H_n > \hat{H}_i \\ 1.0, & \text{else.} \end{cases} \tag{D.29}$$

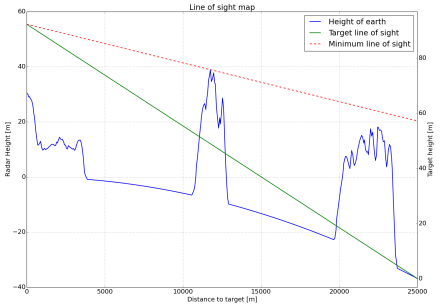
3.4 Position average

An average position feature is introduced to help in the separation of stationary and moving targets (such as wind turbines versus ships). Although stationary, the measurement uncertainty introduces a large fluctuation in the estimated speed of stationary targets. In Fig. D.8, it can be seen the estimated speed is up to 30 m/s for a wind turbine. It is therefore clear that speed alone cannot separate a stationary target from a moving target. We therefore introduce a feature, which is the difference between the last position update

[2] The reason for multiplying the class probability by ϵ rather than zero is due to that H_n and $\hat{H}_i$ are both estimates.



(a) Example of target, which is over sea and the radar, placed at land at a distance of about 25 km.



(b) Blue is height of the landscape with the Earth curvature. Green is the straight line from the radar to the target. The dash line is the minimums height of the target to be within line of sight

Fig. D.7: An example of line of sight. The distance from the radar to the target is approx. 25 km. The map overview can be seen in Fig. D.7a. As it can be seen, the target cannot be seen if it is surface target such as a RIB. From the Fig. D.7b, the target must be at least 57 meters above the sea level to be seen.

and the position median from the last N_s scans. A non-moving target will exhibit a small difference from the median compared to a moving object such as a helicopter. By using the median we are more immune to outliers[3]. We define the set of the previous N_s positions as $\{\mathbf{r}_{n-1}\}_{N_s} = \{\mathbf{r}_{n-1} \dots \mathbf{r}_{n-N_s-1}\}$. The median can then be calculated as

$$\mathbf{M}_p = \begin{bmatrix} \text{med}(\{x_{n-1}\}_{N_s'}) \\ \text{med}(\{y_{n-1}\}_{N_s'}) \end{bmatrix}. \tag{D.30}$$

[3]We define outliers as wrong plot/track association done internal in the radar.

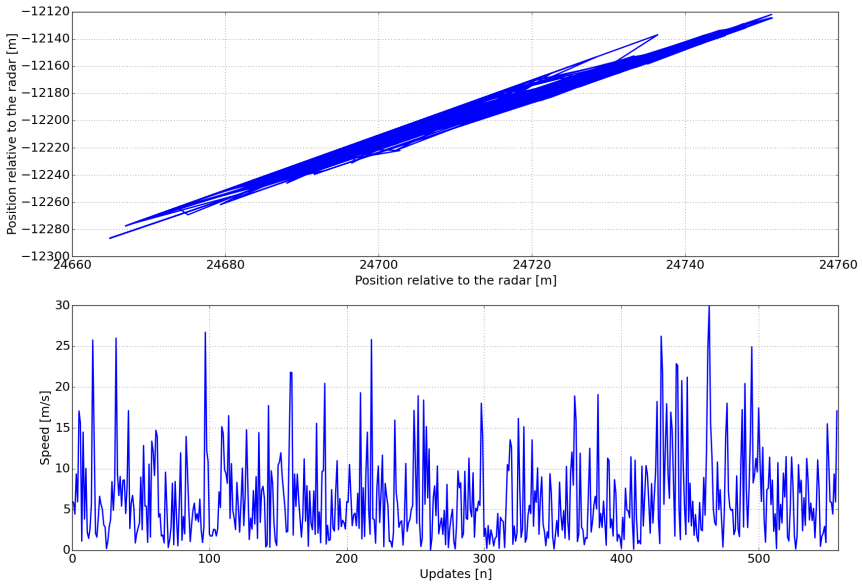


Fig. D.8: In this figure the speed and position for a wind turbines is shown.

We then calculate the magnitude of the difference from the new position $\mathbf{r}_n$ to $\hat{\mathbf{M}}_p$ as

$$\rho_n = \frac{\left|\mathbf{M}_p - \mathbf{r}_n\right|}{N_s'}, \tag{D.31}$$

where if $n \geq N_s'$ then $N_s' = N_s$ otherwise $N_s = n$, and N_s' is a normalization constant.

4 Implementation of the classifier

Our training data is derived from multiple different sensors such as automatic identification system (AIS), automatic dependent surveillance–broadcast (ADS-B), GPS and radars with different measurement accuracy and sampling frequencies. During training, the entire sequence of measurements is available. Therefore we can apply e.g. forward and backward Kalman filtering to obtain an improved estimate of the track positions relative to that obtained by e.g. a forward filtering only. This approach is known as fixed interval smoothing [22]. Fixed interval smoothing reduces the uncertainty in the training data and we therefore treat the data as if they have no uncertainty.

As explained in section 2 our recursive classifier at time n first computes (D.13) and then uses (D.13) in (D.14) to get the class probability. In order to compute (D.13), we therefore need to find $P(V_n|V_{n-1}, c_i)$, $P(c_i, V_{n-1}|\{Z_{n-1}\})$,

$P(V_n|V_{n-1})$,$P(V_{n-1})$ and $P(V_n|Z_n)$. We first note that $P(c_i, V_{n-1}|\{Z_{n-1}\})$ is known since it was found at time $n-1$. Moreover, $P(V_n|Z_n)$ is explicitly given by (D.23). In order to find the remaining terms, we will model the PDF $P(V_n, V_{n-1}|c_i)$ using a Gaussian mixture model (GMM), i.e.

$$P(V_n, V_{n-1}|c_i) = \sum_i \pi_i \mathcal{N}_i(V_n, V_{n-1}|c_i), \tag{D.32}$$

where π_i is the weight for the i'th mixture, $\sum_i \pi_i = 1$ and $\mathcal{N}$ denotes the normal distribution. We can then use Bayes rule to get the following PDF

$$P(V_n|V_{n-1}, c_i) = \frac{P(V_n, V_{n-1}|c_i)}{P(V_{n-1}|c_i)}. \tag{D.33}$$

$P(V_{n-1}|c_i)$, $P(V_n|V_{n-1})$ and $P(V_{n-1})$ are obtained from (D.32) by marginalization via analytical integration. Thus, we have all the terms required to calculate the class probability by using (D.14). It is important to note that in (D.13), the product $P(V_n, V_{n-1}|c_i)P(c_i, V_{n-1}|\{Z_{n-1}\})$ becomes increasingly complex to calculate as time increases. To see this, we note that both terms in the product are represented by GMMs. It follows that the product will also be a GMM with a number of mixtures equal to the product of the number of mixtures in each of the GMMs. Thus, in the next update step, this product is further multiplied by a GMM, and the resulting GMM contains an exponentially increasing number of mixtures. There are at least two approaches, which can be used to handle this. The first is to fit a new GMM, with a predefined number of mixtures, for each update of the probability. However, this is not a good solution as it is computationally expensive as it requires running Kmeans and the EM-algorithm [23] for each update. The second option, which we use, is to solve the equation numerically. This means that we change the integrals in (D.13) and (D.14) to sums and thereby numerically approximate the integrals. In this cases (D.13) is approximated by

$$P(c_i, V_n|\{Z_n\}) \approx \sum_{\{\hat{V}_{n-1}\}} \frac{P(V_n|\hat{V}_{n-1}, c_i)P(c_i, \hat{V}_{n-1}|\{Z_{n-1}\})}{P(V_n, \hat{V}_{n-1})} P(V_n|Z_n)\Delta\hat{V}_{n-1}, \tag{D.34}$$

and (D.14) is approximated by

$$P(c_i|\{Z_n\}) \approx \sum_{\{\hat{V}_n\}} P(c_i, \hat{V}_n|\{Z_n\})\Delta\hat{V}_{n-1}. \tag{D.35}$$

Furthermore $P(V_n|Z_n)$ is normalized such that

$$\sum_{\{\hat{V}_n\}} P(\hat{V}_n|Z_n)\Delta\hat{V}_n = 1. \tag{D.36}$$

The set $\{\hat{V}_n\}$ contains a number of values (speeds) which we sum over. We expect that $P(V_n|Z_n)$ has a smaller span in V_n then $P(V_n, V_{n-1}|c_i)$. It is therefore only necessary to have points where $P(V_n|Z_n) > 0$. We therefore choose to only define the set of $\{\hat{V}_n\}$ where $P(\hat{V}_n|Z_n) > 0$.

To reduce the computational load of each update of the probability, we can reduce the number of points in the sum by sampling closer near the mean of $P(V_n|Z_n)$ and then reduce sampling farther away from the mean. However, to ensure that stationary targets are classified correct, we also sample speed from zero and upwards if the mean of $P(V_n|Z_n)$ is below a threshold, in this case 5 m/s. We use trapezoidal integration [24] as this has shown an adequate accuracy for the approximations.

5 Simulation study

The data used in this work are recorded from real targets and have been acquired using automatic identification system (AIS), automatic dependent surveillance – broadcast (ADS-B), GPS loggers on cooperating targets and radar.

In this section we presents the result of the classifier when using the uncertainty, i.e. by setting $P(V_n|Z_n)$ to a delta function, and when using the uncertainty in which case $P(V_n|Z_n)$ is given by (D.23). We model for each class c_i $P(V_n, V_{n-1}|c_i)$ used in (D.16), by a Gaussian mixture model (GMM). The number of mixtures for the different classes is found by the method of Gmeans [25] and the GMM is trained with Kmeans and the EM-algorithm [23]. Further a boundary box is used in the GMM such that the probability from the GMM is set to zero if the input to the GMM is higher or lower then seen in the training. We also model $P(x_n, x_{n-1}|c_i)$ in (D.25) by GMMs. When the uncertainty is included the number of points, i.e. the size of the set $\{\hat{V}_n\}$ in (D.34), (D.35) and (D.36), is set to minimum 91, as this has shown good accuracy of the numerically integration while keeping the computational load acceptable. The size of the training database and test database is shown in Table D.1, and the number of mixtures for each class is shown in Table D.2. Moreover, we used the heights defined in Table D.3 for $\hat{H}_i$ in (D.29).

In Fig. D.9 the probability for a sea buoy is shown. In Fig. D.10 the probability is shown for the commercial aircraft circling the wind farms from the wind farm scenario shown in Fig. D.1. In the figure, the probability is for the classification without using the uncertainty in the plots and with using the uncertainty.

Table D.1: The size of the training and test database

Classes	Trainings size	Test size
large boat	23130	2462
stationary sea	13662	7492
RIB	23470	720
helicopter	19143	205
commercial aircraft	6725	8840
birds	3289	262

Table D.2: The number of mixtures in the GMM for $P(V_n, V_{n-1}|c_i)$

Class name	Number of mixtures
Large boat	11
Stationary sea	1
Small boat	13
Helicopter	9
Commercial aircraft	6
Birds	1

Table D.3: Predefined height of the different target classes

Target class	predefined height [m]
Large ship	35.0
Stationary sea	200.0
Commercial aircraft	10000.0
Birds	100.0
RIB	2.0
Helicopter	1000.0

6 Discussion

As described in the introduction the probability of the classification, must correspond to the confidence of the classification. In Fig. D.9 the classification for the sea buoy is shown. Here it can be seen that when using uncertainty the classifier is slower to achieve a high confidence that the target is a sea buoy then when the classifier does not utilize the uncertainty. It can also be seen that without the uncertainty the target is classified wrong the most of the time. If we look at Fig. D.10 the probability for the commercial aircraft is shown. When we utilize the uncertainty the classifier continues to believe that the target is a commercial aircraft. We also see that when we do not utilize the uncertainty in the classifier we have to restart the classifier as there are no

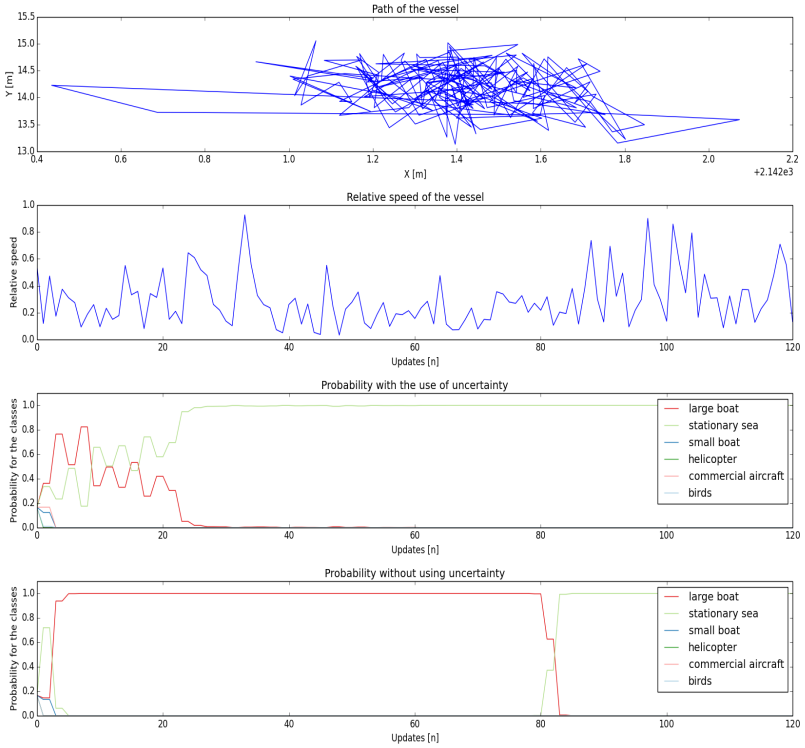


Fig. D.9: The probability curve for a sea buoy.

classes which matches the target. This can be seen around the updates 40, 45, 150, 200 and 240. This is because there is no target in the training database, which can both fly, as fast as around 100 m/s and as low as 40 m/s. From the results shown in figures D.9 and D.10 we show that the classifier which use the uncertainty is more robust and the probability matches better with the confidence of the classification results.

7 Conclusion

In this work we presented a Bayesian recursive classifier, which uses the uncertainty in position measurements to achieve a better match between the probability for the target class and the confidence of the classification. We showed that by exploiting the uncertainty in the position a more robust classification is made as more possible speeds are used in testing. To further improve the classification, we introduced additional features such as surface features (to indicate whether the target is on land or sea), radar plot inten-

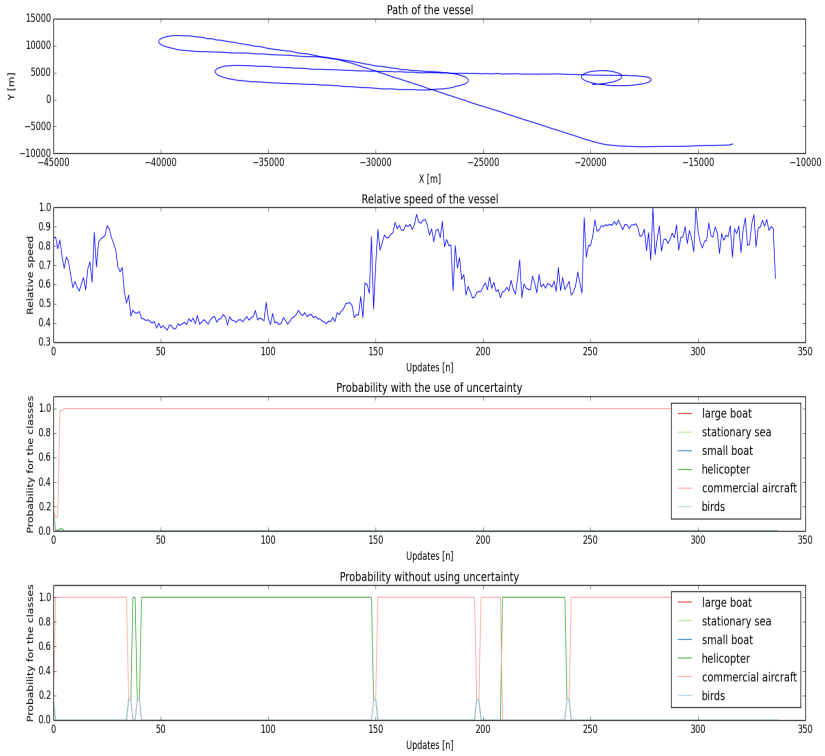


Fig. D.10: The probability development of the commercial aircraft from the Horns rev scenario

sity, line of sight (this is useful since the radar has no height information), and position averaging (to separate stationary targets from moving targets).

References

[1] L. W. Jochumsen, J. Ostergaard, S. H. Jensen, and M. O. Pedersen, "Modelling temporal variations by polynomial regression for classification of radar tracks," in *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European*. IEEE, 2014, pp. 1412–1416.

[2] L. W. Jochumsen, M. Pedersen, K. Hansen, S. H. Jensen, and J. Østergaard, "Recursive bayesian classification of surveillance radar tracks based on kinematic with temporal dynamics and static features," in *Radar Conference (Radar), 2014 International*. IEEE, 2014, pp. 1–6.

[3] J. R. Layne, "Automatic target recognition and tracking filter," DTIC Document, Tech. Rep., 1998.

[4] I. Bilik, J. Tabrikian, and A. Cohen, "GMM-based target classification for ground surveillance doppler radar," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 42, no. 1, pp. 267–278, 2006.

[5] F. Kruse, F. Fölster, M. Ahrholdt, H. Rohling, M.-M. Meinecke, and T.-B. To, "Target classification based on near-distance radar sensors," in *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, 2004, pp. 722–727.

[6] P. Tait, *Introduction to radar target recognition*. IET, 2005, vol. 18.

[7] M. J. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[8] Y.-H. Chao, W.-H. Tsai, and H.-M. Wang, "Discriminative feedback adaptation for GMM-UBM speaker verification," in *Chinese Spoken Language Processing, 2008. ISCSLP'08. 6th International Symposium on*. IEEE, 2008, pp. 1–4.

[9] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley, 2006.

[10] Y. Bar-Shalom, P. Willett, P. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, 2011.

[11] B. Ristic, N. Gordon, and A. Bessell, "On target classification using kinematic data," *Information Fusion*, vol. 5, no. 1, pp. 15–21, 2004.

[12] M. I. Skolnik, *Introduction to Radar Systems 3nd Edition*, 3rd ed. McGraw Hill Book Co., 2001.

[13] Terma. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/291389/sc6k_-_asia.pdf

[14] ——. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/211400/scanter_5000_series__w.pdf

[15] ——. [accessed 26 january 2016]. [Online]. Available: http://www.terma.com/media/211428/scanter_4100_-_naval_air___surface_surveillance_2d_radar.pdf

[16] L. W. Jochumsen, E. Nielsen, J. Østergaard, S. H. Jensen, and M. Ø. Pedersen, "Using position uncertainty in recursive automatic target classification of radar tracks." IIEEE, 2015.

[17] C. F. Karney, "Algorithms for geodesics," *Journal of Geodesy*, vol. 87, no. 1, pp. 43–55, 2013.

[18] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.

[19] G. R. Cooper and C. D. McGillem, *Probabilistic Methods of Signal and System Analysis*, 3rd ed. Oxford University Press Inc., 1999.

[20] K. D. Ward, R. J. A. Tough, and S. Watts, *Sea clutter: Scattering, the K Distribution and Radar Performance*. IET, 2006.

[21] A. Jarvis, H. I. Reuter, A. Nelson, and E. Guevara, "Hole-filled srtm for the globe version 4," *available from the CGIAR-CSI SRTM 90m Database (http://srtm. csi. cgiar. org)*, 2008.

[22] D. E. Catlin, "Fixed interval smoothing," in *Estimation, Control, and the Discrete Kalman Filter*. Springer, 1989, pp. 188–199.

[23] C. M. Bishop, *Pattern recognition and machine learning*. springer New York, 2006, vol. 1.

[24] R. L. Burden and J. D. Faires, *Numerical Analysis*, 9th ed., M. Julet, Ed. Cengage Learning, 2010.

[25] G. Hamerly and C. Elkan, "Learning the k in k-means," in *In Neural Information Processing Systems*. MIT Press, 2003, p. 2003.

Paper E

A Recursive kinematic Random forest and alpha beta filter classifier for 2D radar tracks

Lars W. Jochumsen, Jan Østergaard, Søren H. Jensen, Carmine Clement, Morten Ø. Pedersen

The paper has been published in the
EURASIP, Journal on Advances in Signal Processing, Submitted undergoing review.

©2016 EURASIP
The layout has been revised.

Abstract

In this work, we show that by using a recursive random forest together with an alpha beta filter classifier it is possible to classify radar tracks from the tracks' kinematic data. The kinematic data is from a 2D scanning radar without Doppler or height information. We use random forest as this classifier implicit handles the uncertainty in the position measurements. As stationary targets can have an apparently high speed because of the measurement uncertainty, we use an alpha beta filter classifier to classify stationary targets from moving targets. We show an overall classification rate from simulated data at 82.6 % and from real world data 79.7 %. Additional to the confusion matrix we also show recordings of real world data.

1 Introduction

The increasing demand for protection and surveillance of the coastal areas requires modern coastal surveillance radars. These radars are designed such that small objects can be detected. Therefore, there is an increasing amount of information for the radar observer. Moreover, the number of false and unwanted objects increases as the demand for seeing small objects makes the radar more sensitive. Generally, the false objects can be avoided by using a reliable tracker. However, the tracker does not exclude unwanted objects. The difference between false and unwanted objects are that false objects do not originate from true objects but are mainly noise objects, whereas the unwanted objects originate from true objects but are unwanted in the surveillance image. These objects depend on the purpose of the radar however, for coastal surveillance radars the unwanted objects are normally birds, wakes from large ships etc.

It has been shown in [1] that it is possible to classify tracks by using a recursive classifier where a Gaussian mixture model (GMM) is used to model the probability distribution function (PDF) of targets kinematic behavior. However the classifier does not handle the uncertainty in the measurements from the radar. In [2] the position uncertainty is used as an input to the classifier. The classifier also use a GMM to model the PDF of the kinematic behavior of the target. The problem with this is that it is very computationally expensive. To obtain an easier way to handle uncertainty, joint target tracking and classification can be used, as shown in [3–5]. The problem with joint target tracking and classification is that it is difficult to achieve a high degree of freedom in the filters to separate the classes. For example a car driving 130 km/h on highway is not likely to accelerate but more likely to decelerate. This is very hard to model with a tracking filter. A particle filter can be used but this is computationally expensive. In [6] the authors are describing a method to classify trucks and cars from GPS measurements. The classifier

consists of a support vector machine (SVM) and the features are primarily acceleration and deceleration. The classifier is non-recursive, which means that the complete length of the tracks is required. The measurements from a GPS device is generally more accurate than the position measurements then a radar. In [7] a decision tree is used for a recursive classification of four different target classes. The data are from a radar with height information. The decision tree has the advantage that it in some way implicitly handles the uncertainty. That is, features that do not separate the classes will not be used as much as features, separating the classes. The disadvantage is that the classifier has a high variance of the classification results. In [8] the random forest classifier is introduced. The random forest is a bagging classifier [9] where multiple decision trees are used to reduce the variance of the classification results. For this reason random forest is selected in this work.

In this work we introduce a classifier which uses position measurements to classify radar tracks from a 2D scanning radar. The classifier consists of a alpha beta filter [10] and a random forest classifier. The alpha beta filter is classifying stationary or moving and the random forest classifies the moving targets. The classify is recursive such that the classification results is being updated for each scan of the radar. The classifier performance is shown by using simulated track data and real world radar data.

In section 2.1 we will introduce the random forest classifier by describing the training of a decision tree and then explain how this tree is used in the random forest. In section 2.2 we will explain how we utilize the probability estimates from the random forest in a recursive framework. In section 2.3 we introduce a alpha beta filter classifier, which classifies targets as either stationary or moving. This is introduced because stationary targets can have high speeds because of they fluctuate in the position because of measurements uncertainty or the main scatter points is moving i.e. wind turbine. In section 2.4 we combine the random forest and the alpha beta filter to our proposed classifier. In 2.5 we describe, which features we use in the random forest. The simulation study is shown in section 3 and in section 4 the real world results are shown. We discuss the results in section 5 and conclude the work in section 6.

2 Method

When using a random forest, a feature vector is needed. We define our feature vector as a set of kinematic and geographic features. The feature vector is derived from the radar position measurements. We define this set of position measurements as

$$\{Z_n\}_k = \{Z_n \cdots Z_{n-k}\}, \tag{E.1}$$

where $Z_n = [x_n, y_n]^T$, x and y is the position in a Cartesian coordinate system with the origin at the location of the radar, n is the measurement number index and k is the set size.

2.1 Random forest

In this section, we introduce the random forest classifier [8]. Random forest is a bagging algorithm, which means that the random forest consists of a number of weak classifiers [11], which has zero bias, but high variance of the true value. The weak classifiers are decision tress. We start this section by describing how to grow a decision tree and then move on to the random forest.

A decision tree consists of a number of nodes e.g. $(N_1 \cdots N_3)$ and a number of leafs e.g. $(N_4 \cdots N_7)$. This is shown in Fig. E.1. A node is defined by more then one class existing in the node data, whereas a leaf only have one class. In every node a decision must be made such that we either go left or right in the tree. The decision must always be true or false. A leaf is defined as a node where all of the data in the node only consists of one class therefore no more splits are required.

To train the tree we start with a feature vector F of size $N_s \times D$ where N_s is the number of samples and D is the number of features i.e. dimensions in the feature vector. We now want to split the data such that we make the best separation of the classes by choosing the best feature and feature value. To do this we need to find the best feature to split and the best value to split at. To explain the algorithm we assume that there are only 2 classes so it forms a binary classification problem and that the values of the feature belong to a finite sample space. This is done to make the explanation easier.

We start by assuming that a split already has been made and we want to evaluate how good the split is. For this, we use a normalized the entropy measure to do that [11]. An alternative to the normalized entropy is the more common Gini index [12] however, for this work the normalized entropy as shown better results. We define the set of samples in the parent node as s_1 and the number of samples in the set as $|s_1|$. Similarly we define the set of samples in the children as s_2 and s_3 and the number of samples as $|s_2|$ and $|s_3|$. Further we index the samples belonging to class ℓ by the superscript ℓ such as s_1^ℓ, where $\ell \in \{1, 2\}$. We can calculate the empirical entropy for the children as

$$H(s_i) = -P(s_i^1)\log_2(P(s_i^1)) - P(s_i^2)\log_2(P(s_i^2)), i = \{2, 3\}, \tag{E.2}$$

where $P(s_i^1) = |s_i^1|/|s_i|$ and $P(s_i^2) = |s_i^2|/|s_i|$. It follows that $P(s_i^1) = 1 - P(s_i^2)$. As the entropy does not take into account how many samples there

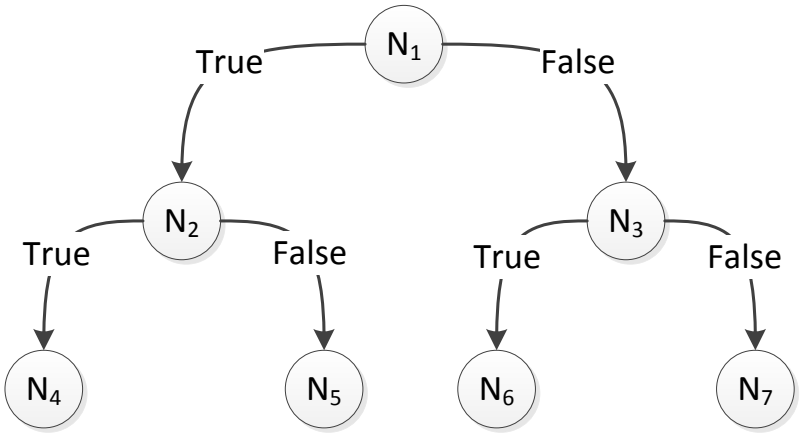


Fig. E.1: An example of a decision tree where N_1 to N_3 is nodes where a decision must be made. An example could be is the ball blue (True or False)

are in each child we normalize the entropy as

$$\hat{H}(s_i) = \frac{|s_i|}{|s_1|} H(s_i), i = \{2, 3\}. \tag{E.3}$$

We can now calculate the information gain from the split as

$$\tilde{H} = H(s_1) - (\hat{H}(s_2) + \hat{H}(s_3)). \tag{E.4}$$

From (E.4) we now have a measure for how good a split is, and now able to optimize each split of the data such that we choose the best feature to split on and the best value of the feature. This is continued until all of the training data is classified correct i.e. all end nodes are leafs. typically for a decision tree, it is necessary to prune the tree back to avoid over fitting. However, an advantage of using random forest is that it is not necessary to prune the decision trees. The random forest is a bagging classifier [9]. This means that the random forest consist of a number of trees N_t where each tree is trained with a random part of the samples and a random part of the features. That is, we draw a random subset of the training data and select a random subset of the features. We then train each tree with these random subsets and we assume that the trees are statically independent of each other. In general the random forest is not a probabilistic classifier. However, by counting the number of trees for each class and normalizing with the total number of trees a empirical probability can be achieved.

$$\hat{P}(c_i|\{Z_n\}_k) = \psi_i / N_t, \tag{E.5}$$

where ψ_i is the number of trees in the random forest which ended up in a leaf for class i.

In the next section we explain how we (E.5) obtained from the random forest to achieve a recursive update of the probability for the class given all the measurements.

2.2 Recursive update of the random forest probability

The empirical probabilities obtained from the random forest classifier are obtained as the fraction of the number of trees which predicts c_i divided with the total number of trees. By this definition, the resolution of the probability estimates is given by the number of trees in the random forest. To prevent that a class is assigned a zero probability, we modify it in the following way:

$$P(c_i|\{Z_n\}_k) = \hat{P}(c_i|\{Z_n\}_k)(1 - 2/N_t) + 1/N_t, \tag{E.6}$$

such that a probability never reaches zero and is restricted below 1.

Based upon the above, we have the probability for the class given the current set of features $P(c_i|\{Z_n\}_k)$. However we want the probability given all measurements, that is $P(c_i|\{Z_n\})$, where $\{Z_n\} = \{Z_n\}_n$. We have, however, not been able to find a simple way to recursively update $P(c_i|\{Z_n\})$ based on the previous $P(c_i|\{Z_{n-1}\})$ and which works for all n. Instead we propose the following recursive function $f(c_i|\{Z_n\})$, which is everywhere non-negative and sum to one. Thus, $f(c_i|\{Z_n\})$ can be considered to be a probability mass function (PMF), which we will use as an approximation for the true $P(c_i|\{Z_n\})$. In particular, we define:

$$f(\{Z_n\}_k, c_i) \triangleq \frac{P(c_i|\{Z_n\}_k)^w}{\phi_n} f(\{Z_{n-1}\}_k, c_i), \tag{E.7}$$

where w is a weighting factor, $P(c_i|\{Z_n\}_k)$ is given by (E.6) and where ϕ_n is the normalization constant such that $\sum_{c_i} f(\{Z_n\}_k = 1$. The introduction of the weighting by w is inspired by the weighted Bayesian classifier used in [13]. In particular, we choose $w = 1/k$ since the features of the random forest is given by a set of measurements where only one out of k measurements is substituted at each update.

In the next section we describe our alpha beta tracking filter. This filter is used to classify if a target is non moving or moving. The reason for applying such a filter is to classify stationary targets, which have high apparent speed due to measurement uncertainties.

2.3 Alpha beta filter

The alpha beta filter is a simple tracking filter [14]. By using the alpha beta filter, we assume that we can describe the target movements with a first order Markov chain. We have the state vector $X_n = [\hat{x}, \hat{y}]^T$ and the measurement

z_n. The alpha beta filter is trying to predict z_n given the speed v_{n-1} at time $n-1$ and the state X_{n-1} as

$$X_n = X_{n-1} + \tau v_{n-1}, \tag{E.8}$$

where τ is the time between z_{n-1} and z_n. Assume the speed is constant between n and $n-1$ that is $v_n = v_{n-1}$. The error can be calculated as

$$r_n = z_n - X_n, \tag{E.9}$$

with the residual we update the estimate of the v_n and X_n as

$$\begin{aligned} X_n &:= X_n + \alpha r_n \\ v_n &:= v_n + \frac{\beta}{\tau} r_n, \end{aligned} \tag{E.10}$$

where α and β are the constants in the alpha beta filter. To calculate the probability for the class c_i we use a multivariate normal distribution

$$\begin{aligned} P_{\alpha\beta}(z_n|c_i) = &\frac{1}{2\pi\sqrt{|\Sigma_n|}} \exp\left(-\frac{1}{2}(X_n - z_n)^T \Sigma_n^{-1}(X_n - z_n)\right) \\ &\frac{1}{2\pi\sqrt{|\Sigma_n|}} \exp\left(-\frac{1}{2} r_n^T \Sigma_n^{-1} r_n\right), \end{aligned} \tag{E.11}$$

where Σ_n is the covariance of the position, and the subscript $\alpha\beta$ is to emphasize that this is the probability for the alpha beta filter. The purpose of the alpha beta filter is to separate non moving targets i.e. stationary targets from moving targets. We therefore define two filters: a stationary filter $P_{\alpha\beta}(c_s|z_n)$ which have the parameters $\alpha = 0.1$ and $\beta = 0.0$, which allows the position part of the state to move slightly but force the speed to be constant at zero. The possibility for a slight movement of the state is because of the possibility for false starting measurements. We define the moving alpha beta filter as $P_{\alpha\beta}(c_m|z_n)$ with the parameters $\alpha = 1.0$ and $\beta = 1.0$ i.e. we hold the speed constant from update to update but allow both the movement and the speed to change with the measured change. For this work we want the alpha beta filter to classify if the target is stationary or non-stationary, we therefore recursively update the probability of the alpha beta filter.

$$P_{\alpha\beta}(c_i|\{Z_n\}) = \frac{P_{\alpha\beta}(Z_n|c_i, \{Z_{n-1}\}) P_{\alpha\beta}(c_i|\{Z_{n-1}\})}{P_{\alpha\beta}(Z_n|\{Z_{n-1}\})}. \tag{E.12}$$

To reduce the computational complexity we assume that the positions are controlled by a first order Markov chain i.e. $Z_n \leftrightarrow Z_{n-1} \leftrightarrow \{Z_{n-2}\}, \forall n$.[1]

$$P_{\alpha\beta}(c_i|\{Z_n\}) = \frac{P_{\alpha\beta}(Z_n|c_i, Z_{n-1}) P_{\alpha\beta}(c_i|\{Z_{n-1}\})}{P_{\alpha\beta}(Z_n|Z_{n-1})}, \tag{E.13}$$

[1] We denote the Markov chain by $a \leftrightarrow b \leftrightarrow c$, such that a is statistically independent of c if we known b.

In the next section we describe how we combine the random forest classifiers and the alpha beta filter classifier such that a classifier, which is a combination of the two classifier are created.

2.4 Combining the alpha beta filter with random forest

In our work we let the alpha beta filter classify if the target is stationary or non-stationary i.e. the alpha beta filter has two classes. The random forest has a stationary class and multiple non-stationary classes. We define for the random forest c_0 to be the stationary class and $c_{1\dots C}$ to be the moving classes, where C is the total number of classes. For the alpha beta filter we have the two classes as c_s and c_m for stationary and non-stationary classes respectively. We include the alpha beta filter as described in (E.14), (E.15).

$$\hat{P}(c_0|\{Z_n\}) = f(\{Z_n\}_k, c_0) P_{\alpha\beta}(c_s|\{z_n\}), \tag{E.14}$$

$$\hat{P}(c_i|\{Z_n\}) = f(\{Z_n\}_k, c_i) P_{\alpha\beta}(c_m|\{z_n\}), i = 1 \cdots C \tag{E.15}$$

We then normalize $\hat{P}(c_i|\{Z_n\})$ as

$$P_c(c_i|\{Z_n\}) = \frac{\hat{P}(c_i|\{Z_n\})}{\hat{\omega}}, \tag{E.16}$$

where $\hat{\omega}$, is a constant such that $\sum_i P_c(c_i|\{Z_n\}) = 1$. By including the alpha beta filter in this manner, we ensure that the alpha beta filter, classifies if a target is stationary or moving and let the random forest classify what type of moving target it is.

In the next section we will describe the features we use for the random forest feature vector, we will also describe how these are derived from the position. We only utilize position dependent features such as speed, acceleration etc.

2.5 Features

For the feature vector, we draw inspiration from [15] for some of the features. In this work, we set the number of position measurements k in (E.1) to 10. The number of measurements used in the feature vector is a compromise between the time it takes to get the number of measurements required for a full feature vector and the amount of information contained in the feature vector. Larger k requires more measurements i.e. more time before a classification results is made whereas for smaller k the first classification result comes earlier albeit with a greeter uncertainty due to the smaller amount of available information. The features and their descriptions can be seen in Table E.1. The set of measurements, $\{Z_n\}_k$, is a sub-set of the total set of measurements containing the latest k measurements. The index i is used to represent the i'th

element in the vector of measurements with length k with $0 \leq i < k$. Likewise we define the set of time stamps of the measurements as $\{t_n\}_k$ with the individual measurement being observed at time t_i. We start by calculating the vectorial distance between the measurements as:

$$\delta_i = z_i - z_{i-1}, \tag{E.17}$$

with the scalar distance given by

$$\Delta_i = |z_i - z_{i-1}|, \tag{E.18}$$

and the time difference between the measurements as

$$\tau_i = t_i - t_{i-1}. \tag{E.19}$$

The 2-point velocity estimate is

$$v_i = \frac{\Delta_i}{\tau_i}, \tag{E.20}$$

for $1 \leq i < k$ and the 3-point acceleration estimate is

$$a_i = \frac{2(v_{i+1} - v_i)}{\tau_{i+1} + \tau_{i-1}}, \tag{E.21}$$

for $1 \leq i < (k-1)$. The normal acceleration $a_i^{\perp}$ is given by the product of the speed and angular velocity

$$a_i^{\perp} = \left(\frac{v_{i+1} + v_i}{2}\right)\left(\frac{2}{t_{i+1} - t_{i-1}}\right)\cos^{-1}\left(\frac{\delta_{i+1} \cdot \delta_i}{\Delta_{i+1}\Delta_i}\right). \tag{E.22}$$

The feature d_i is the shortest distance from the $i'th$ measurement to the nearest coastline where we use the SWBD database from [16] to determine if a given location is over land or sea. Because of errors in the database a hard threshold cannot be used for land and sea. We therefore assign a maximum distance ξ to the coastline from the target. If the target is farther away then ξ we assign ξ to the distance. the sign of the distance decide if it is over land or sea. We set $\xi = 700$ meters to accomodate for errors in the SWBD database.

In the next section we will show some simulation results of the classifier. We will also show some real world results of the classifier.

3 Simulation study

We start by showing the performance of the algorithm versus the number of measurements k which the extracted features is from. The size of the feature vector change by k and the table shown in Table E.1 for $k = 10$.

Table E.1: The feature vector used. The number of measurement has been chosen to be $k = 10$

Feature	Feature description
$\mathrm{std}(\Delta_i)$	Empirical standard deviation of sample-to-sample distances
v_1 $\vdots$ v_k	2-point speed estimate
$\mathrm{mean}(v_i)$	Empirical mean of the speed
$\mathrm{std}(v_i)$	Empiricial standard deviation of the speed
a_1 $\vdots$ a_{k-1}	2-point acceleration estimate
$\mathrm{mean}(a_i)$	Empirical mean of the acceleration
$\mathrm{std}(a_i)$	Empirical standard deviation of the acceleration
$\mathrm{mean}(a_i^{\perp})$	Empirical mean of the normal acceleration
$\mathrm{std}(a_i^{\perp})$	Empirical standard deviation of the normal acceleration
$\lvert z_k - z_0 \rvert$	Total distance moved
d_0 $\vdots$ d_k	Distance to coastline
$\mathrm{mean}(d_i)$	Empirical mean of the distance to coast line

The data we use are simulated data from a controlled random walk. The controlled random walk consist of a three state transition matrix which has a deceleration, steady state and acceleration state. Parameters for maximum and minimum speed are incorporated which changes the probability in the transition matrix if the speed is not within the boundary of the permitted speed range. The data for different targets are generated such that they have nearly the same support in speed and the main difference is the acceleration support. The random walk creates position p_m^x and p_m^y which are extrapolated from some smooths speeds $\hat{V}_m^x$ and $\hat{V}_m^y$ described later.

$$p_m^x = p_{m-1}^x + \Delta t \hat{V}_m^x + \Sigma_m^x \tag{E.23}$$
$$p_m^y = p_{m-1}^y + \Delta t \hat{V}_m^y + \Sigma_m^y, \tag{E.24}$$

where Δt is the time between the updates for m and $m-1$ and Σ_m^x and Σ_m^y are position uncertainty drawn from a distribution.

$$\begin{bmatrix} \Sigma_m^x \\ \Sigma_m^y \end{bmatrix} \sim \mathcal{N}(0, \Sigma_e), \tag{E.25}$$

where Σ_e is the position covariance and $\mathcal{N}$ denotes the normal distribution. The smooth speeds are speeds V_m^x and V_m^y which are convolved with a moving average filter h:

$$\hat{V}_m^x = h * V_m^x \tag{E.26}$$
$$\hat{V}_m^y = h * V_m^y. \tag{E.27}$$

The speeds are extrapolated from accelerations $A_j^x(k)$ and $A_j^y(k)$, where j denotes the depending upon the state j. The speeds are given as

$$\begin{bmatrix} V_m^x \\ V_m^y \end{bmatrix} = \begin{bmatrix} V_{m-1}^x + \Delta t O_j^x(k) \\ V_{m-1}^y + \Delta t O_j^y(k), \end{bmatrix} \tag{E.28}$$

where

$$O_j^x(k) \sim \mathcal{N}(\mu_{x,j}, \sigma_{x,j}^2) \tag{E.29}$$
$$O_j^y(k) \sim \mathcal{N}(\mu_{y,j}, \sigma_{y,j}^y) \tag{E.30}$$

The parameters for the normal distribution $\mu_{x,j}$,$\mu_{y,j}$,$\sigma_{x,j}^2$ and $\sigma_{y,j}^2$ are given from the function $\phi_j(V(k-1), c_i)$ defined as:

$$\phi_j(V(k-1), c_i) = \begin{cases} \psi_j(1) & \text{if } V_{m-1} > \zeta^{max}, \\ \psi_j(2) & \text{if } \zeta^{min} \leq V_{m-1} \leq \zeta^{max}, \\ \psi_j(3) & \text{else,} \end{cases} \tag{E.31}$$

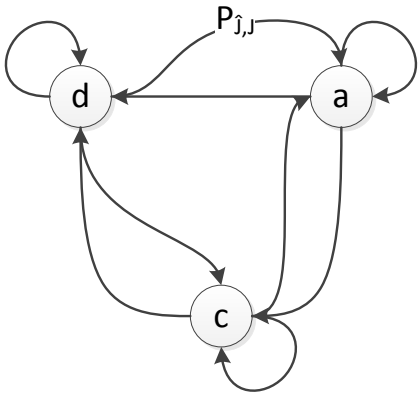


Fig. E.2: The state machine used for the data generation of the simulated data. The state machine has three states, an accelerating a, decelerating d and a constant speed c state. The probability for jumping between the states is controlled with $P_{\hat{j},j}$ which is change depending on the speed.

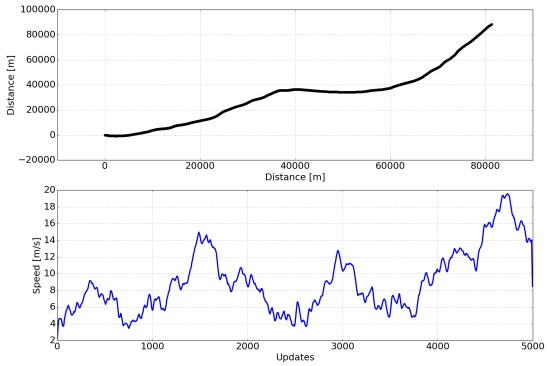
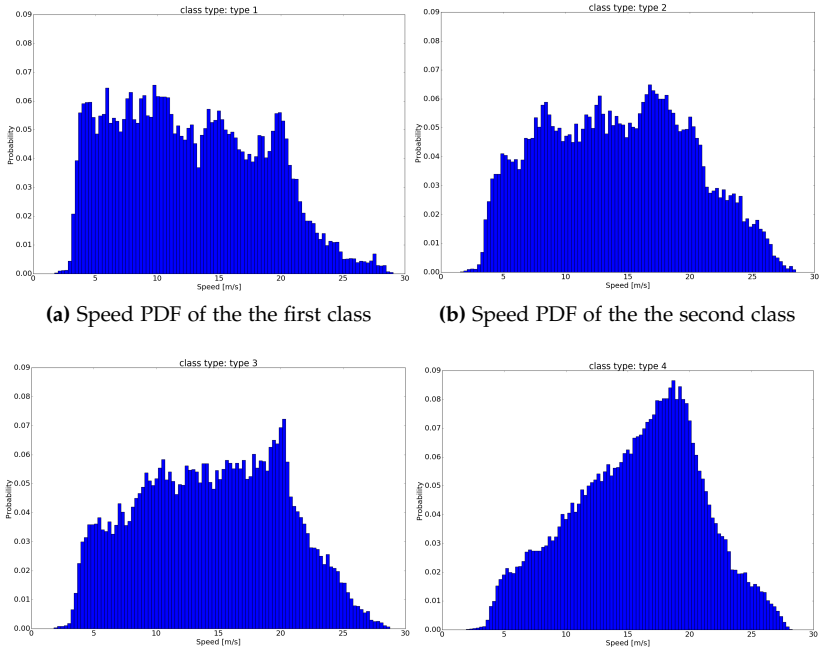


Fig. E.3: An example of a simulated track

where $\psi_j(1)$, $\psi_j(2)$ and $\psi_j(3)$ is the set of parameters $\{\mu_{y,j}\mu_{x,j}, \sigma^2_{y,j}, \sigma^2_{x,j}\}$ used in (E.30). The state machine consists of three states: deceleration (d), constant (c) and acceleration (a) states, see Fig. E.2. The state transition probabilities are given as:

$$P_{\hat{j},j}(V(k-1), c_i) = \begin{cases} \Psi_{\hat{j},j}(1) & \text{if } V_{m-1} > \zeta^{max}, \\ \Psi_{\hat{j},j}(2) & \text{if } \zeta^{min} \leq V_{m-1} \leq \zeta^{max} \\ \Psi_{\hat{j},j}(3) & \text{else,} \end{cases}, \tag{E.32}$$

where $\hat{j}$ is the previous state and $\Psi_{\hat{j},j}$ is the transition probability. An example of a track can be seen in Fig. E.3 The speed PDFs can be seen in Fig. E.4 and the accelerations PDF can be seen in Fig. E.5. The performance of the classifier



(a) Speed PDF of the the first class

(b) Speed PDF of the the second class

(c) Speed PDF of the the third class

(d) Speed PDF of the the fourth class

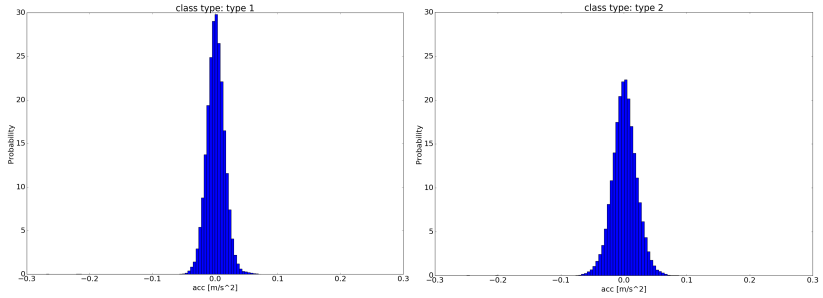
Fig. E.4: The speed PDFs of the four different classes

versus the number of measurement k can be seen in Fig. E.6. Further we show the performance of the classifier vs. the number of trees N_t used in the random forest, see Fig. E.7. The confusion matrix of the classification results for the four classes can be seen in Table E.2, where we have used $k = 10$ and $N_t = 100$.

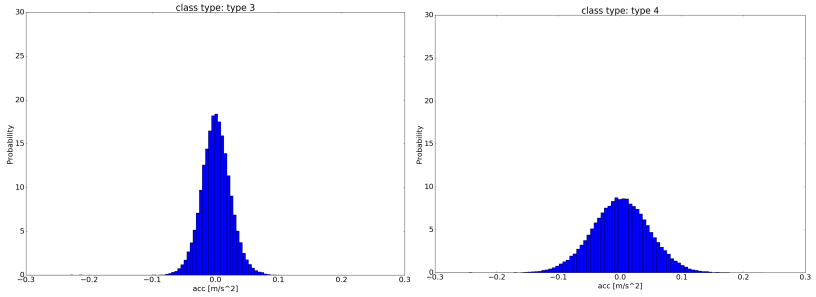
4 Real world results

The data used for this work consist of Automatic Identification System (AIS), which is a broadcast system used for large ships, Automatic Dependent Surveillance-Broadcast (ADS-B) which is a broadcast system used for commercial aircrafts, GPS logs and real world radar data. The classes for this work is typically classes for coastal surveillance e.g. large ships, birds, small boats etc.

We show a confusion matrix for real world data in Table E.3. As a confusion matrix does not take into account how the probability develops over time we also show some real world scenarios. For these scenarios extra classes are used. The scenarios are images showing all tracks within a specific time pe-



(a) Acceleration PDF for the first class **(b)** Acceleration PDF for the second class



(c) Acceleration PDF for the third class **(d)** Acceleration PDF for the fourth class

Fig. E.5: The acceleration PDF of the four classes

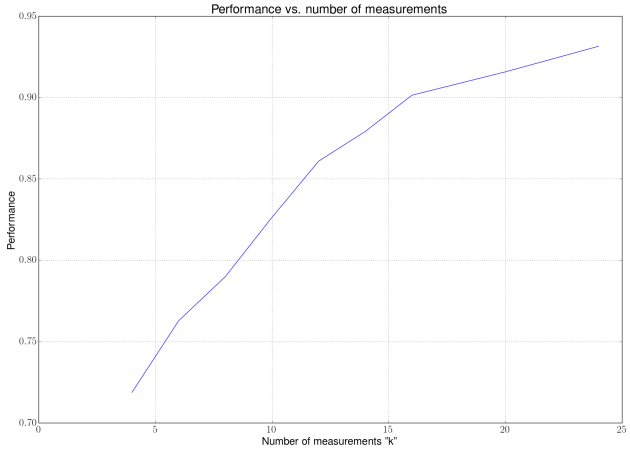


Fig. E.6: The overall performance of the algorithm given the number of measurement used in the feature vector.

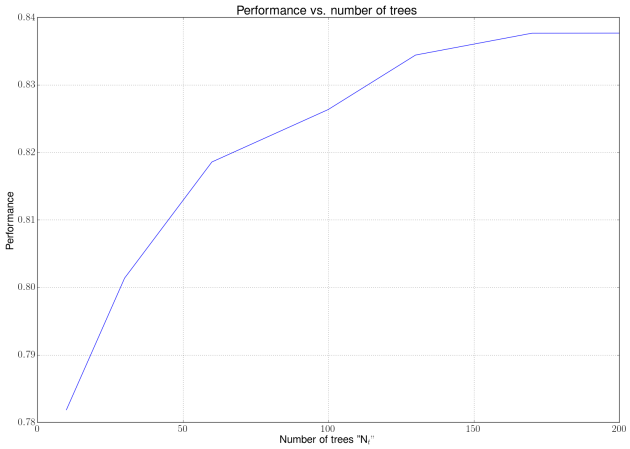


Fig. E.7: The performance of the classifier for the synthetic generated data vs. the number of trees used in the random forest.

Table E.2: The confusion matrix of the simulated data

	Predicted:			
Actual:	type 1	type 2	type 3	type 4
type 1	95.2	4.8	0.0	0.0
type 2	16.7	72.1	11.2	0.0
type 3	1.0	35.6	63.3	0.0
type 4	0.0	0.0	0.0	99.9
Overall performance	82.6			

Table E.3: The confusion matrix for real world data.

	Predicted:					
Actual:	Birds	RIBs	Stationary sea targets	Large ships	Helicopters	Commercial aircrafts
Birds	67.9	9.2	0.0	21.0	1.9	0.0
RIBs	6.4	62.4	0.0	31.2	0.0	0.0
Stationary sea targets	0.5	0.0	99.5	0.0	0.0	0.0
Large ships	21.4	5.1	0.3	61.5	11.6	0.0
Helicopters	12.2	0.0	0.0	0.0	87.8	0.0
Commercial aircrafts	0.8	0.0	0.0	0.0	0.0	99.2
Overall performance	79.7					

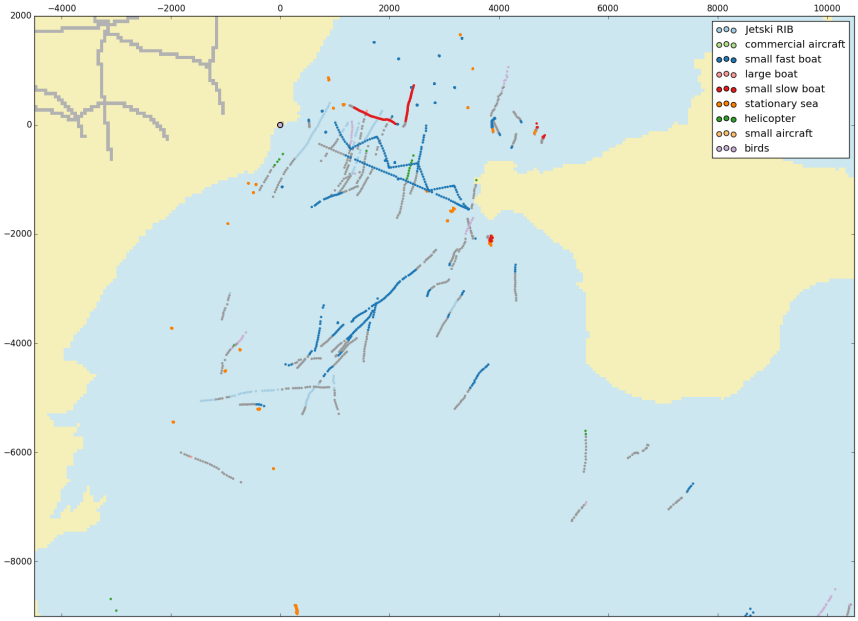


Fig. E.8: The scenario where a RIB is salling out and zigzagging back again, a big amount of birds is present

riod. The scenarios have both known and unknown targets. It is therefore not possible to make a confusion matrix of the scenario however, it is possible to have a good estimate of the performance of the classifier in real world situations. The scenarios are recorded with different radars and antennas, further the sampling rate can be different for the different scenarios. We show two scenarios from coastal surveillance applications. The first coastal surveillance scenario is recorded in Denmark where a rigid inflatable boat (RIB) is sailing from west to east and zigzagging back. Towards the north of the RIB there are two unknown vessels, further there are some sea buoys present both to the north of the RIB but also to the far south. The rest of the tracks are believed to be bird. See the scenario at Fig. E.8. The second scenario is also from Denmark and shows two wind turbines farms. A commercial plane is flying in from the west to the east and a small personal aircraft is circling over first the wind farm to the north then the second wind farm and finally leaving towards the east. Three vessels is present one to the east of the wind farm in the north (above the other wind farm) the second vessel is sailing through the wind farm in the south. The last vessel is sailing from west to east under the south wind farm. The rest of the tracks are believed to be birds, see Fig. E.9. As the majority of previously published results are based on a joint tracking and classification approach, mostly on simulated data, it

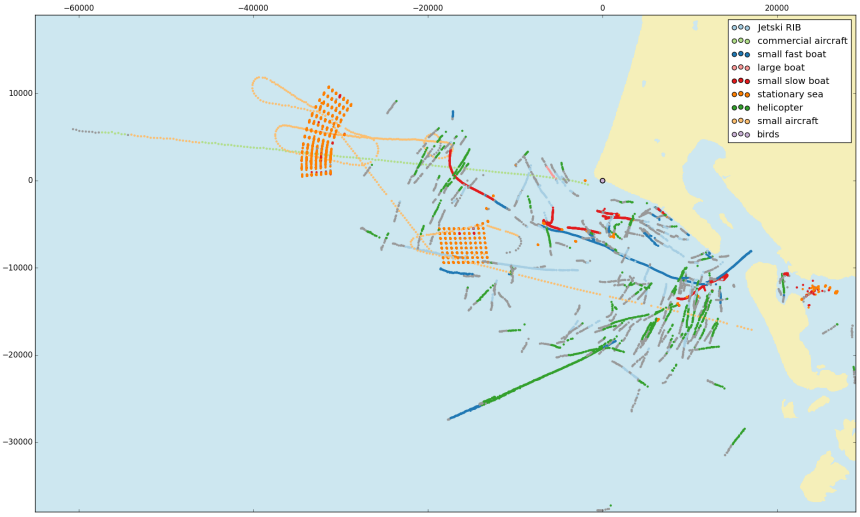


Fig. E.9: Hornsrev

is not directly possible to compare the obtained classification accuracy.

In the next section we will discus the results of the classifier.

5 Discussion

In Fig. E.6 the performance of the classification results for the simulated data set is shown, where we vary the number of measurements k, in (E.1), used to extract the features. The performance is calculated as the mean of the diagonal in the confusion matrix. It is clear the more measurement (longer feature vector) used the better the classification results. This is clear as more information to the classifier gives better estimation of the class and therefore it is more likely to classify correct. The downside of increasing the number of measurements is that it takes longer time from a track is seen until the first probability of the target is shown. For our results, the sampling rate varies between 0.333 to 1 Hz. For 10 measurements this gives, a maximum waiting time of 30 seconds, which we believe for the application in hand, is acceptable. In Fig. E.7 the performance can seen when varying the number of trees used in the random forest. The plot is made with $k = 10$. It can be seen that the performance does not get better after around 170 trees. The increase the number of trees take longer time to train the random forest and is more computational expansive and memory requiring when using the classifier for testing i.e. the purpose of the classifier is to run in real time. The performance of $k = 10$ and $n_t = 100$ can be seen in Table E.2. It is clear that type 2 and

type 3 has the most confusion between them. This is also natural if we look at the speed PDF's and the acceleration PDF's in Fig. E.4 and E.5 respectively as these is very similar. In general the of diagonal numbers in the confusion matrix is at the left side. This is due to the fact the large allowed acceleration still contains smaller acceleration which therefore will be classified as a lower class type.

For the real world scenarios we use $k = 10$ and $n_t = 170$. As it can be seen the confusion matrix in Table E.3 shows relative good performance. Nearly all of the stationary sea targets and commercial aircrafts are classified correct. The helicopters are confused with birds. This can be because of the helicopters can move as slow as birds. The are some confusion between large ships, birds and RIBs. All of these classes has kinematics which are close to each other.

In Fig. E.8 one of the real coastal surveillance scenario is shown. The scenario shows a RIB sailing out from a marina and zigzagging back again. The RIB is classified as a small fast boat. The reason that it is not classified as a jetski/RIB is that it sails more like a fast boat whereas jetski/RIB often makes turns, accelerate and decelerate. The two slow moving vessels to the north of the RIB is classified correctly. Some of the sea buoys are classified correct as stationary targets. Only a few birds are classified correctly. In Fig. E.9 two wind farms can be seen and nearly all of the wind turbines is classified as stationary, while a few are misclassified as small slow moving boats. The commercial aircraft is between commercial aircraft and small aircraft, however the target is primary classified as commercial aircraft. The small aircraft circling the two wind farms is classified correctly even though the aircraft is flying below stall speed. This can be due to the strong winds, and therefore the real airspeed is much larger. The one sea vessel that is sailing between the wind turbines is misclassified as a bird, while the other sea vessels are classified as small slow boats, small fast boats and helicopters. Unfortunately, nearly all the birds are misclassified as either unknown or as helicopter. We believe this is because that the training data do not contain any birds at that distance and speeds (because of the wind). Further the radar used to record this scenario is different from the radars used for the training data.

6 Conclusion

We have shown that it is possible to use a recursive approach to classify radar tracks from kinematic data. We have also showed that it is possible to use an alpha beta filter together with the random forest such that stationary targets are classified as stationary. The study both use simulated data, which is simulated to behave as real targets and real world data. We have shown both scenario and confusion matrix to get an overview of the performance.

References

[1] L. W. Jochumsen, M. Pedersen, K. Hansen, S. H. Jensen, and J. Østergaard, "Recursive bayesian classification of surveillance radar tracks based on kinematic with temporal dynamics and static features," in *Radar Conference (Radar), 2014 International*. IEEE, 2014, pp. 1–6.

[2] L. W. Jochumsen, E. Nielsen, J. Østergaard, S. H. Jensen, and M. Ø. Pedersen, "Using position uncertainty in recursive automatic target classification of radar tracks." IIEEE, 2015.

[3] D. H. Nguyen, J. H. Kay, B. J. Orchard, and R. H. Whiting, "Classification and tracking of moving ground vehicles," *Lincoln Laboratory Journal*, vol. 13, no. 2, pp. 275–308, 2002.

[4] S. Challa and G. W. Pulford, "Joint target tracking and classification using radar and ESM sensors," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 37, no. 3, pp. 1039–1055, 2001.

[5] D. Angelova and L. Mihaylova, "Joint target tracking and classification with particle filtering and mixture kalman filtering using kinematic radar information," *Digital Signal Processing*, vol. 16, no. 2, pp. 180–204, 2006.

[6] Z. Sun and X. J. Ban, "Vehicle classification using GPS data," *Transportation Research Part C: Emerging Technologies*, vol. 37, pp. 102–117, 2013.

[7] M. Garg and U. Singh, "C & R tree based air target classification using kinematics," in *National Conference on Research Trends in Computer Science and Technology (NCRTCST), IJCCT_Vol3Iss1/IJCCT_Paper_3*, 2012.

[8] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.

[10] E. Brookner, *Tracking and Kalman Filtering Made Easy*. Wiley-Interscience, 1998.

[11] C. M. Bishop, *Pattern recognition and machine learning*. springer New York, 2006, vol. 1.

[12] G. James, D. Witten, and T. Hastie, *An Introduction to Statistical Learning: With Applications in R*. Taylor & Francis, 2014.

[13] L. Chen and S. Wang, "Automated feature weighting in naive bayes for high-dimensional data classification," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1243–1252.

[14] J. A. Lawton, R. J. Jesionowski, and P. Zarchan, "Comparison of four filtering options for a radar tracking problem," *Journal of guidance, control, and dynamics*, vol. 21, no. 4, pp. 618–623, 1998.

[15] N. Mohajerin, J. Histon, R. Dizaji, and S. Waslander, "Feature extraction and radar track classification for detecting UAVs in civillian airspace," in *Radar Conference, 2014 IEEE*. IEEE, 2014, pp. 0674–0679.

[16] SWBD. [accessed 4 december 2014]. [Online]. Available: http://dds.cr.usgs.gov/srtm/version2_1/SWBD/

SUMMARY

The topic of this thesis is target classification of radar tracks from a 2D mechanically scanning coastal surveillance radar. The measurements provided by the radar are position data and therefore the classification is mainly based on kinematic data, which is deduced from the position. The target classes used in this work are classes, which are normal for coastal surveillance e.g.~ships, helicopters, birds etc. The classifier must be recursive as all data of a track is not present at any given moment. If all data were available, it would be too late to classify the track, as the track would have been terminated. Therefore, an update of the classification results must be made for each measurement of the target. The data for this work are collected throughout the PhD and are both collected from radars and other sensors such as GPS.

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-545-0

AALBORG **UNIVERSITY PRESS**