Automatic feature detection and terrain mapping using millimeter wave radar for autonomous land vehicle navigation

Jose, Ebi

2009

Jose, E. (2009). Automatic feature detection and terrain mapping using millimeter wave radar for autonomous land vehicle navigation. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/46991>

<https://doi.org/10.32657/10356/46991>

Nanyang Technological University

Downloaded on 26 Apr 2023 20:32:01 SGT

352
24
D/B/ue

Automatic Feature Detection and Terrain Mapping Using Millimeter Wave Radar for Autonomous Land Vehicle Navigation

Ebi Jose

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University in
fulfillment of the requirement for the degree of

Doctor of Philosophy

2009

I would like to dedicate this thesis to my loving parents

Acknowledgements

This work has benefited greatly from the input and support of many people over the past five years, and my gratitude at its completion is extended especially to those mentioned below.

To begin, I would like to thank my supervisor. To Martin Adams, thanks for your availability, expertise and advice, motivation, and encouragement to move forward.

Thank you to Graham Brooker, for your advices, discussions and suggestions on Radar principles and signal processing. Your encouragement is greatly appreciated. To Steve Clark, thanks for the willingness with which you shared your knowledge and technical help on Millimeter Wave Radar.

I would like to extend my appreciation further to Mr. Lim Eng Cheng, Mr Tan Chai Soon and other technical staff of the Robotics Research Centre and Mechanical Engineering workshop for providing me all the technical assistance.

My most valuable support by far has come from my wife Roshni. Thank you Roshni, for your love, support and understanding. To my parents, thank you for your prayers, love and care, and for financing my life and education all those years. To all my friends in NTU, thank you for all the support, understanding and company.

Abstract

Building an incremental map and simultaneously using it for localization, is of prime importance for mobile robot navigation. A prerequisite for autonomous navigation algorithms is the extraction of useful data from sensors and the correct association of that data as a vehicle moves. This thesis focuses on the successful interpretation of range/amplitude data recorded from a scanned millimetre wave (MMW) radar sensor in outdoor terrains.

MMW radars can offer remarkable advantages for autonomous navigation because their performance is less affected by dust, fog, moderate rain or snow and ambient lighting conditions than other sensors. They have a comparatively long range detection capability (typically hundreds of metres), which can enable a vehicle to localize even with sparse features. MMW radar differs from other range sensors as it can provide complete power returns for many points down range. This means that they provide received signal strength values, at all ranges, allowing user defined algorithms to detect where, in range space, the features/obstacles actually are. Such algorithms, or “Automatic target detection methods” should perform optimally in the presence of received noise.

There is a vast amount of literature on automatic target detection methods such as Constant False Alarm Rate (CFAR) algorithms for radar, most of which are applicable to air surveillance where the occurrence of distributed targets and clutter is minimal. However, target detection in outdoor terrains is a challenging task. The radar cross sections (RCS) of targets are comparatively low and there will be clutter, *i.e.*, echo from the terrain, foliage etc. This results in the degradation of the detection performance or an excessive increase in the false alarm rate. These CFAR processor limitations impede the detection of stable, discrete landmarks in outdoor environments, and can degrade vehicle localisation and map building. Therefore, the main purpose of this research is to obtain efficient feature detection algorithms for robot navigation with millimetre wave radar.

The *first* contribution of this thesis is a method to predict the range spectra from a Frequency Modulated Continuous Wave (FMCW) radar. This is very important in robot navigation for generating predictions of sensor observations from different sensor locations and correspondingly, providing an interpretation for observed targets. This is achieved by modeling the FMCW radar receiver sections, with the knowledge of the noise distributions affecting the received power and range estimates. To understand the MMW Radar range spectrum and to predict it accurately, it is necessary to know the noise distributions in the Radar power-range

spectra. A noise analysis during signal absence and presence is therefore carried out in order to understand the noise distributions that affects the power spectra. Knowledge of these noise distributions during target absence and presence also helps in deriving the Kolmogorov-Smirnov (KS) CFAR processor, which is the *second* contribution of this thesis, enabling targets to be identified from radar measurements. The proposed KS-CFAR has an excellent clutter rejection capability compared to the other standard CFAR processors, while successfully resolving closely lying features in natural outdoor terrains.

The *third* contribution is an algorithm, which detects multiple features down range, for each range spectra, based on the received signal-to-noise power. This is referred to as feature detection based on target presence probability (TPP) which provides a probabilistic map of the environment that can also be utilized for feature based robot navigation. Results are shown which compare probability based feature detection with other feature extraction techniques using a constant threshold on raw data and other variations of CFAR processors. The results show the merit of the proposed algorithm, which can detect features in typically cluttered outdoor environments with a higher success rate compared to other feature detection techniques. This work is a step towards robust outdoor robot navigation with MMW radar based continuous power spectra.

A *fourth* contribution is the practical realisation of stochastic SLAM with MMW radar. Automatic feature detection based on probability of feature presence is implemented in an outdoor vehicle navigation system. Feature selection and identification based on target presence probability is implemented with an Extended Kalman Filter - Simultaneous Localisation and Mapping (EKF-SLAM) filter to obtain robust navigation in unstructured, outdoor environments. The newly proposed Kolmogorov-Smirnov Constant False Alarm Rate (KS-CFAR) detector is also implemented with the EKF-SLAM filter and its performance is compared with TPP based outdoor SLAM.

The outdoor navigation results show the merits of the proposed methods which can successfully detect features in typically cluttered outdoor environments with a higher success rate. The contributions in this thesis are a step towards robust outdoor robot navigation with MMW radar in typical outdoor environments.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Objectives	7
1.3	Main Contributions of the Thesis	7
1.4	The Structure of the Thesis	8
2	FMCW Radar Operation - Spectra Analysis and Prediction for Robot	
	Navigation	10
2.1	Introduction	10
2.2	FMCW Radar Operation	12
2.3	Power–Range Spectra Prediction in the Presence of Noise	15
2.4	Comparing Predicted and Actual Spectra	23
2.5	Results: Power–Range Spectra Prediction	24
2.6	Conclusion	45
3	A Constant False Alarm Rate Processor based on Statistical Similarity for	
	Automatic Feature Detection	47
3.1	Introduction	47
3.2	Kolmogorov-Smirnov CFAR Processor Design Overview	52

CONTENTS

3.3	KS-CFAR Processor - Adaptive Threshold Generation and Performance Analysis	60
3.4	KS CFAR Processor in Cluttered and Multiple Target Environments	60
3.5	KS-CFAR Processor Performance Assessments	70
3.6	Conclusion	92
4	Feature Detection based on Target Presence Probability	95
4.1	Introduction	95
4.2	Target Presence Probability	96
4.3	Analysis of the TPP based Feature Detection Algorithm	100
4.4	Results	106
4.5	Performance analysis of TPP based Feature Detector and KS-CFAR processor	123
4.6	Conclusions	130
5	An Augmented State SLAM formulation for Multiple Line-of-Sight Features with Millimetre Wave Radar	132
5.1	Introduction	132
5.2	Radar Power Spectra Prediction	134
5.3	Radar Based Augmented State Vector	136
5.4	Radar Spectra Prediction Results	143
5.5	Conclusions	161
6	Simultaneous Localisation And Mapping in Outdoor Terrains with MMW Radar	164
6.1	Introduction	164
6.2	Multiple Line-of-Sight EKF-SLAM Formulation	168
6.3	Autonomous Navigation with Synthetic Data	190
6.4	Automatic Feature Relevance Determination for Effective Vehicle Navigation	201
6.5	Data Association	214

CONTENTS

6.6	SLAM Results from Outdoor Terrains with the Proposed Feature Detectors	221
6.7	Conclusions	227
7	Conclusions	231
7.1	Summary of Main Contributions	232
7.2	Future Research	237
A	The Radar Range Compensation Filter Effects	239
B	Summary of the Standard CFAR Processors	241
B.1	Introduction	241
B.2	The Concept of Cell-Averaging CFAR Processor	243
B.3	Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor	248
B.4	Analysis of the Greatest-Of and the Smallest-Of CFAR Processors . .	255
B.5	Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor	258
C	The Likelihood Ratio Test for Target Presence Probability	264
C.1	Statistical Decision Theory	264

List of Figures

1.1	<i>A full 360° radar scan obtained from an outdoor environment (car park). Features are marked in the scan.</i>	2
1.2	<i>A full 360° scanning radar mounted on a pick-up truck in a typical outdoor environment. Features include trees, lamp-posts, buildings etc.</i>	5
2.1	<i>Schematic block diagram of a MMW radar Transceiver.</i>	12
2.2	<i>Superimposed range spectra, for a constant radar swash plate bearing angle, with no targets present.</i>	16
2.3	<i>Experimental noise distribution during target absence obtained by keeping the radar pointed towards the sky.</i>	17
2.4	<i>Beat Signals in the time domain (point B in figure 2.1) during target absence, obtained from the spectra with no targets present.</i>	18
2.5	<i>Noise distribution at point B (in figure 2.1) during target absence, obtained from the spectra with no targets present. The distribution is approximately Rayleigh in nature and will be utilised in the spectra prediction method. Superimposed in red is the equivalent Rayleigh distribution with the same mean and variance as the discrete (blue) distribution.</i>	18
2.6	<i>5000 range spectra plotted together for the same bearing angle. Figure (a) shows the power–range spectra in dB. Figure (b) shows the same power spectra with power values linearised, and the range compensation filter effect removed.</i>	21
2.7	<i>Experimental histogram of range noise for an object at 10 metres.</i>	22
2.8	<i>Experimental histogram of range noise for an object at 136.5 metres.</i>	22

LIST OF FIGURES

2.9	<i>Predicted Power versus range spectrum, for a single target, with a 40 dB/decade range compensation filter applied.</i>	25
2.10	<i>Power versus range spectrum obtained from an actual target of RCS 10 m² at 10.25 metres.</i>	25
2.11	<i>Predicted power spectrum, for 2 co-aligned targets with a 40 dB/decade range compensation filter applied.</i>	26
2.12	<i>Actual power spectrum obtained from the 2 co-aligned targets.</i>	27
2.13	<i>Photograph of the car park environment used for comparing predicted and actual radar scans, with certain artefacts and natural objects labelled. Note that the car in the foreground was not present when the radar scans were recorded.</i>	28
2.14	<i>360° radar scan from the initial vehicle location "0" in figure 2.13, showing the power received from range values up to 50m. The scale on the right indicates the received power values in dB. Features include trees, lamp posts, concrete buildings and pillars.</i>	29
2.15	<i>Predicted 360° scan for the next predicted vehicle location, with predicted features labelled.</i>	30
2.16	<i>Actual 360° radar scan from the outdoor environment, 2.3m away (position "1" in figure 2.13) from the initial scan of figure 2.14.</i>	31
2.17	<i>Correlation coefficient of predicted and actual scans obtained from figures 2.15 and 2.16. Note that 0° corresponds to the 0° line marked in figure 2.15.</i>	32
2.18	<i>A zoomed view of a section of the predicted scan of figure 2.15 showing the predicted power versus range spectra in the vicinity of the radar.</i>	33
2.19	<i>A zoomed view of the corresponding section from the actual scan of figure 2.16.</i>	34
2.20	<i>Predicted radar scan from a vehicle location 11 metres away from the original scan of figure 2.14.</i>	35
2.21	<i>Actual 360° radar scan after the vehicle drives 11 metres (to position "2" in figure 2.13) from the original location of figure 2.14.</i>	36
2.22	<i>Correlation coefficient of predicted and actual scans of figures 2.20 and 2.21.</i>	37
2.23	<i>A zoomed section of the predicted scan of figure 2.20 with labelled objects.</i>	38
2.24	<i>A zoomed view of the corresponding section from the actual scan of figure 2.21.</i>	39
2.25	<i>Predicted radar scan 30m from the position of the original scan of figure 2.14.</i>	40
2.26	<i>Actual scan 30m from the position of the original scan of figure 2.14.</i>	41

LIST OF FIGURES

2.27	Correlation coefficient of predicted and actual scans of figures 2.25 and 2.26.	42
2.28	A zoomed section of the predicted scan of figure 2.25 with labelled objects.	43
2.29	A zoomed view of the corresponding section from the actual scan of figure 2.26.	44
3.1	KS-CFAR processor block diagram. There are two main sliding windows in the KS-CFAR processor. The Upper Window contains the power values from the spectrum under test and the Lower Window contain power values from noise-alone spectrum, determined a-priori. The Upper Window is sub divided into Window A and Window B and the Lower Window is sub divided into Window C and Window D. The sum of the power values in Windows A, B, C and D are labelled as Σ_A , Σ_B , Σ_C and Σ_D respectively. The sum of power values in the Upper Window and Lower Window are given as Σ_{AB} and Σ_{CD} respectively.	53
3.2	A spectrum containing only noise is obtained by pointing the radar swash plate towards empty space (pointing towards the sky).	54
3.3	MR values obtained from the VI and KS CFAR processors are plotted along with the spectrum under test. The spectrum contains reflections from land (land clutter) as well as noise. The value of the MR obtained from the VI-CFAR processor does not change significantly in the presence of clutter, which will affect the detection process. The MR values are lower during clutter transitions due to the VI-CFAR processor design. However, the proposed design of the KS CFAR processor reflects the clutter presence efficiently with higher MR values.	58
3.4	MR values obtained from the VI and KS CFAR processors are plotted along with the spectrum under test. The spectrum contains reflections from a low RCS object (tree). The value of the MR obtained from the KS-CFAR processor changes in the presence of the small RCS feature, indicating a feature presence. However, the VI CFAR processor missed the feature as the MR values did not change significantly.	59

LIST OF FIGURES

3.5	<i>Probability of similarity plots obtained from KS CFAR processor (equation 3.5) along with the spectrum. The spectrum contains clutter at approx. 20 metres along with two features at approximately 100 and 125 metres. Note that the power axis not to scale, in order to accommodate the probability of similarity of leading and lagging windows. The probability of similarity values (from equation 3.5) approaches 1.0 at around 20 metres, indicating clutter presence.</i>	63
3.6	<i>Mean ratio (MR) values obtained from the KS-CFAR processor, plotted along with the spectrum.</i>	64
3.7	<i>Hypothesis values obtained from the KS-CFAR processor for two closely lying targets.</i>	66
3.8	<i>MR statistics obtained from the KS-CFAR processor compared with the VI-CFAR processor.</i>	67
3.9	<i>The figure shows the KS-CFAR processor performance on a range spectrum with two closely lying features. The KS-CFAR processor successfully resolves the two features where as the VI-CFAR processor misses it.</i>	68
3.10	<i>The figure shows the KS-CFAR processor performance on a range spectrum compared with the CA-CFAR processor. The spectrum contains two features along with land clutter at a range of 20 metres.</i>	69
3.11	<i>Probability of detection, P_d vs. SNR plot of CFAR processors in a homogeneous environment.</i>	71
3.12	<i>Photo of the test site from where the radar scan is obtained for KS-CFAR processor tuning. Features include vehicles, buildings and walls.</i>	73
3.13	<i>Features detected by the KS-CFAR processor with window width, $N = 22$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.</i>	73
3.14	<i>Features detected by the KS-CFAR processor with window width, $N = 66$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.</i>	74
3.15	<i>Features detected by the KS-CFAR processor with window width, $N = 44$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.</i>	75

LIST OF FIGURES

3.16	<i>Features detected by the KS-CFAR processor (marked in black crosses) are plotted on a 360° radar scan. A 180° degree laser range scan (marked as green crosses) was recorded at the same time and from the same vehicle location. Features include metallic posts, buildings, pillars, trees etc. Features detected by the KS-CFAR processor are plotted in green with the laser scan as shown.</i>	76
3.17	<i>360° radar scan from a car park along with 180° sick laser scans. Features include trees, road curbs, different concrete structures such as buildings and pillars etc. Features detected by the KS-CFAR processor (plotted in green dots) are also shown in the picture.</i>	77
3.18	<i>A spectrum at 278° in the radar scan shown in figure 3.16. The reflection from the bush/foilage appears to be spread across the range and will be rejected by the KS-CFAR processor.</i>	78
3.19	<i>Satellite image of the test area from which the two different data sets - one from the urban road and the other from the off-road - are collected for analysing the proposed KS-CFAR processor performance.</i>	80
3.20	<i>Photo of the test area from where the radar scan is taken. The typical urban environment include features such as parked vehicles, buildings etc.</i>	81
3.21	<i>Figures of OS and KS CFAR processor results. The radar scan is obtained from the robot location at $[0,0,\pi/2]$. Features include vehicles, metallic container etc.</i>	82
3.22	<i>Photo of the test area from where the radar scan is taken. The environment includes features such as metallic object (shed), iron pillars, a building etc.</i>	84
3.23	<i>Figure of OS and KS CFAR processor results from an outdoor terrain. Features include a metallic wall (shed), metal pillars and a building.</i>	85
3.24	<i>Photo of the test area from where the radar scan is taken. The typical off-road environment include features such as road curbs, bush/foilage, sign post, compound wall etc.</i>	86
3.25	<i>Figure of OS and KS CFAR processor results from off-road terrains. The KS-CFAR processor is successful in rejecting most of the land and foliage clutter compared to the standard OS-CFAR processor.</i>	87
3.26	<i>Photo of the test area from where the radar scan is taken. The typical environment include features such as wall, road curbs, bush and foliage.</i>	88

LIST OF FIGURES

3.27	Figure of OS and KS CFAR processor results from an off-road environment. Features include wall, road curbs, bush and foliage.	89
3.28	Photo of the test area from where the radar scan is taken. The typical urban environment include features such as foliage/bush and land terrain.	90
3.29	Figure of OS and KS CFAR processor results obtained from a radar scan obtained from an off-road terrain. Features include foliage/bush and land terrain.	91
4.1	A radar spectrum obtained from a controlled environment. The environment contains two features, a radar reflector of 10 m^2 RCS at 10.5m and a wall at approx. 137m. The corresponding range values obtained from a Reigl laser range finder (range accuracy = 0.04m) are also shown in the figure (as inverted shaded triangles ▼). The distance measurements obtained from the laser range sensor are considered as the reference/ground-truth.	101
4.2	Figure show an ideal probability output generated with the knowledge of ground truth information. This will be the output if the TPP algorithm were perfect.	102
4.3	Analysis of TPP based detection for different parameters of α_s and α_p	104
4.3	Analysis of TPP based detection for different parameters of α_s and α_p	105
4.4	Received, noisy range spectrum, TPP and spectral subtraction, noise reduced spectrum versus range.	107
4.5	Raw radar data and corresponding target presence probability plots obtained from an indoor sports hall.	108
4.6	Raw radar data and corresponding TPP obtained from an outdoor environment (stadium) shown in figure 4.8.	109
4.7	Raw radar data and corresponding TPP obtained from the car park.	111
4.8	The satellite image of the outdoor stadium with significant features labelled.	112
4.9	The car park environment with significant features labelled.	113
4.10	The TPP based feature detection result is plotted with results obtained from four different constant threshold values. A constant power threshold is used to select the closest range value from a raw radar range spectrum. This is compared with the closest range value selected by applying a threshold of 0.8 to the TPP spectra.	114

LIST OF FIGURES

4.11	<i>TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen for the OS-CFAR processor to show the effects of varying window widths for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-4}$.</i>	117
4.12	<i>TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen to show the effects of varying window width for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-6}$.</i>	118
4.13	<i>TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen to show the effects of varying window widths for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-14}$.</i>	119
4.14	<i>TPP based feature detection and OS-CFAR processor results obtained from outdoor terrains with a fixed false alarm probability, $P_{fa} = 1 \times 10^{-6}$.</i>	121
4.15	<i>TPP based feature detection and OS-CFAR processor results obtained from outdoor terrains with a fixed false alarm probability, $P_{fa} = 1 \times 10^{-14}$.</i>	122
4.16	<i>TPP based feature detection plotted with OS-CFAR processor detections. Missed detections and false positives are marked. The result shows that TPP based feature detection shows a superior performance over the OS-CFAR processor in outdoor terrains.</i>	123
4.17	<i>A synthetic data set comprising fifty spectra with two different reflections are shown. The first reflection originates from environmental clutter whereas the second is from a feature of small RCS.</i>	125
4.18	<i>KS-CFAR feature detection output with the adaptive threshold is shown. The adaptive threshold successfully detects the small RCS feature. However, there is a false alarm due to the power reflections from clutter.</i>	126
4.19	<i>TPP based feature detection result is plotted. The result shows that TPP based feature detection have a good clutter rejection capability. The TPP based detector detects the small RCS feature successfully as shown.</i>	126
4.20	<i>A synthetic data of fifty spectra with two different reflections are shown. The first reflection originate from a clutter where as the second reflection is a feature with small RCS.</i>	127

LIST OF FIGURES

4.21	A standard KS-CFAR feature detection output with the adaptive threshold is shown. There are false alarms due to the reflections from land terrain. Due to large clutter presence in the spectrum, the real feature (lamp-post) at approx. 147 m is completely missed by the KS-CFAR processor.	128
4.22	TPP based feature detection result is plotted. The result shows that TPP based feature detection successfully detects the lamp-post. The TPP assign comparatively low probabilities (less than 0.5) to powers' that originated from land terrain.	128
4.23	The figure show the batch KS-CFAR processor results when applied to the smoothed spectrum. The only feature contained in the spectrum (lamp post) at approx. 147 metres is successfully detected by the processor along with two false alarms (due to land clutter).	129
5.1	A 360° radar range spectra obtained from an outdoor field, containing a trees, non-metallic poles, fences and concrete walls. The received power value is represented in colour space, as shown by the right hand colour bar, with power units in dB.	134
5.2	A radar range spectrum, recorded at the bearing angle 231 ° shown in figure 5.1, obtained from the outdoor field with multiple features down range. . . .	135
5.3	Power definitions for reflections, absorptions and transmissions for i multiple line-of-sight features.	140
5.4	Vehicle motion and the features observed/predicted. Features observed/predicted down-range at different bearings are marked.	144
5.5	Radar range spectra (233° azimuth) obtained at the starting robot location. Three features observed down-range are marked.	144
5.6	Predicted radar Range Spectra (at 234° bearing) obtained from the predicted vehicle location.	145
5.7	Actual radar Range Spectra (at 234° bearing) obtained at the next robot location. Features observed down-range are marked.	146
5.8	The difference between predicted and measured range spectra containing two features down-range is shown. This error is compared with the 3 σ noise power bounds.	147

LIST OF FIGURES

5.9	<i>Real 360° radar scan taken from the initial vehicle location. Features include trees, lamp posts, concrete buildings and pillars etc.</i>	149
5.10	<i>A full 360° predicted and real radar scan of an outdoor environment (car park), obtained from a predicted vehicle location 2.4m away approx. from the initial vehicle location. Features include pillars, trees, lamp posts, buildings etc. It is possible to predict distances of features that have a non circular cross section such as buildings, cars etc. However, the predicted power will not be accurate as their RCS values will change when the vehicle/radar moves (the radar signal's angle of incidence will change).</i>	150
5.11	<i>Polar plots of full 360° predicted and real radar scan of an outdoor environment (car park) with features such as pillars, trees, lamp posts etc.</i>	152
5.12	<i>The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 2.4m from the initial vehicle location.</i>	153
5.13	<i>Zoomed in versions of the full 360° predicted (a) and real (b) radar scans of an outdoor environment (car park) with features such trees, buildings, concrete wall etc.</i>	154
5.14	<i>Actual full 360° radar scan obtained from the vehicle location which is 11 metres away from the initial location. Features include walls, pillars, lamp posts, a metallic structure, trees etc. Features are extracted automatically from the scan for spectra prediction.</i>	155
5.15	<i>A full 360° predicted and real radar scan of an outdoor environment (car park), obtained from a predicted vehicle location which is 11 metres away from the initial vehicle location, with features such as pillars, trees, lamp posts etc. It is possible to predict distances of features that have a non circular cross section such as buildings, cars etc. However, the predicted power will not be accurate as their RCS values will change when the vehicle/radar moves (the radar signal's angle of incidence will change).</i>	156
5.16	<i>Polar plots of full 360° predicted and real radar scan of an outdoor environment (car park) at about 11 metres from the initial vehicle location, with features such as pillars, trees, lamp posts etc.</i>	157
5.17	<i>The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 2.4m from the initial vehicle location.</i>	158

LIST OF FIGURES

5.18	<i>Zoomed in versions of the full 360° predicted and real radar scan of an outdoor environment (car park) with features such concrete pillars, metallic objects etc.</i>	159
5.19	<i>Actual radar scan from vehicle location 30 metres away from the initial location.</i>	160
5.20	<i>Predicted scan from the next predicted vehicle location.</i>	160
5.21	<i>Actual full 360° radar scan from the measured vehicle location.</i>	161
5.22	<i>Polar plots of full 360° predicted and real radar scans of an outdoor environment (car park) with features such as pillars, trees, lamp posts etc.</i>	162
5.23	<i>The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 11m from the initial vehicle location.</i>	163
6.1	<i>Single line-of-sight predicted features. The predicted feature locations are spaced is such a manner, relative to the robot's position, that none of them line up.</i>	175
6.2	<i>Multiple line-of-sight features. The shaded yellow regions show angular sectors in which more than one feature approximately line up, with respect to the vehicle's current pose. each shaded region corresponds to a particular predicted bearing angle $\hat{\beta}_i$ and forms a tuple $\mathbf{B}_i$ of Q approximately co-aligned features.</i>	179
6.3	<i>Example: A tuple of 4 features, approximately co-aligned at bearing angle $\hat{\beta}_2$.</i>	180
6.4	<i>Ground truth of the vehicle path and feature locations. The landmarks are marked with red dots. The vehicle path is shown in red solid line. The vehicle starts at co-ordinates (0,0) and the end co-ordinates were (7,2) between which the vehicle had traversed just over a complete circle as shown.</i>	191
6.5	<i>The "true" vehicle path together with the exact feature locations (ground truth) and the estimated vehicle path. Estimated feature locations are also shown.</i>	192
6.6	<i>The spectrum obtained from the initial vehicle location which contains the first feature.</i>	193
6.7	<i>The spectrum obtained from the initial vehicle location which contains the second feature.</i>	194
6.8	<i>The spectrum obtained from the vehicle step 18. The two features are lined up as shown in the spectrum.</i>	194

LIST OF FIGURES

6.9	Actual error in vehicle location estimate in x (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors. . .	195
6.10	Actual error in vehicle location estimate in y (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors. . .	195
6.11	Actual error in vehicle location estimate in θ (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors. . .	196
6.12	Innovation in range together with associated 3σ confidence bounds. . . .	197
6.13	Innovation in bearing together with associated 3σ confidence bounds. . . .	197
6.14	Innovation in Power together with associated 3σ confidence bounds. . . .	198
6.15	Difference between the actual and estimated x location for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines). . . .	198
6.16	Difference between the actual and estimated y location for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines). . . .	199
6.17	Difference between the actual and estimated normalised RCS, Υ_{r_1} for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines). . . .	199
6.18	Difference between the actual and estimated absorption RCS, Υ_{a_1} for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines). . . .	200
6.19	The test environment (NTU car park) with the superimposed vehicle path. Typical features include trees, lamp-posts, parked cars, buildings, concrete pillars etc.	202
6.20	Five full 360° radar scans after constant thresholding, $T = \mu + 2\sigma$ with features marked.	203
6.21	SLAM simulation with synthetic data. Features include single, point features only.	205
6.22	SLAM simulation with synthetic data. Features include single, point features and the reflections from a wall.	206
6.23	The RANSAC algorithm run on data sets obtained from three different feature detectors. Due to the missed detections in the constant threshold feature extractor, the RANSAC algorithm failed in detecting the wall. However, due to the merits of the KS-CFAR processor and the TPP based feature detector, RANSAC was able to find the wall.	210

LIST OF FIGURES

6.24	<i>The RANSAC algorithm run on three data sets obtained from three different feature detectors as shown. Features include metal posts and walls. Due to the missed detections in the constant threshold detector, the RANSAC algorithm was not successful in detecting the wall. However, due to the merits of the other feature detectors, RANSAC was able to find the wall from points extracted by the KS-CFAR processor and the TPP based feature detector. . .</i>	211
6.25	<i>Analysis of the SLAM runs (100 steps) with TPP based feature detection, to verify the effect of the RANSAC based line removal method on vehicle navigation. Scans obtained from the laser are assumed to be the ground truth map and are included in the figure.</i>	212
6.26	<i>Error plots obtained from the SLAM runs (100 steps) with TPP based feature detector, to verify the effect of the RANSAC based line feature removal method on vehicle navigation.</i>	213
6.27	<i>The figure shows the Nearest Neighbour filter gating procedure for data association. The robot with position, $\hat{\mathbf{x}}_v$ and the uncertainty represented by the ellipse has to associate an observation $\mathbf{z}(k+1)$ to landmarks previously observed denoted and now predicted as $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$.</i>	216
6.28	<i>The figure shows the Nearest Neighbour filter gating procedure for data association in 3D. Both predicted features lie close together in x, y space but are well separated in power space.</i>	217
6.29	<i>The normalised innovation covariance obtained by 1000 Monte Carlo runs of the filter without the power information, to obtain a measure of correct data association.</i>	219
6.30	<i>The normalised innovation covariance obtained by 1000 Monte Carlo runs of the filter with the power information, to obtain a measure of correct data association.</i>	220
6.31	<i>The test vehicle with sensors, which include a MMW radar scanner, a 180° SICK laser scanner and odometers attached to its wheels.</i>	221
6.32	<i>SLAM with constant threshold ($\mu + 3\sigma$) based feature detection in the outdoor car park. The solid lines represents the estimated vehicle path and the odometry. Laser scans are also plotted for ground truth.</i>	224
6.33	<i>Vehicle error plots obtained from an EKF-SLAM run with a constant threshold based feature detector.</i>	224

LIST OF FIGURES

6.34	<i>SLAM with the KS-CFAR based feature detection in the outdoor car park. . .</i>	225
6.35	<i>Figure show the EKF-SLAM filter adjusting the vehicle localisation every time it receives the observations.</i>	226
6.36	<i>Vehicle error plots obtained from an EKF-SLAM run with the KS-CFAR processor.</i>	226
6.37	<i>SLAM with TPP based feature detection in an outdoor environment. . . .</i>	228
6.38	<i>Vehicle error plots obtained from an EKF-SLAM run using the TPP-based feature detection. The reduction in vehicle error in theta at approx. 270 steps is due to the observation of few features (concrete pillars).</i>	229
A.1	<i>A radar range spectrum with range compensation filter effect removed. The spectrum is obtained by pointing towards open sky and it does not contain features. A bias due to the constants, K in the radar range equation A.3 is also shown.</i>	240
B.1	<i>Generic CFAR block diagram.</i>	243
B.2	<i>Scale factor, α as a function of reference cells, N. $P_{fa} = 1 \times 10^{-6}$. . . .</i>	247
B.3	<i>The structure of the Cell Averaging-CFAR detector.</i>	249
B.4	<i>The plot shows target detection using a cell averaging CFAR detector. Detected false alarms are marked on the plot.</i>	251
B.5	<i>The plot shows target detection using a cell averaging CFAR detector. Missed detection of features are shown the figure.</i>	252
B.6	<i>A single power spectrum obtained from the radar with two power returns which lie within the window width results in target masking with the CA-CFAR processor.</i>	253
B.7	<i>A power spectrum obtained from the radar, A false alarm due to clutter edge at 20 metres along with other features are detected by the CA-CFAR processor. False alarms due to noise power variation is also shown.</i>	254
B.8	<i>Greatest Of (GO) CFAR processor performance with the adaptive threshold.</i>	257
B.9	<i>The result of an SO-CFAR processor compared with the CA-CFAR processor result.</i>	259
B.10	<i>The false alarm probability versus scale factor graph of an OS-CFAR processor for two different window widths (for $N = 22$ and 44).</i>	260

LIST OF FIGURES

B.11 The figure shows the OS-CFAR processor performance on a range spectrum.
False alarms occur along with the correct feature detection. 261

B.12 The figure shows the OS-CFAR processor performance on a range spectrum
with closely lying features. Closely lying features have been detected which
was missed by the CA-CFAR processor. 262

B.13 The OS-CFAR processor result compared with the standard CA-CFAR pro-
cessor output. 263

Chapter 1

Introduction

1.1 Motivation

The last few years have seen the implementations of autonomous systems in highly challenging application areas such as mining, agriculture, industrial and defence applications. The present successful implementations rely mainly on absolute information obtained from GPS [51] and/or an a priori information [33] about a vehicle's environment.

This thesis addresses the issues pertaining to Simultaneous Localisation and Mapping (SLAM) in outdoor terrains, with a Millimetre Wave (MMW) radar scanning sensor. MMW radar sensors can operate in adverse weather conditions such as dust, fog and light to moderate rain, which makes them ideal for outdoor mobile robot navigation. The sensor used here uses the Frequency Modulated Continuous Wave (FMCW) technique from range estimation, and the whole range profile¹ is obtained which gives an image of the environment. To demonstrate the type of data received, a full 360° radar scan is as shown in figure 1.1. The red colour refers to highly reflective objects

¹Frequency Modulated Continuous Wave radar transmits a frequency sweep, often called a *chirp*. The signal is reflected from distant targets and detected by the receiver where the returned signal is mixed with a copy of the transmitted signal to determine the ranges of the targets *i.e.*, the range profile. More detailed information about the range spectrum is provided in the next chapter.

1.1 Motivation

in the environment - *i.e.* a large received signal strength (In general, metallic objects have higher radar reflectivity compared to other features such as trees, concrete pillars etc.).

Figure 1.1: A full 360° radar scan obtained from an outdoor environment (car park). Features are marked in the scan.

The development of a reliable SLAM system with MMW radar in outdoor environments is dependent on addressing the following issues.

- *Automatic Feature Detection.* Feature detection is essential for any form of successful robot navigation (although there are some exceptions where no absolute

1.1 Motivation

feature extraction is necessary. For eg:- while using grid maps or scan matching, for localisation). Nearly all current stochastic SLAM implementations with range sensors [77], represent landmarks as simple point-locations or line features and so are viable only in environments where such features exist. After successful feature detection, navigation is completely dependent on data association. Therefore, in many outdoor environments, SLAM depends on the development of more accurate feature models, which will aid data association. Instead of just target range values, radar can provide the returned power value from the targets (which is a function of the object's Radar Cross Section (RCS)). Thus a more complete sensor model can be derived for radar based robot navigation.

- *Data Association.* In real-world SLAM problems, the mapping between observations and landmarks is rarely known. There is also no a priori information on the total number of landmarks. Every time the robot makes an observation, that sensor reading must be associated with an existing landmark or considered as coming from a previously unseen landmark. Making the wrong association can cause the SLAM filter to diverge. Two factors that contribute to uncertainty in the SLAM posterior are the measurement noise and vehicle motion noise. Each noise source leads to a different type of data association ambiguity. Noise in the measurement model will result in higher uncertainty in the landmark positions. Uncertain landmark positions will lead to measurement ambiguity, or confusion between nearby landmarks.

Higher motion noise will lead to higher robot pose uncertainty after incorporating control inputs. If this uncertainty is high enough, different plausible poses of the robot will lead to drastically different data association hypotheses for the subsequent observations. Motion ambiguity is easily induced if there is significant angular uncertainty in the robot pose estimate. If multiple observations are incorporated per time step, the motion of the robot will correlate the associations of all of the landmarks. If an algorithm selects the wrong association for a single landmark due to motion ambiguity, with high probability the rest of the associations will also be wrong. This will add a large amount of error to the robot's pose, and often cause the SLAM filter to diverge.

1.1 Motivation

EKF SLAM algorithms commonly determine data association using a maximum likelihood approach. Each observation is associated with the landmark that was most likely to have generated it. At every time step, only the single most probable data association hypothesis is considered. More sophisticated EKF data association algorithms have been developed to consider the best association of all observations simultaneously [60], however these approaches still rely on a single data association hypothesis at every time step. In a scenario with ambiguous data association, an algorithm that maintains a single data association will sometimes pick the wrong association. If the ambiguity is due to the robot's motion, this will lead to the divergence of SLAM. However, the Fast-SLAM algorithm [54] presents an alternative approach to the SLAM problem that can sample over multiple data association hypotheses. It is shown in [55] that Fast-SLAM results in better performance in situations with significant motion ambiguity.

- *Map Management.* There should be a map management scheme to maintain an optimal number of features in order to reduce the computational complexity and the storage memory requirements, but at the same time, maintain constant and accurate state updates.

The main objective of this thesis is to develop solutions to successful feature detection and accurate sensor modelling and demonstrate the functionality of these solutions with experimentation in real outdoor environments.

1.1.1 Autonomous Navigation in Outdoor Environments

The ability of a mobile robot to determine its location in space is a fundamental competence for autonomous navigation. Knowledge of self-location, and the location of other features in the environment is the basic foundation on which all high level navigation operations are built. It enables path planning for tasks such as goal reaching, terrain exploration and obstacle avoidance.

Figure 1.2 show a typical outdoor environment where vehicle navigation is considered. The operating area is often large and sensors that are reliable over extended

1.1 Motivation

distances are required. Automatic feature detection methods are required which can reliably extract features while successfully rejecting clutter¹.

Figure 1.2: A full 360° scanning radar mounted on a pick-up truck in a typical outdoor environment. Features include trees, lamp-posts, buildings etc.

1.1.2 Brief Review on Vehicle Localisation

This section summarises three types of localisation namely dead reckoning, map-aided localisation, and SLAM.

The most basic form of localisation is dead reckoning, which is the estimation of the vehicle pose by integrating estimates of its motion. Most vehicle navigation systems incorporate some component of dead reckoning, usually from encoders linked to the wheels and steering controls. By counting encoder pulses and modelling the vehicle kinematics, the predicted path can be calculated. However, errors quickly accumulate and cannot be reset without some external reference (for example, from Global Positioning System (GPS), or from radar or laser range sensors). Improving

¹clutter is the collective term used for signals originating from objects which are of less importance for navigation. E.g:- reflections from ground, foliage etc. are not reliable features for localisation or map building

1.1 Motivation

sensor and motion model accuracy can serve only to slow, however not prevent, this increase in error. Eventually, the pose estimate becomes so uncertain that it is unusable and, for this reason, dead reckoning is not a preferable solution for long-term vehicle localisation.

Dead reckoning, however, can be used as an auxiliary information source in conjunction with map-aided localisation. In map-aided localisation, a mobile robot equipped with a visual or range sensor, senses its surroundings and compares the information acquired with an a priori map to determine where it is located in the world coordinate frame. Thus, given an ability to sense its surroundings, the robot can obtain absolute pose estimates by registering sensed information with the map. However, there are problems associated with map-aided localisation. First, it is necessary to explore or be informed of the environment in advance, and to have surveyed the landmark locations before the robot can begin to navigate autonomously. Second, construction of an a priori map must be built for each new environment. Third, the resulting map is static and cannot adapt to changes in the environment (map-aided localisation cannot cope with new features/modifications in the environment.) Thus acquiring prior maps of large unexplored terrains is a challenging and difficult task.

The SLAM problem asks if it is possible for a mobile robot to be placed at an unknown location in an unknown environment and for the robot to incrementally build a consistent map of this environment while simultaneously determining its location within this map. A solution to the SLAM problem will provide the means to make a robot truly autonomous. SLAM finds numerous applications in military applications, aerospace, underwater and planetary exploration, mining, agriculture, cargo handling and driver assistance systems. Further, the maps produced by SLAM algorithms can serve as a basis for a multitude of other tasks such as motion planning and exploration.

Most real-time implementations of SLAM are based on Extended Kalman Filters with appropriate models for the sensors and vehicle kinematics [1]. At a theoretical and conceptual level, SLAM is often referred to as a solved problem [25]. However, substantial issues remain such as building and using maps for navigation, and in practically realizing map memory storage ($O(N^2)$) and computational complexity. One of the biggest challenges is navigation in large unstructured environments. In particular, SLAM in natural terrains presents a large number of unsolved problems, mainly

1.2 Objectives

in the area of sensor perception, map building and management (due to memory and processing power constraints).

1.2 Objectives

The objectives of this thesis are to investigate and analyse the issues pertaining to radar sensing for robot navigation in outdoor environments. Noise characteristics and sensor modelling of radar power-range spectra; automatic feature detection methods and SLAM implementations in outdoor terrains with MMW radar are addressed in detail. The results are a step closer towards building accurate maps of the environment while successfully localising the robot in the map.

1.3 Main Contributions of the Thesis

This thesis provides a comprehensive analysis of millimetre wave radar based robot navigation in outdoor environments. The major thesis contributions are the following:

- A detailed analysis of a 77GHz MMW radar has been carried out. Noise spectra during target presence and absence are found. This helps in accurately modelling the predicted spectra. It is also useful for Constant False Alarm Rate (CFAR) processors for automatic feature detection, by incorporating accurate noise models into the design of the processor.
- A new automatic feature detector known as the Kolmogorov-Smirnov Constant False Alarm Rate (KS-CFAR) is proposed which provides excellent clutter rejection in off-road terrains. The merits of the proposed detection method are shown by analysing the performance in various outdoor situations. The KS-CFAR processor performance is compared against the other CFAR processors such as the well known ordered statistic (OS) CFAR processor to show its merits.
- Sensor modelling with the knowledge of noise statistics during target presence and absence is presented. This is essential for predicting the radar scans, which

1.4 The Structure of the Thesis

is useful for an effective radar based SLAM implementation. The prediction of spectra has been accurately modelled by considering the absorption and reflection coefficients of features in the environment.

- A detection method based on probability of target presence, known as Target Presence Probability (TPP), is proposed. Feature detection in outdoor terrains is carried out with the proposed method and its performance has been evaluated. It will be shown that the proposed TPP based feature detector has a superior feature detection capability, compared to other standard feature detectors, especially for low RCS features.
- An Extended Kalman Filter (EKF) SLAM formulation with radar is derived and tested in outdoor terrains, with the proposed prediction and feature detection methods. The merits and the drawbacks of the proposed KS-CFAR processor and the TPP based feature detection are then discussed, in relation to the full SLAM problem.

1.4 The Structure of the Thesis

Chapter 2 gives an introduction to the radar sensor used in this work and the FMCW technique for obtaining target range. This is necessary for radar signal interpretation and for understanding and quantifying the noise in the range/power estimates. This is ultimately used in predicting power-range spectrum observations given the predicted vehicle state, in a mobile robot navigation framework - which is one of the goals of this thesis. The radar spectra prediction is useful for an effective radar based SLAM implementation. The chapter provides tools to accurately interpret and predict the power spectra by describing the FMCW operation, power spectra and the noise effects.

Chapter 3 introduces a new composite CFAR processor, known as the Kolmogorov-Smirnov (KS) CFAR processor and discusses its design, implementation and analysis in detail. A two-sample KS test, along with the Mean Ratio (MR) test are used to obtain an adaptive threshold to reliably detect features while having

1.4 The Structure of the Thesis

low false alarm rates. Results are compared with other standard and well known CFAR processors from a robot navigational perspective. It is shown that the KS-CFAR processor has a superior performance over the other CFAR detectors in resolving multi-targets while rejecting clutter at ground level.

Chapter 4 introduces a new detection method that uses the target presence probability (TPP). This method is appealing compared to CFAR processor and constant threshold methods, as a threshold can be applied to the probability of feature presence level and is shown to perform well at ground level.

Chapter 5 shows the formulation of radar power spectra prediction with multiple line-of-sight features, which is another contribution of this thesis. For feature based SLAM, it is necessary to predict the target/feature locations reliably, given a prediction of the vehicle/radar location. As radar can penetrate certain non-metallic objects it can give multiple range information. A method for accurately predicting the power-range spectra (or range bins) using the radar range equation and knowledge of various noise distributions in the radar is explained in this chapter. A mobile robot SLAM problem is then formulated which estimates a state comprising the robot pose, 2D target positions, normalised target radar cross sections (RCSs), normalised absorption cross sections and the radar power loss, L . The work shown in this chapter is a step towards building feature-rich descriptions with MMW radar in outdoor terrains, which would make data association more reliable.

Chapter 6 shows an EKF-SLAM implementation with MMW radar in outdoor terrains with the proposed prediction and feature detection methods namely the KS-CFAR processor and TPP based feature detection. Methods to improve the data association in *feature-rich* environments are also presented. These are achieved by feature pruning using the RANSAC algorithm and also incorporating added sensory information, such as reflected power measurements, into the standard Nearest Neighbour Filter. SLAM is implemented in typical outdoor terrains with the proposed detectors and their advantages and disadvantages are discussed.

Chapter 2

FMCW Radar Operation - Spectra

Analysis and Prediction for Robot

Navigation

2.1 Introduction

Compared to other sensors commonly used for vehicle navigation such as laser range finders, vision systems or sonar, scanning Millimetre Wave (MMW) radar offers the advantages of higher reliability in bad weather or unpredictable lighting conditions. Some commercially available FMCW radars differ from other range sensors as they can provide complete power returns from an array of discrete range bins, for many points down range. In addition, MMW radar has a comparatively long range (hundreds of metres) which can enable a vehicle to navigate even with sparse features in its proximity.

This chapter provides the tools to predict millimetre wave frequency modulated

2.1 Introduction

continuous wave (FMCW) radar, power–range spectra. This is achieved by theoretically analysing the process of FMCW power versus range spectra generation including the effects of receiver noise on the range and power measurements.

For Bayesian autonomous navigation and map building, it is necessary to predict the target locations accurately given a prediction of the vehicle/radar location. A method for predicting such observations for an FMCW, MMW radar, based on estimates of the range to, and received power from, multiple line-of-sight targets is given in this chapter. This is possible once prior robot and feature estimates are initialised, which include the vehicle’s pose, the estimated feature locations and their radar cross section (RCS) estimates. Such predicted power versus range spectra will also prove extremely useful for simulating real data. The use of such simulated data can aid the quantification of feature detection algorithms, since random noise and clutter can be independently controlled. To achieve this, the simple radar range equation, modelling of the FMCW process and an experimental analysis of the power and range noise statistics are used.

The chapter is organised as follows. In section 2.2, FMCW radar operation is explained, which provides an insight into the power versus range spectra generation and the noise they contain. This is essential for predicting the spectra correctly. Radar spectra prediction, using experimentally determined noise statistics during object absence is demonstrated in section 2.3. The aim is to generate realistic (with simulated noise) predicted radar spectra from predicted vehicle locations, with targets present. Section 2.4 introduces an error metric, based on the Pearson Correlation Coefficient [91], for quantifying the accuracy of predicted power versus range spectra, compared with actual data. The metric is used throughout the results in section 2.5. Firstly, individual predicted and actual power versus range spectra are compared for single and multiple line of sight features, and secondly, full 360° radar power versus range scans are predicted and recorded in a semi-urban, outdoor environment. A comparison of the predicted and actual scans demonstrates the feasibility of the proposed radar spectra prediction techniques for providing spatial and received power predictions for trees, lamp posts, pillars, cars and other artefacts in outdoor environments.

2.2 FMCW Radar Operation

2.2 FMCW Radar Operation

An introduction to the FMCW technique for obtaining target range is now given. This is necessary for radar signal interpretation and quantifying the noise in the range/power estimates. This is ultimately used in predicting range spectra observations given the predicted vehicle state, in a mobile robot navigation framework - which is the goal of this article. By analysing the FMCW technique it will be shown which noise sources affect both the range and received power estimates, and how they are affected.

The transmitted power of the radar used here is 15 dBm and the swept band-width is 600 MHz [15]. Figure 2.1 shows a schematic block diagram of an FMCW radar transceiver. In the figure, the input voltage to the Voltage Control Oscillator (VCO)

Figure 2.1: Schematic block diagram of a MMW radar Transceiver.

is a ramp signal. The VCO generates a signal of linearly increasing frequency δf over a frequency sweep period T_d . This linearly increasing chirp is transmitted via the antenna. An FMCW radar measures the distance to an object by mixing the received signal with a portion of the transmitted signal [72]. The mixer output (point B in figure 2.1) then contains a signal, with beat frequency proportional to the target range. This signal is then passed through a high pass (range compensation) filter for ultimately obtaining a constant returned power value which is independent of range¹, for a target of given RCS. Theoretically, assuming the simple radar equation 2.7 to be true, this would be achieved with a simple 40 dB/decade, high pass filter. This signal is

¹and decreasing the dynamic range of the signal, necessary for further processing.

2.2 FMCW Radar Operation

then convolved with a Blackman window [63], to reduce the spectral leakage in the frequency spectrum output of the Fast Fourier Transform (FFT) processor [14]. Each element of the FFT result is multiplied by its complex conjugate to obtain the signal power spectral density. The logarithm of the power spectral density is then obtained to compress the large dynamic range of the power value estimates [48], which is the output of the radar.

2.2.1 FMCW Radar - Power Spectra Generation

The working principle of a standard FMCW radar receiver, which outputs received power-range spectra, is explained in this section. An in depth understanding of the spectra generation principle is essential for predicting the spectra.

Let the transmitted signal $v_T(t)$, as a function of time t , be represented as

$$\begin{aligned} v_T(t) &= [A_T + a_T(t)] \cos \left[\omega_c t + A_b \int_0^t t dt + \phi(t) \right] \\ &= [A_T + a_T(t)] \cos \left[\omega_c t + \frac{A_b}{2} t^2 + \phi(t) \right] \end{aligned} \quad (2.1)$$

where A_T is the amplitude of the carrier signal, A_b is the amplitude of the modulating signal, ω_c is the carrier frequency (*i.e.* $2\pi \times 77$ GHz in this case), $a_T(t)$ is the amplitude noise and $\phi(t)$ is the phase noise present in the signal which occurs inside the transmitting electronic sections.

At any instant of time, the received echo signal, v_R is shifted in time from the transmitted signal by a round trip time, τ . The received signal is

$$v_R(t-\tau) = [A_R + a_R(t-\tau)] \cos \left[\omega_c(t-\tau) + \frac{A_b}{2}(t-\tau)^2 + \phi(t-\tau) \right] \quad (2.2)$$

where A_R is the received signal amplitude, $a_R(t-\tau)$ is the amplitude noise and $\phi(t-\tau)$ is the phase noise. The sources of noise affecting the signal's amplitude consist of external interference to the radar system (*e.g.*, atmospheric noise, man-made signals) and internally produced noise at the receiver antenna and amplifiers.

2.2 FMCW Radar Operation

In the mixer, the received signal is mixed with a portion of the transmitted signal in an analog multiplier, which can be mathematically described by

$$\begin{aligned} v_T(t) v_R(t-\tau) = & [A_T + a_T(t)] [A_R + a_R(t-\tau)] \left\{ \cos \left[\omega_c t + \frac{A_b}{2} t^2 + \phi(t) \right] \right\} \\ & \times \left\{ \cos \left[\omega_c (t-\tau) + \frac{A_b}{2} (t-\tau)^2 + \phi(t-\tau) \right] \right\}. \end{aligned} \quad (2.3)$$

The output of the mixer, $v_{out}(t)$ is ¹

$$v_{out}(t-\tau) = \frac{[A_T + a_T(t)] [A_R + a_R(t-\tau)]}{2} [B_1 + B_2] \quad (2.4)$$

where $B_1 = \cos \left[(2t-\tau) \left(\omega_c - \frac{A_b \tau}{2} \right) + A_b t^2 + \phi(t) + \phi(t-\tau) \right]$ and $B_2 = \cos \left[(\omega_c - A_b (\frac{\tau}{2} - t)) \tau + \phi(t) - \phi(t-\tau) \right]$.

The second cosine term, B_2 is the signal containing the beat frequency. The output of the low pass filter (which is intrinsic in the mixer stage) consists of this beat frequency component, B_2 and noise components with similar frequencies to the beat frequency, while other frequency components are removed. The beat frequency, f_{bi} caused by the i th target down-range is directly proportional to the round trip time, τ . The relationship between the beat frequency and target range is

$$R_i = \frac{c T_s}{2} \frac{1}{f_s} f_{bi} \quad (2.5)$$

where R_i is the range of the i th object, c is the velocity of the electromagnetic wave, T_s is the frequency sweep period, and f_s is the swept frequency bandwidth [49]. Hence closer objects produce signals with lower beat frequencies and vice-versa.

¹using the trigonometric identity for the product of two sine waves $\cos A \cos B = 0.5[\cos(A+B) + \cos(A-B)]$.

2.3 Power–Range Spectra Prediction in the Presence of Noise

2.3 Power–Range Spectra Prediction in the Presence of Noise

Radar noise is the unwanted power that impedes the performance of the radar. For the accurate prediction of range spectra, the characterisation of noise is important. It is therefore the aim of this section to introduce the type of noise distributions affecting the received chirp, and hence the received power and range estimates. The two main noise components are thermal and phase noise. Thermal noise affects the power reading while phase noise affects the range estimate. Contrary to most sensor noise models within the autonomous robotics literature, the noise in the MMW radar's received signal will be shown to have most affect on its *received power value* rather than its range estimate. This is an important observation which translates to the fact that the received noise mainly affects the uncertainty in the very *existence* of an object being reported by the radar at a particular range, rather than the *range* value of the possible object itself.

The output of the filter within the mixer stage, which selects the difference frequency signal component within equation 2.4 is given by

$$v_{beat}(t, \tau) = \frac{A'}{2} \cos \left\{ \left[\omega_c - A_b \left(\frac{\tau}{2} - t \right) \right] \tau + \Delta\phi(t, \tau) \right\} \quad (2.6)$$

where $A' = [A_T + a_T(t)] [A_R + a_R(t - \tau)]$ is the product of the transmitted and received signal amplitudes, each corrupted with noise. $\Delta\phi(t, \tau) = \phi(t) - \phi(t - \tau)$ is called the differential phase noise which occurs due to the leakage of transmitted signals into the mixer [3].

Next, a brief discussion of the noise components present in the beat signal (shown in equation 2.6) is given. This information is necessary in understanding the radar spectra and an effective representation of features.

2.3 Power–Range Spectra Prediction in the Presence of Noise

2.3.1 Received Power Noise Estimation

As shown in equation 2.4, the beat frequency signal is affected by the thermal noise signal $a_R(t - \tau)$, which contributes to A' in equation 2.6 (point B in figure 2.1). Therefore this noise source directly corrupts the estimated received power.

Since the radar used is essentially a sealed black box, the method adopted to determine this noise (in terms of its distribution and parameter values) is to monitor several recorded power versus range spectra from the logarithmic compressor in figure 2.1 (the output of the radar). This is carried out by keeping the scanning swash plate stationary, and pointing into open sky (target absence). To determine the noise in the received signal, it is then necessary to reverse the effect of all of the signal processing blocks in figure 2.1 between the output and point B .

Received Power Noise Estimation During Target Absence

Figure 2.2 shows 5000 actual superimposed range spectra output from the radar, obtained at a particular radar bearing angle, during target absence. The exponential noise distribution obtained from these spectra is shown in figure 2.3.

Figure 2.2: *Superimposed range spectra, for a constant radar swash plate bearing angle, with no targets present.*

2.3 Power–Range Spectra Prediction in the Presence of Noise

Figure 2.3: Experimental noise distribution during target absence obtained by keeping the radar pointed towards the sky.

The inverse operations of the final filtering stages in the receiver section of figure 2.1 were then applied to these spectra. This results in multiple time domain signals, characterising the noise present, before range compensation and logarithmic compression has taken place (ie. at point *B*, after the mixer stage in figure 2.1). This involves removing the range compensation filter effect. It is then necessary to obtain the real part of the FFT from the power spectra, and apply the inverse Fourier transform to obtain the corresponding time domain signal. Finally, removing the windowing effect (Blackman window) from the signals gives the time-domain signals at *B*. The resulting signals are shown in figure 2.4. These time domain signals, are used to compute the noise statistics in the radar spectra. Due to the slight variation of the weights in the Blackman windowing process, the amplitude of the beat signals are higher at both the ends of the time domain signal. The noise analysis is carried out in the central region of the graph of figure 2.4 where the windowing nonlinearities are minimal. The noise distribution then obtained at point *B* is as shown in figure 2.5. This distribution is approximately Rayleigh in nature¹ and the standard deviation of the Rayleigh noise is estimated as $1.25 V$. As a first approximation it is reasonable to assume that the noise power statistics are independent of range and are dependent on the radar receiver

¹Two Gaussian noise sources, when passed through a mixer, result in Rayleigh noise [71], which explains the shape of the distribution in figure 2.5.

2.3 Power-Range Spectra Prediction in the Presence of Noise

Figure 2.4: Beat Signals in the time domain (point B in figure 2.1) during target absence, obtained from the spectra with no targets present.

Figure 2.5: Noise distribution at point B (in figure 2.1) during target absence, obtained from the spectra with no targets present. The distribution is approximately Rayleigh in nature and will be utilised in the spectra prediction method. Superimposed in red is the equivalent Rayleigh distribution with the same mean and variance as the discrete (blue) distribution.

2.3 Power–Range Spectra Prediction in the Presence of Noise

electronics only, an approximation which has been verified experimentally [59]. The knowledge of the noise statistics during target absence (figure 2.5) is utilised in predicting the power spectra with multiple target returns, which is explained in the next section.

Predicting Noisy Power – Range Spectra with Targets Present

The power-range spectra, when target(s) are present can then be predicted by summing multiple, time domain, sinusoids, with frequencies related to the range of each target given by equation 2.5. The mixer output then contains one or more beat frequencies f_{b_i} which correspond to the target range(s) R_i (given in equation 2.5). The amplitude of each of these sine waves is proportional to P_i , which can be estimated by entering the predicted range R_i and RCS value σ_i into the simple radar equation 2.7:

$$P_i = \frac{P_t G^2 \lambda^2 \sigma_i}{(4\pi)^3 R_i^4 L} \quad (2.7)$$

where P_t is the radar's transmitted power, G is the antenna gain, λ is the wavelength, (*i.e.* 3.89 mm in this case) and L the radar system losses (The total system losses include atmospheric loss, beam-shape loss, beam-width factor, power fluctuation loss, miscellaneous signal processing loss and transmitter and receiver line noise [78]. The value of L is assumed to be 3dB in this work.). These predicted sinusoids, with frequencies f_{b_i} and amplitudes proportional to P_i , are then artificially corrupted with additive noise, generated according to the probability density function (PDF) of the received power range spectra during target absence. Finally, the filtering operations of the receiver, after point B in figure 2.1, are applied to the noise corrupted sinusoid(s) to give the final, noise corrupted, predicted power-range spectrum. This analysis makes the simplifying, but reasonable, assumption that the noise at point B in figure 2.1 primarily results from the components within the radar, and that it is independent of the environmental conditions (*ie.* the same noise source (figure 2.5) is used to corrupt all power values).

2.3 Power–Range Spectra Prediction in the Presence of Noise

2.3.2 Range Noise Estimation

Another source of noise which affects the range spectra is the phase noise $\Delta\phi(t, \tau)$ in equation 2.6. Phase noise is generated by the path leakage of the transmitted signal to the mixer, resulting in a spectrum of frequencies with finite bandwidth instead of a single beat frequency. This introduces noise into the range estimate itself, and experimental data shows that this results in a slight broadening of the power peaks along the range axis. This will be noticeable in many of the actual scans shown in section 2.5.

To demonstrate and estimate the phase noise effect, a large number (5000) of superimposed range spectra obtained for the same radar swash plate bearing angle are plotted together in figure 2.6. The top figure shows the entire range spectra with the returned power in dB¹, while the lower figure shows the same set of range spectra using a linearly scaled power axis, with the effect of the range compensation filter removed. The Spectra contain three dominant features, observed due to the partial occlusion of objects, that lie inside the radar's beam width². To estimate the phase (and hence range) noise distribution, a statistical analysis has been carried out on the targets (the peaks at approximately 10 and 136.5 metres), which also have differing RCS values, to determine any power and/or range dependency of the range noise statistics. The resulting range histograms are shown in figures 2.7 and 2.8. The distributions show only the range values which correspond to the maximum power values recorded from a given target, with the radar swash plate fixed, taken from 5000 range spectra. It is apparent that the resulting range standard deviations are approximately independent of range and returned power, and more importantly, that they are significantly less than the range resolution (0.25 m) for this particular radar. Therefore, in the prediction of the radar range spectra, the broadening of peaks will not be considered significant in comparison with the range resolution of the radar's range bins, and the phase (and hence range) noise is ignored.

It is therefore reasonable to assume that the thermal noise which affects the received power value, plays a crucial role in the correct detection of a target. Feature/target ex-

¹Note that the received power (dB) is a scaled value, which is proportional to the actual received to transmitted power gain.

²In figure 2.6 (top) (received power with a dB scale) three power returns are evident, due to the range compensation filter and logarithmic scaling. Clearly, after the power spectra is linearised and the range compensation filter effect removed, targets at higher range (such as the third target at 150 m) cannot be visualised on a linear scale, as shown in figure 2.6 (lower).

2.3 Power-Range Spectra Prediction in the Presence of Noise

Figure 2.6: 5000 range spectra plotted together for the same bearing angle. Figure (a) shows the power-range spectra in dB. Figure (b) shows the same power spectra with power values linearised, and the range compensation filter effect removed.

2.3 Power-Range Spectra Prediction in the Presence of Noise

Figure 2.7: Experimental histogram of range noise for an object at 10 metres.

Figure 2.8: Experimental histogram of range noise for an object at 136.5 metres.

2.4 Comparing Predicted and Actual Spectra

traction algorithms must be able to minimise the affect of this noise on target existence decisions.

The tools are now complete to predict radar spectra, given a robot navigation framework, capable of predicting the range to a target and its RCS¹. This is possible given prior robot and feature estimates, which include the vehicle's pose, estimated feature locations and their RCS values. A Swerling *I* model is assumed, where the RCS values of objects are assumed to be omni-directionally constant with additive random fluctuations modelling RCS differences caused by the perspective from which the object is scanned [75]. In general this is clearly not a valid assumption, but becomes acceptable for small, cylindrical objects, making their RCS's approximately view-point invariant, such as lamp posts, trees etc, which can be used for outdoor navigation.

2.4 Comparing Predicted and Actual Spectra

To quantify the "similarity" between a predicted power versus range spectrum, computed from previously recorded data from previous vehicle positions, and the current, corresponding real spectrum, the Pearson's coefficient of determination is used [91]. This coefficient provides a measure of the strength of the linear relationship between the predicted spectra X , with spectral values x_i , and the observed range spectra Y , with spectral values y_i , and will provide an indication of how accurate the predicted observations (spectra) are. This coefficient r_{xy}^2 ($0 < r_{xy}^2 < 1$) is given by

$$r_{xy}^2 = \left[\frac{N \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{(N \sum x_i^2 - (\sum x_i)^2) (N \sum y_i^2 - (\sum y_i)^2)}} \right]^2 \quad (2.8)$$

where each summation is taken from $i = 1$ to N with N being the number of bins in each power versus range spectra at a given scanning bearing angle (in this case $N=800$). A value of $r_{xy}^2 = 1$ shows the two spectra are fully correlated and $r_{xy}^2 = 0$ shows that the two spectra are uncorrelated. Before determining this coefficient, the

¹ which is theoretically proportional to the received power, assuming the range compensation, high pass filter to be ideal.

2.5 Results: Power–Range Spectra Prediction

actual received spectrum under comparison is linearised, and then the effect of the range compensation filter removed, assuming it behaves as a 40 dB/decade, high pass filter, as mentioned in section 2.2. This allows comparison with a range uncompensated, linear power-range predicted spectrum. Due to the large dynamic range of the received signals however, for graphical comparison, the predicted and the actual spectra are still shown on a logarithmic scale, with range compensation applied (simulated with a 40 dB/decade, high pass filter on the predicted spectra).

2.5 Results: Power–Range Spectra Prediction

2.5.1 Single Bearing Power-Range Spectra Prediction

Spectra Prediction from a Single Feature

As an example, a single power-range spectrum is to be predicted for an object with a known RCS (10m^2), assumed to be at a distance of 10.25 metres from the radar. This will then be qualitatively and quantitatively compared with a real spectrum, recorded under the same conditions.

To generate the predicted observation, the beat signal corresponding to the object's range is artificially corrupted by Rayleigh noise, with the experimental parameters obtained from figure 2.5. The identical processing of the receiver elements after point *B* in figure 2.1, is then applied to this signal, to produce the returned power in dB shown in figure 2.9. Analysis of the predicted (figure 2.9) and actual range spectrum (figure 2.10) shows a slight mismatch in the noise frequency with respect to range. This mismatch is most likely due to the unmodelled phase noise throughout the entire range spectrum. It should also be noted that the actual radar used here is unable to report received power values for objects at ranges below 5m, as can be seen in figure 2.10. The magnitude of the received power from the target (30dB) corresponds to that actually received in the controlled experiment.

The r_{xy}^2 value (equation 2.8) obtained for figures 2.9 and 2.10 is 0.9741, indicating a close similarity in the predicted and actual spectra.

2.5 Results: Power–Range Spectra Prediction

Figure 2.9: *Predicted Power versus range spectrum, for a single target, with a 40 dB/decade range compensation filter applied.*

Figure 2.10: *Power versus range spectrum obtained from an actual target of RCS 10 m^2 at 10.25 metres.*

2.5 Results: Power–Range Spectra Prediction

Spectra Prediction from Multiple Features

If a predicted range spectra, contains multiple targets down-range, the predicted power from the first (closest) target only will be predicted correctly, since, in this work, no account is taken of the power absorption from the $N - 1$ targets occluding the N th target down-range. The prediction of the reduced power, which would be received from occluded targets has been addressed in [44].

Consider signal returns from multiple features¹. The mixer output then contains multiple frequencies ($f_{b_1}, f_{b_2} \dots, f_{b_N}$) which correspond to the ranges ($R_1, R_2 \dots R_N$) (given by equation 2.5) to different objects. The power spectrum containing reflections from different objects can be predicted, as explained in section 2.5.1. As an example, figures 2.11 (predicted spectrum) and 2.12 (real spectrum) show range compensated radar power spectra with reflections from two features. The first reflection is from a lamp post and the second reflection is from a tree.

Figure 2.11: Predicted power spectrum, for 2 co-aligned targets with a 40 dB/decade range compensation filter applied.

For figures 2.11 and 2.12, $r_{xy}^2 = 0.9807$. This value indicates that the predicted range spectrum has a close “similarity” to the real spectrum with multiple features and

¹ Either due to the beam width or due to the radar wave penetrating closer objects.

2.5 Results: Power–Range Spectra Prediction

Figure 2.12: Actual power spectrum obtained from the 2 co-aligned targets.

that it is possible to predict power spectra with multiple feature returns accurately.

2.5.2 Full 360° Scan Spectra Prediction

This section compares the full 360° predicted radar scans with actual scans, obtained from outdoor terrains. The predicted full scans are then statistically compared with the actual ones to “quantify the goodness” of the proposed prediction method and the results are analysed. Qualitative comparisons are also given, in terms of the known existence and position of objects within the scanned environments.

In autonomous navigation experiments, the radar, which is mounted on to the vehicle, scans the environment. For analysing the proposed prediction method, tests have been carried out in an outdoor car park, shown in figure 2.13.

The real radar scan obtained from the initial vehicle location (marked “0” in the photograph) is shown in figure 2.14¹. The colour scale in the figure indicates the received power values. Features are extracted from this initial, raw scan using a cell averaging CFAR processor [37]. Next, the vehicle is moved and its location is predicted,

¹Note that the 360° scans, can be best visualised in colour.

2.5 Results: Power–Range Spectra Prediction

Figure 2.13: Photograph of the car park environment used for comparing predicted and actual radar scans, with certain artefacts and natural objects labelled. Note that the car in the foreground was not present when the radar scans were recorded.

based on odometry and the vehicle kinematic model [13; 22]. A predicted scan is then calculated from that predicted vehicle location, by translating the previously extracted feature locations through the vehicle motion model, and using the above power-range spectra prediction method. The full predicted 360° scan from the car park is shown in figure 2.15. The vehicle is then commanded to move to that location (position “1” in figure 2.13) which is approximately 2.3 metres away from the initial location, and an actual radar scan is then obtained. This is shown in figure 2.16, in which certain identifiable objects are labelled for comparison with figure 2.15. From a qualitative point of view, it can be seen that artefacts such as the pillars, cars and concrete corner section correspond well between the predicted and actual scans. The most notable differences between them are caused by *spectrum saturation*, *phase noise* and *clutter* present in the actual scan. Spectrum saturation occurs when a very high RCS target is present at a particular angle, which can cause the entire spectral power values, at that bearing, to be high, due to flooding of the receiver amplifier sections [12]. This is evident at the two labelled sections in figure 2.16. As mentioned in section 2.3.2, phase noise, which is also not modelled in our scan predictor, broadens the power peaks, thus increasing

2.5 Results: Power-Range Spectra Prediction

Figure 2.14: 360° radar scan from the initial vehicle location "0" in figure 2.13, showing the power received from range values up to 50m. The scale on the right indicates the received power values in dB. Features include trees, lamp posts, concrete buildings and pillars.

2.5 Results: Power-Range Spectra Prediction

Figure 2.15: Predicted 360° scan for the next predicted vehicle location, with predicted features labelled.

2.5 Results: Power-Range Spectra Prediction

Figure 2.16: Actual 360° radar scan from the outdoor environment, 2.3m away (position "1" in figure 2.13) from the initial scan of figure 2.14.

2.5 Results: Power–Range Spectra Prediction

the uncertainty in the range at which objects are detected. This can be seen, as the received power from the wall, pillars etc are “widened” along their respective range axes of detection in comparison with their predicted values in figure 2.15. Clutter is also evident in the real data, and is not modelled in the predicted scans.

For quantifying the statistical similarity between the predicted and actual scans, again the correlation coefficient is obtained between each actual and predicted spectrum. The correlation coefficient r_{xy}^2 for the full scan is plotted at each bearing angle in figure 2.17, from which it is evident that relatively high (maxima) correlation values exist in the directions of the detected pillars, concrete corner section, large trees and a small section of the left wall. However, not all of the predicted spectra are correlated

Figure 2.17: Correlation coefficient of predicted and actual scans obtained from figures 2.15 and 2.16. Note that 0° corresponds to the 0° line marked in figure 2.15.

(or similar) with the real spectra, due to any of the following reasons:

2.5 Results: Power–Range Spectra Prediction

Figure 2.18: A zoomed view of a section of the predicted scan of figure 2.15 showing the predicted power versus range spectra in the vicinity of the radar.

- There is no clutter modelling incorporated into the prediction model. Hence when a real spectrum, containing land clutter, is compared with the predicted one (with no clutter), a lower correlation results.
- Features may appear/disappear due to either partial occlusion of objects or due to object(s) being outside the radar's maximum visible range, at the position from which the predicted scan was computed.

Figure 2.18 (which is a zoomed view showing the pillars, cars and concrete corner section labelled in figure 2.15) shows a more detailed view of these predicted artefacts. The corresponding zoomed actual scan is shown in figure 2.19. Again, in these scans, the main predicted and detected objects are clear as are the effects of phase noise and clutter in the real data. Importantly, robust objects are reliably predicted, making them suitable as features within robot navigation frameworks.

Figure 2.20 shows the predicted 360° radar scan obtained from a vehicle location which is 11 metres away from the initial location. This is to show the usefulness of

2.5 Results: Power-Range Spectra Prediction

Figure 2.19: A zoomed view of the corresponding section from the actual scan of figure 2.16.

2.5 Results: Power-Range Spectra Prediction

Figure 2.20: Predicted radar scan from a vehicle location 11 metres away from the original scan of figure 2.14.

2.5 Results: Power–Range Spectra Prediction

the predicted observations, for faster moving vehicles, or for slower prediction/update times, as it then becomes necessary to predict a scan at greater distances away from the data set used to compute it. The actual scan from that vehicle location is shown in figure 2.21. The correlation coefficient between each spectrum is shown in figure

Figure 2.21: Actual 360° radar scan after the vehicle drives 11 metres (to position “2” in figure 2.13) from the original location of figure 2.14.

2.22. As the vehicle moves from positions “1” to “2” in figure 2.13, the magnitude of the correlations in figure 2.22 in the directions of the concrete corner section, and the pillars are notably reduced, whilst those in the direction of vehicle motion, (corresponding to trees and lamp posts) increase. Again, importantly, most of the labelled features (which can be seen more clearly in the zoomed views of figures 2.23 and 2.24) are accompanied by relatively high correlation coefficients (labelled in figure 2.22).

2.5 Results: Power-Range Spectra Prediction

Figure 2.22: Correlation coefficient of predicted and actual scans of figures 2.20 and 2.21.

2.5 Results: Power-Range Spectra Prediction

Figure 2.23: A zoomed section of the predicted scan of figure 2.20 with labelled objects.

2.5 Results: Power-Range Spectra Prediction

Figure 2.24: A zoomed view of the corresponding section from the actual scan of figure 2.21.

2.5 Results: Power–Range Spectra Prediction

Finally, a predicted scan is generated for a large distance manoeuvre corresponding to 30m of vehicle motion (placing the vehicle at position "3" in figure 2.13). As described in section 2.5.2, features extracted from the initial position ("0" in fig 2.13) are transformed using the vehicles odometry and the spectra prediction method to produce the predicted scan of figure 2.25. Note that a slight angular shift in the scan is apparent

Figure 2.25: Predicted radar scan 30m from the position of the original scan of figure 2.14.

due to a slight vehicle rotation reported by the odometry.

When traversing such a large distance, one would expect less of the predicted features to match those in the real scan recorded at position "3" due to the occlusion or cluttering of distant objects recorded at position "0". Figure 2.26 shows the actual scan recorded at position "3" along with the correlation coefficient plotted in figure 2.27, where it can be seen that robust features are still correctly predicted. The trees,

2.5 Results: Power-Range Spectra Prediction

Figure 2.26: Actual scan 30m from the position of the original scan of figure 2.14.

2.5 Results: Power-Range Spectra Prediction

Figure 2.27: Correlation coefficient of predicted and actual scans of figures 2.25 and 2.26.

2.5 Results: Power–Range Spectra Prediction

Figure 2.28: A zoomed section of the predicted scan of figure 2.25 with labelled objects.

lamp posts and a sign post, to the rear (between scanning angles 0° and 180°) of the vehicle at position “3”, can be seen in both the predicted and actual scans, and have correspondingly high correlation coefficients. This is logical, as this region is closer to the radar, when the scan was predicted at position “0”. Figures 2.28 and 2.29 show zoomed views in the vicinity of the robot with a clearer view of the successfully predicted trees, lamp posts and sign post, corresponding to the peaks in the correlation coefficient of figure 2.27.

2.5.3 Spectra Prediction: Level of Significance

The spectra prediction method explained here is useful for radar based robot navigation. In order to quantify the level of correlation which is acceptable for a predicted spectrum to be used in robot navigation, *tests of significance* can be deployed. *Tests of significance* can examine whether there exists any “relationship” between predicted and actual spectrum, or whether pure chance could produce the predicted spectrum

2.5 Results: Power-Range Spectra Prediction

Figure 2.29: A zoomed view of the corresponding section from the actual scan of figure 2.26.

closely similar to the observed one.

The tests of significance depends on Pearson's correlation coefficient, r and sample size, N (in this case, $N=800$, which is the number of bins in one spectrum). Probability is typically assessed as the fractional or percentage chance that the result is non-significant (known as examining the *null hypothesis*). The borderline that is most commonly used to calculate the level of significance is $\xi < 0.05$, i.e., the probability of non-significance is less than 5%.

In order to determine if the r value that is calculated meets the above requirement, the critical value table for Pearson's Correlation Coefficient is used. The two pieces of information are required to use the table - the degrees of freedom, df^1 , (which is the total number of samples, $N - 2$) and the correlation coefficient, r .

¹The number of degrees of freedom is the number of values in the final calculation of a statistic that are free to vary. In general, for a statistical sample distribution, if there are N samples and m constraints on the distribution, then there are $N - m$ degrees of freedom. In this case, there are $N=800$ samples in each data set and 2 constraints (as Pearson's correlation assess the relationship between two variables). Thus, the degrees of freedom $df = N - 2 = 798$.

2.6 Conclusion

After calculating the values for r , the critical value table can be used to find the intersection of $\xi < 0.05$ and $N-2$ degrees of freedom, df . The value found at the intersection is the minimum correlation coefficient, r that is required to confidently state 95 times out of a hundred that the relationship found between predicted and measured spectrum ($N-2$) samples are significantly similar.

If the absolute value of your correlation coefficient is higher than the value obtained from the reference table, the null hypothesis is rejected and accept the alternative hypothesis; that is the predicted and measured spectrum are statistically similar.

If the absolute value of your correlation coefficient were less than the value obtained from the reference table, the null hypothesis will be failed to reject: which means that there is no statistical similarity between the predicted and the measured spectrum.

2.6 Conclusion

A technique for predicting scanned millimetre wave radar data, based on the FMCW range measurement technique has been presented. This allows the development of an estimation framework in the raw, high dimensional, measurement space, as opposed to the reduced dimensional post-processor, detection space. This is useful for two reasons.

1) Observations can be predicted for the purpose of data association, based on predicted vehicle manoeuvres. This is a fundamental component of stochastic autonomous navigation techniques, which requires the spatial prediction and then detection of targets, for vehicle pose and map estimation.

2) The methods presented can be adopted to simulate realistic radar scans, in the presence of noise, for which different feature extraction algorithms can be tested, allowing the user to control target RCS values and the corrupting sensor noise parameters.

Qualitative comparisons of predicted and actual scans were shown in the results, however quantified comparisons provide more insight into the prediction quality. Therefore, a metric, based on the Pearson Correlation coefficient was utilised and demonstrated, which provided quantified comparisons, for validating the prediction tech-

2.6 Conclusion

nique. To test the concept, predictions of scans were made at progressively larger distances from the scan used to generate the predictions. Even at a distance of 30m, robust feature prediction was demonstrated.

Further research, in which spectral saturation and clutter modelling would be useful for refining the model. This work is part of the authors' efforts in applying scanned millimetre wave radar to probabilistic autonomous robot navigation in outdoor environments.

Chapter 3

A Constant False Alarm Rate

Processor based on Statistical

Similarity for Automatic Feature

Detection

3.1 Introduction

This chapter deals with design and implementation of an adaptive landmark detection system for radar based robot navigation. The requirements for a feature detection scheme are to reject clutter (originating mainly from foliage, bush and uneven land terrain) and to resolve closely lying features. The clutter rejection of the standard CFAR processor at ground level are not satisfactory, as shown in the appendix A, and hence a new CFAR processor is proposed here. This chapter introduces a new composite

3.1 Introduction

CFAR processor and discusses its design, implementation and analysis in detail. The proposed processor is termed as a Kolmogorov-Smirnov¹ (KS) CFAR processor which is tested on received power-range spectra obtained from the 77 GHz FMCW scanning radar.

While performing radar based robot navigation in outdoor environments, multiple, closely lying features and clutter presence (unwanted reflections, originating mainly from land terrain) are often encountered. Techniques are required to extract the true target distances automatically from the observed radar spectra which are corrupted by noise. Estimating the relative distances of features, while rejecting land clutter reflections, for the successful localisation of mobile platforms, is the main goal of this chapter.

To extract features from radar spectra, previous methods have used a constant threshold (the power return to exceed a certain threshold which gives the object's distance) [16]. Constant threshold based feature detection assumes that the noise samples in the spectra are independent and identically distributed (*i.i.d*). However, the use of a constant threshold in a realistic environment results either in a degradation in detection performance (missed detections) or an excessive increase in false alarm rate (FAR) as a consequence of the data's deviation from the *i.i.d* assumption [46]. Therefore, using an adaptive threshold with the constant false alarm rate (CFAR) property is essential for automatic feature detection in radar [73]. The constant false alarm rate (CFAR) processor involves performing target detection by comparing spectra power returns to an adaptive threshold such that a CFAR is maintained [19]. For achieving this task, a CFAR processor, dynamically determines a detection threshold by estimating the local background noise and/or clutter power and multiplying this estimate by a scaling constant based on the desired probability of false alarm, P_{FA} . Hence CFAR processor performance is maintained in an unknown environment where the background noise/clutter power in the spectra are varying [31]. The most common CFAR processor is the Cell Averaging (CA) CFAR processor, where an average of samples from leading and lagging windows are used to obtain the background noise estimate [31]. The detection performance of the CA-CFAR processor is optimum (maximizes detection probability) in a homogeneous background when the reference cells contain independent and identically distributed (*i.i.d*) observations governed by the background noise

¹The CFAR processor utilises two-sample KS tests, hence the name.

3.1 Introduction

distribution [31], [8] (The radar used in this work has an exponential distribution during target absence, as shown in chapter 2). However, in real situations, the CA-CFAR processor performance deteriorates in cluttered environments and multiple target situations [31]. Also, bins in the spectra obtained from radar are highly correlated which again violates the noise assumptions. Various modifications have been applied to improve the detection performance of the CFAR processor in the above mentioned situations. In order to preserve a constant FAR property at clutter edges, the Greatest-Of (GO) CFAR processor has been proposed [36]. In the GO-CFAR processor the greater average of either the leading or lagging windows are used. Although GO-CFAR can regulate the FAR at clutter transitions to some extent, CFAR loss¹ is higher compared to the CA-CFAR processor, resulting in target masking. The Smallest-Of (SO) CFAR processor can resolve closely lying features, which utilises the smallest average of the leading and lagging windows [85]. SO-CFAR detector performance deteriorates when the *i.i.d* assumption is violated, such as in the presence of *clutter transitions*² in the reference window. It has been reported in the literature that the Ordered Statistics (OS) CFAR processor can handle multiple features much better than the CA-CFAR processor [68]. The OS-CFAR processor can overcome the issue of masking, thereby resolving closely lying features [76]. However poor clutter rejection and false alarm occurrences makes it less attractive for mobile robot navigation. The Trimmed-Mean (TM) CFAR processor is a more generalised, improvement over the OS-CFAR processor which combines ordering and arithmetic averaging [27]. False alarm occurrences in TM-CFAR processors have been reduced compared to the OS-CFAR processors. The performance of the TM-CFAR processor is still not acceptable for outdoor navigation in mobile robotics due to its inability to reject land clutter to an acceptable level [79], which could possibly impede robot navigation.

To extract features under general and cluttered environments accurately, some CFAR processors such as CGO (Censored Greatest-Of) [4; 66], ACMLD (Automatic Censored Mean Level Detector) [39] and OSGO (Ordered Statistics Greatest-Of) [26; 67]

¹CFAR loss is the incremental Signal-to-Noise ratio (SNR) required by a given CFAR threshold estimation scheme to achieve a specified detection probability, P_d , over that required with known background interference.

²As the moving window of the CFAR processor slides past the clutter portion of the spectrum, a portion of the window may contain clutter and the rest may be occupied by noise. This will cause a sudden change in the average power inside the window, which violates the *i.i.d* noise assumption.

3.1 Introduction

have also been proposed. Each processor has some performance disadvantages in any one or all of the above mentioned situations. The performances of CGO and ACMLD detectors at the clutter power transition regions are almost similar to the performance of the GO-CFAR processor. However, for CGO, if the number of interferers exceeds a pre-defined limit, its detection performance degrades severely. ACMLD requires intense computation, due to its relatively complex algorithm [46]. Similarly, the OSGO processor requires a great deal of processing time to decide upon which of the two reference sub windows has to be selected. Furthermore, it suffers from more CFAR loss compared to the other CFAR processors mentioned earlier.

A composite CFAR processor approach is adopted in [79], known as the Variability Index (VI) CFAR processor which utilises the advantages of various standard CFAR techniques. The VI-CFAR processor is composed of the CA, GO and SO CFAR processors. A slight modification in the VI-CFAR architecture is shown in [35], where the SO is replaced by an OS-CFAR processor in the design. Threshold selecting rules are also modified in [35]. The VI-CFAR processor shows superior clutter rejection over other stand-alone CFAR processors [79]. However, detection performance degrades if the interfering targets are not confined to one single half of the reference window [35]. Also when the spectra bins are highly correlated, the performance of the VI-CFAR processor deteriorates considerably as shown in section 3.2.2.

CFAR processors reported to-date are based on various assumptions made on noise and/or clutter statistics (*i.i.d* assumptions). Rohling stated in [68] that, “improvements in target detection and clutter suppression over the present state of the art can be effected only by removing the simplifying assumptions step by step and introducing a more differentiating way of argumentation”.

In this chapter, some of the simplifying assumptions made in existing CFAR techniques are removed in order to develop a new CFAR processor that is more robust in cluttered environments compared to the existing ones. The assumptions about the homogeneity of the reference window environment are discarded and replaced with tests based on statistical similarity with real data from which inferences about the level of homogeneity in the reference sample can be made. The modification will then provide a clear indication of target as well as clutter presence. This information is then utilised for selecting a suitable CFAR processor for detecting/rejecting the features/clutter.

3.1 Introduction

The KS-CFAR processor explained here adopts the same strategy as a composite CFAR processor. In the KS-CFAR the typical sliding window approach in CFAR processors has been modified, as explained in section 3.2. It will be shown here that the KS-CFAR detector shows a more robust performance, over other standard CFAR detectors, in cluttered environments as well as multiple target situations.

The main highlights of the proposed KS-CFAR processor are the following:

- The KS-CFAR processor has a modified sliding window design which verifies the statistical similarity of the spectrum bins with the bins from a spectrum containing no target (data obtained by scanning an open space).
- The new window design accommodates real noisy data collected from the radar for similarity tests. The Kolmogorov-Smirnov¹ (KS), along with the Mean Ratio² (MR) tests³ are used to obtain an adaptive threshold to reliably detect features while having low false alarm rates.
- Results obtained from the KS-CFAR processor are compared with a composite VI-CFAR processor from a robot navigational perspective. It is shown here that the KS-CFAR processor has a superior performance over VI and other standard CFAR detectors in resolving multi-targets while rejecting clutter.

This chapter is organised as follows. The Kolmogorov-Smirnov (KS) CFAR processor architecture is shown in section 3.2, which explains the KS CFAR processor design. An overview of the KS similarity measurement is also given in the same section. Adaptive threshold generation methods of KS-CFAR processors are given in section 3.3. KS-CFAR processor results obtained in different scenarios (multiple target and clutter situations) are shown in section 3.4. Performance and results are discussed in section 3.5. Conclusions are made at the end of the chapter, summarising the advantages of the proposed CFAR processor in outdoor terrains.

¹The Kolmogorov-Smirnov test (often referred to as the KS test) is a goodness of fit test used to determine whether two underlying probability distributions differ based on finite samples. The two-sample KS test is one of the most useful and general nonparametric methods for comparing two samples, as it is sensitive to differences in both location and shape of the empirical cumulative distribution functions of the two samples.

²The mean ratio (MR) can be stated as the ratio between the sum of the power values in the upper window and the sum of the noise power values in the lower window.

³Based on the outcome of the KS and MR tests, different CFAR processors are selected as explained in later sections.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

For the KS-CFAR processor, the generation of an adaptive threshold is carried out by using the Kolmogorov-Smirnov (KS) and Mean Ratio (MR) tests. Statistical similarity of two populations based on the Kolmogorov-Smirnov statistic is carried out with a sample set from the spectrum under test, and a sample set obtained from the noise-only spectrum (spectrum obtained during target absence). The KS and MR hypothesis test results are used to obtain the adaptive threshold for the KS-CFAR processor. The KS-CFAR processor can resolve closely lying features while successfully rejecting clutter originating mainly from land terrain.

The KS-CFAR processor architecture is shown in figure 3.1. The sliding window technique is modified to accommodate actual radar noise-alone data (determined a-priori by pointing the radar towards the sky), in order to utilise two sample KS tests. One sample set will be from the spectrum under test while the other set is from noise-only spectra recorded a-priori as shown in figure 3.2. There are two sliding windows in this design. One sliding window (upper window) moves along the spectrum-under-test. The second sliding window (lower window) moves along the noise-only spectra. Each window is divided into leading and lagging halves. KS and MR test statistics are carried out between these upper and lower windows. It is to be noted here that in this design, separate KS tests are carried out for the leading and lagging windows. This is to detect feature/clutter presence in either or both windows, which will then trigger the selection logic (see figure 3.1 and this will be explained in detail in section 3.3) to choose an appropriate CFAR processor. The KS-CFAR processor consists of CA, GO and TM CFAR processors for feature detection. The GO-CFAR has an advantage in typical transition areas between pure noise and strong clutter or between two different clutter regions. However, it simultaneously reduces the sensitivity of target detection in a homogeneous clutter situation. This is a compromise, which always occurs in radar detection due to the variety of different noise and clutter background situations, which can occur in radar based navigation in outdoor terrains. It is a useful extension of CA-CFAR but shows the same characteristics in multiple target situations. As a stand-alone CFAR processor, the TM-CFAR processor has a superior performance compared to other CFAR processors [31], and is included in the proposed CFAR processor design.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

Figure 3.1: KS-CFAR processor block diagram. There are two main sliding windows in the KS-CFAR processor. The Upper Window contains the power values from the spectrum under test and the Lower Window contain power values from noise-alone spectrum, determined a-priori. The Upper Window is sub divided into Window A and Window B and the Lower Window is sub divided into Window C and Window D. The sum of the power values in Windows A, B, C and D are labelled as Σ_A , Σ_B , Σ_C and Σ_D respectively. The sum of power values in the Upper Window and Lower Window are given as Σ_{AB} and Σ_{CD} respectively.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

Figure 3.2: A spectrum containing only noise is obtained by pointing the radar swash plate towards empty space (pointing towards the sky).

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

The spectra samples obtained from the radar are highly correlated and are exponentially distributed as mentioned earlier. The spectrum in the upper window can have samples from different distributions during signal/clutter presence¹. Let vectors X_1 and X_2 of length $N/2$, be the spectra under test (upper window) and noise alone spectra (lower window) respectively for the leading window. For each potential value x in the upper window, the KS test compares the proportion of X_1 values less than x with proportion of X_2 values less than x . The two-sample KS test uses the maximum difference over all x values is its test statistic. For comparing two different cumulative distribution functions $P_0(x)$ and $P(x)$, where x is the power spectrum bin number, the KS statistic, D is given by equation 3.1.

$$D = \max(|P_0(x) - P(x)|) \quad (3.1)$$

where, $P_0(x)$ is the proportion of X_1 values less than or equal to x and $P(x)$ is the proportion of X_2 values less than or equal to x (*i.e.* P_0 is the Cumulative Distribution function (CDF) of the sample set from the spectra under consideration and P is the reference sample CDF.). *i.e.*, if the samples in the upper window have similar statistics (calculation is shown in the next section) as the samples in the lower window, then it is assumed that both the sample sets are from the same (exponential) distribution. The null hypothesis, H_0 is assumed to be true in this case, else hypothesis, H_1 is assumed to be true. *i.e.*

$$\begin{aligned} H_0 : P &= P_0 \\ H_1 : &\text{otherwise} \end{aligned} \quad (3.2)$$

The KS test is carried out in both the leading and lagging windows separately for obtaining an adaptive threshold for feature selection.

¹Samples from the noise only regions are assumed to be from an exponential distribution [41]. Power returns originating from clutter will have a different distribution. The reflections from features are assumed to follow a Rayleigh distribution.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

3.2.1 Kolmogorov-Smirnov Test Statistics

This section describes how a two sample statistical similarity test is conducted between the spectrum under consideration and a noise only spectrum. The aim of the two-sample KS test is to find whether the spectrum bins inside the upper window (bins from the spectrum-under-test) have similar statistics to that of the lower window, which contains bins from a noise alone spectrum. The KS statistical tests are incorporated into the CFAR design to pave the way for a “true” adaptive CFAR processing algorithm, which utilises real measurements, rather than the assumptions, of noise homogeneity in the reference window.

The Kolmogorov-Smirnov (KS) test is applicable to distributions that are functions of a single independent variable, *i.e.*, to data sets where each data point can be associated with a single parameter [41]. The idea behind this test is first to obtain the cumulative distribution functions (CDF) of the two distributions that need to be compared. The Kolmogorov-Smirnov test statistic, D is a particularly simple measure, which is defined as the maximum value of the absolute difference between two cumulative distribution functions [65].

The Kolmogorov-Smirnov distribution is the distribution of the random variable given by,

$$K = \max |B(t)| \quad (3.3)$$

where $B(t)$ is the “Brownian bridge”¹ and $t \in [0, 1]$.

The statistic D , given in equation 3.1, can then be mapped on to a probability of similarity between the two distributions by the function that calculates the significance as follows [65].

The cumulative distribution function of K is given by,

$$Q_{KS}(x) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2 x^2} \quad (3.4)$$

¹Brownian bridge is a continuous-time stochastic process whose probability distribution is the conditional probability distribution of a Wiener process, $B(t)$ (a mathematical model of Brownian motion) given the condition that $B(0) = B(1) = 0$.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

where $Q_{KS}(x)$ is a monotonic function. The approximate probability of similarity, P [86] is given by

$$P \approx Q_{KS}([\sqrt{N_e} + 0.12 + \frac{0.11}{\sqrt{N_e}} D]) \quad (3.5)$$

where N_e is the effective number of FFT data values from the radar spectrum obtained as,

$$N_e = \frac{N_1 N_2}{N_1 + N_2} \quad (3.6)$$

where N_1 and N_2 are the FFT data values in the signal and noise distributions respectively in either the leading or lagging windows of size $N/2$ (here $N = 44$). The decision to reject the null hypothesis occurs when the significance level, P_{KS} equals or exceeds the value of P in equation 3.5. The value of the significance level, P_{KS} is chosen to be 0.05. *i.e.* the decision to reject the null hypothesis occurs when the significance level, P_{KS} , equals or exceeds the P -value (given in equation 3.5).

3.2.2 Mean Ratio Statistics Calculation

This section shows the mean ratio calculation for the KS-CFAR processor which will be utilised, along with the KS results, for different CFAR processor selection. Since the KS-CFAR processor is a composite CFAR processor, its results are compared to a similar approach, such as the Variable Index (VI) CFAR processor [79], in order to show its merits.

The Mean ratio (MR) can be stated as the ratio between the sum of power values in the upper window (spectrum under test) and the sum of noise power bins in the lower window.

$$MR_{KS} = \frac{\bar{X}_{AB}}{\bar{X}_{CD}} = \frac{\sum_{i \in AB} X_i}{\sum_{i \in CD} X_i} \quad (3.7)$$

Figure 3.3 shows a spectrum containing only land clutter. The MR values obtained

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

Figure 3.3: MR values obtained from the VI and KS CFAR processors are plotted along with the spectrum under test. The spectrum contains reflections from land (land clutter) as well as noise. The value of the MR obtained from the VI-CFAR processor does not change significantly in the presence of clutter, which will affect the detection process. The MR values are lower during clutter transitions due to the VI-CFAR processor design. However, the proposed design of the KS CFAR processor reflects the clutter presence efficiently with higher MR values.

3.2 Kolmogorov-Smirnov CFAR Processor Design Overview

from the VI-CFAR¹ and KS-CFAR processor design are also plotted. From the figure, it can be noted that, the MR values obtained from the KS-CFAR processor indicates a change during clutter where as the MR values of the VI-CFAR processor misclassifies clutter with no significant change in their values. This change in the MR is then utilised for selecting an appropriate CFAR processor in the KS-CFAR processor. As the MR values obtained from the KS CFAR processor reflect the absolute clutter presence, the results obtained are superior to those of the VI-CFAR processor, thereby providing more robust feature detection during clutter and multiple target situations. Figure 3.4 shows a spectrum containing a low RCS feature. The MR values obtained from VI-CFAR and the KS-CFAR processor are plotted in the same figure. The MR values obtained from KS-CFAR indicates a feature presence where as the VI-CFAR missed the feature.

Figure 3.4: MR values obtained from the VI and KS CFAR processors are plotted along with the spectrum under test. The spectrum contains reflections from a low RCS object (tree). The value of the MR obtained from the KS-CFAR processor changes in the presence of the small RCS feature, indicating a feature presence. However, the VI CFAR processor missed the feature as the MR values did not change significantly.

¹The Mean Ratio (MR) statistic in the VI-CFAR processor is the ratio of the sum of the power value's bins in the leading and lagging windows.

3.3 KS-CFAR Processor - Adaptive Threshold Generation and Performance Analysis

3.3 KS-CFAR Processor - Adaptive Threshold Generation and Performance Analysis

The results of the Kolmogorov-Smirnov (KS) and Mean Ratio (MR) tests are used to generate an adaptive threshold as shown in table 3.1. These test results will determine the reference cell selection used to estimate the background power and also the multiplying scale factor, α . The scale factor could be $\alpha_{N/2}$ or α_N . When both the windows are selected, the scale factor, α_N is used. $\alpha_{N/2}$ is used when either the leading or lagging window is selected. The background noise mean estimate is multiplied by the scale factor, to form the adaptive threshold. The sum of power bins in the leading window and lagging windows are represented by Σ_A and Σ_B respectively. When the full (upper) window is selected, the sum of the power spectral values is given by Σ_{AB} . For obtaining the threshold for the TM-CFAR processor, first the power spectral bins inside the full window are sorted in ascending order. Then the smallest and largest values are removed from the sorted power bins, X . This is represented as $\sum_{i=\eta_1+1}^{N-\eta_2} X(i)$ where η_1 and η_2 are the lower and upper trimming values. *i.e.* all the power bins below $\eta_1 + 1$ and above $N - \eta_2$ are trimmed from the sorted data, X . The scale factor for the CA, GO and TM CFAR processor are calculated as shown in [31].

It is to be noted here that table 3.1 has only six cases based on possible KS and MR test outcomes¹. Two cases are omitted from the table as it has been observed that when either of the leading or lagging windows have spectrum bins that are nonhomogeneous, then their MR values will always be different.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

The following analysis explains the KS-CFAR processor performance during clutter and multiple target situations.

¹With three variables, there can be eight possible combinations.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Table 3.1: KS-CFAR processor adaptive detection threshold summary.

Leading Window Variable?	Lagging Window Variable?	Different Means?	KS-CFAR Processor Adaptive Threshold	Equivalent CFAR Method
No	No	No	$\alpha_N \Sigma_{AB}$	CA-CFAR
No	No	Yes	$\alpha_{N/2} \max(\Sigma_A, \Sigma_B)$	GO-CFAR
Yes	No	Yes	$\alpha_{N/2} \Sigma_B$	CA-CFAR
No	Yes	Yes	$\alpha_{N/2} \Sigma_A$	CA-CFAR
Yes	Yes	No	$\alpha_{TM} \sum_{i=\eta_1+1}^{N-\eta_2} X(i)$	TM-CFAR
Yes	Yes	Yes	$\alpha_{TM} \sum_{i=\eta_1+1}^{N-\eta_2} X(i)$	TM-CFAR

3.4.1 Homogeneous (or Uniform) Noisy Environments

If the outcome of the KS hypothesis tests are such that each half reference window is declared non-variable and they also show no change in the mean ratios in both the windows, the procedure decides that the environment is homogeneous and therefore the KS-CFAR processor will select the whole reference window. The selection algorithm will choose the whole window for background estimation according to the first row of Table 3.1.

3.4.2 Cluttered Environments

Clutter mainly arises due to reflections from land terrain. Next, an analysis of the transition of clutter bins through the leading and lagging windows is explained. This will help in understanding how the adaptive thresholding is implemented in the KS-CFAR processor. As the clutter first enters the lagging window, one or more cells will contain a higher power, based on the clutter-to-noise ratio (CNR). The probability of similarity starts changing, which will eventually declare the distribution of bins inside the upper window as different from the lower window (noise) bins. The mean ratio, MR of the leading and lagging windows will also be different and the selection algorithm will choose the window with the greatest mean to prevent excessive false alarms when the test cell contains noise. This corresponds to the third row of Table 3.1.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

As the clutter fills both half windows, each will appear homogeneous and the combined window will be selected as in the first row of Table 3.1. Finally, as the clutter moves into the leading window, the leading window bins will appear to be nonhomogeneous and the lagging window will contain homogeneous noise. In this case, the selection algorithm will choose the homogeneous window for background estimation according to the fourth row of Table 3.1.

Figure 3.5 shows a spectrum containing land clutter at approximately 20 metres along with the probability of similarity plot (from equation 3.5) in the leading and lagging windows, obtained from the KS-CFAR processor. In figure 3.5, the probability of similarity values (from equation 3.5) approaches 1.0 at around 20 metres, indicating clutter presence.

MR statistics for the same spectrum shown in figure 3.5 are shown in figure 3.6. The MR changes during clutter as well as feature presence due to the change in sample statistics in the window, and is as shown in the figure.

3.4.3 Multiple Target Situations

When closely lying features occupy some of the reference window cells in either leading or lagging windows, two possible situations may arise. The first one corresponds to the presence of interfering targets in *one* of the two half windows (leading or the lagging). In this case, the background noise/clutter selection logic will choose the other half for background estimation. This is because, the mean power values of the two windows (leading and lagging windows) will be different and the selection algorithm will choose the window with the greatest mean to prevent excessive false alarms to occur when the cell-under-test contains noise. The second situation arises if the interfering targets (*i.e.* targets that are separated at a distance less than or equal to half the window width) are present in *both* half windows. The probability of similarity, P (equation 3.5) will indicate nonhomogeneous environments in both the leading and lagging windows. In this case the TM-CFAR detector is used for feature detection. This situation corresponds to the fifth and sixth row of KS-CFAR processor selection table (table 3.1). However in situations such as a feature reflection buried inside clutter

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.5: Probability of similarity plots obtained from KS CFAR processor (equation 3.5) along with the spectrum. The spectrum contains clutter at approx. 20 metres along with two features at approximately 100 and 125 metres. Note that the power axis not to scale, in order to accommodate the probability of similarity of leading and lagging windows. The probability of similarity values (from equation 3.5) approaches 1.0 at around 20 metres, indicating clutter presence.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.6: Mean ratio (MR) values obtained from the KS-CFAR processor, plotted along with the spectrum.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

power¹, in order to reject clutter, the KS-CFAR may try to choose the window with the greatest mean power to prevent false alarms. This results in missed detections when the cell-under-test contains the reflection from the feature. This scenario is shown and discussed later in the results section.

Figure 3.7 shows two closely lying features (27 and 33 metres approx. from the radar) and the corresponding probability-of-similarity plot (in the same figure) obtained from the KS test statistics (equation 3.5). The probability-of-similarity plot does not have any significant change due to the presence of closely lying features. However, a feature presence is detected by the Mean Ratio (an increase in the Mean Ratio values) as shown in figure 3.8.

Figure 3.9 shows the adaptive threshold and features detected by KS-CFAR for the closely lying features (same spectrum as shown in figures 3.7 and 3.8).

Figures 3.9 and 3.10 show the KS-CFAR processor performance for two different scenarios (figure 3.9 shows a spectrum containing two closely lying features and figure 3.10 shows a spectrum containing land clutter along with other features). The KS-CFAR processor can successfully reject clutter as shown in figure 3.10. The performance when there are closely lying features has improved compared to VI-CFAR processor. This is shown in figure 3.9 where two closely lying features are successfully resolved. The development of the KS-CFAR processor is a first step in achieving successful feature detection for autonomous navigation in outdoor environments.

Figure 3.9 show the performance of VI and KS CFAR processors when the spectrum contains two closely lying targets and low clutter level from land terrain. As explained above, the KS-CFAR processor compares the spectrum under test with a reference noise alone spectrum (obtained offline from a controlled environment). Thus, when closely lying targets are present in the leading or lagging reference windows, the corresponding KS statistics will indicate a nonhomogeneous or variable environment presence. If this situation occurs for one of the leading or lagging windows, but not both, then the other half window is chosen for background estimation using the CA-CFAR algorithm, resulting in correctly detecting the closely lying features as shown in figure 3.9. However, when the low clutter power (from land terrain) first enters into

¹for example, a spectrum containing two closely lying reflections, such as a reflection from foliage followed by a reflection from an object such as tree or lamp-post. When the targets and clutter reflections fill up one half of the sliding window and the cell-under-test respectively, the KS-CFAR processor will try to reject clutter by choosing the window with the greatest mean power to prevent false alarms.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.7: Hypothesis values obtained from the KS-CFAR processor for two closely lying targets.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.8: MR statistics obtained from the KS-CFAR processor compared with the VI-CFAR processor.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.9: The figure shows the KS-CFAR processor performance on a range spectrum with two closely lying features. The KS-CFAR processor successfully resolves the two features where as the VI-CFAR processor misses it.

3.4 KS CFAR Processor in Cluttered and Multiple Target Environments

Figure 3.10: The figure shows the KS-CFAR processor performance on a range spectrum compared with the CA-CFAR processor. The spectrum contains two features along with land clutter at a range of 20 metres.

3.5 KS-CFAR Processor Performance Assessments

the sliding window, the KS statistics could not correctly detect the non-homogeneity, resulting in a false alarm as shown in figure 3.9.

In the case of VI-CFAR processor, the VI statistics is calculated between leading and lagging windows. In the event that when one of the window contains multiple targets, and the other contains reflections from clutter (low power reflections from land terrain), then the VI statistics will indicate that both half windows are nonhomogeneous. So the split window with the minimum mean will be chosen for background estimation using the SO-CFAR algorithm [79]. This results in decreased P_d loss resulting in missing the two closely lying targets, which is shown in figure 3.9.

3.5 KS-CFAR Processor Performance Assessments

The composite technique adopted in this KS-CFAR processor switches between CA, GO, and TM CFAR processors when either nonhomogeneous (*i.e.* clutter) or closely lying features are encountered. The KS-CFAR detector compares the power return in the cell-under-test to the adaptive threshold obtained as explained in section 3.3. As the decision to switch to any of the CFAR processors are mutually exclusive, the target detection probability can be given as shown in equation 3.8.

$$\begin{aligned}
 P_{dKS} = & P(CA - CFAR) * P_{dCA} \\
 & + P(GO - CFAR) * P_{dGO} \\
 & + P(TM - CFAR) * P_{dTM}
 \end{aligned} \tag{3.8}$$

where P_{dCA} , P_{dGO} and P_{dTM} are the detection probabilities of the CA, GO and TM CFAR processors respectively. $P(CA - CFAR)$, $P(GO - CFAR)$ and $P(TM - CFAR)$ are the probabilities of the respective CFAR processors being selected by the decision logic (table 3.1). The probability of randomly selecting any of these three CFAR processors in the presence of noise has been found empirically by feeding 1000 noise-only spectra (collected from the radar by pointing the swash plate towards open sky) into the KS-CFAR processor. The respective selection probability values obtained

3.5 KS-CFAR Processor Performance Assessments

are 0.83, 0.16 and 0.01 for CA, GO and TM CFAR processors. These values along with the corresponding detection probabilities are used to obtain the P_d for the KS-CFAR processor.

Figure 3.11: Probability of detection, P_d vs. SNR plot of CFAR processors in a homogeneous environment.

Figure 3.11 shows the probability of detection, P_d in a homogeneous environment for the KS-CFAR processor (equation 3.8 along with results for an optimal (Neyman-Pearson) detector, CA-CFAR, GO-CFAR and TM-CFAR processors. All of the CFAR processors have some amount of CFAR loss relative to the optimal detector. The KS-CFAR processor performance is slightly lower compared to the CA-CFAR processor in a homogeneous environment. This is because, in the KS-CFAR real noise data is used in the design, which is correlated. This gives a value which is lower compared to the CA-CFAR processor design based on ideal homogeneous noise assumptions.

3.5 KS-CFAR Processor Performance Assessments

However, the KS-CFAR processor has a high P_d for a given SNR compared to the GO and TM CFAR processors as shown in figure 3.11.

3.5.1 Parameter Selection for Optimising KS-CFAR Processor Performance

The window width and false alarm probability values can be chosen to obtain the optimal KS-CFAR performance in outdoor terrains. Tests have been carried out in different outdoor scenarios with different values for the window width, N and false alarm probability, P_{fa} values. One of the scenes where the tests have been carried out is shown in figure 3.12. The outdoor environment contains features such as vehicles, buildings, walls etc. A radar scan is obtained from the area for fine-tuning the proposed KS-CFAR processor. The aim is to process the radar scan with the proposed KS-CFAR processor, with varying window widths, N , to determine an approximately optimal value for N in terms of feature detection sensitivity and clutter rejection. The KS-CFAR processor is then applied to the data with different window widths and false alarm probability values. Points marked in red are received power values which have exceeded the relevant CFAR thresholds and are considered to be detected features. Figure 3.13 show the KS-CFAR processor detection result with the settings, window width, $N = 22$ and $P_{fa} = 1 \times 10^{-6}$. There is a high number of false alarms in the result as shown in figure 3.13. The false alarms are due to the weak reflections that originate from a metallic mesh (having low RCS values) which is behind the parked vehicles. The reflected power to the radar from the metal mesh however has a similar RCS to that of objects such as small tree trunks. Extracting these low RCS objects from clutter in outdoor terrains is a difficult task as missed detections may occur.

Figure 3.14 shows the KS-CFAR processor result with a window width, $N = 66$ and probability of false alarm, $P_{fa} = 1 \times 10^{-6}$. The results show that there are less false alarms compared to the previous setup. There is however a missed detection due to a higher threshold selection¹. The missed detection of a metallic feature is marked

¹The value chosen for the window width, N is 66, which is higher than the previous KS-CFAR processor implementation with $N=22$ (figure 3.13). The adaptive threshold will increase as the window width increases, thus missing small RCS targets.

3.5 KS-CFAR Processor Performance Assessments

Figure 3.12: Photo of the test site from where the radar scan is obtained for KS-CFAR processor tuning. Features include vehicles, buildings and walls.

Figure 3.13: Features detected by the KS-CFAR processor with window width, $N = 22$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.

3.5 KS-CFAR Processor Performance Assessments

as a missed feature figure in 3.14 which was detected in figure 3.13.

Figure 3.14: Features detected by the KS-CFAR processor with window width, $N = 66$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.

Figure 3.15 shows the KS-CFAR processor result with window width, $N = 44$ and false alarm probability, $P_{fa} = 1 \times 10^{-6}$. The result shows less false alarms compared to the result with the window width, $N = 22$, while successfully detecting all the reliable features, compared to the result obtained with a window width of, $N = 66$, where a missed detection occurred.

Changing the false alarm probability, (in this case from P_{fa} from 1×10^{-6} to 1×10^{-8}), does not have much effect in the KS-CFAR processor result¹. The KS-CFAR processor parameters, after analysing the results with different window widths and false alarm probability values, are fixed as, $N = 44$ and $P_{fa} = 1 \times 10^{-6}$. These values are used in the KS-CFAR processor design, to adaptively extract features from outdoor

¹The adaptive threshold will be more sensitive to change in signal and/or noise level in the spectrum as P_{fa} decreases, resulting in a higher number of false alarms.

3.5 KS-CFAR Processor Performance Assessments

Figure 3.15: Features detected by the KS-CFAR processor with window width, $N = 44$, and false alarm probability, $P_{fa} = 1 \times 10^{-6}$.

3.5 KS-CFAR Processor Performance Assessments

terrains in experimental comparisons with other CFAR processors, in the following sections.

3.5.2 KS-CFAR Processor Performance in Outdoor Terrains

In order to verify the performance of the proposed KS-CFAR processor, data sets were collected from various outdoor environments. Figures 3.16 and 3.17 show the scans collected from an outdoor field (car park). Important features in the environments are labeled in the figures. The KS-CFAR processor detections are also shown in the figures (marked as black crosses).

Figure 3.16: Features detected by the KS-CFAR processor (marked in black crosses) are plotted on a 360° radar scan. A 180° degree laser range scan (marked as green crosses) was recorded at the same time and from the same vehicle location. Features include metallic posts, buildings, pillars, trees etc. Features detected by the KS-CFAR processor are plotted in green with the laser scan as shown.

The radar was mounted on top of a mobile platform and allowed to scan the environment. The radar scan obtained from the outdoor car park (figure 3.16) contains features such as trees, concrete pillars, buildings etc. There is clutter presence, originating mainly due to bush/foilage, as can be seen in the figure. This is a typical outdoor scenario where the CFAR processor is supposed to reject the clutter information which

3.5 KS-CFAR Processor Performance Assessments

Figure 3.17: 360° radar scan from a car park along with 180° sick laser scans. Features include trees, road curbs, different concrete structures such as buildings and pillars etc. Features detected by the KS-CFAR processor (plotted in green dots) are also shown in the picture.

is not useful for robot localisation. The features such as the trees and the buildings were successfully detected by the proposed KS-CFAR processor. However, the bush is not detected by the processor since it would provide limited useful information for robot navigation (which is the desired result). This is due to the spread of the spectra which is typical of clutter. Figure 3.18 shows a spectrum obtained at 278° which is taken from the car park. The reflections from the foliage are spread across the range axis¹ which is a typical characteristic of a clutter reflection. Although most of the CFAR processors are designed to reject clutter, in reality, mainly due to the violations of the homogenous noise assumptions, most of the CFAR processor incorrectly label clutter as features. Although the KS-CFAR processor has a superior clutter rejection, in typical natural environments, some false alarms (especially from land clutter, as will be shown in the later sections) may occur. In figure 3.17, laser scans originating from clutter (bush) are marked where the KS-CFAR processor has successfully rejected it.

Analysing the KS-CFAR processor result in figure 3.17 shows that all of the important features in the environment are detected. Clutter due to reflections from bush

¹ A typical point feature will have a pointed peak in the FFT power spectrum, while clutter will have its power spread over several power bins, along the range axis.

3.5 KS-CFAR Processor Performance Assessments

Figure 3.18: A spectrum at 278° in the radar scan shown in figure 3.16. The reflection from the bush/foliage appears to be spread across the range and will be rejected by the KS-CFAR processor.

3.5 KS-CFAR Processor Performance Assessments

at 11 metres, 270° and also at 13 meters, 295° to the radar, are successfully rejected by the KS-CFAR detector. Laser reflections from the bush are marked in the figure. Other standard CFAR processors will falsely report these clutter as detected features.

The overall performance of the proposed KS-CFAR processor is excellent compared to other standard and well known CFAR processors for land navigation as will be shown later in this section. A more in depth analysis of the proposed KS-CFAR processor under different navigational scenarios will be shown next. This is carried out in order to analyse the KS-CFAR processor's clutter rejection ability, while simultaneously extracting reliable features for navigation. The KS-CFAR processor results will be compared with a standard OS-CFAR processor to analyse the merits of the proposed processor.

In order to further analyze the proposed KS-CFAR processor performance, tests have been carried out in urban and off road environments. The satellite image of a test environments containing both off-road and urban sections (both sections are in close proximity to one another) is shown in figure 3.19. The urban section of the environment contains features such as parked vehicles, buildings, metallic objects such as posts, containers etc. The off road terrain contains features such as bush, mud banks, trees, metallic posts, fences and walls.

Figure 3.20 shows the snapshot of the urban area from which the radar scan is obtained. The features include a metallic container, parked vehicles, buildings etc. The radar data is used to compare the performance of the proposed KS-CFAR processor against the standard OS-CFAR processor¹. The values used for OS-CFAR processor design are, window width, $N=44$ and $P_{fa}=1 \times 10^{-6}$. The results are shown in figures 3.21a and 3.21b respectively. The features detected by the processors are shown as red dots, superimposed on to the raw radar scan. Analysing the OS-CFAR processor result shows few false alarms along with other detections which are marked in figure 3.21a. Other features such as buildings, metallic objects, parked vehicles etc. are successfully detected by both processors. There are only a few false alarms in the OS-CFAR result compared to the KS-CFAR processor. This is mainly due to the nature of the scene from where the radar scan is obtained. The terrain is flat (road surface) and hence there are no reflections from land clutter. The majority of the objects in the scene

¹It is reported in the literature that OS-CFAR processor has an excellent clutter rejection capability [68], which is why it is selected for comparison here.

3.5 KS-CFAR Processor Performance Assessments

Figure 3.19: Satellite image of the test area from which the two different data sets - one from the urban road and the other from the off-road - are collected for analysing the proposed KS-CFAR processor performance.

3.5 KS-CFAR Processor Performance Assessments

are metallic, that provides strong reflections. The results in figure 3.21 demonstrate the performance of both the OS and KS CFAR processors in near “ideal” conditions, where the signal reflections are high enough to be detected in the absence of clutter. Thus most of the objects were successfully detected by both OS and KS CFAR processors. Results from the OS and KS CFAR processors obtained from other terrain conditions will be shown next.

Figure 3.20: Photo of the test area from where the radar scan is taken. The typical urban environment include features such as parked vehicles, buildings etc.

Figure 3.22 shows a scene which consists of a garage with metallic walls. The roof of the garage is supported by two rows of metal pillars as seen in the figure. The scenario is selected to test the ability of the proposed detector in resolving multiple targets. Figure 3.23a shows the results from the OS-CFAR processor. The detector has some false alarms occurring due to spectra saturation. The OS-CFAR processor output also has some false positives (false alarms) which lie behind the metallic wall. Those

3.5 KS-CFAR Processor Performance Assessments

(a) OS-CFAR processor result from an urban environment. Features include parked vehicles, metal container etc. False alarms are marked in the figure.

(b) KS-CFAR processor result from an urban environment. Features include parked vehicles, metal container etc. False alarms due to multi-path effects are also marked.

Figure 3.21: Figures of OS and KS CFAR processor results. The radar scan is obtained from the robot location at $[0,0,\pi/2]$. Features include vehicles, metallic container etc.

3.5 KS-CFAR Processor Performance Assessments

false positives occur due to multi path reflections. Figure 3.23b shows the output from the KS-CFAR processor. There are less false alarms in the KS-CFAR processor output compared to the OS-CFAR processor. As the false alarms due to multi path reflections have the characteristics of clutter (as shown in figure 3.23a), where the power return spreads across the range, the KS-CFAR processor manages to successfully reject it. The false alarms due to multi path reflections are also absent in the KS-CFAR processor. However, one of the metal posts is missed by the KS-CFAR processor. Further analysis reveals that the reflection from the missed pillar is very weak due to partial occlusion by another pillar. Analysis further reveals that the missed detection occurs only when there is a high power return from the first object (*i.e.* normally from any metallic objects) and a low power return from the second object, which is due to occlusion. All the other pillars were successfully detected by the KS-CFAR processor while maintaining low clutter and false alarm levels compared to the standard OS-CFAR processor.

The vehicle was next moved off-road, where the typical features include bush, foliage, mud banks, tree trunks, sign posts and a wall. Figure 3.24 shows the image taken from the off-road track. The results from the OS-CFAR and KS-CFAR processors are shown in figures 3.25a and 3.25b respectively. For navigation, foliage is not a reliable feature and hence is marked as clutter. As the adaptive feature detectors are designed to reject clutter (such as the reflections occurring from land terrain and foliage) the comparative tests in this environment will show the clutter rejection capabilities of the CFAR processors. Results from the OS-CFAR processor are shown in figure 3.25a. The OS-CFAR processor shows false alarms occurring due to noisy bins (due to sudden changes in power values inside the sliding window, mainly due to clutter transitions) and also clutter arising from bush and land terrain. The output from the KS-CFAR processor detects less of the clutter from foliage/grass and from land terrain. This shows the superior clutter rejection capability of the proposed KS-CFAR processor. Also features such as a sign post and off-road curbs (mud banks) have been successfully detected by both the detectors.

Figure 3.26 shows a scene off-road again where the prominent features include walls and road curbs, along with the reflections arising from foliage/bush and uneven terrain. The results from the OS-CFAR processor, given in figure 3.27a show clut-

3.5 KS-CFAR Processor Performance Assessments

Figure 3.22: Photo of the test area from where the radar scan is taken. The environment includes features such as metallic object (shed), iron pillars, a building etc.

3.5 KS-CFAR Processor Performance Assessments

(a) Features detected by the OS-CFAR processor which include metallic object (shed) and pillars, along with false alarms due to multi path.

(b) Features obtained by the KS-CFAR processor, that include metallic wall of a shed, metal pillars. A missed detection occurs due to occlusion of a pillar by the other, which is also marked.

Figure 3.23: Figure of OS and KS CFAR processor results from an outdoor terrain. Features include a metallic wall (shed), metal pillars and a building.

3.5 KS-CFAR Processor Performance Assessments

Figure 3.24: Photo of the test area from where the radar scan is taken. The typical off-road environment include features such as road curbs, bush/foliage, sign post, compound wall etc.

3.5 KS-CFAR Processor Performance Assessments

(a) OS-CFAR processor result obtained from the off-road terrain. There is a large amount of clutter in the radar scan, originating both from land and foliage. The OS-CFAR processor failed in rejecting most of it.

(b) KS-CFAR processor result from an off-road terrain. Most of the clutter was successfully rejected by the KS-CFAR processor.

Figure 3.25: Figure of OS and KS CFAR processor results from off-road terrains. The KS-CFAR processor is successful in rejecting most of the land and foliage clutter compared to the standard OS-CFAR processor.

3.5 KS-CFAR Processor Performance Assessments

ter arising from land and foliage. The clutter rejection of the KS-CFAR processor is superior compared to the OS-CFAR processor as shown in figure 3.27b.

Figure 3.26: Photo of the test area from where the radar scan is taken. The typical environment include features such as wall, road curbs, bush and foliage.

Figure 3.28 shows another off-road scenario where more reflections arise from foliage/bush and land terrain. The clutter rejection capability of the proposed KS-CFAR processor is also tested against the standard OS-CFAR processor. The results from the OS-CFAR processor, shown in figure 3.29a, incorrectly identified clutter (due to grass/bush) as features. This is marked in figure 3.29a. Other features such as road curbs (made of concrete and rock sediments) have also been detected by the processor. The results from the proposed KS-CFAR processor are less sensitive to clutter and false alarms compared to the standard OS-CFAR processor, hence rejecting the reflections from foliage. The wall, however, is missed by the KS-CFAR processor. This is due to the bush/foliage in front of it, which partially occludes the wall there by reducing the power returns from it. The result shows that the performance of the KS-CFAR processor deteriorates in certain environments. There will be a performance

3.5 KS-CFAR Processor Performance Assessments

(a) OS-CFAR processor result from an off-road terrain. Features include wall, road curbs, bush and foliage.

(b) KS-CFAR processor result from an off-road environment. Features include wall, road curbs, bush and foliage.

Figure 3.27: Figure of OS and KS CFAR processor results from an off-road environment. Features include wall, road curbs, bush and foliage.

3.5 KS-CFAR Processor Performance Assessments

degradation if the interfering targets are not confined to one single half of the sliding window, resulting in selecting the TM-CFAR processor (see table 3.1, row 5 and 6). The clutter/interference (due to foliage and land terrain) were missed by the censoring procedure in the TM-CFAR processor [31] resulting in the missed detection of the feature (wall), along with the clutter. For future work, it is proposed that the results will be further improved by further fine-tuning the trimming constants, η_1 and η_2 of the TM-CFAR processor (table 3.1) especially considering the case where both half of the sliding windows are determined to have a feature/clutter. The results from the KS-CFAR processor, given in figure 3.29b, show that the off-road curb (made of concrete and rock sediments) has been successfully detected by the processor, while rejecting clutter from the land and bush.

Figure 3.28: Photo of the test area from where the radar scan is taken. The typical urban environment include features such as foliage/bush and land terrain.

Qualitative tests have been carried out to analyse the performance of the proposed

3.5 KS-CFAR Processor Performance Assessments

(a) OS-CFAR processor result from an off-road terrain. Features include foliage/bush and land terrain.

(b) KS-CFAR processor result from an off-road terrain. Features include foliage/bush and land terrain.

Figure 3.29: Figure of OS and KS CFAR processor results obtained from a radar scan obtained from an off-road terrain. Features include foliage/bush and land terrain.

3.6 Conclusion

KS-CFAR processor in different outdoor scenarios. The results have been compared against a standard OS-CFAR processor to find out the merits and draw backs. The test results reveal that the proposed KS-CFAR processor has a superior clutter rejection and fewer false alarms compared to the well known OS-CFAR processor. Feature misses can only happen at very specific scenarios for the KS-CFAR processor such as the one shown in figure 3.23b, where any feature detector may fail to identify the feature that is either fully or partially occluded. This makes the proposed KS-CFAR processor an ideal adaptive feature detector for outdoor vehicle navigation with MMW radar.

3.6 Conclusion

This chapter introduced a composite CFAR processor known as the Kolmogorov-Smirnov (KS) CFAR processor. The KS-CFAR has a high detection probability during multi target situations and a good clutter rejection ability. Tests conducted with the radar data show that the proposed KS-CFAR processor is computationally less intense compared to the OS-CFAR processor. In the KS-CFAR processor, which has a composite design, the CA, GO and TM CFAR processors are adaptively selected, based on the feature/clutter presence. The TM-CFAR processor, which is a modified OS-CFAR processor, is the most computationally intense processor (due to the sorting method [31]) in the KS-CFAR processor. The TM-CFAR processor will be selected only if one or more interfering targets are present in the leading or lagging windows, as explained earlier. The KS-CFAR processor switches to either the CA or GO CFAR processor in other target/clutter situations. Thus the overall computational complexity of the proposed KS-CFAR processor is slightly less intense compared to the standard OS-CFAR processor.

The Goodness-of-Fit (GoF) CFAR processor [62], which also utilise KS statistics for feature detection, is designed specifically for Weibull and lognormal clutter. The proposed GoF detector structures use the assumed PDF of clutter, and if the assumed and actual clutter distributions are different, both the P_d and P_{fa} of the detector will change. One way to solve the issue of change in false alarm probability, P_{fa} is by using the samples of actual clutter to find the suitable threshold. This value of P_{fa} will remain unchanged regardless the skewness of clutter. However, the detector will still

3.6 Conclusion

suffer from some detection loss. This phenomenon is a weakness of the GoF-CFAR detection method.

The individual CFAR processors, in the KS-CFAR processor design, are assumed to have an exponential noise distribution during target absence, as given in chapter 2. For all standard CFAR processors' performance will deteriorate once the distribution of clutter, present in the spectrum, deviates from the assumed ones. KS-CFAR processor is made up of the CA-CFAR, GO-CFAR and TM-CFAR processors and therefore when each detector is running the same assumptions apply. However, the KS-CFAR detector can have an improved detection robustness, since it monitors changes in clutter and changes in detector accordingly. Results indicate that the new KS-CFAR technique performs well in non-stationary clutter environments of varying size, with a slight detection loss (missed detection), and points to the feasibility of computationally efficient processor for outdoor robot navigation based on FMCW radar.

To process a single radar spectrum of 200 metres, the total CPU time taken by the KS-CFAR processor in Visual C++ was found to be 0.0344 seconds, where as the standard OS-CFAR processor took 0.0359 seconds to process the same spectrum. For processing a full 360° scan (containing 170 spectra) of 200 metres range, the KS-CFAR processor implementation in Visual C++ took approximately 0.78 seconds, which includes the time to load one full radar scan data into the memory, the time to process (feature extraction) the data and, the time to unload/clear the data from the memory. This processing time is almost the same as that of the standard TM-CFAR processor for the full 360° radar scan. Further analysis has been carried out to obtain the computational load of the KS-CFAR processor at various target scenarios. When there is clutter presence, the CFAR processor selects the TM-CFAR processor based on the MR and KS calculations. The TM-CFAR processor requires more computation compared to the CA and GO CFAR processors, due to the sorting and summing section in its design. So while the vehicle manoeuvres in off-road terrains, the selection logic (table 3.1) often switches between the CA and TM-CFAR processors in order to achieve clutter removal arising from foliage and ground terrain.

The disadvantages of the CFAR processors reported in the literature is that, all the implementations are based on the assumptions that noise samples are *i.i.d.* Hence, the standard CFAR processors provide good results in simulations. However, all the standard CFAR processor implementations will have more CFAR loss with real radar

3.6 Conclusion

data due to the violation of the *i.i.d* noise assumption. A new composite CFAR processor design is introduced here which accommodates real noisy spectra for carrying out statistical similarity tests in order to set an adaptive threshold. The CFAR processor which is selected however, is tuned based on *i.i.d* assumptions. In future work, a rigorous analysis should be carried out for accommodating the correlation of spectral samples to exclude noise assumptions in CFAR processor design.

Chapter 4

Feature Detection based on Target Presence Probability

4.1 Introduction

The previous chapter introduced an adaptive Constant False Alarm Rate (CFAR) processor based on statistical similarity, termed as the Kolmogorov-Smirnov (KS) CFAR processor. In the KS-CFAR processor, the average noise power is estimated online based on the power bins inside a moving window. This estimated noise, along with a scale factor, forms an adaptive threshold. The results of the KS-CFAR processor show improved clutter rejection compared to other known CFAR processors in off-road terrains. The proposed KS-CFAR processor decides feature presence/absence on a per spectrum basis. This chapter explains an algorithm which makes optimal estimates of the range to multiple targets down range, for each range spectra based on received signal-to-noise power. This is referred to as feature detection based on target presence probability (TPP). In TPP based feature detection, the feature presence/absence is decided after considering a selected number (> 1) of spectra, instead of a single spectrum (as in standard CFAR processors, including the proposed KS-CFAR proces-

4.2 Target Presence Probability

sor). This improves feature detection in cluttered environments, as shown in the results section. The noise estimate in the radar spectra is obtained by averaging past power spectral values for the same azimuth, using a smoothing parameter that is adjusted by the target presence probability. The target presence probability is determined by the ratio between the local power of the noise and its minimum within a specified number of spectra. Results are shown which compare probability based feature detection with other feature extraction techniques such as constant threshold on raw data and Constant False Alarm Rate (CFAR) techniques including the KS-CFAR processor proposed in chapter 3. The results show the merit of the proposed algorithm which can detect features in typically cluttered outdoor environments with a higher success rate compared to other feature detection techniques.

This chapter presents a method to automatically detect features based on their probability of presence. The following are the main contributions of this chapter:

- The TPP based feature detector is introduced, which estimates local noise by recursively averaging the previous power spectra, using a smoothing parameter that is adjusted by the estimated target presence probability in spectra bins.
- TPP based feature detection can quickly adapt to the changes in the spectra (which may occur due to clutter).
- Evaluation of TPP based feature detection is performed under various outdoor scenarios. It is shown here that the TPP detector is superior compared to other adaptive feature detectors, particularly in detecting low RCS features.

4.2 Target Presence Probability

For typical outdoor environments, the radar cross section (RCS) of objects may be small. The reflected signals will be corrupted by noise. For reducing the noise and extracting smaller signal returns along with the higher power returns, a method is now introduced which uses the probability of target presence [17] for feature detection [42]. This method is appealing compared to CFAR and constant threshold methods at ground level, as a threshold can be applied to the TPP.

4.2 Target Presence Probability

4.2.1 The Target Presence Probability Algorithm

The detection problem described here can be stated formally as a binary hypothesis testing problem [84]. Feature detection can be achieved by estimating the noise power contained in the range spectra, by averaging past spectral power values and using a smoothing parameter. This smoothing parameter is adjusted by the TPP in the range spectra. The TPP is obtained by taking the ratio between the local power of the range spectra containing noise and its minimum value. The noise power thus estimated can then be subtracted from the range spectra to give a reduced noise range spectra.

Let the power of the noisy spectra be smoothed by a first order recursive averaging technique:

$$\check{P}(k, l) = \alpha_s \check{P}(k, l-1) + (1 - \alpha_s) \dot{P}(k, l) \quad (4.1)$$

where α_s is a weighting parameter ($0 \leq \alpha_s \leq 1$)¹, $\check{P}(k, l)$ is the smoothed spectrum bin, $\dot{P}(k, l)$ is the raw spectrum bin, k is the bin number of the spectrum and l is the spectrum number. First the minimum and temporary values of the local power are initialised to $P_{min}(k, 0) = P_{imp}(k, 0) = \check{P}(k, 0)$. Then a range spectrum wise comparison is performed with the present spectrum l and the previous spectrum $l - 1$.

$$P_{min}(k, l) = \min \{P_{min}(k, l-1), \check{P}(k, l)\} \quad (4.2)$$

$$P_{imp}(k, l) = \min \{P_{imp}(k, l-1), \check{P}(k, l)\} \quad (4.3)$$

When a predefined number of range spectra have been recorded at the same vehicle location, and the same sensor azimuth, the temporary variable, $P_{imp}(k, l)$ is re-initialised² as

¹ An analysis into the choice of α_s and other TPP parameters will be given in section 4.3

² The noise power calculation is re-initialised after every 4 iterations. This procedure controls the mean noise power, preventing an upward bias, in heavily cluttered environments (uneven terrains).

4.2 Target Presence Probability

$$P_{min}(k, l) = \min \{P_{imp}(k, l-1), \check{P}(k, l)\} \quad (4.4)$$

$$P_{imp}(k, l) = \check{P}(k, l) \quad (4.5)$$

Let the signal-to-noise power, $P_{SNP}(k, l) = \frac{\check{P}(k, l)}{P_{min}(k, l)}$ be the ratio between the linear¹ local noisy power value and its derived minimum. A Bayes minimum-cost decision rule (explained in appendix C) is given by

$$\frac{p(S_r|H_1)}{p(S_r|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P(H_0)(C_{10} - C_{00})}{P(H_1)(C_{01} - C_{11})} \quad (4.6)$$

where $P(H_0)$ and $P(H_1)$ are the *a priori* probabilities for the speech absence and presence, respectively, and C_{ij} is the cost for deciding H_i when H_j .

In the Neyman-Pearson theory of binary hypothesis testing, one must decide between a null hypothesis and an alternative hypothesis. A level of significance (known as the test of size) is imposed on the false alarm (type I error) probability, and a test must satisfy this constraint while minimizing the miss detection (type II error) probability, or equivalently, maximizing the detection probability. The Neyman-Pearson lemma specifies necessary and sufficient conditions [84] for the most powerful test of size, provided the *likelihood ratio* is a monotonic function (explained in appendix C) [50],[64], the decision rule of 4.6 can be expressed as

$$\frac{p(P_{SNP}|H_1)}{p(P_{SNP}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \delta \quad (4.7)$$

where δ is a threshold and H_0 and H_1 designate hypothetical target absence and presence respectively. $p(P_{SNP}|H_0)$ and $p(P_{SNP}|H_1)$ in equation 4.7 are the conditional probability density functions. The decision rule of equation 4.7 can be expressed as

¹after removing the high pass filter and logarithmic amplification effect from the power spectra

4.2 Target Presence Probability

$$P_{SNP}(k, l) \underset{H_0}{\overset{H_1}{\gtrless}} \delta \quad (4.8)$$

An indicator function, $I(k, l)$ is defined where, $I(k, l) = 1$ for $P_{SNP} > \delta$ and $I(k, l) = 0$ otherwise. The estimate of the conditional TPP¹, $\hat{p}'(k, l)$ is

$$\hat{p}'(k, l) = \alpha_p \hat{p}'(k, l-1) + (1 - \alpha_p) I(k, l) \quad (4.9)$$

This TPP can be used as a target likelihood within a mobile robot navigation formulations. α_p is a smoothing parameter ($0 \leq \alpha_p \leq 1$). The value of α_p is chosen in such a way that the probability of target presence in the previous range spectrum has very small correlation with the next range spectrum.

It is of interest to note that, as a consequence of the above analysis (calculated recursively from the smoothed spectral power bins (equations 4.1-4.9)), the noise power, $\hat{\lambda}_d(k, l)$ in k -th range spectrum is given by

$$\hat{\lambda}_d(k, l) = \tilde{\alpha}_d(k, l) \hat{\lambda}_d(k, l-1) + [(1 - \tilde{\alpha}_d(k, l))] \check{P}(k, l) \quad (4.10)$$

where

$$\tilde{\alpha}_d(k, l) = \alpha_d + (1 - \alpha_d) \hat{p}'(k, l) \quad (4.11)$$

and α_d is a smoothing parameter ($0 \leq \alpha_d \leq 1$). This can be used to obtain a noise reduced spectrum, $\hat{P}_{NR}(k, l)$ using the method of power spectral subtraction [10]. In the basic spectral subtraction algorithm, the average noise power, $\hat{\lambda}_d(k, l)$ is

¹conditioned on the indicator function $I(k, l)$ [34].

4.3 Analysis of the TPP based Feature Detection Algorithm

subtracted from the noisy range spectrum. To overcome the inaccuracies in the noise power estimate, and to keep the power estimate within the working range of the radar, the following method can be used [9].

$$\hat{P}_{NR}(k, l) = \begin{cases} \check{P}(k, l) - c \times \hat{\lambda}_d(k, l) & \text{if } \check{P}(k, l) > c \times \hat{\lambda}_d(k, l) \\ d \times \hat{\lambda}_d(k, l) & \text{otherwise} \end{cases} \quad (4.12)$$

where c , is an over-subtraction factor ($c \geq 1$) and d is spectral floor parameter ($0 < d < 1$). The values of c and d are empirically determined for obtaining an optimal noise subtraction level at all ranges and here are set to be 4 and 0.01. Although a reduced noise range spectrum can be useful in other detection methods, the TPP estimate (equation 4.9), will be demonstrated further in the results here.

4.3 Analysis of the TPP based Feature Detection Algorithm

There are various parameters in the Target Presence Probability (TPP) algorithm, which critically determine the smoothing and detection process. Here, an analysis is carried out to find out the effects of these parameters. This is essential for the accurate tuning of the filter to obtain the best detection results in outdoor environments.

In equation 4.1, a fraction of the current spectra bin is added to the complementary fraction of the last spectra bin, *i.e.*, each power value from the old spectra has less and less influence as it recedes into the past. α_s in equation 4.1, is a smoothing parameter which controls this and whose value can be anywhere in between 0 and 1. Technically, data smoothing is a form of low pass filtering, which means that it blocks out the high frequency noise components in order to emphasise the low frequencies. The higher the value of α_s , the greater the smoothing will be. It is to be noted that, excessive smoothing will suppress the important frequency components in the spectrum, thereby reducing the detection ability.

The optimisation procedure for TPP based feature detection consists of the following steps. First, power spectra are collected from a controlled environment with the

4.3 Analysis of the TPP based Feature Detection Algorithm

Figure 4.1: A radar spectrum obtained from a controlled environment. The environment contains two features, a radar reflector of 10 m^2 RCS at 10.5m and a wall at approx. 137m. The corresponding range values obtained from a Reigl laser range finder (range accuracy = 0.04m) are also shown in the figure (as inverted shaded triangles ▼). The distance measurements obtained from the laser range sensor are considered as the reference/ground-truth.

number of features known a priori. The data thus collected contains reflections from two features. The first feature is a 10 m^2 reflector at 10.5 metres and the another is a building at 137 metres¹. The radar power spectrum thus obtained is shown in figure 4.1.

Proper validation of TPP based feature detection necessitates radar spectra with known ground-truth data obtained from a laser range finder, which will be compared to the outputs of the proposed algorithm in order to compute the errors. The Reigl laser range sensor used in the test has a range accuracy of 0.04m. The ground-truth measurements (thus obtained from the laser), along with the radar spectrum are shown in figure 4.1. The radar spectra, along with the laser reference data, are used to fine-tune the TPP parameters to obtain the best possible results, (*i.e.* fine tuning the TPP parameters to obtain feature presence probability values close to 1.0 and for the feature absence values close to 0).

First, an initial value for α_s is chosen, usually the smallest value the parameter can

¹Due to beam-width of the radar, a portion of the transmitted signal hits the first feature and reflects back to the receiver while the remaining portion of the transmitted signal continue to traverse till it gets reflected back from the second target which will then give the second “peak”.

4.3 Analysis of the TPP based Feature Detection Algorithm

Figure 4.2: Figure show an ideal probability output generated with the knowledge of ground truth information. This will be the output if the TPP algorithm were perfect.

take (which is 0.01 here), and the target presence probability is calculated for the given signals. The TPP result is compared against the ideal case, which is shown in figure 4.2. The value for α_s is increased at 0.01 steps and the resultant probability of feature presence is obtained. The resultant probability plot is compared against the ideal one to obtain a measure of closeness. The value for α_s , which makes the probability plot close to the ideal output, is selected for TPP algorithm. The chosen value is discussed in the results in section 4.4.

Another recursive smoother is implemented in equation 4.9. The smoothing parameter, α_p can have a value anywhere in between 0 and 1. Recursive smoothing is carried out with the previous target presence probability value and with the indicator function. The value of α_p is selected to reduce the correlation between previous and present probability values as mentioned in the previous section. The value of α_p is incremented from the initial value 0.01 in steps of 0.01. The resultant probability output will be then compared against the ideal output and the best value for α_p is chosen, once the error between the probability plot and the ideal plot is minimal.

Figure 4.3 shows TPP plots obtained for different values of α_s and α_p . Analysing the plots, shows that, for smaller values of α_s and α_p , the smoothing does not prevent noise components from entering into the subsequent stages. The target presence probability plot appears closer to the optimal output (shown in figure 4.2) as the values of

4.3 Analysis of the TPP based Feature Detection Algorithm

both α_s and α_p are closer to 1. This is as shown in figure 4.3e, where the probability plot resembles the ideal output plot. By empirical analysis, the values for α_s and α_p are fixed as 0.99 and 0.95 respectively. There is a third peak in the resultant output which is at about 155 metres. This peak is also present in the raw radar signals and is due to a multi path reflection.

The smoothing parameters, α_s and α_p are independent of radar characteristics and can have the same values as given in the thesis even with a different scanning radar. A low δ value in equation 4.7 increases the SNR gain, thereby increasing the sensitivity of the target presence probability. This will degrade the TPP calculation resulting in an increase in false positives (high probability (> 0.5) during target absence). A higher value of δ results in missed detections (low probability values for land marks). The calculation of the threshold value, δ in equation 4.7 is set to be 4.0, and is shown in appendix C. P_{tmp} is re-initialised every 4 spectra as given in equation 4.5. The TPP based feature detection is tuned to receive a minimum of 28 spectra¹ at every azimuth for successful feature detection.

The TPP based feature detector has a structure of a batch filter and hence it requires more than one scan to accurately detect targets/features. As the radar used here is a slow scanning one (scanning rate is 2 Hz), the robot has to stop and collect sufficient data in order to perform the TPP based detection. The robot can have a continuous motion with TPP detection if the radar has a higher scanning rate (> 30 Hz) from the present 2 Hz.

It is to be noted here that the TPP based detection results can be repeated with any other MMW radar, and in any type of environments, as the threshold, δ is the only sensor dependent parameter. In order to tune the TPP algorithm, collect the radar spectra from a controlled environment, with few features at known locations and no clutter present. Then these signals have to be processed offline using TPP to find the optimal value of the threshold, δ . Once the value of δ is found offline, it can be used for TPP based detection in real-time.

¹The minimum number of spectra required was found offline, with the spectra collected in controlled environments.

4.3 Analysis of the TPP based Feature Detection Algorithm

(a) Figure show the target presence probability with $\alpha_s = 0.11$ and $\alpha_p = 0.99$.

(b) Figure show the target presence probability with $\alpha_s = 0.55$ and $\alpha_p = 0.99$.

Figure 4.3: Analysis of TPP based detection for different parameters of α_s and α_p .

4.3 Analysis of the TPP based Feature Detection Algorithm

(c) Figure show the target presence probability with $\alpha_s = 0.99$ and $\alpha_p = 0.11$.

(d) Figure show the target presence probability with $\alpha_s = 0.99$ and $\alpha_p = 0.55$.

(e) Figure show the target presence probability with $\alpha_s = 0.99$ and $\alpha_p = 0.95$.

Figure 4.3: Analysis of TPP based detection for different parameters of α_s and α_p .

4.4 Results

In this section, a comparison of the proposed TPP technique with constant threshold methods and CA-CFAR will be shown. First, comparisons of complete processed spectra, with no thresholding applied, will be given. Second, range scans which result from applying thresholds to raw data and TPP data will be compared and finally, CA-CFAR and the KS-CFAR processor estimated range scans will be compared with the TPP algorithm.

4.4.1 Comparison of TPP with Raw Data Thresholding

In figure 4.4 a noisy radar range spectrum (figure 4.4a), the corresponding estimated target presence probability (figure 4.4b, from equation 4.9) and the reduced noise range spectra (figure 4.4c, from equation 4.12) have been plotted.

In figure 4.4a, the range spectrum contains three distinct peaks of differing power values, whereas the TPP plot (figure 4.4b) shows the three peaks with a more uniform range width and similar probabilistic values. This result shows that although the returned power values vary from different objects, the corresponding TPP values are similar. Note that it is not necessary to subtract any bias signal from the received range spectrum.

As a by-product of this algorithm, figure 4.4c is the result of the spectral subtraction method (equation 4.12) which could be used, instead of the raw range spectra, with constant threshold detection methods. However, the selection of targets based on TPP, remains the focus of this chapter.

The TPP based feature detector is easier to interpret as shown in figures 4.5, 4.6 and 4.7 where TPP plots are shown along with the corresponding raw radar data.

Figures 4.5a and 4.6a show the raw radar data obtained in an indoor sports hall and outdoor sports field respectively. The corresponding TPPs are shown in figures 4.5b and 4.6b respectively. In figure 4.5b, the four walls of the stadium are clearly obtained by the proposed algorithm. The other probability values at higher ranges arise due to multi-path effects. The satellite image of the outdoor environment (outdoor stadium), shown in chapter 2, is reproduced here (figure 4.8) for convenience. The clutter shown

4.4 Results

Figure 4.4: Received, noisy range spectrum, TPP and spectral subtraction, noise reduced spectrum versus range.

4.4 Results

(a) Power versus range of a two-dimensional radar scan from an indoor environment.

(b) TPP versus range of a two-dimensional radar scan in an indoor environment. The probability of the detected targets (i.e. walls) are shown in the figure.

Figure 4.5: Raw radar data and corresponding target presence probability plots obtained from an indoor sports hall.

4.4 Results

(a) Power versus Range of a two-dimensional radar scan from an outdoor environment.

(b) TPP versus Range of a two-dimensional radar scan in an outdoor environment. The probability of the targets detected (i.e. radar reflectors, wall and tree) are shown in the figure.

Figure 4.6: Raw radar data and corresponding TPP obtained from an outdoor environment (stadium) shown in figure 4.8.

4.4 Results

in figure 4.6b is obtained when the radar beam hits the ground due the unevenness of the field surface.

Figure 4.9 show a car park environment from where a full 360° radar scan is obtained. Salient features in the environment are marked in the picture. Figure 4.7a shows the full radar scan obtained from the car park. The corresponding target presence probability plot is shown in figure 4.7b. Selected Features, labelled in figure 4.9 are marked in the TPP figure (figure 4.7b) along with the detection of other objects in the environment. Analysing the raw scans obtained from the radar, it is evident that feature detection in typical outdoor environments is a difficult task, as the environment contains features with different radar cross section (RCS) values. Feature detection based on target presence probability is comparatively easier, as will be shown in the next section. There is no clutter rejection mechanism in the TPP based feature detector. Hence the 'clutter' detected by the TPP based feature detectors are considered here as probable features.

For actual feature selection, a comparison of constant threshold, CA-CFAR and thresholded TPP methods will now be given. To apply a constant power threshold to a received range spectrum, four different threshold values, $\mu + 1\sigma$, $\mu + 2\sigma$, $\mu + 3\sigma$ and $\mu + 4\sigma$ are used. μ and σ are the range dependent mean and standard deviation obtained by averaging a large number of range spectra by keeping the radar swash plate stationary, pointing towards the sky where there is no reflection.

The TPP algorithm is initially compared to the constant threshold detection algorithm in an indoor environment to demonstrate the effects of varying the power thresholds in the constant power threshold algorithm. The merits of the proposed algorithm are shown in figures 4.10a - 4.10d where plots obtained using different power thresholds applied to raw radar range spectra are shown and compared with a single threshold (0.8) applied to the target presence probability plot. Here, a complete 360° radar sweep of an indoor badminton court with a single open door at coordinates (0, -10) was taken. Ground truth was hand measured, and coincides with the dominant four walls shown in the figures. The figures show the comparison of two-dimensional plots obtained by choosing a suitable threshold applied to the raw radar data (labelled as "Constant Threshold") and selecting the closest range value to exceed a target presence probability threshold of 0.8 (labelled as "Target Presence Probability").

4.4 Results

(a) Power versus Range of a two-dimensional radar scan from a car park environment.

(b) TPP versus Range of a two-dimensional radar scan in a car park. The probability of the targets detected (i.e. wall, trees, lamp post etc.) are shown in the figure.

Figure 4.7: Raw radar data and corresponding TPP obtained from the car park.

4.4 Results

Figure 4.8: The satellite image of the outdoor stadium with significant features labelled.

4.4 Results

Figure 4.9: The car park environment with significant features labelled.

4.4 Results

(a) TPP based feature detection and constant threshold of $T = \mu + 1\sigma$ results.

(b) TPP based feature detection and constant threshold of $T = \mu + 2\sigma$ results.

(c) TPP based feature detection and constant threshold of $T = \mu + 3\sigma$ results.

(d) TPP based feature detection and constant threshold of $T = \mu + 4\sigma$ results.

Figure 4.10: The TPP based feature detection result is plotted with results obtained from four different constant threshold values. A constant power threshold is used to select the closest range value from a raw radar range spectrum. This is compared with the closest range value selected by applying a threshold of 0.8 to the TPP spectra.

4.4 Results

The result in figure 4.10a shows that a constant threshold detection technique, with the particular threshold $(\mu + \sigma)$, has a slightly better performance over the proposed TPP based feature detector. However, it should be noted here that, a “good” constant threshold value (in this case $\mu + 1\sigma$ dB) is difficult to predict, and in this particular case, was selected after analysing the returned power, obtained from the controlled environment. The results can significantly deteriorate if the threshold selected is different as shown in figures 4.10c or 4.10d.

In figures 4.10c and 4.10d, the constant threshold values are raised, resulting in more missed detections. The selection of a “good” threshold in an outdoor environment, where objects have significantly different RCS values, is difficult to determine. In such situations, feature detection based on a constant power threshold is likely to produce many missed detections and/or false alarms as will be shown in the outdoor results.

Further results conducted show the TPP of actual objects will be approximately the same and is found to be more than 0.8. Feature detection using TPP is then carried out by keeping a threshold at 0.8. The results shown in figures 4.5, 4.6, 4.7 and 4.10 show that TPP based feature detection is easier to interpret and, in general, produces less false alarms compared to constant threshold based feature detection in the typical indoor and outdoor environments tested [43]. This will be demonstrated further in chapter 6 where SLAM with TPP based feature detection and KS-CFAR processor based feature detection will be demonstrated.

4.4.2 Comparison of Target Presence Probability (TPP) based feature detection with OS-CFAR Processor Results

In OS-CFAR processors, the power spectral bins in the sliding window (width of the sliding window = N) are first arranged in ascending order and then the k th spectral bin (usually $k = 0.75 \times N$) is selected. The median spectral bin is a function of the adaptive threshold. The test window compares the adaptive threshold value with the power of the signal and classifies the cell content as signal or noise. In the standard OS-CFAR processor implementation, the window width, N and the false alarm probability, P_{fa}

4.4 Results

are selected to maximize detection probability, P_d , while rejecting clutter. In order to find the best OS-CFAR processor parameters for navigation in outdoor environments, results are obtained with different window widths and false alarm probability values. The results thus obtained show how critical the window width, N and the probability of false alarm, P_{fa} values are.

Figures 4.11, 4.12 and 4.13 show features obtained with the TPP method and the OS-CFAR processor technique from an indoor environment as used in figures 4.5 and 4.10, with different window widths ($N = 22, 44$ and 66) and false alarm probabilities ($P_{fa} = 1 \times 10^{-4}$, 1×10^{-6} and 1×10^{-14}). The targets obtained by the thresholded (0.8) TPP are shown by a solid line while the dotted lines are those obtained from the OS-CFAR based target detector.

Analysing the results obtained from the indoor stadium shows that as the window width, N becomes larger for a fixed false alarm probability, P_{fa} , the adaptive threshold tends to become less sensitive to the change in power spectral values, resulting in missed detections. The OS-CFAR processor results are shown in figures 4.13, for a fixed probability of false alarm value¹, $P_{fa} = 1 \times 10^{-14}$ with different window widths, N . Analysing results from three different window widths, N ($N=22, 44$ and 66) shows that, for $N=22$, the result is more sensitive to the change in spectral power changes compared to the result obtained by selecting the window width, $N=66$. The results obtained by selecting a lower false alarm probability, P_{fa} results in a higher detection probability, P_d for any given window width. This is evident from results shown in figures, 4.11b, 4.12b and 4.13b where, for a given window width, $N=44$, the probability of false alarm, P_{fa} selection yields different results. The lower the values of P_{fa} , the lower the missed detections will be. The TPP based feature detection, however, is based on a constant threshold applied against the target presence probability values. Since the probability values will be higher than 0.5 regardless of the target RCS values, applying a constant threshold on the TPP values will provide good estimates of feature distances in both indoor as well as outdoor situations.

The OS-CFAR processor was also applied in an outdoor environment (outdoor stadium) and compared with TPP based feature detector. For a fixed false alarm rate, results are obtained with three different window widths ($N=22, 44$ and 66). Two sets

¹the false alarm probability value is chosen to be small to show the effect of P_{fa} on the adaptive threshold.

4.4 Results

(a) TPP based feature detection and OS-CFAR processor result with $N=22$ and $P_{fa} = 1 \times 10^{-4}$.

(b) TPP based feature detection and OS-CFAR processor result with $N=44$ and $P_{fa} = 1 \times 10^{-4}$.

(c) TPP based feature detection and OS-CFAR processor result with $N=66$ and $P_{fa} = 1 \times 10^{-4}$.

Figure 4.11: TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen for the OS-CFAR processor to show the effects of varying window widths for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-4}$.

4.4 Results

(a) TPP based feature detection and OS-CFAR processor result with $N=22$ and $P_{fa} = 1 \times 10^{-6}$.

(b) TPP based feature detection and OS-CFAR processor result with $N=44$ and $P_{fa} = 1 \times 10^{-6}$.

(c) TPP based feature detection and OS-CFAR processor result with $N=66$ and $P_{fa} = 1 \times 10^{-6}$.

Figure 4.12: TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen to show the effects of varying window width for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-6}$.

4.4 Results

(a) TPP based feature detection and OS-CFAR processor result with $N=22$ and $P_{fa} = 1 \times 10^{-14}$.

(b) TPP based feature detection and OS-CFAR processor result with $N=44$ and $P_{fa} = 1 \times 10^{-14}$.

(c) TPP based feature detection and OS-CFAR processor result with $N=66$ and $P_{fa} = 1 \times 10^{-14}$.

Figure 4.13: TPP based feature detection and OS-CFAR processor results. Three different window widths, $N = 22, 44$ and 66 are chosen to show the effects of varying window widths for a fixed false alarm probability, $P_{fa} = 1 \times 10^{-14}$.

4.4 Results

of results are obtained for $P_{fa} = 1 \times 10^{-6}$ and 1×10^{-14} to determine how the change in false alarm selection affects the OS-CFAR processor results in this different scenario. The TPP based feature detection and the OS-CFAR processor results are shown in figures 4.14 and 4.15. The TPP based feature results are marked with large dots while small dots represent features detected by the OS-CFAR processor. Analysing the results shows that for a smaller false alarm probability ($P_{fa} = 1 \times 10^{-14}$) and a moderately sized window ($N=44$), the OS-CFAR processor produces the best possible results in built and natural terrains. However, a close comparison of the TPP based feature detection and feature detections from a “well-tuned” OS-CFAR processor reveals that the TPP based detection is superior. Figure 4.16 shows this comparison. The false positives and the missed detections by the OS-CFAR processor are marked in the figure.

From the above results it can be seen that target presence based feature detection has a superior performance to OS-CFAR detector in both of the indoor and outdoor environments tested. In figure 4.16, the OS-CFAR processor, failed to detect some point features (trees) as marked in the figure. This is due to the lower power returns received, resulting in missed detections. There are also false alarms, detected by the OS-CFAR processor. In the experiments conducted, TPP based feature detection produces a lower number of false alarms compared to the “well-tuned” OS-CFAR processor as is evident in the figure. The proposed algorithm for feature extraction appears to outperform the OS-CFAR method because the CFAR processor finds the noise locally, while the TPP based feature detection algorithm estimates the noise power by considering more than one range spectrum (equation 4.5). The TPP based feature extraction method, unlike the CFAR detector, is not a binary detection process as is shown in figure 4.4(c). This method of feature detection is useful as the feature representation is probabilistic.

4.4 Results

(a) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=22$ and $P_{fa} = 1 \times 10^{-6}$.

(b) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=44$ and $P_{fa} = 1 \times 10^{-6}$.

(c) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=66$ and $P_{fa} = 1 \times 10^{-6}$.

Figure 4.14: TPP based feature detection and OS-CFAR processor results obtained from outdoor terrains with a fixed false alarm probability, $P_{fa} = 1 \times 10^{-6}$.

4.4 Results

(a) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=22$ and $P_{fa} = 1 \times 10^{-14}$.

(b) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=44$ and $P_{fa} = 1 \times 10^{-14}$.

(c) TPP based feature detection and OS-CFAR processor result obtained from outdoor terrains with $N=66$ and $P_{fa} = 1 \times 10^{-14}$.

Figure 4.15: TPP based feature detection and OS-CFAR processor results obtained from outdoor terrains with a fixed false alarm probability, $P_{fa} = 1 \times 10^{-14}$.

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

Figure 4.16: TPP based feature detection plotted with OS-CFAR processor detections. Missed detections and false positives are marked. The result shows that TPP based feature detection shows a superior performance over the OS-CFAR processor in outdoor terrains.

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

Unlike the single-input-single-output CFAR algorithms which decide upon target presence/absence hypotheses based on a single power-range spectrum input, the proposed TPP method is a batch algorithm. The trade-off between batch vs. non-batch approaches is that non-batch approaches typically require a broad range of a priori distribution assumptions on both the noise and environmental clutter distributions (except for KS-CFAR processor). While noise signal distributions may be obtainable using offline learning methods, clutter signal distributions are environmentally dependent, and are therefore impossible to obtain a priori. For standard CFAR models which assume a given set of signal models, but from which the realistic models differ significantly, catastrophic failure is likely. For fully autonomous systems, the amount of a priori information requirement should be kept to a minimum, and therein lies the merits of the proposed TPP based detector and the KS-CFAR processor. This section analyses the performance of the TPP based detector in comparison to the proposed KS-CFAR detector. Due to its previously outlined advantages and lack of a priori noise and clutter

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

signal model assumptions, both TPP detector and KS-CFAR processor can outperform the non-batch approaches (*i.e.* all standard CFAR processor designs) especially in situations of (noise and clutter models which deviate from the assumed distributions) closely lying features and clutter signals. The comparison of TPP based detector and the KS-CFAR detector is demonstrated in this section in simulated trials. Figure 4.17 depicts the simulated raw data¹ which contain 50 superimposed spectra comprising noise, clutter and target signals (of low RCS).

Results from the KS-CFAR processor and a TPP based feature detector are shown in figures 4.18 and 4.19 respectively. It can be seen that despite the seemingly impressive results from the KS-CFAR detector, false alarms do persist in the clutter region due to power fluctuation (different distribution to the Swerling *I* model). The KS-CFAR processor, although it rejects clutter most of the times, sometime results in false alarms due to clutter power fluctuation. One instance of false alarm due to clutter is shown in figure 4.18. Analysing the result from the TPP based feature detection, shown in figure 4.19 reveals that the low RCS feature is successfully and continually detected while the clutter signals are simultaneously repressed. This demonstrates the advantages of TPP based feature detection in natural terrains, with no clutter modeling. This eliminates the requirement of a priori data needed about the environment (for clutter modeling), thus increases the robustness of localization and map building. The drawback of the method is that given its batch structure, it can be more computationally intensive than classical CFAR approaches.

To process 28 radar spectra (number chosen arbitrarily) of 200 metres, the total CPU time taken by the TPP based feature detector in Visual C++ was found to be 0.0365 seconds. For processing a full 360 degree scan (containing 170×28 spectra) of 200 metres range, the TPP detector implementation in Visual C++ took approximately 0.97 seconds, which includes the time to load one full radar scan data into the memory, the time to process (feature extraction) the data and, the time to unload/clear the data from the memory. This processing time is slightly higher than the proposed KS-CFAR processor (shown in chapter 3) for the full 360 degree radar scan. The above results show that the processing time of TPP based detector is comparable to that of any

¹The radar spectra is simulated using the method explained in chapter 2, FMCW Radar Operation - Spectra Analysis and Prediction for Robot Navigation

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

standard CFAR processors and can be implemented for real-time feature detection with any faster scanning radars.

Figure 4.17: A synthetic data set comprising fifty spectra with two different reflections are shown. The first reflection originates from environmental clutter whereas the second is from a feature of small RCS.

In order to compare the performance of the proposed feature detectors (the KS-CFAR processor and the TPP based feature detector) in real environments, radar spectra are obtained from a heavily cluttered environment (outdoor environment with uneven terrains). The spectra contains reflections from an uneven terrain and a lamp post. The spectra are shown in figure 4.20. The lamp-post is approximately at 147 metres from the radar. Clutter reflections (reflections from uneven terrain) are present between 30 and 70 metres from the radar unit as shown in the figure. The power values of the spectral bins are highly fluctuating as shown in figure 4.20. The clutter statistics of land terrain is difficult to model, as mentioned earlier in chapter 2. As previously highlighted, in an unknown outdoor terrain, it is extremely challenging to obtain a priori distributions of the clutter signals. Given the poorly modelled signal distributions, as expected, the KS-CFAR output is inferior to the batch approach of TPP which can adaptively deal with unknown signal distributions. Thus the result obtained by the KS-CFAR processor contains few false alarms, along with a missed detection, as shown in figure 4.21. However, the TPP based feature detection successfully detects the real feature (lamp post at 147 meters) by with a higher probability of its presence. The

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

Figure 4.18: KS-CFAR feature detection output with the adaptive threshold is shown. The adaptive threshold successfully detects the small RCS feature. However, there is a false alarm due to the power reflections from clutter.

Figure 4.19: TPP based feature detection result is plotted. The result shows that TPP based feature detection have a good clutter rejection capability. The TPP based detector detects the small RCS feature successfully as shown.

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

clutter reflections has been assigned a lower probability compared to the real feature (lamp-post) as shown in figure 4.22. The results show that the TPP based feature detection offer a superior feature detection in outdoor terrains, with out any a priori clutter information.

Figure 4.20: A synthetic data of fifty spectra with two different reflections are shown. The first reflection originate from a clutter where as the second reflection is a feature with small RCS.

4.5.1 Comparison of TPP based Feature Detector with KS-CFAR

Processor in Batch Mode

In order to have a fair comparison of the TPP detector (which has a batch filter design) with the proposed KS-CFAR processor, the latter is redesigned to operate in batch mode (referred to as a “*batch* KS-CFAR processor” here). A *batch* KS-CFAR processor design will be explained here and its performance will be analysed against the TPP based feature detector in real world scenarios.

The spectra shown in figure 4.20 are used here for analysing the *batch* KS-CFAR processor performance. The *batch* KS-CFAR processor utilises a weighted recursive averaging filter. The recursive filter implemented in the *batch* KS-CFAR processor is

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

Figure 4.21: A standard KS-CFAR feature detection output with the adaptive threshold is shown. There are false alarms due to the reflections from land terrain. Due to large clutter presence in the spectrum, the real feature (lamp-post) at approx. 147 m is completely missed by the KS-CFAR processor.

Figure 4.22: TPP based feature detection result is plotted. The result shows that TPP based feature detection successfully detects the lamp-post. The TPP assign comparatively low probabilities (less than 0.5) to powers' that originated from land terrain.

4.5 Performance analysis of TPP based Feature Detector and KS-CFAR processor

shown in equation 4.13

$$P_{\mu}(k, l) = \varphi \times P_{\mu}(k, l - 1) + (1 - \varphi) \times P_r(k, l) \quad (4.13)$$

where P_r is the raw radar spectrum, P_{μ} is the smoothed (weighted average) power spectrum, k is the spectrum bin number, l is the spectrum number and φ is the weighting parameter ($0 \leq \varphi \leq 1$) of the filter. Notice that the equation in 4.13 uses two sources of data to calculate each point in the output, $P_{\mu}(k, l)$: points from the input, $P_r(k, l)$ and previously calculated points from the output, $P_{\mu}(k, l - 1)$, both multiplied by constants (*i.e.* weighting parameters) φ and $(1 - \varphi)$ respectively. This algorithm is simple and faster than other digital filters for several reasons. First, there are only three computations per point (two multiplications (with the constants), and one addition), regardless of the length of the filter. Second, the indexing scheme is very simple. Each index in equation 4.13 is found by adding or subtracting integer constants that can be calculated before the filtering starts. Third, the entire algorithm can be carried out with integer representation. Depending on the hardware/processor used, integers can be more than an order of magnitude faster than floating point variables.

Figure 4.23: The figure show the batch KS-CFAR processor results when applied to the smoothed spectrum. The only feature contained in the spectrum (lamp post) at approx. 147 metres is successfully detected by the processor along with two false alarms (due to land clutter).

The value of φ is selected to be 0.9 in order to reject high noise fluctuations in the

4.6 Conclusions

spectra. The output of the recursive average filter will be a single smoothed spectrum. The resultant smoothed spectrum is then fed to a standard KS-CFAR processor for feature detection. The result obtained from the KS-CFAR processor is shown in figure 4.23. The P_{fa} of the KS-CFAR processor is set as 1×10^{-6} and the window width, $N=44$.

The result in figure 4.23 shows that the *batch* KS-CFAR processor was able to successfully detect the lamp-post at 147 meters. This feature was missed when a standard KS-CFAR processor is applied as shown in the previous section (figure 4.21). Although the detection of the KS-CFAR processor has improved (with the *batch* filter design), the issues of false alarm occurrences still remains. Comparing the *batch* KS-CFAR processor result with the TPP based detection (shown in figure 4.22) shows that the performance of the TPP based detection is superior in heavily cluttered environments. Thus this analysis show the importance of the TPP based detector over the existing CFAR processors.

4.6 Conclusions

This chapter introduced a feature detector based on target presence probability. Results were shown which compared this method with other feature extraction techniques such as a constant threshold on raw data and Constant False Alarm Rate (CFAR) techniques. A difficult compromise in the CA-CFAR method is the choice of the window size which results in a play-off between false alarms and missed detections. Variants of the CFAR method exist, which can be tuned to overcome the problem of missed detections, but the problem of false alarms remains.

The TPP algorithm presented here, does not rely on adaptive threshold techniques, but estimates the probability of target presence based on local signal to noise power estimates, found from several range spectra. The technique is also insensitive to any non target noise biases present in the range spectra. The results show that the algorithm can detect features in the typically cluttered outdoor environments tested, with a higher success rate compared to the constant threshold and CFAR feature detection techniques. An a-priori information of noise and clutter statistics is not essential for

4.6 Conclusions

the TPP based feature detector, which makes it an attractive feature detector in outdoor terrains. Further methods of including the target presence probability of feature estimates into a SLAM formulation are investigated in chapter 6.

Chapter 5

An Augmented State SLAM

formulation for Multiple Line-of-Sight

Features with Millimetre Wave Radar

5.1 Introduction

The results of object penetration by radar waves shown in chapter 2 motivates further development of power spectra prediction with multiple line-of-sight features which is one of the contributions of this chapter. For feature based SLAM, it is necessary to predict the target/feature locations reliably, given a prediction of the vehicle/radar location. As radar can penetrate certain non-metallic objects it can give multiple range information. A method for accurately predicting range spectra from new robot positions, given an estimate of the vehicle and target states, is explained in this chapter.

A mobile robot SLAM problem is then formulated which estimates a state comprising the robot pose, 2D target positions, normalised target radar cross sections (RCSs),

5.1 Introduction

normalised absorption cross sections and the radar power loss, L . The term “Normalised” is used here to indicate that the effect of the actual RCS is incorporated into a reflectivity parameter, Υ_r . Normalisation results as it is assumed that the sum of this reflectivity parameter and the absorption and transmittance parameters is unity. Predicted observations are then formed using the predicted state, the radar equation, and a noise analysis, from previous work shown in chapter 3 and in [44]. These predicted observations take the form of predicted range spectra which are compared with actual received power/range readings from the radar [42], [43]. The radar is able to continuously sweep the radar wave through 360° in bearing, and acquire entire power-range spectra every 1.8° .

This chapter describes a new approach in predicting radar range spectra which is essential for simultaneous localisation and map building (SLAM) with millimetre wave radar. The following topics regarding the power spectra prediction for radar based SLAM are presented in this chapter.

- A SLAM formulation using an augmented state vector which includes the normalised radar cross sections and absorption cross sections of features, as well as the usual feature Cartesian coordinates, is introduced. This is intended to aid the data association process, so that features need not just be associated based on their Cartesian coordinates, but account can be taken of their estimated normalised reflection and absorption cross sections also.
- The second contribution is a predictive model of the form and magnitudes of the power-range spectra from differing vehicle locations, for multiple line-of-sight targets. This forms a predicted power-range observation, based on estimates of the augmented SLAM state.
- The formulation of power returns from multiple objects down-range is explained and predicted radar range spectra are compared with real spectra, recorded outdoors. This prediction of power-range spectra is a step towards a full, radar based SLAM framework.

This chapter is organised as follows. A method for predicting radar power spectra from new robot locations, given an estimate of the vehicle and target states, is explained

5.2 Radar Power Spectra Prediction

in section 5.2. An augmented state vector is introduced in section 5.3 where, along with the vehicle and feature positions, normalised radar cross sections and absorption cross sections of features are added together with the radar losses, L . Finally section 5.4 shows full predicted power spectra and the results are compared with the measured power spectra in the initial stages of a simple SLAM formulation. A full EKF SLAM implementation with the proposed spectra prediction is shown in chapter 6.

5.2 Radar Power Spectra Prediction

For SLAM, the measurements taken from the radar used here are the range, R , bearing, θ and the received power, P_R from the target at range R . One of the contributions of this chapter is to predict range spectra from new robot positions given an estimate of the vehicle and target states. A new augmented state vector is introduced here which, along with the usual feature coordinates x and y , contains that feature's normalised RCS, Υ_r and absorption RCS, Υ_a and the radar losses, L .

Figure 5.1: A 360° radar range spectra obtained from an outdoor field, containing a trees, non-metallic poles, fences and concrete walls. The received power value is represented in colour space, as shown by the right hand colour bar, with power units in dB.

5.2 Radar Power Spectra Prediction

To illustrate this, figure 5.1 shows a 360° radar scan obtained from an outdoor field. Objects in the environment consist of lamp-posts, trees, fences, and concrete steps. The radar penetrates some of the non-metallic objects (at 77 GHz the attenuation through paper, fibreglass, plastic, wood, glass, foliage etc. are relatively low, while attenuation through brick and concrete is high [28].), and can observe multiple targets down line (as shown in chapter 2). This is shown in figure 5.2, which is the received power versus range for the particular bearing of 231° marked in figure 5.1.

Figure 5.2: A radar range spectrum, recorded at the bearing angle 231° shown in figure 5.1, obtained from the outdoor field with multiple features down range.

Multiple targets down range can occur due to either the beam width of the transmitted wave intersecting two or more objects at differing ranges or due to penetration of the waves through certain objects. The radar used here is a pencil beam device, with a beam width of 1.8°. This means that multiple returns within the range spectra occur mostly due to penetration and occasionally due to feature occlusion. Therefore a model for predicting entire range spectra, based on target penetration is now given.

5.3 Radar Based Augmented State Vector

5.3 Radar Based Augmented State Vector

The state vector consists of the normalised radar cross section, Υ_r , absorption cross section, Υ_a and the radar loss constants, L , along with the vehicle state and feature locations. The variables, Υ_r , Υ_a and L are assumed unique to a particular feature/radar. Hence, this SLAM formulation makes the (very) simplified assumption that all features are stationary and that the changes in the normalised values of RCS and absorption cross sections of features when sensed from different angles, can be modeled using Gaussian random variables v_{Υ_i} ¹. This is a reasonable assumption only for small circular cross sectioned objects such as trees, lamp posts and pillars, however, as will be shown the method produces good results in semi-structured environments even for targets which do not conform to these assumptions.

5.3.1 Process Model

A simple vehicle predictive state model is assumed with stationary features surrounding it. The vehicle state, $\mathbf{x}_v(k)$ is given by $\mathbf{x}_v(k) = [x(k), y(k), \theta_R(k)]^T$ where $x(k)$, $y(k)$ and $\theta_R(k)$ are the local position and orientation of the vehicle at time k . The vehicle state, $\mathbf{x}_v(k)$ is propagated to time $(k + 1)$ through a simple steering process model [40].

The model, with control inputs, $\mathbf{u}(k)$ predicts the vehicle state at time $(k + 1)$ together with the uncertainty in vehicle location represented in the covariance matrix $\mathbf{P}(k + 1|k)$ [7].

$$\mathbf{x}_v(k + 1) = \mathbf{f}(\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k)) \quad (5.1)$$

$\mathbf{u}(k) = [v(k), \alpha(k)]$. $v(k)$ is the velocity of the vehicle at time k and $\alpha(k)$ is the

¹A Swerling I model [81] has been shown to be a good approximation for modeling the RCS of point features such as trees, lamp-posts etc, which are common features in outdoor terrains. [73]

5.3 Radar Based Augmented State Vector

steering angle. In full, the predicted state at time, $(k + 1)$ becomes

$$\begin{bmatrix} \hat{x}_v(k+1|k) \\ \hat{y}_v(k+1|k) \\ \hat{\theta}_v(k+1|k) \\ \hat{x}_{f_1}(k+1|k) \\ \hat{y}_{f_1}(k+1|k) \\ \hat{\Upsilon}_{r_1}(k+1|k) \\ \hat{\Upsilon}_{a_1}(k+1|k) \\ \vdots \\ \hat{x}_{f_i}(k+1|k) \\ \hat{y}_{f_i}(k+1|k) \\ \hat{\Upsilon}_{r_i}(k+1|k) \\ \hat{\Upsilon}_{a_i}(k+1|k) \\ \vdots \\ \hat{x}_{f_N}(k+1|k) \\ \hat{y}_{f_N}(k+1|k) \\ \hat{\Upsilon}_{r_N}(k+1|k) \\ \hat{\Upsilon}_{a_N}(k+1|k) \\ \hat{L}(k+1|k) \end{bmatrix} = \begin{bmatrix} \hat{x}_v(k|k) \\ \hat{y}_v(k|k) \\ \hat{\theta}_v(k|k) \\ \hat{x}_{f_1}(k|k) \\ \hat{y}_{f_1}(k|k) \\ \hat{\Upsilon}_{r_1}(k|k) \\ \hat{\Upsilon}_{a_1}(k|k) \\ \vdots \\ \hat{x}_{f_i}(k|k) \\ \hat{y}_{f_i}(k|k) \\ \hat{\Upsilon}_{r_i}(k|k) \\ \hat{\Upsilon}_{a_i}(k|k) \\ \vdots \\ \hat{x}_{f_N}(k|k) \\ \hat{y}_{f_N}(k|k) \\ \hat{\Upsilon}_{r_N}(k|k) \\ \hat{\Upsilon}_{a_N}(k|k) \\ \hat{L}(k|k) \end{bmatrix} + \begin{bmatrix} \Delta x_v(k) \\ \Delta y_v(k) \\ \alpha(k) \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (5.2)$$

where $\Delta x_v(k) = V(k)\Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k))$, $\Delta y_v(k) = V(k)\Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k))$ and ΔT is the sampling time. $\hat{x}_{f_i}(k|k)$ and $\hat{y}_{f_i}(k|k)$ are the estimated coordinates of the i^{th} feature. $\hat{\Upsilon}_{r_i}(k|k)$ and $\hat{\Upsilon}_{a_i}(k|k)$ are the estimated reflection and absorption coefficient values of the i^{th} feature at time k .

The augmented state vector is then

$$\mathbf{x}(k) = [\mathbf{x}_v(k), \{\mathbf{F}_1, \Upsilon_{r_1}, \Upsilon_{a_1}\} \dots \{\mathbf{F}_i, \Upsilon_{r_i}, \Upsilon_{a_i}\} \dots \{\mathbf{F}_N, \Upsilon_{r_N}, \Upsilon_{a_N}\}, L]^T \quad (5.3)$$

where $\mathbf{x}_v$ is the vehicle's pose $\mathbf{F}_i = [x_{f_i}, y_{f_i}]^T$ is the i -th feature's location, where $1 \leq i \leq N$. $\hat{\Upsilon}_{r_i}$ is the normalised radar cross section of the i -th feature, $\hat{\Upsilon}_{a_i}$ is its normalised absorption cross section, $\hat{L}$ represents the radar loss and $\mathbf{v}(k) = [\mathbf{v}_v(k), 0, 0, v_{\Upsilon_{r_1}}, v_{\Upsilon_{a_1}}, \dots, 0, 0, v_{\Upsilon_{r_i}}, v_{\Upsilon_{a_i}}, \dots, 0, 0, v_{\Upsilon_{r_N}}, v_{\Upsilon_{a_N}}, 0]^T$. $\mathbf{v}(k)$ is a zero mean uncorrelated Gaus-

5.3 Radar Based Augmented State Vector

sian motion disturbance with covariance $\mathbf{Q}(k)$.

5.3.2 Observation (Measurement) Model

For a detailed explanation on the proposed multiple-line-of-sight prediction model, an observation model is shown here. The radar observation is used to estimate the vehicle's state once the vehicle's pose is predicted. During filter update, the prediction and estimation are fused. For each of the features in the map, the predicted range, $\hat{R}_i(k+1|k)$, the radar bearing angle, $\hat{\beta}_i(k+1|k)$ and the power, $\hat{P}_i(k+1|k)$ are to be predicted from the predicted state in equation (5.2). The predicted range and bearing observations are similar to the ordinary SLAM formulation - *i.e.*

$$\hat{R}_i(k+1|k) = \sqrt{[\hat{x}_{f_i}(k+1|k) - \hat{x}_v(k+1|k)]^2 + [\hat{y}_{f_i}(k+1|k) - \hat{y}_v(k+1|k)]^2} \quad (5.4)$$

$$\hat{\beta}_i(k+1|k) = \hat{\theta}_v(k+1|k) - \tan^{-1} \left[\frac{\hat{y}_{f_i}(k+1|k) - \hat{y}_v(k+1|k)}{\hat{x}_{f_i}(k+1|k) - \hat{x}_v(k+1|k)} \right] \quad (5.5)$$

The predicted power for all targets, such as those in figure 5.2, is the fundamental difference offered in this chapter, and is explained below.

Predicted Power Observation Formulation

The assumptions made in the predicted power model are as follows.

- The environmental features of interest are assumed to have small circular cross-sections, so that the estimated normalised radar cross sections and absorption coefficients are approximately the same in all directions with respect to that feature (*i.e.* features are assumed to obey Swerling I models).

5.3 Radar Based Augmented State Vector

- The measured returned power should be independent on range according to the classical radar equation. If the radar has a built-in range compensation filter, this filter must first be removed or post-filtered to remove its effect, to produce range dependent power returns from all objects [42].
- The beam-width of the radar wave does not increase considerably with range (a pencil beam radar is assumed)¹.

A target is assumed to affect the incident electro-magnetic radiation in three possible ways:

1. A portion of the incident energy Υ_r , $0 \leq \Upsilon_r \leq 1$, is reflected and scattered.
2. A portion of the incident energy Υ_a , $0 \leq \Upsilon_a \leq 1$, is absorbed by the target.
3. A portion of the incident energy $1 - (\Upsilon_r + \Upsilon_a)$ is further transmitted through the target.

Υ_r is thus referred to as the “Normalised” radar cross section. Figure 5.3 represents the radar in an environment with i -features down-range at a particular bearing. The following terms are used in formulating the predicted power observation.

- P_{INCi} = Power incident on the i^{th} feature.
- P_{REFi} = Power reflected from the i^{th} feature.
- P_{TRANi} = Power transmitted through the i^{th} feature.
- P_{INC11} = Power incident on the first feature which is reflected from the i^{th} feature.

¹As radar signals travel away from the antenna, the pencil beam takes on a cone-like appearance and expands in all directions. This expansion or beam broadening increases signal volume, resulting in decreased signal strength. Distant targets appear distorted, or they may not be seen at all. Beam broadening also causes “*partial occlusion*” (or *partial beam filling*), which implies that distant targets occupy proportionally less of an expanded beam. Thus, the true characteristics of a feature may be hidden or altered in the signal returns. Beam broadening reduces azimuthal resolution of the radar. As the beam diameter increases with distance, closely spaced targets may occupy the beam simultaneously and appear as one reflection. In short, multiple targets at a distance are difficult to resolve correctly.

5.3 Radar Based Augmented State Vector

- P_{REFi1} = Power reflected back towards the i^{th} feature from the first feature. This component will not reach the radar receiver directly and is not considered in this formulation (and thus crossed out in figure 5.3).
- P_{TRANi1} = Power transmitted through the first feature which is the reflection from the i^{th} feature.

Figure 5.3: Power definitions for reflections, absorptions and transmissions for i multiple line-of-sight features.

The power incident at the first feature is given by

$$P_{INC1} = \frac{P_t G A_I}{4\pi \hat{R}_1^2} \quad (5.6)$$

where P_t is the power transmitted by the radar, G is the antenna gain and $\hat{R}_1$ is the distance between radar and the first feature and A_I is the area of the object illuminated by the radar wave. Let Υ_{r1} be the normalised radar cross section and Υ_{a1} be the

5.3 Radar Based Augmented State Vector

normalised absorption cross section of the first feature. The power received by the radar receiver from the first feature is given by $P'_{REF1} = P_{REF1} A_e / (4\pi \hat{R}_1^2)$

$$P'_{REF1} = \frac{P_t G \hat{\Upsilon}_{r1} A_I}{(4\pi)^2 \hat{R}_1^4} A_e \quad (5.7)$$

where A_e is the theoretical antenna aperture. It is shown in the radar literature that $A_e = G\lambda^2/4\pi$ [73]. Substituting for A_e in equation 5.7, the power return from the first feature is

$$P'_{REF1} = \frac{P_t G^2 \lambda^2 \hat{\Upsilon}_{r1} A_I}{(4\pi)^3 \hat{R}_1^4} \quad (5.8)$$

The power P_{TRAN1} that passes through the first feature is given by

$$P_{TRAN1} = \frac{P_t G A_I (1 - [\hat{\Upsilon}_{r1} + \hat{\Upsilon}_{a1}])}{(4\pi) \hat{R}_1^2} \quad (5.9)$$

The power reflected from the second feature, P_{REF2} is given by

$$P_{REF2} = \frac{P_t G A_I^2 \hat{\Upsilon}_{r2} (1 - [\hat{\Upsilon}_{r1} + \hat{\Upsilon}_{a1}])}{(4\pi)^2 \hat{R}_1^2 (\hat{R}_2 - \hat{R}_1)^2} \quad (5.10)$$

The power then transmitted back to the first feature from the second feature is given by

$$P_{INC21} = \frac{P_t G A_I^3 \hat{\Upsilon}_{r2} (1 - [\hat{\Upsilon}_{r1} + \hat{\Upsilon}_{a1}])}{(4\pi)^3 \hat{R}_1^3 (\hat{R}_2 - \hat{R}_1)^4} \quad (5.11)$$

5.3 Radar Based Augmented State Vector

The power, P_{INC21} then passes through feature 1 and is given by

$$P_{TRAN21} = P_{INC21}(1 - [\hat{\Upsilon}_{r_1} + \hat{\Upsilon}_{a_1}]) \quad (5.12)$$

The power returned from the second feature is then $P'_{TRAN21} = P_{TRAN21}A_e/(4\pi\hat{R}_1^2)$

$$P'_{TRAN21} = \frac{P_tGA_I^3A_e\hat{\Upsilon}_{r_2}(1 - [\hat{\Upsilon}_{r_1} + \hat{\Upsilon}_{a_1}])^2}{(4\pi)^4\hat{R}_1^4(\hat{R}_2 - \hat{R}_1)^4} \quad (5.13)$$

In general, the predicted observed power from the i^{th} feature can be written as

$$\hat{P}'_{TRANi1}(k+1|k) = \frac{KA_I^{(2i-1)}\hat{\Upsilon}_{r_i}(k+1|k)}{(4\pi)^{2i}} \times \frac{\prod_{j=0}^{i-1}[1 - (\hat{\Upsilon}_{r_j}(k+1|k) + \hat{\Upsilon}_{a_j}(k+1|k))]}{\prod_{j=0}^{i-1}(\hat{R}_{j+1}(k+1|k) - \hat{R}_j(k+1|k))^4} \quad (5.14)$$

where $K = P_tGA_e$, $A_e = \frac{G\lambda^2}{4\pi}$, $\hat{\Upsilon}_{r_0} = \hat{\Upsilon}_{a_0} = \hat{R}_0 = 0$ and, for the $i - th$ feature, $\hat{R}_i(k+1|k)$ is related to the augmented state by equation 5.4.

Equations 5.4, 5.5 and 5.14 between them comprise the predicted observation. In order to generate realistic predictions of the range spectra, knowledge of the power and range noise distributions is necessary. This has been studied extensively in previous work, and can be found in chapter 2 and also in [42].

The range and power noise are experimentally obtained [42]. The noise in range is the phase noise, which is obtained by observing the range spectra containing reflections from objects with different RCSs at different locations. The noise statistics in power is obtained during both target presence and absence.

5.4 Radar Spectra Prediction Results

The angular standard deviation is assumed to be 1° as the radar wave is a pencil beam with beam width 1.8° . The observation model is then given by

$$\begin{aligned}\mathbf{z}_i(k+1) &= [\mathbf{R}_i(k+1), \beta_i(k+1), P_i(k+1)]^T + \mathbf{w}_i(k+1) \\ &= \mathbf{h}(\mathbf{x}(k+1)) + \mathbf{w}_i(k+1)\end{aligned}\quad (5.15)$$

where $\mathbf{z}_i(k+1)$ is the observation, and $\mathbf{w}_i(k+1)$ is the additive observation noise given by

$$\mathbf{w}_i(k+1) = [v_R(k+1)v_\beta(k+1)v_P(k+1)]^T \quad (5.16)$$

and $\mathbf{h}$ is the non-linear observation function defined by equations 5.4, 5.5 and 5.14.

5.4 Radar Spectra Prediction Results

To validate the formulation for realistically predicting multiple line-of-site target range spectra, tests using a scanning radar unit were carried out. Initially the vehicle was positioned at pose $\mathbf{x}_v(k)$ as demonstrated in figure 5.4.

The full 360° radar scan obtained from this vehicle location is shown in figure 5.1. Range spectra obtained from the initial vehicle location at two different bearing angles are shown in figures 5.2 and 5.5.

Figure 5.2 is obtained at azimuth 231° and is indicated by the black line in figure 5.1. Features in the environment are marked in the figures. The next predicted vehicle location is calculated using the vehicle model and system inputs (equation 5.2). This corresponds to the new predicted vehicle pose $\hat{\mathbf{x}}_v(k+1 | k)$ in figure 5.4. The range spectra in all directions are then predicted from the new predicted vehicle location. For example, in the range spectrum predicted at angle $\hat{\beta}(k+1 | k)$ in figure 5.4, the predicted values for the range, bearing and received power of features A and D are calculated according to equations 5.4, 5.5 and 5.14.

A single predicted spectrum obtained from the predicted vehicle location $\mathbf{x}_v(k+1 | k)$ is shown in figure 5.6 having two features down-range. Equation 5.14 can be used to

5.4 Radar Spectra Prediction Results

Figure 5.4: Vehicle motion and the features observed/predicted. Features observed/predicted down-range at different bearings are marked.

Figure 5.5: Radar range spectra (233° azimuth) obtained at the starting robot location. Three features observed down-range are marked.

5.4 Radar Spectra Prediction Results

predict the received power as long as the power bias as a function of range incorporated into the radar electronics must be taken into account. This simply requires knowledge of the radar's high pass filter circuitry which in an FMCW radar compensates for the fourth power of range loss, expected according to the simple radar equation [73] [42].

Figure 5.6: Predicted radar Range Spectra (at 234° bearing) obtained from the predicted vehicle location.

The actual observation is obtained from the next vehicle location and is shown in figure 5.7 which shows power peaks in close proximity to those predicted in figure 5.6.

The predicted and actual received powers from the target at A are in close agreement in both figures whereas, the predicted value for the received power (30 dB) of the target at 58 metres (feature D in figure 5.6) is slightly less than the actual received power (38 dB) in figure 5.7. The discrepancy for feature D can be due to violation of some of the assumptions made in the formulation - in particular that the normalised reflection and absorption cross sections remain constant, independent of the radar to target angle of incidence.

Figure 5.8 shows the results of a chi-squared test to determine any bias or inconsistency in the power-range spectrum predictions. The difference between the measured

5.4 Radar Spectra Prediction Results

Figure 5.7: Actual radar Range Spectra (at 234° bearing) obtained at the next robot location. Features observed down-range are marked.

5.4 Radar Spectra Prediction Results

and the predicted range spectra is plotted together with 99% confidence interval. The value of 99% bound, $= \pm 16.35 \text{ dB}$, has been found experimentally by recording several noisy power-range spectra in target absence (radar pointing towards open space) [42]. Close analysis of figure 5.8 shows that the error has a negative bias. This is due to the inaccuracy of the high pass filter gain assumption for sensor modelling compared to the actual gain. A detailed explanation on the high pass filter gain is given in appendix A. The result shows that this approximation for the high pass filter gain is acceptable, as a large portion of the error plot lies within the 3σ limits.

Figure 5.8: The difference between predicted and measured range spectra containing two features down-range is shown. This error is compared with the 3σ noise power bounds.

5.4.1 Full 360 Degree Radar Spectra Prediction For Autonomous Navigation

This section demonstrates a full 360° radar spectra prediction for effective robot navigation. This is based on given predictions of the range to, and RCS values of, multiple targets down-range for a spectrum at every azimuth. The sensor model, explained in

5.4 Radar Spectra Prediction Results

this chapter, not only predicts feature locations, but also predicts power returns of features (which is a function of object's RCS values) [44]. The prediction model given in chapter 2 cannot predict multiple targets as it is carried out by modeling the actual sensor receiver electronics. The prediction results given in this chapter are more accurately modelled compared to the spectra prediction method shown in chapter 2. This is evident by comparing the correlation coefficient values obtained using the two prediction methods. The correlation coefficients for each method will now be examined.

First, a 360° power spectra, recorded from the radar will be shown, along with its interpretation. This is carried out in order to fully understand the spectra obtained from outdoor environment. The scan is taken from the radar mounted on top of a vehicle. Next, a predicted 360° spectra is generated from the next predicted vehicle location as explained in the earlier sections. This is possible after detecting the feature range(s) by running a KS-CFAR processor. After the vehicle is moved to the next location, a full 360° radar scan is obtained. The scan is then compared with the predicted one to quantify how close the predicted scan is to the actual one. Due to occlusions and the radar's maximum range, features may appear/disappear. Also, clutter models are not incorporated into spectra prediction where as in a real radar scan, there could be the presence of clutter. So, a real quantification of the predicted and actual spectra is not always feasible in real outdoor environments. However, a spectrum quantification is indeed possible (using the data obtained from controlled environments) as shown in the above section and also in [44].

A full 360° real radar scan obtained from the initial vehicle location is shown in figure 5.9. Features are extracted from the raw scan by a KS-CFAR processor (explained in chapter 3) for predicting features. A full 360° predicted radar spectra, given a robot navigation framework, capable of predicting the range to a target and its RCS is shown in figure 5.10a. A scan from the actual robot location is then obtained and is shown in figure 5.10b for visual comparison. Full spectra prediction is possible given prior robot and feature estimates, which include the vehicle's pose, estimated feature locations and their RCS values. A restriction of this method is that, as a mobile robot moves, range spectra prediction is only possible, assuming that the RCS does not change even though, in general, the radar-to-target angle of incidence changes. In general this is clearly not a valid assumption, but becomes acceptable for small, cylin-

5.4 Radar Spectra Prediction Results

drical objects, making their RCS's approximately view-point invariant, such as lamp posts, trees etc, which can be used for outdoor navigation.

Figure 5.9: Real 360° radar scan taken from the initial vehicle location. Features include trees, lamp posts, concrete buildings and pillars etc.

In order to quantify the statistical similarity between the predicted and actual scan, the correlation coefficient of each spectrum are obtained. The correlation coefficient, r_{xy}^2 is computed as given in equation 5.17 (also given in chapter 2):

The correlation coefficient provides a measure of the strength of the linear relationship between the predicted spectra X , with spectral values x_i , and the observed range spectra Y , with spectral values y_i , and will provide an indication of how accurate the predicted observations (spectra) are. This coefficient r_{xy}^2 ($0 < r_{xy}^2 < 1$) is given by

$$r_{xy}^2 = \left[\frac{N \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{(N \sum x_i^2 - (\sum x_i)^2) (N \sum y_i^2 - (\sum y_i)^2)}} \right]^2 \quad (5.17)$$

where each summation is taken from $i = 1$ to N with N being the number of bins in each power versus range spectra at a given scanning bearing angle (in this case $N=800$).

5.4 Radar Spectra Prediction Results

(a) A full 360° predicted radar scan of a car park environment.

(b) A full 360° real radar scan obtained from the car park environment.

Figure 5.10: A full 360° predicted and real radar scan of an outdoor environment (car park), obtained from a predicted vehicle location 2.4m away approx. from the initial vehicle location. Features include pillars, trees, lamp posts, buildings etc. It is possible to predict distances of features that have a non circular cross section such as buildings, cars etc. However, the predicted power will not be accurate as their RCS values will change when the vehicle/radar moves (the radar signal's angle of incidence will change).

5.4 Radar Spectra Prediction Results

In order to obtain the correlation coefficient values, polar plots of scans corresponding to figures 5.10a and 5.10b are obtained, and are shown in figures 5.11. Next, the correlation coefficient of the predicted and measured spectra are obtained and are shown in figure 5.12. It is evident from figure 5.12 that, not all of the predicted radar spectrum is highly correlated with the real spectrum. The low correlation could be either due to the following reasons:

- As the KS-CFAR processor for automatic feature detection, used in the prediction model, has excellent clutter rejection capabilities, the predicted spectra contains reflections only from “real features”. Thus, the clutter reflections will be absent in the predicted spectra. Hence, when a real spectrum that contains reflections from land terrain (land clutter), is compared against the predicted one, a lower correlation results.
- Features may appear/disappear due to either occlusion of objects or due to object(s) being outside the radar’s maximum visible range depending on the displacement between the scan used to generate a predicted observation, and the next actual observation.

Figure 5.13 shows the zoomed versions of the predicted and actual scans 5.10a and 5.10b respectively. Features such as trees at around 68° azimuth are successfully predicted as shown in figure 5.13a along with the predicted reflections from a nearby building. In figure 5.13b, features such as a concrete wall and trees are successfully predicted. Although one tree partially occludes the other, the sensor model still manages to predict it, however, with a reduced predicted power value¹. Analysis of the predicted and actual spectra at that area (which is around azimuth 270°) provides a lower correlation coefficient as shown in figure 5.12. In order to demonstrate the ability of the predictive observation model to correctly predict observations, larger distances (11 metres and 30 metres) between the scan used to generate a prediction, and the location of the predicted scan are tested.

Figure 5.14 shows the full 360° radar scan obtained from a vehicle location which is 11 metres away from the initial vehicle location. The predicted and actual scans from the vehicle location are shown in figures 5.15a and 5.15b respectively. The most important features include walls, pillars, lamp posts, a metallic structure and trees.

¹due to partial occlusion, the returned power will be lower than the actual returned power

5.4 Radar Spectra Prediction Results

(a) Polar plot of the predicted radar scan of a car park environment.

(b) A full 360° real radar scan obtained from the car park environment.

Figure 5.11: Polar plots of full 360° predicted and real radar scan of an outdoor environment (car park) with features such as pillars, trees, lamp posts etc.

5.4 Radar Spectra Prediction Results

Figure 5.12: The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 2.4m from the initial vehicle location.

Figures 5.16 show the polar plots of the full 360° predicted and actual scans. The correlation coefficient between each spectrum is shown in figure 5.17. The correlation coefficient value is comparatively high at around 5° azimuth which is due to the matching of predicted and measured scans from a metallic gas chamber of high RCS. This is shown in figures 5.18a and 5.18b. At azimuth 179°, the correlation coefficient is very low compared to other regions. At 179°, the concrete pillars are successfully predicted and observed in both the scans. However, due to the slight change in vehicle azimuth, feature locations in the observed scan are different from the feature locations in the predicted scan, thus lowering the correlation coefficient.

Next, the vehicle is moved to approximately 30 metres away from the initial vehicle location and a radar scan is obtained. The scan is shown in figure 5.19. A predicted scan is then obtained from the next predicted vehicle location. This is shown in figure 5.20. The vehicle is then moved to the next vehicle location and a full 360° scan is obtained as shown in figure 5.21.

For performing a quantitative analysis the predicted and actual scans are plotted in polar co-ordinates as shown in figure 5.22 and a measure of correlation is again

5.4 Radar Spectra Prediction Results

(a) A building and other features such as trees, which are predicted successfully from the predicted vehicle location.

(b) A concrete wall perceived by the radar from the observed vehicle location along with trees.

Figure 5.13: Zoomed in versions of the full 360° predicted (a) and real (b) radar scans of an outdoor environment (car park) with features such trees, buildings, concrete wall etc.

5.4 Radar Spectra Prediction Results

Figure 5.14: Actual full 360° radar scan obtained from the vehicle location which is 11 metres away from the initial location. Features include walls, pillars, lamp posts, a metallic structure, trees etc. Features are extracted automatically from the scan for spectra prediction.

5.4 Radar Spectra Prediction Results

(a) A full 360° predicted radar scan of a car park environment.

(b) A full 360° real radar scan obtained from the car park environment.

Figure 5.15: A full 360° predicted and real radar scan of an outdoor environment (car park), obtained from a predicted vehicle location which is 11 metres away from the initial vehicle location, with features such as pillars, trees, lamp posts etc. It is possible to predict distances of features that have a non circular cross section such as buildings, cars etc. However, the predicted power will not be accurate as their RCS values will change when the vehicle/radar moves (the radar signal's angle of incidence will change).

5.4 Radar Spectra Prediction Results

(a) Polar plot of the predicted radar scan of a car park environment.

(b) A full 360° real radar scan obtained from the car park environment.

Figure 5.16: Polar plots of full 360° predicted and real radar scan of an outdoor environment (car park) at about 11 metres from the initial vehicle location, with features such as pillars, trees, lamp posts etc.

5.4 Radar Spectra Prediction Results

Figure 5.17: The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 2.4m from the initial vehicle location.

obtained. This correlation co-efficient between predicted and actual scans are shown in figure 5.23.

At an azimuth of about 90°, the correlation coefficient is comparatively low due to the missed detection of a concrete pillar and predicting a tree and a concrete wall. The predicted and actual scans, thus, are different which provides a lower correlation.

This formulation and analysis shows the initial stages necessary in implementing an augmented state, feature rich SLAM formulation with millimetre wave radar. The next chapter will address the ease with which data association can be carried out using the multi-dimensional feature state estimates, and a full SLAM implementation in outdoor environments, with the proposed KS-CFAR processor and the TPP based feature detector.

5.4 Radar Spectra Prediction Results

(a) A metallic gas chamber of high RCS, which is predicted successfully from the predicted vehicle location along with concrete pillars.

(b) The high RCS metallic gas chamber perceived by the radar from the observed vehicle location along with concrete pillars.

Figure 5.18: Zoomed in versions of the full 360° predicted and real radar scan of an outdoor environment (car park) with features such concrete pillars, metallic objects etc.

5.4 Radar Spectra Prediction Results

Figure 5.19: Actual radar scan from vehicle location 30 metres away from the initial location.

Figure 5.20: Predicted scan from the next predicted vehicle location.

5.5 Conclusions

Figure 5.21: Actual full 360° radar scan from the measured vehicle location.

5.5 Conclusions

This chapter described a new approach in predicting radar range spectra which is essential for simultaneous localisation and map building (SLAM) with millimetre wave radar.

A SLAM formulation using an augmented state vector which includes the normalised radar cross sections and absorption cross sections of features, as well as the usual feature Cartesian coordinates, was introduced. This is intended to aid the data association process, so that features need not just be associated based on their Cartesian coordinates, but account can be taken of their estimated normalised reflection and absorption cross sections also.

The second contribution is a predictive model of the form and magnitudes of the power-range spectra from differing vehicle locations, for multiple line-of-sight targets. This forms a predicted power-range observation, based on estimates of the augmented SLAM state.

5.5 Conclusions

(a) Polar plot of the predicted radar scan of a car park environment.

(b) A full 360° real radar scan obtained from the car park environment.

Figure 5.22: Polar plots of full 360° predicted and real radar scans of an outdoor environment (car park) with features such as pillars, trees, lamp posts etc.

5.5 Conclusions

Figure 5.23: The correlation coefficient values obtained by comparing the predicted and actual spectra obtained at 11m from the initial vehicle location.

The formulation of power returns from multiple objects down-range is explained and predicted radar range spectra are compared with real spectra, recorded outdoors. This prediction of power-range spectra is a step towards a full, radar based SLAM framework.

Chapter 6

Simultaneous Localisation And Mapping in Outdoor Terrains with MMW Radar

6.1 Introduction

The simultaneous localization and mapping (SLAM) problem has received tremendous attention in the robotics literature. The SLAM problem involves a moving vehicle attempting to recover a spatial map of its environment, while simultaneously estimating its own pose (location and orientation). SLAM problems arise in the navigation of mobile robots through unknown environments for which no accurate map is available. The problem of acquiring a map with known robot poses [82], [56] is significantly easier than the general SLAM problem. When the robot location is unknown (for example, if the GPS data - which is used to find vehicle coordinates - is unavailable), the vehicle will accrue pose errors during mapping which will induce errors in the map it-

6.1 Introduction

self. SLAM addresses this very problem of acquiring a map without an external source of vehicle pose information.

A first approach to the SLAM problem was introduced in a seminal paper by Smith, Self, and Cheeseman [80]. It was first developed into an implemented system by Moutarlier and Chatila [58], [57]. This approach used an extended Kalman filter (EKF) for estimating the posterior distribution over the map and the robot pose. The EKF represents the map of feature locations and the robot pose estimate as a multi-dimensional Gaussian. The off-diagonal elements in the covariance matrix of this multivariate Gaussian represent the correlations between errors in the vehicle pose and the features in the map [90], as well as the correlations between the map estimates themselves. When dealing with non-linear models in the state and measurement equations and a non-Gaussian noise assumption, the EKF may lead to non-optimal solutions due to *linearisation* errors [45]. In recent years, Julier and Uhlmann developed an unscented Kalman filter (UKF) that can compute the mean and covariance of a non-linear function $x = g(y)$ up to the second order of the Taylor series expansion of $g()$ [45]. The UKF avoids the calculation of Jacobians, and transforms individual “sigma-points” through the non-linear function g in order to generate predicted means and covariances. While the UKF is significantly better than the EKF in density statistics estimation, it still assumes a Gaussian parametric form of the posterior, and thus cannot handle multi-modal distributions.

An optimal solution to SLAM would result from a full Bayesian filtering algorithm, which the EKF and UKF attempt to approximate to certain levels of accuracy. Unfortunately, a fully Bayesian representation of the SLAM state is typically, computationally intractable due to the high dimensionality of the map state representation. This makes the full Bayesian solution difficult to apply in real-time SLAM implementations. Sequential Monte Carlo methods, or particle filters [5], which are again approximations to Bayesian filtering, can provide general solutions to problems where linearisation is intractable or would result in low performance. The number of particles necessary to attain a certain level of state estimation accuracy however, is still an ill-defined problem [23].

A novel algorithm named FastSLAM was presented in [54] which addressed the real time implementation of the SLAM problem from a Bayesian point of view. The

6.1 Introduction

authors used a particle filter [23] to estimate the vehicle pose and EKF filters to estimate the location of features in the map. This algorithm partitions SLAM into a factored solution containing both pose estimates and map estimates, conditioned on each pose estimate. FastSLAM uses a modified particle filter to estimate the robot pose. Each particle has an independent EKF running for each landmark in the map to estimate its position. The performance of FastSLAM is dependent upon the number of particles chosen, and can suffer from the problem of particle depletion [30]. EKF-SLAM [89] is therefore used in this work to demonstrate the merits of the proposed prediction and feature detection methods, explained in previous chapters.

A contribution of this chapter is the formulation of a full EKF solution to SLAM based on *multiple line-of-sight* sensing. It was shown in chapter 2 that, to a limited extent, MMW radar has the ability to detect multiple targets down range, due to its unique penetrating ability. Hence this chapter begins with a thorough derivation of the predicted measurement equations, and the corresponding measurement Jacobian matrices necessary to implement EKF SLAM with multiple line-of-sight sensing. It should be noted that the derived formulation is actually applicable to any exteroceptive sensing technique which is capable of returning multiple line-of-sight measurements. The theory could be extended to X-ray devices and thermal imaging sensors. The focus of the work here however remains that of MMW radar, and the actual received power measurement model is therefore based on the radar range equation.

It is also true that one of the most difficult problems for any outdoor SLAM implementation using radar, is that of successive reliable feature detection, which is required for consistent feature validation and data association hypotheses. Hence, a reliable feature extraction method is a prerequisite to any successful radar based autonomous robot navigation, which was the subject of chapters 3 and 4. It will be demonstrated here that SLAM can be achieved with the information extracted from the radar scans using the proposed KS-CFAR and TPP feature detectors from chapters 3 and 4, while attempting SLAM with the classically used fixed power thresholding method fails in all but extremely simple environments. SLAM implementations, based on the derived multiple line-of-sight, EKF algorithm, with these three different feature detectors, in outdoor terrains, are presented and their merits and drawbacks are analysed in this chapter.

The important contributions of this chapter are the following:

6.1 Introduction

- Closed form solutions for the measurement model and measurement Jacobian matrices, under an EKF framework, are derived for the multiple line-of-sight imaging qualities of MMW radar. This incorporates the received power measurements into the framework, as well as the usual range and bearing measurements considered in most other SLAM formulations. The correct measurement and measurement Jacobian matrix formulations allow even occluded features to reduce the error in the overall SLAM state estimate.
- Features are represented not only by their x , y coordinates, but also by their normalised radar reflection and absorption coefficients. This provides two extra dimensions, compared with the more usual SLAM formulations, under which feature data association can take place.
- The received power predictions and measurements, as well as the usual range and bearing predictions/measurements, provide extra measurement information, which is optimally fused with the SLAM state estimate.
- A multi-model feature extraction strategy is implemented based on the RANSAC algorithm and point clustering, to identify both line and point features.
- Automatic feature detectors, based on the previously introduced TPP detector and the KS-CFAR processor, are implemented to generate the feature observations for SLAM in outdoor terrains. The performance of the proposed feature detectors, and subsequent effects of the SLAM filter are analysed.

Section 6.2 provides a brief introduction to feature based EKF-SLAM and then shows the derivation of the appropriate received power measurement model the corresponding measurement Jacobian matrices. Due to the coupled relationship between the received power from an occluded feature and the state estimates of *other* features which occlude it, a new measurement Jacobian matrix formulation which correctly includes this interaction is presented.

A multi-model feature extraction strategy based on the RANSAC algorithm and a point feature clustering method is explained in section 6.4. Section 6.5 shows an efficient Nearest Neighbour (NN) data association strategy by utilising the additional information, *i.e.* the reflected power from features. EKF-SLAM implementations with

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

naturally occurring features in outdoor, semi-built environments are shown in section 6.6. These are based on the KS-CFAR processor and TPP based feature detection algorithms. This section analyses the merits and drawbacks of the proposed processors in a robot navigation perspective.

Conclusions are drawn in section 6.7, which summarise that the multiple line-of-sight feature model, together with the proposed feature detectors indeed improve SLAM performance (both in terms of consistent localisation estimates and accurate feature maps) with radar in outdoor terrains.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

Probabilistic SLAM involves finding an appropriate representation for the joint uncertainty introduced by both a vehicle's motion model and its exteroceptive sensor(s) measurement model(s). The solution should then yield the consistent computation of the vehicle pose, and map feature coordinates, joint posterior distributions [24].

An EKF is used here for analysing the proposed sensor model (explained in chapter 5) and various feature detectors' performances (explained in chapters 3 and 4) for radar based SLAM. In the EKF-SLAM formulation shown here, the vehicle state, $\mathbf{x}_v(k)$ is given by $\mathbf{x}_v(k) = [x_v(k), y_v(k), \theta_v(k)]^T$ where $x_v(k)$, $y_v(k)$ and $\theta_v(k)$ are the global position and orientation of the vehicle at time k . At time instant, $(k + 1)$, the previous vehicle state, $\mathbf{x}_v(k)$, is propagated through a simple vehicle process model [40]. The model, with control inputs, $\mathbf{u}(k) = [V(k) \ \alpha(k)]$, where $V(k)$ and $\alpha(k)$ are the forward velocity and steering angle of the vehicle at time k , predicts the vehicle state at time $(k + 1)$ as well as the corresponding uncertainty in vehicle location, represented by a covariance matrix $\mathbf{P}(k + 1|k)$ [7]. Hence,

$$\mathbf{x}_v(k + 1) = \mathbf{f}(\mathbf{x}_v(k), \mathbf{u}(k), \mathbf{v}(k)) \quad (6.1)$$

where $\mathbf{f}(\cdot)$ models the vehicle's kinematics and $\mathbf{v}(k)$ is a zero mean uncorrelated Gaussian motion disturbance with covariance $\mathbf{Q}(k)$.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

The predicted augmented state, $\hat{\mathbf{x}}(k+1|k) = [\hat{\mathbf{x}}_v^T(k+1|k) \hat{\mathbf{m}}^T(k+1|k)]^T$ at time, $(k+1)$, as shown previously in equation 5.2, chapter 5, is written as,

$$\begin{bmatrix} \hat{x}_v(k+1|k) \\ \hat{y}_v(k+1|k) \\ \hat{\theta}_v(k+1|k) \\ \hat{x}_{f_1}(k+1|k) \\ \hat{y}_{f_1}(k+1|k) \\ \hat{\Upsilon}_{r_1}(k+1|k) \\ \hat{\Upsilon}_{a_1}(k+1|k) \\ \vdots \\ \hat{x}_{f_i}(k+1|k) \\ \hat{y}_{f_i}(k+1|k) \\ \hat{\Upsilon}_{r_i}(k+1|k) \\ \hat{\Upsilon}_{a_i}(k+1|k) \\ \vdots \\ \hat{x}_{f_N}(k+1|k) \\ \hat{y}_{f_N}(k+1|k) \\ \hat{\Upsilon}_{r_N}(k+1|k) \\ \hat{\Upsilon}_{a_N}(k+1|k) \\ \hat{L}(k+1|k) \end{bmatrix} = \begin{bmatrix} \hat{x}_v(k|k) \\ \hat{y}_v(k|k) \\ \hat{\theta}_v(k|k) \\ \hat{x}_{f_1}(k|k) \\ \hat{y}_{f_1}(k|k) \\ \hat{\Upsilon}_{r_1}(k|k) \\ \hat{\Upsilon}_{a_1}(k|k) \\ \vdots \\ \hat{x}_{f_i}(k|k) \\ \hat{y}_{f_i}(k|k) \\ \hat{\Upsilon}_{r_i}(k|k) \\ \hat{\Upsilon}_{a_i}(k|k) \\ \vdots \\ \hat{x}_{f_N}(k|k) \\ \hat{y}_{f_N}(k|k) \\ \hat{\Upsilon}_{r_N}(k|k) \\ \hat{\Upsilon}_{a_N}(k|k) \\ \hat{L}(k|k) \end{bmatrix} + \begin{bmatrix} \Delta x_v(k) \\ \Delta y_v(k) \\ \alpha(k) \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (6.2)$$

where $\Delta x_v(k) = V(k)\Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k))$, $\Delta y_v(k) = V(k)\Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k))$ and ΔT is the sampling time. $\hat{x}_{f_i}(k|k)$ and $\hat{y}_{f_i}(k|k)$ are the estimated coordinates of the i^{th} feature. $\hat{\Upsilon}_{r_i}(k|k)$ and $\hat{\Upsilon}_{a_i}(k|k)$ are the estimated reflection and absorption coefficient values of the i^{th} feature at time k (See chapter 5, section 5.3.1). Note that for a total of N features, the dimension of the state vector is $4N + 4$.

For each of the features in the map, the predicted range, $\hat{R}_i(k+1|k)$, bearing angle, $\hat{\beta}_i(k+1|k)$ and received power, $\hat{P}_i(k+1|k)$ are to be determined using the measurement model and predicted states in equation (6.2). The predicted range and bearing observations are similar to other SLAM formulations - *i.e.*

$$\hat{R}_i(k+1|k) = \sqrt{[\hat{x}_{f_i}(k+1|k) - \hat{x}_v(k+1|k)]^2 + [\hat{y}_{f_i}(k+1|k) - \hat{y}_v(k+1|k)]^2} \quad (6.3)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

$$\hat{\beta}_i(k+1|k) = \hat{\theta}_v(k+1|k) - \tan^{-1} \left[\frac{\hat{y}_{f_i}(k+1|k) - \hat{y}_v(k+1|k)}{\hat{x}_{f_i}(k+1|k) - \hat{x}_v(k+1|k)} \right] \quad (6.4)$$

The predicted observed power (shown in chapter 5, equation 5.14) from the i^{th} feature depends on whether any of the features occlude one another. If not (*i.e.* if there are no multiple line-of-sight features), then $i = 1$ in equation 5.14 and the predicted received power $\hat{P}_i(k+1|k)$ can be written as:

$$\hat{P}_i(k+1|k) = \frac{KA_I \hat{\Upsilon}_{r_i}(k+1|k)}{(4\pi)^2 \hat{R}_i^4(k+1|k)} \quad (6.5)$$

where $K = P_t G A_e$, $A_e = \frac{G\lambda^2}{4\pi}$ and, for the i^{th} feature, $\hat{R}_i(k+1|k)$ is related to the augmented state by equation 6.3. The terms P_t , G and λ were defined in chapter 5 as the transmitted power, antenna gain and wavelength respectively.

Equations 6.3, 6.4 and 6.5 comprise the predicted observation. The observation model is:

$$\begin{aligned} \mathbf{z}_i(k+1) &= [R_i(k+1), \beta_i(k+1), P_i(k+1)]^T + \mathbf{w}_i(k+1) \\ &= \mathbf{h}_i(\mathbf{x}(k+1)) + \mathbf{w}_i(k+1). \end{aligned} \quad (6.6)$$

Here $\mathbf{z}_i(k+1)$ is the observation, $\mathbf{h}_i$ is the non-linear observation model vector of functions, with components defined by equations 6.3, 6.4 and 6.5 (or as will be shown later, equation 6.33 as appropriate), and $\mathbf{w}_i(k+1)$ is the additive observation noise given by

$$\mathbf{w}_i(k+1) = [v_R(k+1) \ v_\beta(k+1) \ v_P(k+1)]^T \quad (6.7)$$

where $v_R(k+1)$, $v_\beta(k+1)$ and $v_P(k+1)$ represent the measurement noise in range, bearing and received power respectively. For implementation purposes, the range, bearing and received power standard deviations are assumed to be $0.05m$, 1° ¹ and $0.52dB$ respectively.

With an initialised state estimate $\hat{\mathbf{x}}(k|k)$ and its estimated covariance matrix $\mathbf{P}(k|k)$,

¹as the radar is a pencil beam type system.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

the defined, augmented state vector, $\mathbf{x}(k+1)$ and its covariance matrix, $\mathbf{P}(k+1)$, and the predicted observation of each feature, $\hat{\mathbf{z}}_i(k+1|k)$, the EKF recursion can be implemented. Adopting the notation of Bar-Shalom [7], the prediction equations are

$$\begin{aligned}\hat{\mathbf{x}}(k+1|k) &= [\mathbf{f}^T(\hat{\mathbf{x}}_v(k|k), \mathbf{u}(k)) \ \hat{\mathbf{m}}^T(k|k)]^T \\ \mathbf{P}(k+1|k) &= \nabla_{\mathbf{x}} \mathbf{f}[\hat{\mathbf{x}}(k|k)] \mathbf{P}(k|k) \nabla_{\mathbf{x}}^T \mathbf{f}[\hat{\mathbf{x}}(k|k)] + \nabla_{\mathbf{u}} \mathbf{f} \mathbf{Q}(k) \nabla_{\mathbf{u}}^T \mathbf{f}\end{aligned}\quad (6.8)$$

where $\nabla_{\mathbf{x}} \mathbf{f}[\hat{\mathbf{x}}(k|k)]$ is the Jacobian of the augmented process (vehicle and map) model (given in equation 6.2) at time k (defined as $\nabla_{\mathbf{x}} \mathbf{f}[\hat{\mathbf{x}}(k|k)] = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \big|_{\mathbf{x}=\hat{\mathbf{x}}(k|k)}$), $\mathbf{Q}(k)$ is the augmented process noise covariance matrix and $\hat{\mathbf{m}}_i(k+1|k)$ is the predicted coordinate vector of the i^{th} feature. For ease of notation, from this point on, $\nabla_{\mathbf{x}} \mathbf{f}[\hat{\mathbf{x}}(k|k)]$ is simply referred to as $\nabla_{\mathbf{x}} \mathbf{f}$. Based on the full state prediction above, individual predicted observations of each stored map feature can be made

$$\hat{\mathbf{z}}_i(k+1|k) = \mathbf{h}_i(\hat{\mathbf{x}}(k+1|k)) = \mathbf{h}_i(\hat{\mathbf{x}}_v(k+1|k), \hat{\mathbf{m}}_i(k+1|k)) \quad (6.9)$$

At time, $k+1$, feature observations, $\mathbf{z}_i(k+1)$, are made and if the same feature is already stored and associated using suitable data association methods, an innovation for each feature $\nu_i(k+1)$ can be determined as follows

$$\nu_i(k+1) = \mathbf{z}_i(k+1) - \hat{\mathbf{z}}_i(k+1|k) \quad (6.10)$$

For N observed, and matched features, a full innovation vector $\nu(k+1)$ can be defined

$$\nu(k+1) = \begin{bmatrix} \nu_1(k+1) \\ \nu_2(k+1) \\ \nu_3(k+1) \\ \vdots \\ \nu_i(k+1) \\ \vdots \\ \nu_N(k+1) \end{bmatrix} \quad (6.11)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

The full state EKF update equations can then be written as follows:

$$\begin{aligned}\hat{\mathbf{x}}(k+1|k+1) &= \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\nu(k+1) \\ \mathbf{S}(k+1) &= \nabla_{\mathbf{x}}\mathbf{h}[\hat{\mathbf{x}}(k|k)] \mathbf{P}(k+1|k) \nabla_{\mathbf{x}}^T\mathbf{h}[\hat{\mathbf{x}}(k|k)] + \mathbf{R}(k+1) \\ \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1)\mathbf{W}^T(k+1)\end{aligned}\quad (6.12)$$

where $\mathbf{S}(k+1)$ is its innovation covariance matrix. $\mathbf{R}(k+1)$ is the measurement noise covariance matrix, $\nabla_{\mathbf{x}}\mathbf{h}[\hat{\mathbf{x}}(k|k)] = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \big|_{\mathbf{x}=\hat{\mathbf{x}}(k|k)}$ is the measurement model Jacobian which is formed as a stacked matrix of the individual feature measurement Jacobian sub-matrices as follows

$$\nabla_{\mathbf{x}}\mathbf{h}[\hat{\mathbf{x}}(k|k)] = \begin{bmatrix} \nabla_{\mathbf{x}}\mathbf{h}_1[\hat{\mathbf{x}}(k|k)] \\ \nabla_{\mathbf{x}}\mathbf{h}_2[\hat{\mathbf{x}}(k|k)] \\ \nabla_{\mathbf{x}}\mathbf{h}_3[\hat{\mathbf{x}}(k|k)] \\ \vdots \\ \nabla_{\mathbf{x}}\mathbf{h}_i[\hat{\mathbf{x}}(k|k)] \\ \vdots \\ \nabla_{\mathbf{x}}\mathbf{h}_N[\hat{\mathbf{x}}(k|k)] \end{bmatrix} \quad (6.13)$$

and $\mathbf{W}(k+1)$ is the Kalman gain which is given by

$$\mathbf{W}(k+1) = \mathbf{P}(k+1|k) \nabla_{\mathbf{x}}^T\mathbf{h}[\hat{\mathbf{x}}(k|k)] \mathbf{S}^{-1}(k+1). \quad (6.14)$$

Again, for ease of notation, from this point on, $\nabla_{\mathbf{x}}\mathbf{h}[\hat{\mathbf{x}}(k|k)]$ and each of its individual feature measurement Jacobian sub-matrices $\nabla_{\mathbf{x}}\mathbf{h}_i[\hat{\mathbf{x}}(k|k)]$ are simply referred to as $\nabla_{\mathbf{x}}\mathbf{h}$ and $\nabla_{\mathbf{x}}\mathbf{h}_i$ respectively¹.

In order to implement SLAM in outdoor environments with the proposed multiple-line-of-sight prediction method and the automatic feature detectors (KS-CFAR pro-

¹Note that in this section, we often refer to an individual i^{th} feature's measurement Jacobian so that we can derive suitable individual measurement Jacobians $\nabla_{\mathbf{x}}\mathbf{h}_i$ which will be of particular use when considering multiple line-of-sight features. It should be noted however that the full measurement Jacobian used in the EKF update, is then given by the stacked matrix of these individual feature measurement Jacobians, given by equation 6.13.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

cessor and the TPP based feature detector), a simple three state vehicle model is used. The three states comprise the Cartesian coordinates (x_v, y_v) of the vehicle, centered midway between the rear wheels, and the vehicle orientation, θ_v . The model is based on a rear wheel drive vehicle. $V(k)$ is the measured velocity of the rear wheels. ΔT is the time interval between states at times, k and $k + 1$. The one step vehicle prediction can be described as

$$\begin{bmatrix} \hat{x}_v(k+1|k) \\ \hat{y}_v(k+1|k) \\ \hat{\theta}_v(k+1|k) \end{bmatrix} = \begin{bmatrix} \hat{x}_v(k|k) \\ \hat{y}_v(k|k) \\ \hat{\theta}_v(k|k) \end{bmatrix} + \begin{bmatrix} V(k)\Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k)) \\ V(k)\Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k)) \\ \alpha(k) \end{bmatrix} \quad (6.15)$$

The extended Kalman filter requires the linearisation of the process model about the vehicle location. This is calculated by determining the Jacobians of the process model with respect to the full augmented state (equation 6.2) denoted $\nabla_{\mathbf{x}} \mathbf{f}$ and with respect to the input control vector $\nabla_{\mathbf{u}} \mathbf{f}[\hat{\mathbf{x}}(k|k)] = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}|_{\mathbf{x}=\hat{\mathbf{x}}(k|k)}$, which again for notational convenience is referred to as $\nabla_{\mathbf{u}} \mathbf{f}$. Thus, covariances are propagated through the process model in the extended Kalman filter form for non-linear systems according to:

$$\mathbf{P}(k+1|k) = \nabla_{\mathbf{x}} \mathbf{f} \mathbf{P}(k|k) \nabla_{\mathbf{x}}^T \mathbf{f} + \nabla_{\mathbf{u}} \mathbf{f} \mathbf{Q}(k) \nabla_{\mathbf{u}}^T \mathbf{f} \quad (6.16)$$

where $\mathbf{Q}(k) = \begin{bmatrix} \sigma_V^2 & 0 \\ 0 & \sigma_\alpha^2 \end{bmatrix}$ includes the variances of the forward velocity (σ_V^2) and the steering angle (σ_α^2), and it has been assumed that the process noise is uncorrelated with the vehicle state.

The $(4N + 4) \times (4N + 4)$ Jacobian of the process model, with respect to the full augmented state vector is

$$\nabla_{\mathbf{x}} \mathbf{f} = \begin{bmatrix} 1 & 0 & -V(k)\Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k)) & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 1 & V(k)\Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k)) & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.17)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

where the Jacobian matrix has been evaluated about the previous state estimate - *i.e.* $\mathbf{x} = \hat{\mathbf{x}}(k|k)$ and the previous input vector $\mathbf{u}(k)$.

The $(4N + 4) \times 2$ Jacobian of the process model, with respect to the control vector, is

$$\nabla_{\mathbf{u}} \mathbf{f} = \begin{bmatrix} \Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k)) & -V(k) \Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k)) \\ \Delta T \sin(\hat{\theta}_v(k|k) + \alpha(k)) & V(k) \Delta T \cos(\hat{\theta}_v(k|k) + \alpha(k)) \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \quad (6.18)$$

where this Jacobian matrix has also been evaluated about the previous state estimate - *i.e.* $\mathbf{x} = \hat{\mathbf{x}}(k|k)$ and the previous input vector $\mathbf{u}(k)$.

Equation 6.12 shows that the predicted observation covariance matrix $\mathbf{S}(k + 1)$, depends on the Jacobian matrix $\nabla_{\mathbf{x}} \mathbf{h}$ which is a stacked matrix of rows corresponding to $\nabla_{\mathbf{x}} \mathbf{h}_i$, containing the partial derivatives of the vehicle to i^{th} feature measurement model, with respect to the state $\mathbf{x}$, evaluated about the previously estimated state vector $\hat{\mathbf{x}}(k|k)$. This Jacobian consists of both the vehicle, $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ component (3×3) sub-matrix, single feature component, $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ (3×4) sub-matrix and (in the multiple line-of-sight features) the i^{th} measurement with respect to j^{th} feature state (3×4) partial derivative sub-matrix $\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i$. The off-line calculation of these measurement Jacobians requires a separate analysis, depending on whether or not multiple line-of-sight features are detected. This is the subject of the next two sections.

6.2.1 Single Line of Sight Feature Observations

Suppose at a particular pose, N features are expected. If $\hat{\beta}_i(k + 1|k) \neq \hat{\beta}_j(k + 1|k) \quad \forall 1 \leq i, j \leq N, \quad i \neq j$, then none of the features are predicted to line up. For clarity, this scenario is demonstrated in figure 6.1. In this case the i^{th} feature's

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

Figure 6.1: Single line-of-sight predicted features. The predicted feature locations are spaced in such a manner, relative to the robot's position, that none of them line up.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

$3 \times (4N + 4)$ observation Jacobian sub-matrix is given by

$$\nabla_{\mathbf{x}} \mathbf{h}_i = [\nabla_{\mathbf{x}_v} \mathbf{h}_i \quad \mathbf{0} \quad \mathbf{0} \quad \cdots \quad \nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i \quad \cdots \quad \mathbf{0}] \quad (6.19)$$

where the $\mathbf{0}$ terms represent 3×4 sub-matrices of zeros. $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ represents the 3×3 measurement Jacobian sub-matrix with respect to the vehicle state and $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ represents the 3×4 measurement Jacobian sub-matrix with respect to the i^{th} feature state. For clarity, the $(k + 1|k)$ indices were dropped in the above equation (hence the Jacobians were referred to as $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ and $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$) and will remain so in the rest of this section. Also, $\hat{R}_i(k + 1|k)$, $\hat{\beta}_i(k + 1|k)$ and $\hat{P}_i(k + 1|k)$ will be represented as $\hat{R}_i$, $\hat{\beta}_i$ and $\hat{P}_i$ respectively. Predicted feature locations $\hat{x}_{f_i}(k + 1|k)$ and $\hat{y}_{f_i}(k + 1|k)$, and their predicted normalised reflection and absorption cross sections, $\hat{\Upsilon}_{r_i}(k + 1|k)$ and $\hat{\Upsilon}_{a_i}(k + 1|k)$ will be represented as $\hat{x}_{f_i}$, $\hat{y}_{f_i}$, $\hat{\Upsilon}_{r_i}$ and $\hat{\Upsilon}_{a_i}$ respectively, and the predicted vehicle pose will be referred to as $\hat{x}_v$, $\hat{y}_v$ and $\hat{\theta}_v$.

Calculation of $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ — Single Line-of-Sight Features

In equation 6.19, the sub-matrix Jacobian $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ is given by:

$$\nabla_{\mathbf{x}_v} \mathbf{h}_i = \begin{bmatrix} \frac{\partial \hat{R}_i}{\partial \hat{x}_v} & \frac{\partial \hat{R}_i}{\partial \hat{y}_v} & \frac{\partial \hat{R}_i}{\partial \hat{\theta}_v} \\ \frac{\partial \hat{\beta}_i}{\partial \hat{x}_v} & \frac{\partial \hat{\beta}_i}{\partial \hat{y}_v} & \frac{\partial \hat{\beta}_i}{\partial \hat{\theta}_v} \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_v} & \frac{\partial \hat{P}_i}{\partial \hat{y}_v} & \frac{\partial \hat{P}_i}{\partial \hat{\theta}_v} \end{bmatrix} \quad (6.20)$$

where from equations 6.3 and 6.4

$$\frac{\partial \hat{R}_i}{\partial \hat{x}_v} = \frac{\hat{x}_v - \hat{x}_{f_i}}{\hat{R}_i}; \quad \frac{\partial \hat{R}_i}{\partial \hat{y}_v} = \frac{\hat{y}_v - \hat{y}_{f_i}}{\hat{R}_i}; \quad \frac{\partial \hat{R}_i}{\partial \hat{\theta}_v} = 0; \quad (6.21)$$

$$\frac{\partial \hat{\beta}_i}{\partial \hat{x}_v} = \frac{\hat{y}_v - \hat{y}_{f_i}}{\hat{R}_i^2}; \quad \frac{\partial \hat{\beta}_i}{\partial \hat{y}_v} = \frac{\hat{x}_v - \hat{x}_{f_i}}{\hat{R}_i^2}; \quad \frac{\partial \hat{\beta}_i}{\partial \hat{\theta}_v} = 1 \quad (6.22)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

and from equation 6.5, assuming that $\hat{\Upsilon}_{r_i}$ does not vary with the vehicle's pose:

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_v} = \frac{\partial \hat{P}_i}{\partial \hat{R}_i} \frac{\partial \hat{R}_i}{\partial \hat{x}_v} = \frac{K A_I \hat{\Upsilon}_{r_i} (\hat{x}_{f_i} - \hat{x}_v)}{4\pi^2 \hat{R}_i^6}; \quad \frac{\partial \hat{P}_i}{\partial \hat{y}_v} = \frac{\partial \hat{P}_i}{\partial \hat{R}_i} \frac{\partial \hat{R}_i}{\partial \hat{y}_v} = \frac{K A_I \hat{\Upsilon}_{r_i} (\hat{y}_{f_i} - \hat{y}_v)}{4\pi^2 \hat{R}_i^6}; \quad (6.23)$$

$$\frac{\partial \hat{P}_i}{\partial \hat{\theta}_v} = \frac{\partial \hat{P}_i}{\partial \hat{R}_i} \frac{\partial \hat{R}_i}{\partial \hat{\theta}_v} = 0 \quad (6.24)$$

Note that at large predicted range values $\hat{R}_i$, $\frac{\partial \hat{P}_i}{\partial \hat{x}_v}$ and $\frac{\partial \hat{P}_i}{\partial \hat{y}_v} \approx 0$.

Calculation of $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ — Single Line-of-Sight Features

In equation 6.19, the sub-matrix Jacobian $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ is given by:

$$\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i = \begin{bmatrix} \frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{R}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{a_i}} \\ \frac{\partial \hat{\beta}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{a_i}} \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{P}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_i}} \end{bmatrix} \quad (6.25)$$

where from equations 6.3, 6.4 and 6.5:

$$\frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} = \frac{\hat{x}_{f_i} - \hat{x}_v}{\hat{R}_i}; \quad \frac{\partial \hat{R}_i}{\partial \hat{y}_{f_i}} = \frac{\hat{y}_{f_i} - \hat{y}_v}{\hat{R}_i}; \quad \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{r_i}} = 0; \quad \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{a_i}} = 0; \quad (6.26)$$

$$\frac{\partial \hat{\beta}_i}{\partial \hat{x}_{f_i}} = \frac{\hat{y}_{f_i} - \hat{y}_v}{\hat{R}_i^2}; \quad \frac{\partial \hat{\beta}_i}{\partial \hat{y}_{f_i}} = \frac{\hat{x}_v - \hat{x}_{f_i}}{\hat{R}_i^2}; \quad \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{r_i}} = 0; \quad \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{a_i}} = 0; \quad (6.27)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_{f_i}} = \frac{\partial \hat{P}_i}{\partial \hat{R}_i} \frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} = \frac{K A_I \hat{\Upsilon}_{r_i} (x_v - x_{f_i})}{4\pi^2 \hat{R}_i^6}; \quad \frac{\partial \hat{P}_i}{\partial \hat{y}_{f_i}} = \frac{\partial \hat{P}_i}{\partial \hat{R}_i} \frac{\partial \hat{R}_i}{\partial \hat{y}_{f_i}} = \frac{K A_I \hat{\Upsilon}_{r_i} (y_v - y_{f_i})}{4\pi^2 \hat{R}_i^6}; \quad (6.28)$$

$$\frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_i}} = \frac{K A_I}{(4\pi)^2 \hat{R}_i^4}; \quad \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_i}} = 0 \quad (6.29)$$

All of the partial derivatives derived in this section, allow the full i^{th} observation Jacobian matrix of equation 6.19 be symbolically calculated off-line, and numerically updated on each cycle of the EKF. The full Jacobian measurement matrix $\nabla_{\mathbf{x}} \mathbf{h}$ is then a stacked matrix of all the individual measurement Jacobian matrices $\nabla_{\mathbf{x}} \mathbf{h}_i$ for all $1 \leq i \leq N$.

6.2.2 Multiple Line of Sight Feature Observations

Suppose at a particular pose, N features are expected. If

$$\hat{\beta}_i \approx \hat{\beta}_j \approx \dots \approx \hat{\beta}_M \quad \forall 1 \leq i, j, \dots, M \leq N, \quad i \neq j \neq \dots \neq M, \quad (6.30)$$

then at least two (but possibly more) of the features are predicted to line up. This is possible at multiple angles, as shown in figure 6.2. Each group of approximately co-aligned predicted features, is associated with a particular predicted bearing angle $\hat{\beta}_i$ and forms a Q -tuple $\mathbf{B}_i = \{f_i \ f_j \ \dots \ f_M\}_{\hat{\beta}=\hat{\beta}_i}$ where Q is the number of features which co-align at that bearing angle, and hence the number of elements in $\mathbf{B}_i$. If equation 6.30 is satisfied, then each of the features within $\mathbf{B}_i$ must have its original predicted range $\hat{R}_i$ and state $\mathbf{F}_i = [\hat{x}_{f_i} \ \hat{y}_{f_i} \ \hat{\Upsilon}_{r_i} \ \hat{\Upsilon}_{a_i}]_{\hat{\beta}_i}$ temporarily copied into a new predicted range $\hat{d}_{q(i)}$ and state $\mathbf{G}_{q(i)} = [\hat{\mathcal{X}}_{q(i)} \ \hat{\mathcal{Y}}_{q(i)} \ \hat{\Gamma}_{r_{q(i)}} \ \hat{\Gamma}_{a_{q(i)}}]_{\hat{\beta}_i}$, where $\hat{d}_{q(i)} = \hat{R}_i$, $\hat{\mathcal{X}}_{q(i)} = \hat{x}_{f_i}$, $\hat{\mathcal{Y}}_{q(i)} = \hat{y}_{f_i}$, $\hat{\Gamma}_{r_{q(i)}} = \hat{\Upsilon}_{r_i}$, $\hat{\Gamma}_{a_{q(i)}} = \hat{\Upsilon}_{a_i}$ and $q(i)$ is the ranking $q(i) = 1, 2, 3, \dots, Q$ of the feature f_i in $\mathbf{B}_i$, according to its predicted distance $\hat{R}_i$ from the vehicle. This transformation is necessary, since the received power from an occluded feature, depends on the states of the occluding features, as well as the state of the occluded feature of interest. Hence, all occluding features, must have their states temporarily redefined in *ascending range*

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

Figure 6.2: Multiple line-of-sight features. The shaded yellow regions show angular sectors in which more than one feature approximately line up, with respect to the vehicle's current pose. each shaded region corresponds to a particular predicted bearing angle $\hat{\beta}_i$ and forms a tuple $\mathbf{B}_i$ of Q approximately co-aligned features.

order from the vehicle. Hence, if feature f_i has predicted range $\hat{R}_i$, and this is the minimum predicted range of all the features in Q -tuple $\mathbf{B}_i$, then for feature f_i , $q(i) = 1$; if it has the second smallest range value, $q(i) = 2$; if it has the maximum predicted range value of the tuple, $q(i) = Q$ etc. The above method, allows us to form a mapping from the $\mathbf{F}_i$ feature state space to the $\mathbf{G}_{q(i)}$ feature state space as follows

$$\begin{array}{llll}
 \mathbf{F}_i \text{ Space} & \longrightarrow & \mathbf{G}_{q(i)} \text{ Space} & (6.31) \\
 \text{Feature: Range:} & \text{Feature State:} & \longrightarrow & q(i) \text{ Range: Feature State:} \\
 f_i & \hat{R}_i & [\hat{x}_{f_i} \ \hat{y}_{f_i} \ \hat{\Upsilon}_{r_i} \ \hat{\Upsilon}_{a_i}]^T & \longrightarrow \ q \ \hat{d}_{q(i)} \ [\hat{\mathcal{X}}_{q(i)} \ \hat{\mathcal{Y}}_{q(i)} \ \hat{\Gamma}_{r_{q(i)}} \ \hat{\Gamma}_{a_{q(i)}}]^T
 \end{array}$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

As an illustration, figure 6.3 shows an angular sector which contains four approximately co-aligned features. From the figure it can be seen that at predicted bearing

Figure 6.3: Example: A tuple of 4 features, approximately co-aligned at bearing angle $\hat{\beta}_2$.

angle $\hat{\beta} \approx \hat{\beta}_2$, 4 features, f_4 , f_2 , f_9 and f_7 are approximately co-aligned. Hence, in this case, the Q -tuple B_i is a *quadruple* B_2 and the mapping from the F_i feature space to the $G_{q(i)}$ feature space as follows

F_i Space				$\longrightarrow$	$G_{q(i)}$ Space		(6.32)
Feature:	Range:	Feature State:	$\longrightarrow$	$q(i)$	Range:	Feature State:	
f_4	$\hat{R}_4$	$[\hat{x}_{f_4} \ \hat{y}_{f_4} \ \hat{\Upsilon}_{r_4} \ \hat{\Upsilon}_{a_4}]^T$	$\longrightarrow$	1	$\hat{d}_1$	$[\hat{\mathcal{X}}_1 \ \hat{\mathcal{Y}}_1 \ \hat{\Gamma}_{r_1} \ \hat{\Gamma}_{a_1}]^T$	
f_2	$\hat{R}_2$	$[\hat{x}_{f_2} \ \hat{y}_{f_2} \ \hat{\Upsilon}_{r_2} \ \hat{\Upsilon}_{a_2}]^T$	$\longrightarrow$	2	$\hat{d}_2$	$[\hat{\mathcal{X}}_2 \ \hat{\mathcal{Y}}_2 \ \hat{\Gamma}_{r_2} \ \hat{\Gamma}_{a_2}]^T$	
f_9	$\hat{R}_9$	$[\hat{x}_{f_9} \ \hat{y}_{f_9} \ \hat{\Upsilon}_{r_9} \ \hat{\Upsilon}_{a_9}]^T$	$\longrightarrow$	3	$\hat{d}_3$	$[\hat{\mathcal{X}}_3 \ \hat{\mathcal{Y}}_3 \ \hat{\Gamma}_{r_3} \ \hat{\Gamma}_{a_3}]^T$	
f_7	$\hat{R}_7$	$[\hat{x}_{f_7} \ \hat{y}_{f_7} \ \hat{\Upsilon}_{r_7} \ \hat{\Upsilon}_{a_7}]^T$	$\longrightarrow$	4	$\hat{d}_4$	$[\hat{\mathcal{X}}_4 \ \hat{\mathcal{Y}}_4 \ \hat{\Gamma}_{r_4} \ \hat{\Gamma}_{a_4}]^T$	

This formulation allows us to correctly predict the received power $\hat{P}_i$ from the

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

i -th feature, when it is occluded by other features, under a multiple-line-of sight scenario. If equation 6.30 is satisfied (there are co-aligned features), the above mapping from $\mathbf{F}_i$ space to $\mathbf{G}_{q(i)}$ space must be carried out for each feature f_i in each Q -tuple $\mathbf{B}_i$ (each angular segment, at angle $\hat{\beta}_i$, in which features approximately co-align). This means that each such feature f_i , will have a corresponding value of $q(i)$ and a corresponding feature in $\mathbf{G}_{q(i)}$ feature space with predicted range $\hat{d}_{q(i)}$ and state $[\hat{\mathbf{x}}_{q(i)} \ \hat{\mathbf{y}}_{q(i)} \ \hat{\Gamma}_{r_{q(i)}} \ \hat{\Gamma}_{a_{q(i)}}]_{\hat{\beta}_i}^T$.

In chapter 5, equation 5.14 the predicted power received from an occluded target, in terms of its range, reflection and absorption coefficients, and those of the objects occluding it, was derived. This equation can now be applied, and used to replace the the predicted power received from single line-of-sight features in equation 6.5, for the more general case of multiple line-of-sight features. For such features, equations 6.3 and 6.4 still apply for the predicted range and bearing of any occluded feature, but equation 6.5 must now be replaced by the following

$$\begin{aligned} \hat{P}_i(k+1|k) = & \frac{KA_I^{(2q(i)-1)}\hat{\Gamma}_{r_{q(i)}}(k+1|k)}{(4\pi)^{2q(i)}} \\ & \times \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l}(k+1|k) + \hat{\Gamma}_{a_l}(k+1|k))]^2}{(\hat{d}_{l+1}(k+1|k) - \hat{d}_l(k+1|k))^4} \end{aligned} \quad (6.33)$$

where $\hat{\Gamma}_{r_0}(k+1|k) = \hat{\Gamma}_{a_0}(k+1|k) = \hat{d}_0(k+1|k) = 0$. With the relaxed notation introduced earlier (dropping the $(k+1|k)$ indices)

$$\hat{P}_i = \frac{KA_I^{(2q(i)-1)}\hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \times \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l} + \hat{\Gamma}_{a_l})]^2}{(\hat{d}_{l+1} - \hat{d}_l)^4} \quad (6.34)$$

where $\hat{\Gamma}_{r_0} = \hat{\Gamma}_{a_0} = \hat{d}_0 = 0$. Note that equation 6.5 in section 6.2.1 is a particular case of equation 6.34 when there is only a single line-of-sight feature - i.e. $q(i) = 1$. Note that equation 6.34 operates on the *transformed* feature state space $\mathbf{G}_{q(i)}$, since the range order of the features which occlude f_i is necessary. Equations 6.3, 6.4 and 6.34 outline the *multiple-line-of-sight measurement model* for a general imaging radar. Due to the

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

multiple range dependencies in equation 6.34, the received power is actually a function of the vehicle and multiple feature states. This extra observation term (the received power), not normally considered in general SLAM formulations, therefore introduces further correlations between states and may be used to improve the accuracy of SLAM.

Measurement Jacobians – Multiple Line-of-Sight Features

In the multiple line-of-sight case, the off-line calculation of the terms related to the predicted received power in the measurement Jacobian becomes somewhat more complicated. This is because in the single line-of-sight feature case, the observation of a particular feature f_i affected *only* the particular Jacobian sub-matrix $\nabla_{\mathbf{x}} \mathbf{h}_i$. For multiple line-of-sight features, this is no longer true. If a feature f_i is sensed, and determined to be part of a Q -tuple $\mathbf{B}_i$, then *all* of the Q Jacobian sub-matrices $\nabla_{\mathbf{x}} \mathbf{h}_i, \nabla_{\mathbf{x}} \mathbf{h}_j, \dots, \nabla_{\mathbf{x}} \mathbf{h}_M$ within tuple $\mathbf{B}_i$ must be *jointly* calculated and updated. Each Jacobian sub-matrix is now dependent not only on the single feature of interest's (and vehicle's) state estimates but also on the state estimates of those other features which occlude it. This is a consequence of equation 6.34, since the received power from feature f_i depends not only on the state parameters of f_i , but on those of the other occluding features. The $3 \times (4N + 4)$ observation Jacobian for the i^{th} feature is therefore given by

$$\nabla_{\mathbf{x}} \mathbf{h}_i = [\nabla_{\mathbf{x}_v} \mathbf{h}_i \quad 0 \cdots 0 \quad \nabla_{\mathbf{x}_{f_1}} \mathbf{h}_i \quad 0 \cdots 0 \quad \nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i \quad 0 \cdots 0 \quad \nabla_{\mathbf{x}_{f_M}} \mathbf{h}_i \quad 0 \cdots 0] \quad (6.35)$$

where features $f_i, f_j, \dots, f_M$ are considered to be approximately co-aligned, according to equation 6.30. $\nabla_{\mathbf{x}_{f_1}} \mathbf{h}_i, \nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i, \dots, \nabla_{\mathbf{x}_{f_M}} \mathbf{h}_i$ represent the 3×4 Jacobian sub-matrices of the measurement from feature f_i , with respect to the states of features $f_i, f_j, \dots, f_M$, since there is now interaction. These must be calculated *jointly*. The derivation of the Jacobian sub matrices $\nabla_{\mathbf{x}_v} \mathbf{h}_i, \nabla_{\mathbf{x}_{f_1}} \mathbf{h}_i, \dots, \nabla_{\mathbf{x}_{f_M}} \mathbf{h}_i$, for the multiple line-of-sight case now follows.

Calculation of $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ — Multiple Line-of-Sight Features

As in section 6.2.1, in equation 6.35, the sub-matrix Jacobian $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ is given by

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

(repeated here for convenience):

$$\nabla_{\mathbf{x}_v} \mathbf{h}_i = \begin{bmatrix} \frac{\partial \hat{R}_i}{\partial \hat{x}_v} & \frac{\partial \hat{R}_i}{\partial \hat{y}_v} & \frac{\partial \hat{R}_i}{\partial \hat{\theta}_v} \\ \frac{\partial \hat{\beta}_i}{\partial \hat{x}_v} & \frac{\partial \hat{\beta}_i}{\partial \hat{y}_v} & \frac{\partial \hat{\beta}_i}{\partial \hat{\theta}_v} \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_v} & \frac{\partial \hat{P}_i}{\partial \hat{y}_v} & \frac{\partial \hat{P}_i}{\partial \hat{\theta}_v} \end{bmatrix} \quad (6.36)$$

the first two rows of which are still given by equations 6.21 and 6.22.

To calculate the partial derivatives in the third row, each multiple line-of-sight feature f_i in a particular Q -tuple $\mathbf{B}_i$ must be transformed from its $\mathbf{F}_i$ space, into its equivalent $\mathbf{G}_{q(i)}$ space. This results in a set of feature states $[\hat{\mathcal{X}}_{q(i)} \quad \hat{\mathcal{Y}}_{q(i)} \quad \hat{\Gamma}_{r_{q(i)}} \quad \hat{\Gamma}_{a_{q(i)}}]^T$ arranged in ascending predicted range order $\hat{d}_{q(i)}$ ($q(i) = 1, 2, 3, \dots, Q$) from the vehicle. Using this transformation, the chain rule of differentiation must be applied to equation 6.34 to determine the last row of partial derivatives in equation 6.36. This is because $\hat{P}_i$ depends on the range values of *all* the features which occlude feature f_i , as well as the range of feature f_i itself which, in turn, all depend on the predicted vehicle pose $\hat{\mathbf{x}}_v$. Hence the first term, in the last row of equation 6.36, for the i^{th} feature, can be written in terms of its transformed range and state variables, $\hat{d}_{q(i)}$ and $[\hat{\mathcal{X}}_{q(i)} \quad \hat{\mathcal{Y}}_{q(i)} \quad \hat{\Gamma}_{r_{q(i)}} \quad \hat{\Gamma}_{a_{q(i)}}]^T$

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_v} = \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_1} \right) \left(\frac{\partial \hat{d}_1}{\partial \hat{x}_v} \right) + \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_2} \right) \left(\frac{\partial \hat{d}_2}{\partial \hat{x}_v} \right) + \dots + \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(i)}} \right) \left(\frac{\partial \hat{d}_{q(i)}}{\partial \hat{x}_v} \right) \quad (6.37)$$

From equation 6.34

$$\frac{\partial \hat{P}_i}{\partial \hat{d}_1} = \frac{4KA_I \hat{\Gamma}_{r_1}}{(4\pi)} \frac{[1 - (\hat{\Gamma}_{r_0} + \hat{\Gamma}_{a_0})]^2}{(\hat{d}_1 - \hat{d}_0)^4 (\hat{d}_1 - \hat{d}_0)}. \quad (6.38)$$

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

and from equation 6.21 with the application of the above notation

$$\frac{\partial \hat{d}_1}{\partial \hat{x}_v} = \left(\frac{\hat{\mathcal{X}}_1 - \hat{x}_v}{\hat{d}_1} \right) \quad (6.39)$$

Hence

$$\begin{aligned} \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_1} \right) \left(\frac{\partial \hat{d}_1}{\partial \hat{x}_v} \right) &= \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_1} \right) \left(\frac{\partial \hat{R}_{i(1)}}{\partial \hat{x}_v} \right) \\ &= \frac{4K A_I \hat{\Gamma}_{r_1}}{(4\pi)} \frac{[1 - (\hat{\Gamma}_{r_0} + \hat{\Gamma}_{a_0})]^2}{(\hat{d}_1 - \hat{d}_0)^4 (\hat{d}_1 - \hat{d}_0)} \cdot \left(\frac{\hat{\mathcal{X}}_1 - \hat{x}_v}{\hat{d}_1} \right) \end{aligned} \quad (6.40)$$

Similarly

$$\left(\frac{\partial \hat{P}_i}{\partial \hat{d}_2} \right) \left(\frac{\partial \hat{d}_2}{\partial \hat{x}_v} \right) = \frac{4K A_I^3 \hat{\Gamma}_{r_2}}{(4\pi)^2} \frac{[1 - (\hat{\Gamma}_{r_0} + \hat{\Gamma}_{a_0})]^2 [1 - (\hat{\Gamma}_{r_1} + \hat{\Gamma}_{a_1})]^2}{(\hat{d}_1 - \hat{d}_0)^4 (\hat{d}_2 - \hat{d}_1)^4 (\hat{d}_2 - \hat{d}_1)} \cdot \left(\frac{\hat{\mathcal{X}}_2 - \hat{x}_v}{\hat{d}_2} \right) \quad (6.41)$$

and

$$\begin{aligned} \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(i)}} \right) \left(\frac{\partial \hat{d}_{q(i)}}{\partial \hat{x}_v} \right) &= \frac{4K A_I^{(2q(i)-1)} \hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{(2q(i))}} \\ &\times \frac{[1 - (\hat{\Gamma}_{r_0} + \hat{\Gamma}_{a_0})]^2 [1 - (\hat{\Gamma}_{r_1} + \hat{\Gamma}_{a_1})]^2 \dots [1 - (\hat{\Gamma}_{r_{q(i)-1}} + \hat{\Gamma}_{a_{q(i)-1}})]^2}{(\hat{d}_1 - \hat{d}_0)^4 (\hat{d}_2 - \hat{d}_1)^4 (\hat{d}_3 - \hat{d}_2)^4 \dots (\hat{d}_{q(i)} - \hat{d}_{q(i)-1})^4 (\hat{d}_{q(i)} - \hat{d}_{q(i)-1})} \\ &\times \left(\frac{\hat{\mathcal{X}}_{q(i)} - \hat{x}_v}{\hat{d}_{q(i)}} \right) \end{aligned} \quad (6.42)$$

Hence equation 6.37 forms a sum of products, and in general $\frac{\partial \hat{P}_i}{\partial \hat{x}_v}$ can be written in

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

compact form as

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_v} = 4K \sum_{m=1}^{q(i)} \left\{ \frac{A_I^{(2m-1)} \Gamma_{r_m}}{(4\pi)^{2m}} \left(\frac{\hat{x}_m - \hat{x}_v}{\hat{d}_m(\hat{d}_m - \hat{d}_{m-1})} \right) \prod_{l=1}^m \frac{[1 - (\hat{\Gamma}_{r_{l-1}} + \hat{\Gamma}_{a_{l-1}})]^2}{(\hat{d}_l - \hat{d}_{l-1})^4} \right\} \quad (6.43)$$

With a similar analysis, it can be shown that

$$\frac{\partial \hat{P}_i}{\partial \hat{y}_v} = 4K \sum_{m=1}^{q(i)} \left\{ \frac{A_I^{(2m-1)} \Gamma_{r_m}}{(4\pi)^{2m}} \left(\frac{\hat{y}_m - \hat{y}_v}{\hat{d}_m(\hat{d}_m - \hat{d}_{m-1})} \right) \prod_{l=1}^m \frac{[1 - (\hat{\Gamma}_{r_{l-1}} + \hat{\Gamma}_{a_{l-1}})]^2}{(\hat{d}_l - \hat{d}_{l-1})^4} \right\} \quad (6.44)$$

where, as before, $\hat{\Gamma}_{r_0} = \hat{\Gamma}_{a_0} = \hat{d}_0 = 0$. Finally, since none of the variables in equation 6.34 depend on $\hat{\theta}_v$

$$\frac{\partial \hat{P}_i}{\partial \hat{\theta}_v} = 0 \quad (6.45)$$

Equations 6.21, 6.22, 6.43, 6.44 and 6.45 between them enable the Jacobian sub-matrix $\nabla_{\mathbf{x}_v} \mathbf{h}_i$ in equation 6.35 to be symbolically calculated off-line, and updated during EKF iteration.

Calculation of $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ — Multiple Line-of-Sight Features

As in section 6.2.1, in equation 6.35, the sub-matrix Jacobian $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ is given by (repeated here for convenience):

$$\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i = \begin{bmatrix} \frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{R}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{a_i}} \\ \frac{\partial \hat{\beta}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{a_i}} \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_{f_i}} & \frac{\partial \hat{P}_i}{\partial \hat{y}_{f_i}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_i}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_i}} \end{bmatrix} \quad (6.46)$$

the first two rows of which are still given by equations 6.26 and 6.27.

As with the calculation of $\nabla_{\mathbf{x}_v} \mathbf{h}_i$, to calculate the partial derivatives in the third row,

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

each multiple line-of-sight feature f_i in a particular Q -tuple B_i must be transformed from its F_i space, into its equivalent $G_{q(i)}$ space, again resulting in a set of features states $\{\hat{\mathcal{X}}_{q(i)} \quad \{\hat{y}_{q(i)} \quad \hat{\Gamma}_{r_{q(i)}} \quad \hat{\Gamma}_{a_{q(i)}}\}$ arranged in ascending predicted range order $\hat{d}_{q(i)}$ ($q(i) = 1, 2, 3, \dots, Q$) from the vehicle.

For the first term in the last row of the matrix equation 6.46,

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_{f_i}} = \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(i)}} \right) \left(\frac{\partial \hat{d}_{q(i)}}{\partial \hat{x}_{f_i}} \right) = \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(i)}} \right) \left(\frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} \right) \quad (6.47)$$

Note that, unlike section 6.2.2, the chain rule is not required here, since only the range value $\hat{d}_{q(i)}$ in equation 6.34 depends on $\hat{x}_{f_i}$. It can be shown in a similar manner to equation 6.42 (and with the use of equation 6.26)

$$\begin{aligned} \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(i)}} \right) \left(\frac{\partial \hat{R}_i}{\partial \hat{x}_{f_i}} \right) &= \frac{4K A_I^{(2q(i)-1)} \hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{(2q(i))}} \\ &\times \frac{[1 - (\hat{\Gamma}_{r_0} + \hat{\Gamma}_{a_0})]^2 [1 - (\hat{\Gamma}_{r_1} + \hat{\Gamma}_{a_1})]^2 \dots [1 - (\hat{\Gamma}_{r_{q(i)-1}} + \hat{\Gamma}_{a_{q(i)-1}})]^2}{(\hat{d}_1 - \hat{d}_0)^4 (\hat{d}_2 - \hat{d}_1)^4 (\hat{d}_3 - \hat{d}_2)^4 \dots (\hat{d}_{q(i)} - \hat{d}_{q(i)-1})^4 (\hat{d}_{q(i)} - \hat{d}_{q(i)-1})} \\ &\times \left(\frac{\hat{x}_v - \hat{x}_{f_i}}{\hat{d}_{q(i)}} \right) \end{aligned} \quad (6.48)$$

so that

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_{f_i}} = 4K \frac{A_I^{(2q(i)-1)} \hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \left(\frac{\hat{x}_{f_i} - \hat{x}_v}{\hat{d}_{q(i)}(\hat{d}_{q(i)} - \hat{d}_{q(i)-1})} \right) \prod_{l=1}^{q(i)} \frac{[1 - (\hat{\Gamma}_{r_{l-1}} + \hat{\Gamma}_{a_{l-1}})]^2}{(\hat{d}_l - \hat{d}_{l-1})^4} \quad (6.49)$$

With a similar analysis

$$\frac{\partial \hat{P}_i}{\partial \hat{y}_{f_i}} = 4K \frac{A_I^{(2q(i)-1)} \hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \left(\frac{\hat{y}_{f_i} - \hat{y}_v}{\hat{d}_{q(i)}(\hat{d}_{q(i)} - \hat{d}_{q(i)-1})} \right) \prod_{l=1}^{q(i)} \frac{[1 - (\hat{\Gamma}_{r_{l-1}} + \hat{\Gamma}_{a_{l-1}})]^2}{(\hat{d}_l - \hat{d}_{l-1})^4} \quad (6.50)$$

The third partial derivative of the last row of $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ can be determined directly from

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

equation 6.34

$$\frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_i}} = \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_{q(i)}}} = \frac{K A_I^{(2q(i)-1)}}{(4\pi)^{2q(i)}} \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l} + \hat{\Gamma}_{a_l})]^2}{(\hat{d}_{l+1} - \hat{d}_l)^4} \quad (6.51)$$

and (also directly from equation 6.34)

$$\frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_i}} = \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_{q(i)}}} = 0 \quad (6.52)$$

Equations 6.26, 6.27, 6.49, 6.50, 6.51 and 6.52 between them enable the Jacobian sub-matrix $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$ in equation 6.35 to be symbolically calculated off-line, and updated during EKF iteration.

Calculation of $\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i$, ($j \neq i$) — Multiple Line-of-Sight Features

Finally, to calculate the sub-matrix Jacobian terms in equation 6.35 other than $\nabla_{\mathbf{x}_i} \mathbf{h}_i$ and $\nabla_{\mathbf{x}_{f_i}} \mathbf{h}_i$, which arise due to the interaction between the received power observation and the multi-feature states in equation 6.34, it is necessary to now determine $\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i$, $j \neq i$ given by

$$\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i = \begin{bmatrix} \frac{\partial \hat{R}_i}{\partial \hat{x}_{f_j}} & \frac{\partial \hat{R}_i}{\partial \hat{y}_{f_j}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{r_j}} & \frac{\partial \hat{R}_i}{\partial \hat{\Upsilon}_{a_j}} \\ \frac{\partial \hat{\beta}_i}{\partial \hat{x}_{f_j}} & \frac{\partial \hat{\beta}_i}{\partial \hat{y}_{f_j}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{r_j}} & \frac{\partial \hat{\beta}_i}{\partial \hat{\Upsilon}_{a_j}} \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_{f_j}} & \frac{\partial \hat{P}_i}{\partial \hat{y}_{f_j}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_j}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_j}} \end{bmatrix} \quad (6.53)$$

For $i \neq j$, predicted measurements $\hat{R}_i$ and $\hat{\beta}_i$ are independent of the state variables of the j^{th} feature, meaning that all partial derivatives in the first two rows of equation 6.53 are zero. In this situation however, the last row of $\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i$ depends on $q(j)$, the

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

range order of feature f_j from the current vehicle's predicted pose, in relation to that of feature f_i .

if $q(j) > q(i)$:

$$\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i = \mathbf{0} \quad (6.54)$$

since, from the i^{th} measurement, no information is provided about features which are further down-line from the vehicle, as can be seen in equation 6.34¹. However

if $q(j) < q(i)$:

In this case, the i^{th} received power prediction is dependent on the state parameters of all features f_j which occlude feature f_i . The first term of the last row of equation 6.53 can be determined from

$$\frac{\partial \hat{P}_i}{\partial \hat{x}_{f_j}} = \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(j)}} \right) \left(\frac{\partial \hat{d}_{q(j)}}{\partial \hat{x}_{f_j}} \right) = \left(\frac{\partial \hat{P}_i}{\partial \hat{d}_{q(j)}} \right) \left(\frac{\partial \hat{R}_j}{\partial \hat{x}_{f_j}} \right) \quad (6.55)$$

which, by partially differentiating equation 6.34 with respect to $\hat{d}_{q(j)}$ ² and the use of equation 6.26, yields

$$\begin{aligned} \frac{\partial \hat{P}_i}{\partial \hat{x}_{f_j}} &= \frac{4KA_I^{(2q(i)-1)} \hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \left(\frac{(\hat{d}_{q(j)+1} - \hat{d}_{q(j)-1})(\hat{x}_v - \hat{x}_{f_j})}{\hat{d}_{q(j)}(\hat{d}_{q(j)} - \hat{d}_{q(j)-1})(\hat{d}_{q(j)+1} - \hat{d}_{q(j)})} \right) \\ &\quad \times \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l} + \hat{\Gamma}_{a_l})]^2}{(\hat{d}_{l+1} - \hat{d}_l)^4} \end{aligned} \quad (6.56)$$

¹Equation 6.34 contains only the state variables of features up to $q(i)$ in range order from the vehicle, which excludes the state variables of feature f_j , since $q(j) > q(i)$.

²since $q(j) < q(i)$, equation 6.34 must be differentiated as a product.

6.2 Multiple Line-of-Sight EKF-SLAM Formulation

and similarly

$$\frac{\partial \hat{P}_i}{\partial \hat{y}_{f_j}} = \frac{4KA_I^{(2q(i)-1)}\hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \left(\frac{(\hat{d}_{q(j)+1} - \hat{d}_{q(j)-1})(\hat{y}_v - \hat{y}_{f_j})}{\hat{d}_{q(j)}(\hat{d}_{q(j)} - \hat{d}_{q(j)-1})(\hat{d}_{q(j)+1} - \hat{d}_{q(j)})} \right) \quad (6.57)$$

$$\times \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l} + \hat{\Gamma}_{a_l})]^2}{(\hat{d}_{l+1} - \hat{d}_l)^4}$$

Direct partial differentiation of equation 6.34 with respect to $\hat{\Gamma}_{r_{q(j)}}$ and $\hat{\Gamma}_{a_{q(j)}}$ yields the third and fourth terms of the last row in equation 6.53

$$\frac{\partial \hat{P}_i}{\partial \hat{\Gamma}_{r_{q(j)}}} = \frac{\partial \hat{P}_i}{\partial \hat{\Gamma}_{a_{q(j)}}} = \frac{-2KA_I^{(2q(i)-1)}\hat{\Gamma}_{r_{q(i)}}}{(4\pi)^{2q(i)}} \left(\frac{1}{1 - (\hat{\Gamma}_{r_{q(j)}} + \hat{\Gamma}_{a_{q(j)}})} \right) \quad (6.58)$$

$$\times \prod_{l=0}^{q(i)-1} \frac{[1 - (\hat{\Gamma}_{r_l} + \hat{\Gamma}_{a_l})]^2}{(\hat{d}_{l+1} - \hat{d}_l)^4}$$

Hence the Jacobian sub-matrix $\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i$ in equation 6.35 is

$$\nabla_{\mathbf{x}_{f_j}} \mathbf{h}_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\partial \hat{P}_i}{\partial \hat{x}_{f_j}} & \frac{\partial \hat{P}_i}{\partial \hat{y}_{f_j}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{r_j}} & \frac{\partial \hat{P}_i}{\partial \hat{\Upsilon}_{a_j}} \end{bmatrix} \quad (6.59)$$

where, if $q(j) > q(i)$ all terms in the last row are zero, and if $q(j) < q(i)$, equations 6.56, 6.57 and 6.58 must be used, which now allows every term in equation 6.35 to be symbolically calculated off-line, and updated during EKF iteration.

The Jacobian $\nabla_{\mathbf{x}} \mathbf{h}$ consists of many zero terms enabling efficient innovation covariance and Kalman gain update calculations given in equations 6.11 and 6.14. The non-zero terms align with the positions of the vehicle states and the observed feature states in the augmented state vector along with the occluded feature states. The *a pos-*

6.3 Autonomous Navigation with Synthetic Data

terior SLAM estimate is subsequently determined from the update equations 6.12. The observation model in equations 6.3, 6.4 and 6.33 correlates the feature estimate to the vehicle pose estimate and serves to reduce the uncertainty of both. While the Jacobians highlight the correlation between the vehicle location and the power measurements, the Υ_{r_i} and Υ_{a_i} predictions will be incorporated into the data association decisions shown later in section 6.5.

The next section demonstrates the key theoretical contributions of the radar based SLAM, explained so far. The implementation also serves to highlight a number of key properties of SLAM with MMW radar, and in particular, the utilisation of the additional information such as the estimated normalised RCS $\hat{\Upsilon}_{r_i}$ and the absorption RCS, $\hat{\Upsilon}_{a_i}$. The SLAM implementation shows how generally non-linear vehicle and radar observation models will be incorporated in the algorithm and how landmarks are initialised and tracked as the filter proceeds.

6.3 Autonomous Navigation with Synthetic Data

To demonstrate the theory explained in the previous sections, a SLAM experiment is conducted using synthetic data. The synthetic data set has the advantage that the ground truth is known (which is often unknown or difficult to determine for the real-world data sets), and that it facilitates experiments on scaling the approach to different environment sizes and complexities. The environment consists of two (non-metallic) features located at (20,10) and (30,10). The features are omni-directional point landmarks in the radar scans, having constant RCSs. The normalised RCS, Υ_{r_1} and absorption RCS, Υ_{a_1} (ground truth) values of the first feature (feature at location (20,10)) are 0.2 and 0.5 respectively. The normalised RCS, Υ_{r_2} and absorption RCS, Υ_{a_2} (ground truth) values of the second feature (feature at location (30,10)) are 0.3 and 0.4 respectively.

Figure 6.4 shows the ground truth of the vehicle path and the locations that correspond to the reflections received by the radar. The two point features reflect the radar signals to produce range, bearing and power observations (as given in the observation model, equation 6.6). The state is initialised by using the inverse observation equations (equations 6.3, 6.4 and 6.5) to get x_f , y_f , Υ_r and Υ_a states of features. The

6.3 Autonomous Navigation with Synthetic Data

Figure 6.4: Ground truth of the vehicle path and feature locations. The landmarks are marked with red dots. The vehicle path is shown in red solid line. The vehicle starts at co-ordinates (0,0) and the end co-ordinates were (7,2) between which the vehicle had traversed just over a complete circle as shown.

6.3 Autonomous Navigation with Synthetic Data

radar observations are generated as explained in chapter 5, by using the ground truth information and adding Gaussian noise onto (range equation 6.3, bearing equation 6.4 and power equation 6.5) observations. The standard deviation of radar noise in range, bearing and power are 0.5m, 0.5° and 1.5 dB respectively. The standard deviation of vehicle acceleration and heading are assumed to be 0.1 m/s^2 and $0.5^\circ/\text{s}$.

Figure 6.5: The “true” vehicle path together with the exact feature locations (ground truth) and the estimated vehicle path. Estimated feature locations are also shown.

The vehicle starts at location (0,0), and executes a circular loop, while scanning the radar beam 360° in azimuth before stopping at location (7,2). The synthetic radar data obtained contains amplitude signals (power spectral density), corresponding to returns at different ranges, at angular increments of approximately 1.8° . Two spectra that contains reflections from the features are shown in figures 6.6 and 6.7. These spectra are obtained from a radar scan taken from the origin. The spectra are passed through an adaptive feature detector (in this particular experiment, the proposed KS-CFAR processor (explained in chapter 3) is used) to obtain a measurement of range, bearing and power to a target. Figure 6.5 shows the overall results of the SLAM algorithm. The estimated landmark locations are designated by blue plus (+) signs. The actual

6.3 Autonomous Navigation with Synthetic Data

(true) landmark locations are designated by red dots. The true vehicle path is shown with a red solid line. The estimated vehicle path is also shown with blue solid lines.

Figure 6.6: The spectrum obtained from the initial vehicle location which contains the first feature.

When the two features are not lined up (*i.e.* $\hat{\beta}_i \neq \hat{\beta}_j$), then the observation Jacobian sub-matrix is given by equation 6.19. The calculation of $\nabla_{\mathbf{x}} \mathbf{h}_i$ is carried out as shown in subsection 6.2.1. However, when the vehicle reaches step numbers 18 and 58, the two features will line up together. *i.e.* at vehicle steps 18 and 58, $\hat{\beta}_i \approx \hat{\beta}_j$ as shown in equation 6.30. The radar spectrum obtained from location 18 is shown in figure 6.8, which show the reflections from both the features. The measurement Jacobians, $\nabla_{\mathbf{x}} \mathbf{h}_i$ are calculated as explained in subsection 6.2.2. The measurement Jacobians will change from the single line-of-sight features to multiple line-of-sight features at vehicle steps 18 and 58. The instances where the features are lined up (vehicle steps 18 and 58), are marked in the localisation and map building error plots.

6.3.1 Vehicle Localisation Results

The differences between the "true" vehicle path shown in figure 6.4 and the path estimated from the SLAM algorithm are too small to be seen on the scale used in figure 6.5. Figures 6.9, 6.10 and 6.11 shows the error between true and SLAM estimated vehicle positions in more detail. These figures show the actual error in estimated vehicle

6.3 Autonomous Navigation with Synthetic Data

Figure 6.7: The spectrum obtained from the initial vehicle location which contains the second feature.

Figure 6.8: The spectrum obtained from the vehicle step 18. The two features are lined up as shown in the spectrum.

6.3 Autonomous Navigation with Synthetic Data

Figure 6.9: Actual error in vehicle location estimate in x (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors.

Figure 6.10: Actual error in vehicle location estimate in y (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors.

6.3 Autonomous Navigation with Synthetic Data

Figure 6.11: Actual error in vehicle location estimate in θ (solid line) with 3σ confidence bounds (dotted lines) obtained from the estimated vehicle location errors.

location in x , y and θ as a function of time steps (the central solid line). The figure also shows 99.73% (3σ) confidence limits in the estimate error. These confidence bounds are derived from the state estimate co-variance matrix, given in equation 6.12, and represent the estimated vehicle error. The actual vehicle error is clearly bounded by the confidence limits of estimated vehicle error. The estimate produced by the SLAM algorithm is thus consistent. The estimated vehicle error as defined by the confidence bounds does not diverge so the estimates produced are stable.

6.3.2 Map Building Results

Figures 6.12, 6.13 and 6.14 shows the innovations in range, bearing and power observations (equation 6.12) together with the estimated 99.73% confidence limits. The innovation plots are useful in the sense that, they are the only available measure for analysing on-line filter behaviour when true vehicle states are unavailable. The innovations here indicate that the filter and models employed are consistent.

Figures 6.15, 6.16, 6.17 and 6.18 show the error between the actual and estimated landmark information for one of the radar reflectors (feature at location (20,20)). Figures also show the associated 99.73% (3σ) confidence limits, which are calculated

6.3 Autonomous Navigation with Synthetic Data

Figure 6.12: Innovation in range together with associated 3σ confidence bounds.

Figure 6.13: Innovation in bearing together with associated 3σ confidence bounds.

6.3 Autonomous Navigation with Synthetic Data

Figure 6.14: Innovation in Power together with associated 3σ confidence bounds.

Figure 6.15: Difference between the actual and estimated x location for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines).

6.3 Autonomous Navigation with Synthetic Data

Figure 6.16: Difference between the actual and estimated y location for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines).

Figure 6.17: Difference between the actual and estimated normalised RCS, Υ_{r_1} for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines).

6.3 Autonomous Navigation with Synthetic Data

Figure 6.18: Difference between the actual and estimated absorption RCS, Υ_{a1} for the first landmark. The 3σ confidence limit of the difference is also shown (dotted lines).

using the estimated landmark covariances. The error between normalised RCS estimate, $\hat{\Upsilon}_{r1}$ and its true value are well inside the 3σ limit as shown in figure 6.17. The landmark location and the normalised RCS estimates are thus consistent (and conservative) with actual landmark location errors being smaller than the estimated error as shown in the figures. The consistency in the normalised RCS is directly related to the initial uncertainty. If the initial variance of the normalised RCS is large, the estimates will become inconsistent, within few vehicle steps. However, if the initial variance is small, inconsistency grows slowly. The estimates will eventually become inconsistent if the SLAM is carried out for longer runs [6]. The difference between the absorption RCS estimate, $\hat{\Upsilon}_{a1}$ and its true value, with the 3σ confidence limits are shown in 6.18. As explained in section 6.2, the absorption RCS values are updated only when the features are lined up (features are lined up at vehicle steps 18 and 58). Thus the normalised absorption estimate error grows after an update at vehicle step 18 as shown in figure 6.18. The absorption RCS estimate is updated again at vehicle location 58, which reduces the absorption RCS estimate error as shown in the figure.

The implementation described here is, however, only a first step in the realisation of a fully autonomous navigation system using MMW radar. A number of substantive further issues such as reliable feature detection, data association and map management are discussed further in following sections.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

The proposed multiple-line-of-sight SLAM framework of this section incorporates, for the first time, the reflectivity and absorbability feature states into a general recursive estimation theoretic for any general feature (point, line, corner etc.). Since the power measurement from a feature is dependent on the viewing aspect of the sensor [47], the incorporation of power measurements into the proposed filter for such a variety of feature types requires some intuitive observations. In the case of a point target, of non-fluctuating mean power and assuming that the radar power-range equation 2.7 is satisfied, the power measurement from such a feature (equation 6.33) can be assumed omni-directionally constant. For a line feature, the perpendicular power measurement can be assumed constant, whereas diffuse/specular reflections as a result of off-normal measurements may influence the power measurement associated with that feature. Equally, the power measurement from a corner feature is constant at the point of intersection of the two lines, as at this point a corner reflector is emulated. Consequently, while non-point features may return multiple range/bearing/power measurements, the generic multi-model multi-line-of-sight filtering framework proposed in this section, can be realized.

To incorporate multi-model features into the framework, suitable feature extraction methods are required. The following section discusses the case of extracting line features from a radar scan.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

Figure 6.19 shows the test area within the campus of Nanyang Technological University (NTU) with the superimposed vehicle path obtained from the odometers. This outdoor test area provides a semi-structured environment for the evaluation of SLAM with the proposed prediction and feature detection methods. The objects present in the environment include trees, lamp-posts, pillars and walls. Power returns from bush and foliage are considered as clutter. It is the aim of this section to extract line features from the radar reflections of the buildings and walls. The Nearest Neighbor filter can

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

Figure 6.19: The test environment (NTU car park) with the superimposed vehicle path. Typical features include trees, lamp-posts, parked cars, buildings, concrete pillars etc.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

fail in providing correct data association, for closely spaced features. Conflict situations arise when multiple measurements fall within a single data association gate, or when a single measurement falls within the gates of more than one feature estimate. This leads to incorrect data association and will cause SLAM to diverge [61].

Figure 6.20: Five full 360° radar scans after constant thresholding, $T = \mu + 2\sigma$ with features marked.

In order to show a typical feature set, the robot maneuvers in the test environment for 20 time steps (3m approx). Radar scans are collected from the test environment by mounting the radar on to a vehicle, which can be remotely operated. A feature set from the test site (landmark positions) is shown in figure 6.20. The features were detected using a constant threshold detector with the threshold, $T = \mu + 2\sigma$. The constant

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

threshold with a relatively low T is used here to show all the probable features (trees, lamp-posts and pillars) along with clutter objects (bush and foliage). The feature and clutter reflections are marked in figure 6.20. To synthesise an environment of closely lying features, a point-feature only environment is assumed. A point feature clustering algorithm, with a low inter-cluster distance threshold, may be used to ‘create’ an environment with multiple closely lying features. These extracted features (reflections which originate from walls, buildings, fences etc) in the map, will cause incorrect data association and will cause SLAM to diverge.

To analyse the issues arising from closely lying targets, SLAM experiments with synthetic data are carried out here. SLAM is performed with two different data sets, one with point-features and the other with the same point-features, and some added reflections from a wall. The synthetic data was generated by the proposed spectra prediction method, explained in chapter 2. The test scenario consists of 8 landmarks lying in a square of dimensions $100 \times 100m^2$. The vehicle explores the environment from west to east and moves about 100m from its starting location. The maximum radar detection range is set at 50m. Figure 6.21a shows the SLAM results with point features only (8 point targets). The variance in the error in the vehicle are shown in figures 6.21b, 6.21c and 6.21d. The error plots show that the vehicle errors were bounded and the estimates remain consistent through out the experiment.

In order to find out the effect of closely lying features (which could be the reflections originating from line features such as walls and fences), a SLAM experiment is conducted with synthetic data, that contains point features (at the same location as the previous SLAM experiment), along with a wall. The SLAM run is shown in figure 6.22a. The vehicle’s x_v , y_v and θ_v error plots are shown in figures 6.22b, 6.22c and 6.22d respectively, which clearly shows SLAM divergence due to incorrect data associations. The standard Nearest Neighbour filter was unable to correctly associate features, causing divergence. This above result shows the necessity for multi-model filters for autonomous navigation. Multi-model measurement classifiers can be applied as in [53]. These often included lines, corners/edges and cylindrical objects. Several measurements are necessary from successive vehicle poses to ensure correct classification, which leads to navigation based on multiple hypotheses [18] [11] of the feature type. Often multiple filters must be spawned together, requiring largely heuristic pruning

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

Figure 6.21: SLAM simulation with synthetic data. Features include single, point features only.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

Figure 6.22: SLAM simulation with synthetic data. Features include single, point features and the reflections from a wall.

methods to avoid the computational complexity increasing without bound. Incorporating a feature selection method will, thus, reduce the computational complexity of SLAM with radar in outdoor environments. To extract line features from the data the well known Random Sample Consensus (RANSAC) algorithm is used in this work. It will be demonstrated next that the RANSAC algorithm can effectively identify radar reflections that originate from walls and buildings.

In this work, the feature extraction section for SLAM has two main modules: 1) data acquisition and target/feature detection (using the proposed KS-CFAR and TPP based feature detection methods), and 2) line feature detection using RANSAC. The feature detection steps can be summarised as follows. Firstly, feature locations are extracted from the raw radar scans (using the proposed feature detectors explained in chapter 3 and 4). The RANSAC algorithm is then applied to the extracted data, which

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

identifies reflections originating from line landmarks such as walls, buildings etc. The results show that the proposed approach is computationally efficient in un-structured as well as semi-structured outdoor environments. In the next section, the RANSAC algorithm for line detection is provided and the results obtained from various outdoor environments are shown. The efficiency of the RANSAC algorithm on radar data is then discussed.

6.4.1 Line Feature Detection using the RANSAC Algorithm

For detecting lines in radar scans, either the well known Hough transform or the RANSAC algorithm can be deployed. The RANSAC algorithm is chosen here as it has been reported in the literature [74], [2] that it is computationally less intensive compared to the Hough transform especially in heavily cluttered (reflections from uneven terrains, bush, foliage etc.) environments. The structure of the RANSAC algorithm is summarised here. Repeatedly, sampled point subsets are randomly selected from the input data and model parameters fitting the sample are computed [29]. The size of the random samples is the smallest sufficient for determining the model parameters. In a second step, the quality of the model parameters is evaluated on the full data set. Different cost functions may be used [83] for the evaluation, the standard being the number of inliers, *i.e.* the number of data points consistent with the model. The process is terminated when the likelihood of finding a better model becomes low. The strength of the method stems from the fact that, to find a good solution, it is sufficient to select a single random sample not contaminated by outliers. Depending on the complexity of the model (the size of random samples) RANSAC can handle contamination levels well above 50%, which is commonly assumed to be a practical limit in robust statistics [70]. The speed of RANSAC depends on two factors. Firstly, the level of contamination determines the number of random samples that have to be taken to guarantee a certain confidence in the optimality of the solution. Secondly, the time spent evaluating the quality of each of the hypothesized model parameters is proportional to the size N of the data set. Typically, a very large number of erroneous model parameters obtained from contaminated samples are evaluated. Such models are consistent with

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

only a small fraction of the data. This observation can be exploited to significantly increase the speed of the RANSAC algorithm.

The RANSAC algorithm can be summarised as follows:

Inputs:

$U = \{x_i | i = 1 \dots N\}$, where U is the set of data points, $|U| = N$

$f(S) : S \rightarrow p$ A function f computes model parameters, p given a sample S from U .

$\psi(p, x)$ is the cost function for a single data point x .

Output:

$\hat{p}$, parameters of the model maximizing the cost function, $\hat{C}$.

RANSAC Algorithm

- $k = 0$
- Repeat until $P\{\text{better solution exists}\} < \eta$ (a function of $\hat{C}$ and no. of steps k)
- $k = k + 1$
 - I. Hypothesis
 - 1) select randomly set $S_k \subset U, |S_k| = m$
 - 2) compute parameters $p_k = f(S_k)$
 - II. Verification
 - 3) compute cost $C_k = \sum_{x \in U} \psi(p_k, x)$
 - 4) if $\hat{C} < C_k$ then $\hat{C} = C_k, \hat{p} = p_k$
- end

To extract lines, the feature detectors derived in chapters 3 and 4 (*i.e.* constant threshold, KS-CFAR processor and TPP based feature detectors) are firstly applied to the complete radar scans. The features which are detected, are then used in the RANSAC algorithm to extract lines. In doing so RANSAC checks how many such

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

features lie close enough to this best fit line. If the number is above a particular threshold (≥ 8 in this implementation) then it is assumed that a line has been observed. This threshold is called the consensus. The algorithm is designed to iterate until the best line fit is found. The number of iterations, k , is chosen high enough to ensure that the probability p (set to 0.99) is such that at least one of the sets of random samples does not include an outlier [38]. Let u represent the probability that any selected data point is an inlier and $v = 1 - u$ the probability of observing an outlier. The number of iterations, k is given by

$$k = \frac{\log(1 - p)}{\log(1 - (1 - v)^m)} \quad (6.60)$$

where m is the minimum number of feature points selected at random, and $1 - p = (1 - u^m)^k$.

Next, the performance of the RANSAC algorithm, applied to features extracted (using a constant threshold ($T = \mu + 3\sigma$), KS-CFAR processor and the TPP based feature detectors) is shown. The RANSAC algorithm is applied to different radar data sets, collected from two different outdoor scenarios. If reflections from line features such as walls, buildings etc. are present in the data set then, after every successful line fit, those data are removed and RANSAC is repeated. Figure 6.23 shows the RANSAC result on radar data obtained from an outdoor stadium. The features in the data set include lamp-posts, trees, wall etc. It is shown in figure 6.23 that, the RANSAC was unable to find the wall from the data obtained using a constant threshold feature detector, due to the missed detections. The wall was successfully detected by RANSAC from the data obtained with the KS-CFAR processor and TPP based feature detectors as shown in 6.23.

RANSAC was also run on data collected from an outdoor field as shown in figure 6.24. The features include walls, metallic posts etc. The constant threshold ($T = \mu + 3\sigma$) missed the wall completely where as the KS-CFAR processor and the TPP based feature detector successfully detected it.

The results show that the RANSAC algorithm can be successfully applied to the radar data in order to achieve effective feature pruning. The features remaining after pruning will generally be point features and hence the proposed prediction model can be effectively applied.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

(a) RANSAC algorithm run on the data points extracted using a constant threshold ($T = \mu + 3\sigma$) from an outdoor radar scan. Due to higher number of missed detections (compared with the KS-CFAR and TPP methods), the RANSAC algorithm failed to identify the wall as shown.

(b) A wall is successfully detected by the RANSAC algorithm from an outdoor stadium. Features extracted by the KS-CFAR processor.

(c) A wall is successfully detected by the RANSAC algorithm from an outdoor stadium. Features were extracted using a TPP based feature detector.

Figure 6.23: The RANSAC algorithm run on data sets obtained from three different feature detectors. Due to the missed detections in the constant threshold feature extractor, the RANSAC algorithm failed in detecting the wall. However, due to the merits of the KS-CFAR processor and the TPP based feature detector, RANSAC was able to find the wall.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

(a) RANSAC algorithm run on the data points extracted using a constant threshold ($T = \mu + 3\sigma$) from an outdoor radar scan. Due to higher number of missed detections (compared with the KS-CFAR and TPP methods), the RANSAC algorithm failed to identify the wall as shown.

(b) A wall successfully detected by the RANSAC algorithm from an outdoor field. Features were extracted using a KS-CFAR processor.

(c) A wall successfully detected by the RANSAC algorithm from an outdoor field. Features were extracted using a TPP based feature detector.

Figure 6.24: The RANSAC algorithm run on three data sets obtained from three different feature detectors as shown. Features include metallic posts and walls. Due to the missed detections in the constant threshold detector, the RANSAC algorithm was not successful in detecting the wall. However, due to the merits of the other feature detectors, RANSAC was able to find the wall from points extracted by the KS-CFAR processor and the TPP based feature detector.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

(a) SLAM with the TPP based feature detector, with no line-clustering algorithm.

(b) SLAM with the TPP based feature detector, with the line-clustering algorithm to reject reflections from lines.

Figure 6.25: Analysis of the SLAM runs (100 steps) with TPP based feature detection, to verify the effect of the RANSAC based line removal method on vehicle navigation. Scans obtained from the laser are assumed to be the ground truth map and are included in the figure.

6.4 Automatic Feature Relevance Determination for Effective Vehicle Navigation

(a) Estimated errors and 3σ bounds obtained from EKF-SLAM with TPP based feature detection with no line-clustering algorithm.

(b) Estimated errors and 3σ bounds obtained from EKF-SLAM with RANSAC based line-removal to reject reflections from buildings.

Figure 6.26: Error plots obtained from the SLAM runs (100 steps) with TPP based feature detector, to verify the effect of the RANSAC based line feature removal method on vehicle navigation.

6.5 Data Association

Figures 6.25 and 6.26 show two SLAM runs, one without the RANSAC based line pruning and the other with the pruning, respectively. Features are detected using the proposed TPP based feature detector. The SLAM results with and without the line feature pruning method are shown for a short run (100 steps or approx. 9.6 metres) here. The SLAM results without the line feature removal scheme is shown in figure 6.25a. Scans obtained from the laser are considered to be the ground truth and are included in the figure. Figure 6.25b shows SLAM results with joint line and point feature extraction, which is included into the TPP based feature detection. Figure 6.25b shows an efficient feature management approach over the EKF-SLAM filter without line feature pruning. The estimated vehicle path and the odometry information are marked in the figures. The multi-model method based on RANSAC, reduces the number of point landmarks in the map from 38 (Figure 6.25a) to 10 (Figure 6.25b). Figures 6.26a and 6.26b show the error plots of the vehicle without and with the line feature extraction method respectively. Although the vehicle localisation errors are well within the 3σ limits, the vehicle error in the x and y axes fluctuates more in the SLAM result without line feature extraction method (6.26a) compared to the SLAM result with line feature extraction method (6.26b). These results along with the SLAM results using the synthetic data (provided earlier in section 6.4), show that the line feature extraction, based on the RANSAC algorithm, makes the SLAM implementation more robust to outdoor terrain navigation with MMW radar.

6.5 Data Association

General outdoor environments generally contain a wide variety of features which may be closely spaced resulting in association difficulties as outlined in the previous section. Successful data association involves association of the correct measurement with the correct feature state, detecting and initialising new tracks and rejecting clutter (bush and foliage). There are different techniques available for implementing the data association process [7]. Most existing techniques are based on an innovation and its corresponding covariance. The innovation $\nu_i(k+1)$ relates observations $\mathbf{z}_i(k+1)$ to the underlying state estimates. In particular, it is defined as the difference between the

6.5 Data Association

observation $\mathbf{z}_i(k+1)$ and the predicted observation based on the observation model and the predicted states $\hat{\mathbf{z}}_i(k+1|k)$ as shown in equation 6.8.

Thus, the normalised innovation is can be written as,

$$\mathbf{d}^2(k+1) = \nu^T(k+1)\mathbf{S}^{-1}(k+1)\nu(k+1) \quad (6.61)$$

Here $\mathbf{S}(k+1)$ is the innovation covariance matrix. It is well known that if the innovations $\nu(k+1)$ have a Gaussian distribution, then $\nu(k+1)\nu^T(k+1)$ will have a χ^2 distribution [7]. The innovation sequence is the basis of the ‘gate validation’ technique, which accepts the correct observations as the association with a given probability. This determination is achieved by comparing the scalar obtained in 6.61 with a threshold value that is determined by fixing the region of acceptance of the χ^2 distribution (for example, 95%).

One of the most common data association techniques used in the SLAM literature is the Nearest Neighbour (NN) filter. The standard NN filter uses the gate validation test to initially determine which landmark states are valid candidates to update with the observation received. Of all the possible landmarks within the validation gate, the one that is nearest (smallest distance, d^2) to the observation is selected. All the other possible hypotheses are ignored. The NN filter can fail to recover the true data association when validation gates overlap, and the observations fall within this overlapped region. In such cases, the NN filter can associate the observation with an incorrect state. Such false data associations are known to induce catastrophic failures in SLAM, and are common, especially in environments of closely lying features.

Additional information, *i.e.* the reflected power (or the reflectivity) from features can, however, be utilized to aid the Nearest Neighbour data association to correctly associate measurements with the states. This is verified here by evaluating the NN data association decision for two closely lying features. The scenario is as shown in figure 6.27 where the robot obtains a single measurement at every time step. The measurements thus obtained could have originated from either one of the features. The 3D representation of the same scenario is shown in figure 6.28.

Let $\mathbf{z}(k+1)$ denote an N -dimensional measurement of the feature state, which could have originated from either a target or a false alarm. Let $G_e = \{\nu^T(k+1)\mathbf{S}^{-1}(k+1)\nu(k+1) \leq \gamma\}$ be a validation gate and $\sqrt{\gamma}$ be the gate size [52]. A measurement

6.5 Data Association

Figure 6.27: The figure shows the Nearest Neighbour filter gating procedure for data association. The robot with position, $\hat{x}_v$ and the uncertainty represented by the ellipse has to associate an observation $z(k+1)$ to landmarks previously observed denoted and now predicted as $\hat{x}_1$ and $\hat{x}_2$.

falling inside the gate is referred to as a validated measurement. However, if the measurement is falling inside the region shared by both the gates, then the NN filter may wrongly associate the measurement to an incorrect predicted state.

The performance of NN data association is evaluated by carrying out multiple Monte Carlo runs. The normalised innovation covariance, G_e is then used to characterise the NN filter performance through 1000 Monte Carlo runs. Under the hypothesis that the filter is consistent, G_e is χ^2 (chi-square) distributed with $\dim\{z(k)\}$ degrees of freedom. In the case of a 3-dimensional measurement (range/bearing/power) then $\dim\{z(k)\} = 3$. Two separate filter runs were carried out in order to verify the usefulness of adding the additional reflected power information for successful data association. The first filter run was carried out with only the range-bearing measurements as is commonly the case, and the second with measurements that include the range, bearing and the reflected power measurements of the features, such as is the case for the reflectivity based recursion proposed in this section.

Figure 6.29a shows the normalised innovation covariance plots obtained from the filter with 1000 Monte Carlo steps. The measurement contains only range and bearing information from the first feature, thus any time the measurement is associated with

6.5 Data Association

Figure 6.28: The figure shows the Nearest Neighbour filter gating procedure for data association in 3D. Both predicted features lie close together in x, y space but are well separated in power space.

6.5 Data Association

feature 2, it is in error. The thick (red colour) line is the normalised innovation distance between the measurement and the first feature prediction ($\hat{x}_1$) and the thin (blue colour) line is the distance to the second feature prediction ($\hat{x}_2$). There are a large number of incorrect data associations as shown in the figure. The zoomed-in version of the same plot is shown in 6.29b. From the figures it is clear that the Nearest Neighbour filter has made numerous incorrect data associations.

Figure 6.30 shows the normalised innovation distance plots obtained from 1000 Monte Carlo filter runs when the observations include power returns, along with the range and bearing information. The plots show that the Nearest Neighbour filter could correctly associate the measurement with the first feature in all cases.

A comprehensive analysis of the additional information aiding the NN data association has been presented in this section. These results provide valuable insight into the problem of navigation in unstructured outdoor environments and the improvements that can be achieved by incorporating added sensory information.

The entire SLAM problem depends on maintaining complete information of the cross-correlation between landmark estimates. It is shown in [22] that as the robot moves through the environment, the errors in the feature estimates will become more correlated. Hence the covariance matrix update is important in SLAM. However, as the number of features in the state vector increases, the computation required to update the covariance matrix also increases [21]. The next section describes a method applicable to millimetre wave radar based SLAM to reduce the computational complexity of EKF-SLAM by feature management/selection.

In the next section, the proposed KS-CFAR and TPP based feature detectors are analysed from a SLAM perspective along with SLAM using constant threshold filtering. The merits of incorporating the augmented feature state measurement vector, coupled with the data association improvements depicted in this section, are highlighted in an outdoor environment.

6.5 Data Association

(a) The innovation covariance plots obtained from a SLAM run of 1000 steps without power information. The measurement $\mathbf{z}(k+1)$ has only the range-bearing information of one of the features. The innovation covariance plot shows that the nearest neighbour filter wrongly associates the measurements most of the time.

(b) Zoomed view of figure 6.29a with x coordinates (0,100) and y coordinates (0,16) respectively is shown here for better clarity.

Figure 6.29: The normalised innovation covariance obtained by 1000 Monte Carlo runs of the filter without the power information, to obtain a measure of correct data association.

6.5 Data Association

(a) The innovation covariance plots obtained from a SLAM run of 1000 steps with power information. The measurement $\mathbf{z}(k+1)$ has the range-bearing and reflected power information of one of the features. The innovation covariance plot shows that the nearest neighbour filter successfully associates the feature to its state more than 95% of the time.

(b) The zoomed in version of figure 6.30a of x and y coordinates (220,300) and (0,160) respectively is shown for more clarity, the area chosen arbitrarily.

Figure 6.30: The normalised innovation covariance obtained by 1000 Monte Carlo runs of the filter with the power information, to obtain a measure of correct data association.

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

Figure 6.31: The test vehicle with sensors, which include a MMW radar scanner, a 180° SICK laser scanner and odometers attached to its wheels.

The navigation system was tested with a vehicle fitted with a scanning radar, a forward-looking SICK laser sensor, odometers and a gyro which provides heading information. The vehicle used for the experiment is shown in Figure 6.31. The laser and radar are mounted in front and on top of the vehicle, respectively. The experimental run was performed in the car park of the university campus. The photo of the test environment is shown in figure 6.19. The total length of the trajectory of the robot was around 65m, lasting around 10 minutes, with an average speed of 12.5 cm/sec. During the robot motion, the radar sensor and the laser scanner were acquiring full range scans at average frequencies of 2Hz and 75Hz, respectively.

Next, SLAM with constant threshold based feature detection is shown. The constant threshold based feature detector is shown here as it has the lowest level of complexity among all feature detectors. The results are analysed and its merits and drawbacks are analysed. The discussion carried out in the next section will explain more

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

on the requirement of having a superior automatic feature detector which can perform well in cluttered and multi-target situations.

6.6.1 Outdoor Vehicle Navigation using Constant Threshold based Feature Detection

As mentioned above, the constant threshold based feature detection method has the lowest level of complexity, and hence requires less processing time. Since several power returns may exceed a threshold in a spectrum, the power return which is closest to the radar is usually taken. This reflected power is considered to originate from a valid feature and is considered for navigation as it's location is well defined.

The results in figure 6.32 correspond to the localization and mapping obtained using a constant threshold detector. A constant threshold of $\mu + 3\sigma$ is applied to the power spectra recorded while the vehicle is continuously moving at approx. 12.5cm/sec. After recording a full 360° scan, the feature detector is applied to the raw data to obtain the probable feature locations and the corresponding received power values. In order to prune the reflections occurring from walls and buildings, a RANSAC algorithm is applied to the feature set (obtained using the constant threshold).

The ellipsi in figure 6.32 indicate the 3σ uncertainty bound of each landmark and also the vehicle uncertainty. Figure 6.33 presents the 3σ (or 99.7%) confidence bounds of the estimated position of the vehicle with the true error¹, shown as continuous and dotted lines respectively. The uncertainty in the vehicles position estimate is found from the eigen values of the 2×2 x_v and y_v sub-matrix within the vehicle component

¹The true error is the difference between the vehicles estimated and, ground truth positions. The ground truth trajectory of the vehicle is obtained using manual scan matching by aligning the laser scans.

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

sub-matrix in $P(k+1|k+1)$ which is shown in equation 6.62,

$$P(k+1|k+1) = \begin{bmatrix} \begin{bmatrix} P_{x_v x_v} & P_{x_v y_v} \\ P_{y_v x_v} & P_{y_v y_v} \end{bmatrix} & P_{x_v \theta_v} & P_{x_v x_{f_1}} & \cdots & P_{x_v \gamma_{f_N}} & P_{x_v L} \\ \begin{bmatrix} P_{y_v x_v} & P_{y_v y_v} \end{bmatrix} & P_{y_v \theta_v} & P_{y_v x_{f_1}} & \cdots & P_{y_v \gamma_{f_N}} & P_{y_v L} \\ P_{\theta_v x_v} & P_{\theta_v y_v} & P_{\theta_v \theta_v} & P_{\theta_v x_{f_1}} & \cdots & P_{\theta_v \gamma_{f_N}} & P_{\theta_v L} \\ P_{x_{f_1} x_v} & P_{x_{f_1} y_v} & P_{x_{f_1} \theta_v} & P_{x_{f_1} x_{f_1}} & \cdots & P_{x_{f_1} \gamma_{f_N}} & P_{x_{f_1} L} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{\gamma_{f_N} x_v} & P_{\gamma_{f_N} y_v} & P_{\gamma_{f_N} \theta_v} & P_{\gamma_{f_N} x_{f_1}} & \cdots & P_{\gamma_{f_N} \gamma_{f_N}} & P_{\gamma_{f_N} L} \\ P_{L x_v} & P_{L y_v} & P_{L \theta_v} & P_{L x_{f_1}} & \cdots & P_{L \gamma_{f_N}} & P_{L L} \end{bmatrix} \quad (6.62)$$

It can be seen from the figure that the y_v and heading errors grow in an unbounded manner beyond the 3σ confidence bounds estimated by the filter. This is due to the incorrect inclusion of clutter targets, such as foliage and bushes, into the state vector. As mentioned earlier, constant threshold feature detection does not have any way to detect and reject clutter and hence incorrectly marks clutter as features. Also, in figure 6.32, when the vehicle is turning, the vehicle error grows larger. This is due to the approximate vehicle model and it is expected to have some systematic errors due to slip and steering non-linearities. Hence, systematic errors present in the detection and association of the landmarks and motion model errors also cause SLAM divergence.

Feature detection in the presence of clutter is challenging and the proposed feature detectors such as KS-CFAR processor and the TPP based feature detector out perform the constant threshold due to their superior feature detection/clutter rejection abilities. This will be shown in the next sections.

6.6.2 Outdoor Vehicle Navigation using the KS-CFAR Processor based Feature Detection

The second set of experimental results shows SLAM using features extracted by the KS-CFAR processor. The location and power return information are obtained by the KS-CFAR processor while the vehicle moves. The final accuracy of the SLAM map estimate is determined by the initial uncertainty of the vehicle and the quality of the

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

Figure 6.32: SLAM with constant threshold ($\mu + 3\sigma$) based feature detection in the outdoor car park. The solid lines represents the estimated vehicle path and the odometry. Laser scans are also plotted for ground truth.

Figure 6.33: Vehicle error plots obtained from an EKF-SLAM run with a constant threshold based feature detector.

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

combination of dead reckoning and external sensors and the ability of the KS-CFAR processor to successfully extract only *reliable features*¹. In these experimental results, the initial vehicle location is considered to be known, *i.e.* the initial vehicle uncertainty will be zero. Figure 6.34 shows the 500 vehicle steps (65m approx.) of the experimental run with features detected. The actual and the estimated vehicle trajectories are plotted as continuous and dotted lines respectively. Figure 6.35 shows how the EKF-filter tries to correct the vehicle localisation every time it receives a radar scan. The correction in vehicle locations are marked in the figure.

The vehicle error and the tenth (for demonstration purposes) detected feature's innovation sequence are shown in figure 6.36. The vehicle error plots show that the vehicle location estimate remains consistent throughout the SLAM run. The landmarks originally have an initial uncertainty which is a function of vehicle and sensor covariances. The positions are then updated and its uncertainties are reduced as shown in figure 6.36.

Figure 6.34: SLAM with the KS-CFAR based feature detection in the outdoor car park.

¹Reliable features are the ones which can be successfully observed from all vehicle locations which can be then used for obtaining a robust localisation.

6.6 SLAM Results from Outdoor Terrains with the Proposed Feature Detectors

Figure 6.35: Figure show the EKF-SLAM filter adjusting the vehicle localisation every time it receives the observations.

Figure 6.36: Vehicle error plots obtained from an EKF-SLAM run with the KS-CFAR processor.

6.7 Conclusions

6.6.3 Outdoor Vehicle Navigation using Target Presence Probability based Feature Detection

The final experimental results correspond to outdoor SLAM using target presence probability based feature detection. Features are extracted using the TPP based detector (explained in chapter 4). The RANSAC based line feature removal algorithm removes reflections occurring from walls and buildings. The resultant feature map will be then used by the navigation filter. The vehicle builds a map of the environment, maintains it and localizes itself. As mentioned in earlier sections, the accuracy of this map is determined by the initial uncertainty of the vehicle, the extracted features' quality and their frequency of observations and the accuracy of the dead reckoning sensors installed on the vehicle. In the experimental results an initial vehicle location is assumed to be known. Figure 6.37 shows the 500 vehicle steps (65m approximately) of the experimental run along with the estimated vehicle path and the estimates of landmark locations. The odometry readings are plotted along with the estimated vehicle trajectory as shown in figure 6.37. Figure 6.38 presents the absolute error¹ with the predicted standard deviation (3σ bounds, 99.7% confidence bounds). The error plots show that the estimate remains consistent through out the SLAM run. It is also to be noted here that the uncertainty in vehicle positions are almost similar to the vehicle position uncertainty obtained using KS-CFAR processor. This is due to detecting larger number of landmarks by the TPP based detector that incorporates more information to the filter. The features detected are also frequently being re-observed. The characteristic 3σ plots in figure 6.38 reflect the filter's attempt to correct/modify the vehicle and map error.

6.7 Conclusions

This work presented SLAM with natural features with the proposed KS-CFAR processor and Target Presence Probability based feature detector algorithms for radar. Meth-

¹The absolute error is the difference between the vehicle's estimated and ground truth positions, the latter being obtained using manual scan matching, by aligning laser scans.

6.7 Conclusions

Figure 6.37: SLAM with TPP based feature detection in an outdoor environment.

6.7 Conclusions

Figure 6.38: Vehicle error plots obtained from an EKF-SLAM run using the TPP-based feature detection. The reduction in vehicle error in theta at approx. 270 steps is due to the observation of few features (concrete pillars).

ods to reject line features using a well known RANSAC algorithm were also presented and the results were shown. Feature pruning is carried out to demonstrate the proposed multiple-line-of-sight feature prediction method in outdoor terrains, which considers objects as point features. A summary of the proposed prediction model is also given in this chapter. A full implementation of SLAM using natural features is then presented, first with the KS-CFAR processor and then with the TPP based detection. An outdoor SLAM implementation using the constant threshold filter was also shown as it is the simplest among all the feature detectors and requires the lowest computational requirements.

The first, and best known, difficulty associated with feature based EKF-SLAM is its scalability requirements. For a map of N landmarks, the storage is proportional to $O(N^2)$, and the computational complexity of the updates are $O(N^2)$ because each observation of a single landmark causes the entire map to update. This means that the EKF-SLAM algorithm can only be applied in real time for relatively small maps and relatively low radar sensor update rates (a few tens of updates per second). Of the two, it is arguable that the high update cost is a more significant issue than the storage cost.

6.7 Conclusions

This is because it is typically easier to increase the amount of storage available than it is to increase the processing speed. This justifies the use of a relatively slow scanning sensor such as radar for outdoor vehicle navigation¹, apart from its other merits such as longer range (compared to other proximity sensors such as laser and sonar), multiple target detection for the same azimuth and its reliability in adverse weather conditions. The addition of more information (*i.e.* returned power/TPP, apart from the feature location) allows for the representation of *high-dimensional landmarks* which makes the association process very robust [61].

It is shown here that the proposed feature detectors, along with the feature pruning technique based on RANSAC, produces feature rich maps, which will aid the robot to accurately localise. It was also demonstrated that the algorithms can successfully build and maintain a map for long runs (approximately 65 metres) with a few hundred features (figure 6.37 shows 138 feature estimates in the outdoor environment (car-park)).

¹The scanning rate for this particular radar is 2Hz where as the scanning rate for the sick laser is around 75Hz.

Chapter 7

Conclusions

This chapter summarises the research undertaken in this thesis, namely the issues pertaining to autonomous radar sensing for robot navigation in natural terrains, and draws conclusions. The work carried outlined in this thesis addresses the major theoretical and practical issues of autonomous navigation with MMW radar. From this study, the primary difficulties concerning radar based autonomous navigation in complex large-scale outdoor environments were,

- Accurate noise characteristic derivation and sensor modeling of the power-range spectra of MMW radar.
- Automatic feature detection method(s) to correctly identify landmarks in large and *feature-rich* environments.
- Efficient map management and clutter (detections from bushes and foliages) rejection.
- Sensor modeling to effectively exploit the information content of the MMW Radar spectra characteristics
- Reliable data association given large uncertainties in the vehicle position.
- Successful implementation of autonomous navigation with MMW radar in large, unstructured environments.

This thesis presents solutions to these problems and verifies their practical advantages through experimentation in outdoor environments. This chapter summaries the contributions of this thesis and proposes directions of future research.

7.1 Summary of Main Contributions

7.1 Summary of Main Contributions

There are five major theoretical contributions in this thesis: A complete analysis of the MMW radar, the KS-CFAR processor, the TPP based feature detector, the multiple-line-of-sight spectra prediction, 3 state data association and an effective SLAM implementation in an outdoor terrain with the MMW radar. These contributions, and their practical implications, are summarized in the following sections.

7.1.1 FMCW Radar Operation and Spectra Analysis

For consistent localisation and map building, accurate prediction of target locations is fundamental, given a prediction of the vehicle/radar location. This thesis emphasises the merits of generating a predicted measurement of equal dimensionality to that of the raw sensor measurement. This is in contrast to general approaches, which typically predict only a reduced dimensionality representation (perhaps through feature representations) of the sensor data. This thesis therefore develops a method for predicting this high dimensionality observation for an FMCW MMW radar, based on estimates of the range and power measurements from multiple line-of-sight targets. This is achieved through exploitation of the prior robot pose and high dimensional feature estimates. The 360° raw measurement prediction of power-range spectra is incorporated into later chapters, in which the newly proposed feature detection methods are rigorously analyzed on simulated (as well as real) data. The use of such simulated data allows for performance quantification of the newly proposed feature detectors, as the random noise and clutter interferences can be independently controlled. This requires the classical radar range equation, modeling of the internal range compensation filter and an experimental analysis of the power and range noise statistics. Furthermore, the high dimensional measurement space allows for increased robustness in the data association stage, specifically under high feature density scenarios.

7.1 Summary of Main Contributions

Contributions

- FMCW radar operation is explained in detail, which provides insights into spectra generation and the noise signals embedded within. This is essential for the development of automatic feature detection methods, as well as for consistent measurement prediction.
- A detailed description of the radar spectra and signal characteristics, such as beam width and penetration capabilities, are discussed and verified through experimentation.
- A radar spectra prediction method, under single and multiple feature situations, is presented, exploiting knowledge of the noise signal statistics during feature presence and absence.

7.1.2 The CFAR Processor based on Statistical Similarity (KS-CFAR Processor)

The second major contribution of this thesis is the design and implementation of an adaptive landmark detection system for radar based robot navigation. The clutter rejection performance of the standard CFAR processors at ground level is shown to be unsatisfactory, and hence a new CFAR processor is proposed. The proposed processor is termed as a Kolmogorov-Smirnov¹ (KS) CFAR processor which is tested on raw power-range spectra obtained from the 77 GHz FMCW scanning radar.

While performing radar based robot navigation in outdoor environments, multiple and potentially closely lying features as well as clutter (unwanted reflections, primarily from land terrain) are often encountered. Techniques are required to extract the true target distances automatically from the observed radar spectra, which is corrupted by noise. Estimating the true relative distances of features, while rejecting land clutter reflections, for the successful localisation of mobile platforms, is the main motivation for the proposed detector.

¹The proposed CFAR processor utilises two-sample KS tests.

7.1 Summary of Main Contributions

In the KS-CFAR processor, the typical sliding window approach for CFAR processors has been modified. A more robust detector performance of the KS-CFAR detector is shown when compared to other standard CFAR detectors, in both cluttered environments as well as in multiple target situations.

Contributions

The main highlights of the proposed KS-CFAR processor are the following:

- The KS-CFAR processor has a modified sliding window design which verifies the statistical similarity of measurement sample bins with the bins from a spectrum containing no target (data obtained by scanning an open space).
- The new window design incorporates actual noisy data collected from the radar for similarity tests. The Kolmogorov-Smirnov¹ (KS), along with the Mean Ratio² (MR) tests³ are used to obtain an adaptive threshold to reliably detect features, while simultaneously maintaining low false alarm rates.

7.1.3 Automatic Landmark Detection based on Probability

The thesis also presents an algorithm which makes optimal estimates of the range to multiple targets down range based on the received signal-to-noise power. This is referred to, in this thesis, as feature detection based on target presence probability (TPP). In TPP based feature detection, the feature presence/absence hypothesis is chosen after considering a selected number of spectra, instead of a single spectrum (as in standard CFAR processors, including the proposed KS-CFAR processor). This batch signal processing method improves feature detection in cluttered environments. The noise estimate in the radar spectra is obtained by averaging the batch power spectral values at the same azimuth, using a smoothing parameter that is adjusted by the target

¹The Kolmogorov-Smirnov test (often referred to as the KS test) is a goodness of fit test used to determine whether two underlying probability distributions differ based on finite samples. The two-sample KS test is one of the most useful and general nonparametric methods for comparing two samples, as it is sensitive to differences in both location and shape of the empirical cumulative distribution functions of the two samples.

²The mean ratio (MR) can be stated as the ratio between the sum of power values in the upper window and the sum of noise power bins in the lower window.

³Based on the outcome of the KS and MR tests, different CFAR processors are selected as explained.

7.1 Summary of Main Contributions

presence probability. The target presence probability is determined by the ratio between the local power of the noise and its local minimum within a specified number of spectra. Results are shown which compare probability based feature detection with other feature extraction techniques such as constant threshold on raw data and Constant False Alarm Rate (CFAR) techniques. The results show the merits of the proposed algorithm, which can reliably detect features in cluttered outdoor environments with a higher success rate when compared to previous feature detection techniques.

Contributions

The following are the main highlights of the proposed algorithm:

- The TPP based feature detector is introduced, which estimates the local noise signal through recursive batch signal averaging, and using a smoothing parameter that is adjusted by the locally estimated target presence probability.
- TPP based feature detection can quickly adapt to the changes in the measurement signal statics (due to clutter interference).
- Evaluation of the TPP based feature detector is performed in various outdoor scenarios. It is shown that the TPP detector is superior when compared to previous adaptive feature detectors, specifically in its ability to detect features of low RCS.

7.1.4 Multiple Line-of-Sight Radar Spectra Prediction

A method for accurately predicting the power received from multiple down range targets is another contribution of this thesis, which is an extension of the high-dimensional measurement prediction outlined previously. A new SLAM formulation is proposed, which estimates an augmented state comprising the robot pose, 2D target positions, normalised target radar cross sections (RCSs), normalised absorption cross sections and the radar power loss. The term “normalised” is used here to indicate that the effect of the actual RCS is incorporated into a reflectivity parameter. Normalisation is introduced as it is assumed that the sum of this reflectivity parameter and the absorption and transmittance parameters is unity. Predicted observations are then formed using the predicted state, the radar equation, and noise analysis.

7.1 Summary of Main Contributions

Contributions

- A SLAM formulation using an augmented state vector which includes the normalised radar cross sections and absorption cross sections of features, as well as the usual feature Cartesian coordinates, is introduced. The increased state dimension is exploited to improve the data association process in feature rich environments.
- The second contribution is a predictive model for the magnitudes of the power-range spectra for multiple line-of-sight targets. This forms a predicted power-range observation, based on estimates from the augmented SLAM state.
- The formulation of power returns from multiple objects down-range is explained and predicted radar range spectra are compared with real spectra, recorded outdoors. This high dimensional prediction of power-range spectra further increases the robustness and performance of radar based SLAM frameworks.

7.1.5 SLAM with MMW Radar in Outdoor Terrains

Finally, this research presents SLAM with natural features using the proposed KS-CFAR processor and Target Presence Probability based feature detector algorithms for radar. A method to detect line features using a well known RANSAC algorithm was also presented and the results were shown. The multi-model feature detector can be incorporated in the multiple-line-of-sight feature prediction algorithm. A full implementation of SLAM using natural features is then presented, with both the proposed KS-CFAR and TPP based feature detection algorithms. An outdoor SLAM implementation using a constant threshold filter was also shown for comparison as it is the most simplistic and computationally attractive detector.

In order to obtain an accurate map for radar based navigation, the two proposed feature detectors, the KS-CFAR processor and TPP based feature detector, were used to extract features from outdoor terrains. The performance of the constant threshold, the KS-CFAR processor and the TPP based feature detector were analysed from a SLAM perspective. The performance of the proposed feature detection algorithms

7.2 Future Research

were verified and contrasted with the radar data collected from outdoor environments containing features of various RCSs.

Contributions

- Data association using the standard Nearest Neighbour filter has been improved in *feature-rich* environments with the utilisation of additional information (such as the reflected power/probability of target presence) of features, and exploiting high dimensional measurements.
- A multi-model point and line feature extraction strategy using the RANSAC algorithm and point clustering, to identify both line and point features (the reflections originating from walls, buildings, fences etc.) was presented.
- Automatic feature detectors, such as the TPP based feature detector and the KS-CFAR processor, were implemented to generate measurement inputs to the vehicle localisation and mapping filter in outdoor terrains. The performance of the proposed feature detectors, and subsequent effects on the SLAM filter were analysed.

7.2 Future Research

This section proposes areas of interesting and compelling further research. The contributions in this thesis result in certain research directions, which are discussed here.

7.2.1 Geometric Feature Model Extensions

The use of a feature based maps remains an open research topic. The environments under consideration for autonomous deployment were typical outdoor scenarios such as car parks, urban roads and off-roads. These environments are dominated by a small number of possible feature types for which mathematical models can be built a priori. In such a representation, the map of the robot vehicle's world is assumed to consist of geometric structures such as lines, cylinders, edges and corner features. Mathematical models for these features could be developed. These features are considered as targets

7.2 Future Research

by the map building filter. The map building process will comprise three main stages. First, information is extracted from the raw radar scans. Secondly, tentative feature hypotheses and map associations are generated for each processed observation. Finally, this information is then tracked through time to resolve identity ambiguities and to refine feature location estimates. The problem of landmark modeling is one of the most difficult challenges for the implementation of SLAM with any commonly used sensors (such as laser, radar, vision etc.).

7.2.2 Multi-line-of-sight Sensor Model Improvements

The multiple-line-of-sight model proposed in this thesis can be extended to fully utilise the correlation between the power measurement and the vehicle pose. This added correlation can be used to improve the vehicle location estimate accuracy, as increased spatial information is available to the filter through the power measurements. For the implementation presented in this work, the adopted radar sensor possesses a range compensation filter, which removes the range dependency of the power measurement. By removing the effects of the range compensation filter, the range dependency of reflected power readings can be retained thereby preserving the correlation between vehicle pose and the reflected power. Exploiting this added correlation, coupled with an increased dimensionality data association filter, presents compelling potential for the improvement of the performance of the proposed SLAM filter.

Appendix A

The Radar Range Compensation

Filter Effects

In the radar range equation 2.7, the transmitted power, P_t , antenna gain, G , wavelength, λ and the radar system losses, L are constants for a particular radar. Thus equation 2.7 can be written as

$$P_r = \frac{K\sigma}{R^4} \quad (\text{A.1})$$

where, K encompasses all radar constants. Equation A.1 expressed in decibels becomes

$$P_{r_{dB}} = 10 \log K + 10 \log \sigma - 40 \log R \quad (\text{A.2})$$

A perfectly tuned high-pass filter compensates the returned power loss due to range attenuation, reducing the above equation to

$$P_{r_{dB}} = 10 \log K + 10 \log \sigma \quad (\text{A.3})$$

This results in same power bias for all range values and will appear as a straight line along the range axis. However, in this particular radar, there is an imbalance in the range compensation filter. This results in an increasing bias along the range axis.

A power spectrum, with range compensation removed, obtained during target absence is shown in figure A.1 along with the bias. The decrease in power between 150 to 200 metres is due to the low pass filter (figure A.1).

Figure A.1: A radar range spectrum with range compensation filter effect removed. The spectrum is obtained by pointing towards open sky and it does not contain features. A bias due to the constants, K in the radar range equation A.3 is also shown.

In this work, the radar power spectra are linearised and the high pass filter effect is removed before carrying out noise analysis and automatic feature detection.

Appendix B

Summary of the Standard CFAR Processors

B.1 Introduction

Using an adaptive threshold with the constant false alarm rate (CFAR) property is essential for automatic feature detection in radar. This appendix summarises the standard CFAR processors reported in the radar literature. The constant false alarm rate (CFAR) problem involves performing target detection by comparing radar returns to an adaptive threshold such that a CFAR is maintained. For achieving this task, a CFAR processor, dynamically determines a detection threshold by estimating the local background noise power and multiplying this estimate by a scaling constant based in the desired probability of false alarm, P_{fa} . Hence CFAR performance is maintained in an unknown environment where the noise power in the spectra are varying. The function of CFAR processors is to maintain a constant and low rate of false alarms in detecting true range values [31]. Here, four important CFAR processors that varies the threshold adaptively for a predefined false alarm probability are presented.

In outdoor radar detection, many different noise and background clutter situations can occur, resulting in missed detections and/or false alarms. This necessitates a method for feature detection, operating with an *adaptive* threshold, instead of a con-

B.1 Introduction

stant one. In CFAR detectors, the unknown noise or clutter power is estimated by analysing the received power within a fixed window. All of the data inside this window is used to estimate the unknown statistical parameters of the background noise as well as clutter, and to calculate an adaptive threshold for target detection. The noise-plus-clutter power, thus estimated, is multiplied with a scale factor, which is used to achieve a desired constant false alarm probability for the given window size. The detection method has to distinguish between returns from objects and clutter/noise.

The thresholds of various CFAR methods can be generalized as shown in equation B.1.

$$\eta = T f(x_1, x_2, \dots, x_N) \quad (\text{B.1})$$

where T is the threshold and $f(\cdot)$ is a function that defines the CFAR properties. η will then be compared with the cell-under-test (CUT) to decide whether it is a possible target or not. The function, $f(\cdot)$ and the threshold, T calculations will be explained in the later sections.

B.1.1 Optimal Detector

The radar detection problem is a binary detection hypothesis-testing problem in which H_0 denotes the hypothesis that no target is present and H_1 is the hypothesis that the target is present. From a radar detection perspective, the most appropriate criteria to solve this problem is the Neyman-Pearson. This criterion maximizes the probability of detection, P_d for a given probability of false alarm, P_{fa} by comparing the likelihood ratio, Λ given in equation C.28 to an appropriate threshold, T which determines the P_{fa} .

$$\Lambda(x_1, x_2, \dots, x_n) = \frac{p(x_1, x_2, \dots, x_n | H_1)}{p(x_1, x_2, \dots, x_n | H_0)} \geq T \quad (\text{B.2})$$

A target is declared present if the likelihood ratio, $\Lambda(x_1, x_2, \dots, x_n) \geq T$.

There are four fundamental CFAR techniques - Cell Averaging (CA) [32], Greatest Of (GO) [88], Smallest Of (SO) [85] and Order Statistics (OS) [68], and much of the

B.2 The Concept of Cell-Averaging CFAR Processor

Figure B.1: Generic CFAR block diagram.

literature presents combinations or modifications of these methods. Each method has its own advantages and disadvantages.

B.2 The Concept of Cell-Averaging CFAR Processor

Figure B.1 shows a generic radar detection processor. The detection shows a one dimensional vector of range cells in making a decision of target presence or absence. The detector will test each power value in the range bin for the presence or absence of a target. The cell under test, denoted by x_i in figure B.1 is compared against a threshold determined by the interference power. If the value of the data in the test cell exceeds the threshold, the processor declares a target present at the particular range. The next cell is then tested and so forth until a target present/absent decision has been made for all cells of interest.

B.2 The Concept of Cell-Averaging CFAR Processor

To set the threshold for testing cell, x_i , the interference power in the same cell must be known. Since it may be variable, it must be estimated from the data. The approach used in CFAR processing is based on two major assumptions.

- The neighboring cells contain interference with the same statistics as the cell under test (known as homogeneous interference), so that they are representative of the interference that is competing with the potential target.
- The neighboring cells do not contain any targets, they are interference only.

Under these conditions, the interference statistics in the cell under test can be estimated from the measured samples in the adjoining cells.

The statistics that must be estimated are determined by the statistics needed to implement the threshold test. The interference during target absence is exponential noise as shown in chapter 1. The interference probability density function (pdf) has only one free parameter, the mean interference power. Thus, the CFAR processor must estimate the mean interference power in the cell under test by using the measured data in the adjoining cells.

The pdf of a cell, x_i , assuming the interference is *independent and identically distributed (i.i.d)* in the power spectra with noise power, P_n , is then

$$p(x_i) = \frac{1}{P_0} \exp(-x_i/P_0) \quad (\text{B.3})$$

The exact knowledge of the noise power, P_0 is required to set the threshold and since the exact knowledge of P_0 is unknown, it must be estimated.

Assume that N cells in the vicinity of the cell under test are used to estimate P_0 , and the interference in each is independent and identically distributed. The joint pdf of a vector $\mathbf{x}$ of N such power bins is

$$p(\mathbf{x}) = \frac{1}{P_0^N} \prod_{i=1}^N \exp(-x_i/P_0) = \frac{1}{P_0^N} \exp - \left(\left(\sum_{i=1}^N x_i \right) / P_0 \right) \quad (\text{B.4})$$

Equation B.4 is the likelihood function, Λ for the observed data vector, $\mathbf{x}$. the maximum likelihood estimate of P_0 is obtained by maximizing equation B.4 with $\sum x_i$

B.2 The Concept of Cell-Averaging CFAR Processor

held constant. It is convenient to maximize the log-likelihood function as shown in equation B.5.

$$\ln \Lambda = -N \ln(P_0) - \frac{1}{P_0} \left(\sum_{i=1}^N x_i \right) \quad (\text{B.5})$$

Obtaining the derivative of equation B.5 with respect to P_0 and equate to zero gives

$$\frac{d(\ln \Lambda)}{d(P_0)} = 0 = -N(1/P_0) - (-1/P_0^2) \left(\sum_{i=1}^N x_i \right) \quad (\text{B.6})$$

Solving equation B.6 for P_0 gives

$$\hat{P}_0 = \frac{1}{N} \sum_{i=1}^N x_i \quad (\text{B.7})$$

The required threshold is then estimated as a multiple of the estimated interference power

$$\hat{T} = \alpha \hat{P}_0 \quad (\text{B.8})$$

Combining equations B.7 and B.8 gives the expression for the estimated threshold

$$\hat{T} = \frac{\alpha}{N} \sum_{i=1}^N x_i \quad (\text{B.9})$$

Let $z_i = (\alpha/N)x_i$. Then $\hat{T} = \sum_{i=1}^N z_i$. From equation B.3, the pdf of z_i can be written as

$$p(z_i) = \frac{N}{\alpha P_0} \exp \frac{-N z_i}{\alpha P_0} \quad (\text{B.10})$$

The pdf of $\hat{T}$ is the Erlang density, which is a special case of gamma density and is given by

$$p(\hat{T}) = \left(\frac{N}{\alpha P_0} \right)^N \frac{\hat{T}^{N-1}}{(N-1)!} \exp \frac{-N \hat{T}}{\alpha P_0} \quad (\text{B.11})$$

B.2 The Concept of Cell-Averaging CFAR Processor

The P_{fa} observed with the estimated threshold will be $\exp(-\hat{T}/\alpha P_0)$. This is now also a random variable and its expected value can be written as

$$\begin{aligned}\bar{P}_{FA} &= \int_{-\infty}^{+\infty} \exp\left(-\frac{\hat{T}}{\alpha P_0}\right) p(\hat{T}) d\hat{T} \\ &= \left(\frac{N}{\alpha P_0}\right)^N \frac{1}{(N-1)!} \int_{-\infty}^{+\infty} \hat{T}^{N-1} \exp\left[-\frac{(N/\alpha) + 1}{P_0} \hat{T}\right] d\hat{T}\end{aligned}\quad (\text{B.12})$$

Completing the standard integral and performing some algebraic manipulations gives the final result

$$\bar{P}_{FA} = \left(1 + \frac{\alpha}{N}\right)^{-N} \quad (\text{B.13})$$

For a given $\bar{P}_{FA}$, the required threshold multiplier is obtained by solving B.13 to obtain

$$\alpha = N \left(\bar{P}_{FA}^{-1/N} - 1 \right) \quad (\text{B.14})$$

$\bar{P}_{FA}$ does not depend on the interference power, P_0 , but only on the number, N of neighboring cells averaged and the multiplier. Thus, the cell-averaging technique exhibits CFAR behavior.

The scale factor, α gives exactly the designed false alarm probability, P_{fa} . When the window width, N increases, the scale factor, α decreases. In other words, when the number of reference samples is large, less scaling is required. This is shown in figure B.2.

B.2.1 Relation Between Reference Cell Number and Noise Estimation

As the number of reference cells, N becomes large, the estimate $\hat{P}_0$ should converge to the true value, P_0 . To verify this, first, obtain the natural logarithm of equation B.13

B.2 The Concept of Cell-Averaging CFAR Processor

Figure B.2: Scale factor, α as a function of reference cells, N . $P_{fa} = 1 \times 10^{-6}$.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

on both sides which gives

$$\begin{aligned}
 \ln \bar{P}_{fa} &= \ln \left\{ \left(1 + \frac{\alpha}{N} \right)^{-N} \right\} \\
 &= -N \ln \left\{ 1 + \frac{\alpha}{N} \right\} \\
 &= -N \left\{ \frac{\alpha}{N} - \frac{1}{2} \left\{ \frac{\alpha}{N} \right\}^2 + \dots \right\}
 \end{aligned} \tag{B.15}$$

Let $N \rightarrow \infty$, then

$$\lim_{N \rightarrow \infty} \{\ln \bar{P}_{fa}\} = -N \left\{ \frac{\alpha}{N} \right\} \implies \lim_{N \rightarrow \infty} \bar{P}_{fa} = \exp -\alpha \tag{B.16}$$

Similarly

$$\lim_{N \rightarrow \infty} \{\bar{P}_d\} = \exp \frac{-\alpha}{(1 + SNR)} \tag{B.17}$$

Combining equations B.16 and B.17 gives the relation

$$\bar{P}_d = (\bar{P}_{fa})^{\frac{1}{(1+SNR)}} \tag{B.18}$$

This shows the relation between the detection probability, P_d as a function of the probability of false alarm, P_{fa} and the SNR .

Next, the four important CFAR processors will be explained and their results will be analysed.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

A cell averaging (CA) detector is useful for maintaining a constant false alarm rate (CFAR) where the noise power-plus-clutter observations $x = x_1, \dots, x_i, \dots, x_n$ follow an exponential random distribution. For comparison purposes, a CA-CFAR processor is tested in this work, and its structure is as shown in figure B.3.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

Figure B.3: The structure of the Cell Averaging-CFAR detector.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

This figure shows $N/2$ reference cells (where $N = 44$) on each side of the cell, Y , under investigation. Guard cells are present to account for the broadened target reflection [20]. A moving window, of width $N = 44$ range points is then used to sum the local noisy power values in the range spectrum as shown in equation B.19. [32].

$$Z = \sum_{i=1}^N x_i \quad (\text{B.19})$$

This sum is multiplied by a threshold, τ , for later comparison with a test sample power value. The value for τ is chosen for achieving the desired value of P_{fa} , the design false alarm probability, in the absence of targets [87]. The scalar τ is a function of the number of reference cells N (here $N = 44$) and P_{fa} is set to allow a value of 1×10^{-6} for the radar used here [14] and is given by equation B.20.

$$\tau = P_{fa}^{-1/N} - 1 \quad (\text{B.20})$$

The test sample Y is either a noise-plus-clutter observation or a target return. The variable threshold τZ is compared with Y . A target is declared to be present if

$$Y > \tau Z \quad (\text{B.21})$$

The CA-CFAR method has been applied to the range spectrum obtained from an outdoor environment. A full 200 metre spectrum with CA-CFAR feature detection is shown in figure B.4 and also another spectrum with CA-CFAR feature detection is shown in figure B.5.

B.3.1 CA-CFAR Processor Drawbacks

- The cell averaging CFAR detectors can successfully detect two closely lying targets, that is separated by the window width.
- The bins of the power spectra are assumed to be independent and identically distributed (*i.i.d*), or, in other words, the noise in the power spectra is assumed to be homogeneous and the distribution is assumed to be the same in the range bin containing the target.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

Figure B.4: The plot shows target detection using a cell averaging CFAR detector. Detected false alarms are marked on the plot.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

Figure B.5: The plot shows target detection using a cell averaging CFAR detector. Missed detection of features are shown the figure.

In realistic situations, both the assumptions are violated due to the fact that noise is often nonhomogeneous. The radar often encounter clutter originate from land, that makes clutter power appear inside the reference window. These results in missed detection due to target masking and false alarms due to clutter edges.

Missed Detections Due to Target Masking

The feature can be missed by the CA-CFAR processor when one or more feature appear in the reference cells, with another feature in the test cell. Assuming that the power of the target in the reference cell exceeds that of the surrounding interference. This will raise the noise power estimate thereby increasing the threshold. The feature(s) in the reference window will cause a reduction in the probability of detection, P_D due to increase in threshold. Figure B.6 shows two closely lying features (first feature is approximately at 25 metres and the second one is at 32 metres) separated by less than the window width, resulting in masking of one target¹ by the CA-CFAR detector.

¹ usually the feature with smaller RCS will be masked.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

Figure B.6: A single power spectrum obtained from the radar with two power returns which lie within the window width results in target masking with the CA-CFAR processor.

B.3 Analysis of Cell Averaging Constant False Alarm Rate (CA-CFAR) Processor

Figure B.7: A power spectrum obtained from the radar. A false alarm due to clutter edge at 20 metres along with other features are detected by the CA-CFAR processor. False alarms due to noise power variation is also shown.

B.4 Analysis of the Greatest-Of and the Smallest-Of CFAR Processors

False Alarms due to Clutter Edges

Figure B.7 shows the result from an environment, containing clutter from the terrain at approximately 20 metres and a lamp post at approximately 105 metres. CA-CFAR processor misclassifies a clutter edge to be a target, which is a false alarm. A lamp-post at approximately 105 metres is detected along with other false alarms, that arise due to interference variations.

The CA-CFAR detection technique finds features regardless of range as shown in this section. Problems still arise however, as CFAR processor may misclassify targets as noise (missed-detections) and noise as targets (false-alarms). Both of these are evident and labeled in the CFAR processor results of figures B.6 and B.7 respectively.

B.4 Analysis of the Greatest-Of and the Smallest-Of CFAR Processors

Excessive number for false alarms and missed detections in multiple target environments in the CA-CFAR detector results in new CFAR detectors. Modifications of CA-CFAR processor results in Greatest-Of (GO) CFAR and Smallest-Of (SO) CFAR detectors. However, each of these detectors can overcome only one of the problems of CA-CFAR processor.

B.4.1 The Greatest-Of CFAR Processor

The noise and clutter power are varying in time and it can fluctuate in different range cells. Consider a scenario wherein a clutter power transition (a transition between noise and clutter power) occurs in the sliding window of the CA-CFAR processor. This violates the *i.i.d* assumptions of the CA-CFAR processor causing an increase in false alarm rate. In order to overcome the false alarms due to the clutter situations, Hansen [37] proposed the Greatest-Of (GO) CFAR processor.

The GO-CFAR processor is capable of reducing the number of excessive false alarms at the clutter edges. In GO-CFAR detector, the leading and lagging windows

B.4 Analysis of the Greatest-Of and the Smallest-Of CFAR Processors

are averaged separately to obtain two independent noise estimates, P_{0_1} and P_{0_2} . The threshold is then computed from the largest of the two noise estimates as shown in equation B.22

$$P_0 = \alpha_{GO} \max(P_{0_1}, P_{0_2}) \quad (\text{B.22})$$

where, $P_{0_1} = \sum_{i=1}^{N/2} x_i$ and $P_{0_2} = \sum_{i=N/2+1}^N x_i$. The false alarm probability of the GO-CFAR processor is found by computing the moment generating function (mgf) of P_0 [37] and is given in equation B.23.

$$P_{fa} = 2(1+T)^{-\frac{N}{2}} - 2 \sum_{i=0}^{\frac{N}{2}-1} \binom{\frac{N}{2}+i-1}{i} (2+T)^{-\left(\frac{N}{2}+i\right)} \quad (\text{B.23})$$

The performance analysis of GO-CFAR processor is carried out against the CA-CFAR detector and is shown in figure B.8. The spectrum shown is the same as shown in figure B.7. The GO-CFAR processor successfully rejects the clutter edge at approximately 20 metres. However, the false alarm occurrences deteriorates the GO-CFAR processor performance. This is shown in figure B.8. Furthermore, target masking will occur due to the elevated adaptive threshold near two closely lying features. Hence GO-CFAR processor, although an extension of the CA-CFAR detector, however have the same performance characteristics in multiple target situations.

B.4.2 The Smallest-Of CFAR Processor

In [88] an extension of Greatest Of (GO) CFAR processor has been proposed for obtaining a superior performance in resolving multiple target situations. In [88], the cell averaging CFAR processor is designed with smallest of the arithmetic mean of the two reference windows and showed the performance analysis in multiple target situations. These CFAR processors are known as Smallest Of (SO) CFAR processors.

The Smallest-Of CFAR processor can resolve closely lying features leading to two or more targets appear inside the reference window. In the SO-CFAR detector, the

B.4 Analysis of the Greatest-Of and the Smallest-Of CFAR Processors

Figure B.8: *Greatest Of (GO) CFAR processor performance with the adaptive threshold.*

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

smallest of the noise power estimate among the leading and lagging windows are chosen. That is

$$P_0 = \alpha_{SO} \min(P_{01}, P_{02}) \quad (\text{B.24})$$

The false alarm probability of the SO-CFAR processor is given in [31] and is shown in equation B.25.

$$P_{fa} = 2 \sum_{i=0}^{\frac{N}{2}-1} \binom{\frac{N}{2} + i - 1}{i} (2 + T)^{-(\frac{N}{2} + i)} \quad (\text{B.25})$$

SO-CFAR detector performance deteriorates when the *i.i.d* assumption is violated in the presence of clutter transition in the reference window. Figure B.9 shows the performance comparison of the SO-CFAR against the CA-CFAR processor. The SO-CFAR processor can resolve close features successfully. However, the higher false alarm rate makes this processor less attractive.

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

It has been reported in the literature that OS-CFAR processor [68] can handle multiple features much better than CA-CFAR processor. In OS-CFAR processor, N power samples are taken and ranked in an ascending order as shown in equation B.26.

$$x_1 \leq x_2 \leq \dots \leq x_i \leq \dots \leq x_j \leq \dots \leq x_N \quad (\text{B.26})$$

It is assumed that the power values are truly random in nature and are statistically independent and identically distributed (*i.i.d*). Therefore, for any given false alarm probability, P_{fa} , an adaptive threshold, T can be derived.

The threshold T , which is a function of the k^{th} ranked power bin sample, k , the

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

Figure B.9: The result of an SO-CFAR processor compared with the CA-CFAR processor result.

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

Figure B.10: The false alarm probability versus scale factor graph of an OS-CFAR processor for two different window widths (for $N = 22$ and 44).

window width, N and the false alarm probability, P_{fa} is given by equation B.27 [69].

$$P_{fa} = k \binom{N}{k} \frac{(k-1)!(T+N-k)!}{(T+N)!} \quad (\text{B.27})$$

Figure B.10 shows the false alarm probability, P_{fa} as a function of the scale factor, α_{OS} for two different window width, one with $N = 22$ and other with $N = 44$. The values of k are chosen to be 16 and 33. From this plot, the scale factor, α_{OS} can be chosen for a desired value of P_{fa} and window width, N .

Figures B.11 and B.12 shows the typical performance of an OS-CFAR processor in multiple target situations. The closely lying features are detected by OS-CFAR processor as shown in figure. This was missed by the CA-CFAR detector as shown in figure B.5. The threshold of OS-CFAR processor follows the noise floor with a certain safety distance, however, false alarms also occur in this detection process as shown the figures B.11 and B.12.

The window width, N is again set to 44 and the value of k is chosen to be 33th order statistic ($=0.75N$) in order to set the threshold. Guard cells are of less importance in OS-CFAR processor as the rank ordering process will not be affected by the beam

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

spreading outside of the cell under test. OS-CFAR processor can resolve closely lying features as long as the number of range bins containing interfering targets does not exceed $N - k$.

In [68], the effect of window width and the choice of the value of k are explained. It is shown in [68] that if $k \leq N/2$, then occurrence of false alarms will increase at clutter transitions. In [69], it is suggested that k be set to $(3/4)N$ in order to provide acceptable performance in cluttered environments.

Figure B.11: The figure shows the OS-CFAR processor performance on a range spectrum. False alarms occur along with the correct feature detection.

A performance comparison between CA-CFAR and OS-CFAR processors has been carried out and is shown in figure B.13. The window width, N has chosen to 44 and the value of k , to be 33 for setting the threshold. OS-CFAR processor can overcome the issue of masking, thereby resolving closely lying features.

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

Figure B.12: The figure shows the OS-CFAR processor performance on a range spectrum with closely lying features. Closely lying features have been detected which was missed by the CA-CFAR processor.

B.5 Ordered Statistics - Constant False Alarm Rate (OS-CFAR) Processor

Figure B.13: The OS-CFAR processor result compared with the standard CA-CFAR processor output.

Appendix C

The Likelihood Ratio Test for Target Presence Probability

C.1 Statistical Decision Theory

Assume a source, which emits two possible outputs at various instants of time. The outputs are referred to as *hypotheses*. The *null hypothesis* H_0 represents a zero (target not present) while the *alternate hypothesis* H_1 represents a one (target present).

Each hypothesis corresponds to one or more observations that are represented by random variables. Based on the observation values of these random variables, the detector decides which hypothesis (H_0 or H_1) is true. Assume that the detector is to make a decision based on a single observation of the received signal. The range of values that the random variable, Y takes constitutes the observation space, Z . The observation space is partitioned into two regions Z_0 and Z_1 , such that if Y lies in Z_0 , the detector decides in favor of H_0 ; while if Y lies in Z_1 , the detector decides in favor of H_1 . The observation space Z is the union of Z_0 and Z_1 ; that is,

$$Z = Z_0 \cup Z_1 \quad (\text{C.1})$$

The probability density functions of Y corresponding to each hypothesis are $f_{Y|H_0}(y|H_0)$

C.1 Statistical Decision Theory

and $y_{Y|H_1}(y|H_1)$, where y is a particular value of the random variable Y .

Each time a decision is made, based on some criterion, for this binary hypothesis testing problem, four possible cases can occur:

1. Decide H_0 when H_0 is true.
2. Decide H_0 when H_1 is true.
3. Decide H_1 when H_0 is true.
4. Decide H_1 when H_1 is true.

For cases (i) and (iv), the detector makes a correct decision, while for cases (ii) and (iii), the detector makes an error. From a radar detection nomenclature, case (i) is known as no detection, case (ii) is known as miss detection, case (iii) a false alarm, and case (iv) a detection.

C.1.1 Bayes Criterion - Binary Hypothesis Testing

In using Bayes criterion, two assumptions are made. *First*, the probability of occurrence of the two source outputs is known. They are the a priori probabilities $P(H_0)$ and $P(H_1)$. $P(H_0)$ is the probability of occurrence of hypothesis H_0 , while $P(H_1)$ is the probability of occurrence of hypothesis H_1 . Denoting the a priori probabilities $P(H_0)$ and $P(H_1)$ by P_0 and P_1 respectively, and since either hypothesis H_0 or H_1 will always occur, we have

$$P_0 + P_1 = 1 \quad (\text{C.2})$$

The *second* assumption is that a cost is assigned to each possible decision. The cost is due to the fact that some action will be taken based on a decision made. The consequences of one decision are different from the consequences of another. In a radar detection problem, the consequences of miss are not the same as the consequences of false alarm. If we let D_i , $i = 0, 1$, where D_0 denotes “decide H_0 ” and D_1 denotes

C.1 Statistical Decision Theory

“decide H_1 ”, we can define C_{ij} , $i, j = 0, 1$, as the cost associated with the decision D_i , given that the true hypothesis is H_j . That is,

$$P(\text{incurring cost } C_{ij}) = P(\text{decide } D_i, H_j \text{ true}), \quad i, j = 0, 1 \quad (\text{C.3})$$

In particular, the costs for this binary hypothesis testing problem are C_{00} for case (i), C_{01} for case (ii), C_{10} for case (iii), and C_{11} for case (iv). The goal in Bayes criterion is to determine the decision rule so that the average cost $E[\mathcal{C}]$, also known as risk $\mathfrak{R}$, is minimized. The operation $E[\mathcal{C}]$ denotes expected value. It is also assumed that the cost of making a wrong decision is greater than the cost of making a correct decision. That is,

$$C_{01} > C_{11} \quad (\text{C.4})$$

and

$$C_{10} > C_{00} \quad (\text{C.5})$$

Given $P(D_i, H_j)$, the joint probability that we decide D_i , and that the hypothesis H_j is true, the average cost is

$$\mathfrak{R} = E[\mathcal{C}] = C_{00}P(D_0, H_0) + C_{01}P(D_0, H_1) + C_{10}P(D_1, H_0) + C_{11}P(D_1, H_1) \quad (\text{C.6})$$

From Bayes rule, we have

$$P(D_i, H_j) = P(D_i|H_j)P(H_j) \quad (\text{C.7})$$

The conditional density functions $P(D_i|H_j)$, $i, j = 0, 1$, can be written as

$$P(D_0|H_0) \equiv P(\text{decide } H_0|H_0 \text{ true}) = \int_{Z_0} f_{Y|H_0}(y|H_0)dy \quad (\text{C.8})$$

$$P(D_0|H_1) \equiv P(\text{decide } H_0|H_1 \text{ true}) = \int_{Z_0} f_{Y|H_1}(y|H_1)dy \quad (\text{C.9})$$

C.1 Statistical Decision Theory

$$P(D_1|H_0) \equiv P(\text{decide } H_1|H_0 \text{ true}) = \int_{Z_1} f_{Y|H_0}(y|H_0)dy \quad (\text{C.10})$$

and

$$P(D_1|H_1) \equiv P(\text{decide } H_1|H_1 \text{ true}) = \int_{Z_1} f_{Y|H_1}(y|H_1)dy \quad (\text{C.11})$$

The probabilities $P(D_0|H_1)$, $P(D_1|H_0)$, and $P(D_1|H_1)$ represent the probability of miss, P_m , the probability of false alarm, P_{fa} , and the probability of detection, P_D , respectively. We also observe that

$$P_m = 1 - P_D \quad (\text{C.12})$$

and

$$P(D_0|H_0) = 1 - P_{fa} \quad (\text{C.13})$$

Consequently, the probability of a correct decision is given by

$$\begin{aligned} P(\text{correct decision}) &= P(c) = P(D_0, H_0) + P(D_1, H_1) \\ &= P(D_0|H_0)P(H_0) + P(D_1|H_1)P(H_1) \\ &= (1 - P_{fa})P_0 + P_D P_1 \end{aligned} \quad (\text{C.14})$$

and the probability of error is given by

$$\begin{aligned} P(\text{error}) &= P(\varepsilon) = P(D_0, H_1) + P(D_1, H_0) \\ &= P(D_0|H_1)P(H_1) + P(D_1|H_0)P(H_0) \\ &= P_m P_1 + P_{fa} P_0 \end{aligned} \quad (\text{C.15})$$

C.1 Statistical Decision Theory

The average cost now becomes

$$\mathfrak{R} = E[C] = C_{00}(1 - P_{fa})P_0 + C_{01}(1 - P_D)P_1 + C_{10}P_{fa}P_0 + C_{11}P_DP_1 \quad (C.16)$$

In terms of the decision regions defined in equation C.8 to equation C.11, the average cost is

$$\begin{aligned} \mathfrak{R} = & P_0C_{00} \int_{Z_0} f_{Y|H_0}(y|H_0)dy + P_1C_{01} \int_{Z_0} f_{Y|H_1}(y|H_1)dy \\ & + P_0C_{10} \int_{Z_1} f_{Y|H_0}(y|H_0)dy + P_1C_{11} \int_{Z_1} f_{Y|H_1}(y|H_1)dy \end{aligned} \quad (C.17)$$

Using equation C.1 and the fact that

$$\int_Z f_{Y|H_0}(y|H_0)dy = \int_Z f_{Y|H_1}(y|H_1)dy = 1 \quad (C.18)$$

the above equation can be rewritten as

$$\int_{Z_1} f_{Y|H_j}(y|H_j)dy = 1 - \int_{Z_0} f_{Y|H_j}(y|H_j)dy, \quad j = 0, 1 \quad (C.19)$$

where $f_{Y|H_j}(y|H_j)$, $j = 0, 1$ is the probability density function of Y corresponding to each hypothesis. Substituting for equation C.19 in equation C.16, obtains:

$$\begin{aligned} \mathfrak{R} = & P_0C_{10} + P_1C_{11} \\ & + \int_{Z_0} \{[P_1(C_{01} - C_{11})f_{Y|H_1}(y|H_1)] - [P_0(C_{10} - C_{00})f_{Y|H_0}(y|H_0)]\}dy \end{aligned} \quad (C.20)$$

It is observed that the quantity $P_0C_{10} + P_1C_{11}$ is constant, independent of how the points are assigned in the observation space, and that the only variable quantity is the region of integration Z_0 . From equations C.4 and C.5, the terms inside the brackets of equation C.20, *i.e.* $[P_1(C_{01} - C_{11})f_{Y|H_1}(y|H_1)]$ and $[P_0(C_{10} - C_{00})f_{Y|H_0}(y|H_0)]$, are both positive. Consequently, the risk is minimized by selecting the decision region Z_0

C.1 Statistical Decision Theory

to include only those points of Y for which the second term is larger, and hence the integrand is negative. Specifically, we assign to the region Z_0 those points for which

$$P_1(C_{01} - C_{11})f_{Y|H_1}(y|H_1) < P_0(C_{10} - C_{00})f_{Y|H_0}(y|H_0) \quad (\text{C.21})$$

All values for which the second term is greater will be excluded from Z_0 and assigned to Z_1 . The values for which the two terms are equal do not affect the risk, and can be assigned to either Z_0 or Z_1 . Consequently, we say if

$$P_1(C_{01} - C_{11})f_{Y|H_1}(y|H_1) > P_0(C_{10} - C_{00})f_{Y|H_0}(y|H_0) \quad (\text{C.22})$$

then we decide H_1 . Otherwise, we decide H_0 . Hence, the decision rule resulting from the Bayes criterion is

$$\frac{f_{Y|H_1}(y|H_1)}{f_{Y|H_0}(y|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0(C_{10} - C_{00})}{P_1(C_{01} - C_{11})} \quad (\text{C.23})$$

The ratio of $f_{Y|H_1}(y|H_1)$ over $f_{Y|H_0}(y|H_0)$ is known as the *likelihood ratio* and is denoted by $\Lambda(y)$. That is

$$\Lambda(y) = \frac{f_{Y|H_1}(y|H_1)}{f_{Y|H_0}(y|H_0)} \quad (\text{C.24})$$

It should be noted that if we have K observations, for example, K samples of a received waveform, $Y_1, Y_1, \dots, Y_k$, based on which we make the decision, the *likelihood ratio* can be expressed as

$$\Lambda(\mathbf{y}) = \frac{f_{\mathbf{Y}|H_1}(\mathbf{y}|H_1)}{f_{\mathbf{Y}|H_0}(\mathbf{y}|H_0)} \quad (\text{C.25})$$

C.1 Statistical Decision Theory

where $\mathbf{Y}$, the received vector, is

$$\mathbf{Y}^T = [Y_1, Y_2, \dots, Y_k] \quad (\text{C.26})$$

The *likelihood statistic* $\Lambda(\mathbf{Y})$ is a random variable since it is a function of the random variable $\mathbf{Y}$.

The threshold is

$$\delta = \frac{P_0(C_{10} - C_{00})}{P_1(C_{01} - C_{11})} \quad (\text{C.27})$$

Therefore, Bayes criterion, which minimizes the average cost, results in the *likelihood ratio* test

$$\Lambda(\mathbf{y}) \underset{H_0}{\overset{H_1}{\geq}} \delta \quad (\text{C.28})$$

An important observation here is that the *likelihood ratio* test is performed by simply processing the receiving vector to yield the *likelihood ratio* and comparing it with the threshold. Thus, in practical situations where the a priori probabilities and the cost may change, only the threshold changes, but the computation of likelihood ratio is not affected.

Because the natural logarithm is a monotonically increasing function, and since the likelihood ratio $\Lambda(\mathbf{y})$ and the threshold δ are nonnegative, an equivalent decision rule to equation C.28 is

$$\ln \Lambda(\mathbf{y}) \underset{H_0}{\overset{H_1}{\geq}} \ln \delta \quad (\text{C.29})$$

if we select the cost of an error to be one and the cost of a correct decision to be zero; that is,

$$C_{01} = C_{10} = 1 \quad (\text{C.30})$$

C.1 Statistical Decision Theory

and

$$C_{00} = C_{11} = 0 \quad (\text{C.31})$$

then the risk function given in equation C.16 reduces to

$$\mathfrak{R} = P_m P_1 + P_{fa} P_0 = P(\varepsilon) \quad (\text{C.32})$$

Thus, in this case, minimizing the average cost is equivalent to minimizing the probability of error, $P(\varepsilon)$. Detectors for such cost assignment are known *minimum probability of error detectors*. The threshold thus reduces to

$$\delta = \frac{P_0}{P_1} \quad (\text{C.33})$$

If the a priori probabilities are equal, (δ will equal to one, based on equation C.33), then the *likelihood ratio* test uses a threshold equal to one. That is equation C.28 becomes:

$$\Lambda(\mathbf{y}) \underset{H_0}{\overset{H_1}{\geq}} 1 \quad (\text{C.34})$$

C.1.2 Radar Likelihood Ratio Test

Consider the radar output, under hypothesis H_1 is the signal during target presence (Gaussian distribution) while H_0 is the hypothesis under target absence (exponential distribution). μ is the mean and σ^2 is the variance of the distributions.

Under hypothesis H_0 ,

$$f_{y|H_0}(y|H_0) = \frac{1}{\sqrt{2}\sigma} \exp\left(\frac{-\sqrt{2}y}{\sigma}\right) \quad (\text{C.35})$$

and under hypothesis H_1 ,

$$f_{y|H_1}(y|H_1) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-1}{2} \frac{(y - \mu)^2}{\sigma^2}\right) \quad (\text{C.36})$$

C.1 Statistical Decision Theory

The likelihood ratio test is

$$\begin{aligned}\Lambda(\mathbf{y}) &= \frac{f_{y|H_1}(y|H_1)}{f_{y|H_0}(y|H_0)} \\ &= \frac{\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2} \frac{(y-\mu)^2}{\sigma^2}\right)}{\frac{1}{\sqrt{2}\sigma} \exp\left(\frac{-\sqrt{2}y}{\sigma}\right)}\end{aligned}\quad (\text{C.37})$$

$$\Lambda(\mathbf{y}) = \frac{1}{\sqrt{\pi}} \exp\left(\frac{\sqrt{2}y}{\sigma} - \frac{1}{2} \frac{(y-\mu)^2}{\sigma^2}\right) \quad (\text{C.38})$$

From equation C.34, the decision test can be written as

$$\frac{1}{\sqrt{\pi}} \exp\left(\frac{\sqrt{2}y}{\sigma} - \frac{1}{2} \frac{(y-\mu)^2}{\sigma^2}\right) \underset{H_0}{\overset{H_1}{\gtrless}} 1 \quad (\text{C.39})$$

$$\frac{1}{\sqrt{\pi}} \exp\left(\frac{2\sqrt{2}\sigma y - (y-\mu)^2}{2\sigma^2}\right) \underset{H_0}{\overset{H_1}{\gtrless}} 1 \quad (\text{C.40})$$

$$\exp\left(\frac{2\sqrt{2}\sigma y - (y-\mu)^2}{2\sigma^2}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \sqrt{\pi} \quad (\text{C.41})$$

$$\exp\left(\frac{-y^2 + 2y(\sqrt{2}\sigma + \mu) - \mu^2}{2\sigma^2}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \sqrt{\pi} \quad (\text{C.42})$$

Rearranging after taking $\ln$ on both sides to have,

$$y^2 - 2y(\sqrt{2}\sigma + \mu) + \mu^2 \underset{H_0}{\overset{H_1}{\gtrless}} -2\sigma^2 \ln(\sqrt{\pi}) \quad (\text{C.43})$$

$$y^2 - 2y(\sqrt{2}\sigma + \mu) + \mu^2 + 2\sigma^2 \ln(\sqrt{\pi}) \underset{H_0}{\overset{H_1}{\gtrless}} 0 \quad (\text{C.44})$$

C.1 Statistical Decision Theory

Equation C.44 is a quadratic equation having solutions:

$$y \underset{H_0}{\overset{H_1}{\gtrless}} \frac{2(\sqrt{2}\sigma + \mu) \pm \sqrt{4(\sqrt{2}\sigma + \mu)^2 - 8\sigma^2 \ln(\sqrt{\pi})}}{2} \quad (\text{C.45})$$

Rearranging,

$$y \underset{H_0}{\overset{H_1}{\gtrless}} (\sqrt{2}\sigma + \mu) \pm \sqrt{2\sigma^2(1 - \ln(\sqrt{\pi})) + \mu^2 + 2\sqrt{2}\sigma\mu} \quad (\text{C.46})$$

The exponential noise distribution during target absence (shown in chapter 2), has a mean, $\mu = \sqrt{2}$ and variance $\sigma^2 = 2$. Substituting,

$$|y| \underset{H_0}{\overset{H_1}{\gtrless}} 6.68 \quad (\text{C.47})$$

$\delta = 6.68$ is the theoretical value for the threshold. In real environments, the signal and noise distributions may vary and hence the threshold found above (*i.e.* $\delta = 6.68$) may not always produce an optimal detection result. Hence, a threshold of 4 is used in this work after carefully analysing the detection results obtained from controlled environments.

Author's Publication List

Book Chapter:

- Martin D. Adams and Ebi Jose, "Autonomous Mobile Robots: Sensing, Control, Decision-Making, and Applications" Series in Control Engineering, Chapter Title: "Millimetre Wave RADAR Power $\hat{U}$ Range Spectra Interpretation for Multiple Line of Sight Feature Detection". CRC Press, 2005.

Journals:

- John Mullane, Ebi Jose, Martin D. Adams and Wijerupage Sardha Wijesoma, "Including Probabilistic Target detection Attributes into Map Representations", Journal of Robotics and Autonomous Systems. March 2006.
- Ebi Jose and Martin D. Adams, "Millimetre Wave RADAR Spectra Simulation and Interpretation for Outdoor SLAM" , Journal of Robotic Research Centre, Nanyang Technological University, Singapore. July 2004.
- Ebi Jose and Martin D. Adams and John Mullane, "Predicting Millimetre Wave Radar Spectra for Autonomous Navigation", IEEE Sensors Journal, June 2009. (Accepted).

Conferences:

- Ebi Jose and Martin D. Adams, “Millimetre Wave RADAR Spectra Simulation and Interpretation for Outdoor SLAM” , The 2004 IEEE International Conference on Robotics and Automation (ICRA) 2004, New Orleans, USA. April 2004.
- Ebi Jose and Martin D. Adams, “Relative RADAR Cross Section based Feature Identification with Millimetre Wave RADAR for Outdoor SLAM” , IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2004, Sendai, Japan. September 2004.
- Ebi Jose and Martin D. Adams, “Feature Identification by Estimating Relative RADAR Cross Section with Millimetre Wave RADAR”, The Eighth International Conference on Control, Automation, Robotics and Vision (ICARCV) 2004, Kunming, China, December 2004.
- Ebi Jose and Martin D. Adams, “An Augmented State SLAM Formulation for Multiple Line-of-Sight Features with Millimetre Wave RADAR”, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2005, Canada. August 2005.

References

- [1] MARTIN ADAMS. Special issue on slam - algorithmic advances, loop closing, measurement classification and outdoor implementations. *Robotics and Autonomous Systems*, 2006. 6
- [2] M.D. ADAMS, F. TANG, C. SOK, AND W.S. WIJESOMA. Convergent smoothing and segmentation of noisy range data in multi-scale space. In *IEEE Transactions on Robotics*, **24**, pages 746–753, June 2008. 207
- [3] M.E. ADAMSKI, K.S. KULPA, M. NALECZ, AND A. WOJTKIEWICZ. Phase noise in two-dimensional spectrum of video signal in fmcw homodyne radar. In *13th International Conference on Microwaves, Radar and Wireless Communications, (MIKON-2000)*, **2**, pages 645–648, 2000. 15
- [4] E.K. AL-HUSSAINI. Performance of the greater-of and censored greater-of detectors in multiple target environments. *Radar and Signal Processing, IEE Proceedings F*, **135**:193–198, 1988. 49
- [5] S. ARULAMPALAM, S. MASKELL, N. GORDON, AND T. CLAPP. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *IEEE Transactions of Signal Processing*, **50**(2):174–188, February 2002. 165
- [6] TIM BAILEY, JUAN NIETO, JOSE GUIVANT, MICHAEL STEVENS, AND EDUARDO NEBOT. Consistency of the ekf-slam algorithm. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 200
- [7] Y. BAR-SHALOM AND T.E. FORTMANN. *Tracking and Data Association*. Academic Press, 1988. 136, 168, 171, 214, 215

REFERENCES

- [8] M. BARKAT. *Signal Detection and Estimation*. Artech House, Norwood, MA, 2nd edition, 2005. 49
- [9] M. BEROUTI, R. SCHWARTZ, AND J. MAKHOUL. Enhancement of speech corrupted by acoustic noise. In *Proceedings of the IEEE ICASSP'79*, pages 208–211, 1979. 100
- [10] S.F. BOLL. Supression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustic, Speech and Signal Processing*, **ASPP-27**(2):113–120, April 1979. 99
- [11] S. BORTHWICK. *A Multi-Hypothesis Approach to Autonomous Vehicle Navigation and Map Building*. PhD thesis, Department of Engineering Science, University of Oxford, 1995. 204
- [12] G.M. BROOKER, S. SCHEDING, M.V. BISHOP, AND R.C. HENNESSY. Development and application of millimeter wave radar sensors for underground mining. *Sensors Journal, IEEE*, **5**. 28
- [13] KOK SENG CHONG AND LINDSAY KLEEMAN. Accurate odometry and error modelling for a mobile robot. In *IEEE International Conference on Robotics & Automation*, page 2783–2788, 1997. 28
- [14] STEVE CLARK. *Autonomous Land Vehicle Navigation Using Millimetre Wave Radar*. PhD thesis, Australian Centre for Field Robotics, University of Sydney, 1999. 13, 250
- [15] STEVE CLARK AND HUGH F. DURRANT-WHYTE. The design of a high performance mmw radar system for autonomous land vehicle navigation. *FSR'97 Proceedings of the International Conference on Field and Service Robotics*, A Zelinsky, eds. Australian Robotic Association Inc Sydney NSW, Australia, pages 292–299, 1997. 12
- [16] STEVE CLARK AND G.DISSANAYAKE. Simultaneous localisation and map building using millimetre wave radar to extract natural features. In *International Conference on Robotics and Automation (ICRA)*, pages 1316–1321, Detroit, Michigan, May 1999. IEEE. 48

REFERENCES

- [17] ISRAEL COHEN AND BARUCH BERDUGO. Noise estimation by minima controlled recursive averaging for robust speech enhancement. *IEEE Signal Processing Letters*, **9**(1), 2002. 96
- [18] INGEMAR J. COX AND JOHN J. LEONARD. Modeling a dynamic environment using a bayesian multiple hypothesis approach. *Artif. Intell.*, **66**(2):311–344, 1994. 204
- [19] N.C CURRIE AND C.E. BROWN. *Principles and Applications of MMW Radar*. Artech House, Dedham, MA, 1987. 48
- [20] G. DAVIDSON, H.D GRIFFITHS, AND S. ABLETT. Analysis of high-resolution land clutter. *IEE proceedings - Vis. Image Signal Processing*, **151**(1), February 2004. 250
- [21] GAMINI DISSANAYAKE, STEFAN B. WILLIAMS, HUGH DURRANT-WHYTE, AND TIM BAILEY. Map management for efficient simultaneous localization and mapping (SLAM). *Autonomous Robots*, **12**:267–286, 2002. 218
- [22] M.W.M.G. DISSANAYAKE, P. NEWMAN, S. CLARK, DURRANT-WHYTE H.F., AND M. CSORBA. A solution to the simultaneous localization and map building (slam) problem. *Robotics and Automation, IEEE Transactions on*, **17**-3, June 2001. 28, 218
- [23] ARNAUD DOUCET AND NANDO DE FREITAS. *Sequential Monte Carlo Methods in Practice*, **XIV** of *Information Science and Statistics*. Springer, 2001. 165, 166
- [24] H.F. DURRANT-WHYTE, J.I. NIETO, AND F.T. RAMOS. Combining object recognition and slam for extended map representations. *Springer Tracts in Advanced Robotics- Experimental Robotics*, 2008. 168
- [25] HUGH DURRANT-WHYTE AND TIM BAILEY. Simultaneous localisation and mapping (slam): Part i the essential algorithms. *Robotics and Automation Magazine*, June 2006. 6
- [26] ELIAS-FUST⁺ E, A. R., M. G. DEMERCADO, AND E. R. DAVO. Analysis of some modified ordered statistic cfar: Osgo and osso cfar. *IEEE Transactions on Aerospace and Electronic Systems*, **26**:197–202, Jan. 1990. 49

REFERENCES

- [27] M.B. EL MASHADE. Detection performance of the trimmed-mean cfar processor with noncoherent integration. *Radar, Sonar and Navigation, IEE Proceedings*, **142**(1):18–24, Feb 1995. 49
- [28] JR. FERRIS, D.D. AND N.C. CURRIE. Microwave and millimeter-wave systems for wall penetration. *Proceedings of the SPIE - The International Society for Optical Engineering*, **3375**:269 – 79, 1998. 135
- [29] M.A. FISCHLER AND R.C. BOLLES. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, **24**(6):381–395, June 1981. 207
- [30] DIETER FOX, WOLFRAM BURGARD, AND FRANK DELLAERT. Monte carlo localization: Efficient position estimation for mobile robots. In *In Proc. of the National Conference on Artificial Intelligence (AAAI)*, pages 343–349, 1999. 166
- [31] P.P. GANDHI AND S.A. KASSAM. Analysis of cfar processors in nonhomogeneous background. *IEEE Transactions on AES*, **4**(24):427–445, July 1988. 48, 49, 52, 60, 90, 92, 241, 258
- [32] P.P. GANDHI AND S.A. KASSAM. Optimality of the cell averaging cfar detector. In *IEEE Transactions on Information Theory*, **40**, pages 1226–1228, July 1994. 242, 250
- [33] JUAN PABLO GONZALEZ AND ANTHONY (TONY) STENTZ. Planning with uncertainty in position using high-resolution maps. In *In Proceedings of the IEEE Int. Conference on Robotics & Automation (ICRA) 2007*, April 2007. 1
- [34] ROBERT M. GRAY AND LEE D. DAVISSON. *Introduction to Statistical Signal Processing*. Cambridge University Press, December 2004. 99
- [35] ZOHEIR HAMMOUDI AND FAOUZI SOLTANI. Distributed ivi-cfar detection in non-homogeneous environments. *Signal Process.*, **84**(7):1231–1237, 2004. 50
- [36] V.G. HANSEN. Constant false alarm rate processing in search radars. In *Proceedings of IEE International Radar Conference*, pages 325–332, London, 1973. 49

REFERENCES

-
- [37] V.G. HANSEN AND J.H SAWYER. Detectability loss due to greatest of selection in a cell-averaging cfar. *IEEE Transactions on AES*, **16**:115–118, 1980. 27, 255, 256
 - [38] R. HARTLEY AND A. ZISSERMAN. *Multiple View Geometry in Computer Vision*. University Press, Cambridge, 2001. 209
 - [39] S.D. HIMONAS AND M. BARKAT. Automatic censored cfar detection for non-homogeneous environments. *IEEE Transactions on Aerospace and Electronic Systems*, **28**, 1992. 49
 - [40] GUIVANT J, NEBOT E, AND BAIKER S. Autonomous navigation and map building using laser range sensors in outdoor applications. *Journal of Robotic Systems*, **17**(10):565–283, October 2000. 136, 168
 - [41] EBI JOSE AND MARTIN D. ADAMS. Feature identification by estimating relative radar cross section with millimetre wave radar. In *The Eighth International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Kunming, China., December 2004. 55, 56
 - [42] EBI JOSE AND MARTIN D. ADAMS. Millimetre wave radar spectra simulation and interpretation for outdoor slam. In *International Conference on Robotics and Automation (ICRA)*, New Orleans, USA, April 2004. 96, 133, 139, 142, 145, 147
 - [43] EBI JOSE AND MARTIN D. ADAMS. Relative radar cross section based feature identification with millimetre wave radar for outdoor slam. In *International Conference on Intelligent Robots and Systems (IROS)*, Sendai, Japan, September 2004. 115, 133
 - [44] EBI JOSE AND MARTIN D. ADAMS. An augmented state slam formulation for multiple line-of-sight features with millimetre wave radar. In *International Conference on Intelligent Robots and Systems (IROS)*, Alberta, Canada, August 2005. 26, 133, 148
 - [45] S. JULIER AND J. UHLMANN. A new extension of the Kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls, Orlando, FL*, 1997. 165

REFERENCES

- [46] M.A. KHALIGHI AND M.H. BASTANI. Adaptive cfar processor for nonhomogeneous environments. *Aerospace and Electronic Systems, IEEE Transactions on*, **36**, 2000. 48, 50
- [47] L. KLEEMAN AND R. KUC. Mobile robot sonar for target localization and classification. In *The International Journal of Robotics Research*, **14**, Aug 1995. 201
- [48] DIRK LANGER. An integrated mmw radar system for outdoor navigation. In *International Conference on Robotics and Automation*, pages 417–422, Minneapolis, Minnesota, April 1996. IEEE. 13
- [49] DIRK LANGER. *An Integrated MMW Radar System for Outdoor Navigation*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, January 1997. 14
- [50] E. LEHMANN. *Testing Statistical Hypotheses*. New York: Wiley, 1986. 98
- [51] S. LI AND A. HAYASHI. Robot navigation in outdoor environments by using gps information and panoramic views. In *Intelligent Robots and Systems, IEEE/RSJ International Conference on*, **1**, pages 570–575, 1998. 1
- [52] X.R. LI. Tracking in clutter with strongest neighbor measurements—part i: Theoretical analysis. In *IEEE. Trans. Automat. Control*, **43**, pages 1560–1578, 1998. 215
- [53] D. MAK SAROV AND H. DURRANT-WHYTE. Mobile vehicle navigation in unknown environments: a multiple hypothesis approach. In *Control Theory and Applications, IEE Proceedings - Dept. of Eng. Sci., Oxford Univ.*, **142**, pages 385–400, July 1995. 204
- [54] M. MONTEMERLO, S. THRUN, D. KOLLER, AND B. WEGBREIT. Fastslam: A factored solution to the simultaneous localization and mapping problem. In *Proc. AAAI National Conference on Artificial Intelligence*, pages 593–598, 2004. 4, 165

REFERENCES

- [55] MICHAEL MONTEMERLO AND SEBASTIAN THRUN. Simultaneous localization and mapping with unknown data association using fastslam. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, pages 1985–1991, 2003. 4
- [56] HANS MORAVEC. Certainty grids for sensor fusion in mobile robots. *Sensor Devices and Systems for Robotics*, pages 243–276, 1989. 164
- [57] P. MOUTARLIER AND R. CHATILA. Stochastic multisensory data fusion for mobile robot location and environment modeling. In *5th Int. Symposium on Robotics Research, Tokyo*. 165
- [58] P. MOUTARLIER AND R. CHATILA. An experimental system for incremental environment modeling by an autonomous mobile robot. *1st International Symposium on Experimental Robotics*, June 1989. 165
- [59] JOHN MULLANE, EBI JOSE, MARTIN DAVID ADAMS, AND WIJERUPAGE SARDHA WIJESOMA. Including probabilistic target detection attributes into map representations. In *Robotics and Autonomous Systems*, number 1 in 55, pages 72–85, 2007. 19
- [60] JOSE NEIRA AND JUAN D. TARDOS. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, December 2001. 4
- [61] JUAN NIETO, TIM BAILEY, AND EDUARDO NEBOT. Recursive scan-matching slam. *Robotics and Autonomous Systems*, 55(1):39–49, 2007. 203, 230
- [62] Y. NOROUZI, F.GINI, M.M. NAYEBI, AND M.GRECO. Non-coherent radar cfar detection based on goodness-of-fit tests. In *IET Radar, Sonar & Navigation*, 1, pages 98–105, April 2007. 92
- [63] ALAN V. OPPENHEIM AND RONALD W. SCHAFER. *Digital Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1975. 13
- [64] H. V. POOR. *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1988. 98

REFERENCES

- [65] WILLIAM H. PRESS, SAUL A. TEUKOLSKY, WILLIAM T. VETTERLING, AND BRIAN P. FLANNERY. Numerical recipes in C: The art of scientific computing. *Cambridge Univ. Press, N.Y.*, **2**, 1992. 56
- [66] J.A. RITCEY AND J.L. HINES. Performance of max-mean level detector with and without censoring. *IEEE Transactions on Aerospace and Electronic Systems*, **25**, 1989. 49
- [67] J.A. RITCEY AND J.L. HINES. Performance of max family of order-statistic cfar detectors. *IEEE Transactions on Aerospace and Electronic Systems*, **27**, 1991. 49
- [68] H. ROHLING. Radar cfar thresholding in clutter and multiple target situations. In *IEEE Transactions, AES-19*, pages 608–621, 1983. 49, 50, 79, 242, 258, 261
- [69] H. ROHLING AND R. MENDE. Os cfar performance in a 77 ghz radar sensor for car application. In *CIE International Conference of Radar*, pages 109 – 114, October 1996. 260, 261
- [70] PETER J. ROUSSEEUW AND ANNICK M. LEROY. *Robust Regression and Outlier Detection*. Wiley, 1987. 207
- [71] CA) BIRCSAK GARIN S. (NORTH HOLLYWOOD CA) SCHAAF, GREGORY (VAN NUYS. Radar clutter simulator. Technical Report 4644357, February 1987. 17
- [72] S. SCHEDING, G. BROOKER, M. BISHOP, R. HENNESSY, AND A. MACLEAN. Terrain imaging millimetre wave radar. In *International Conference on Control, Automation, Robotics and Vision*, Singapore, November 2002. 12
- [73] MERRILL I SCOLNIK. *Introduction to Radar Systems*. McGraw Hill, New York, 1982. 48, 136, 141, 145
- [74] STEPHEN SE, DAVID LOWE, AND JIM LITTLE. Global localization using distinctive visual features. In *In International Conference on Intelligent Robots and Systems, IROS 2002*, pages 226–231, 2002. 207
- [75] D.A. SHNIDMAN. Expanded swerling target models. In *Aerospace and Electronic Systems, IEEE Transactions on*, **39**, pages 1059– 1069, July 2003. 23

REFERENCES

- [76] M. SHOR AND N. LEVANON. Performances of order statistics cfar. *IEEE Trans. on Aerospace and Electronic Systems*, **AES-27**(2):214–224, 1991. 49
- [77] SANJIV SINGH AND JAY WEST. Cyclone: A laser scanner for mobile robot navigation. Technical Report CMU-RI-TR-91-18, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 1991. 3
- [78] MERRILL I. SKOLNIK. *Radar Handbook (2nd Edition)*. McGraw-Hill, 1990. 19
- [79] M.E. SMITH AND P.K. VARSHNEY. Intelligent cfar processor based on data variability. In *IEEE Trans. Aerospace and Electronic Systems*, **36**, pages 837–847, 2000. 49, 50, 57, 70
- [80] R.C. SMITH AND P. CHEESEMAN. On the representation and estimation of spatial uncertainty. *International Journal of Robotics Research*, 1986. 165
- [81] P. SWERLING. Probability of detection for fluctuating targets. *IRE Trans. Inform. Theory*, **6**(271):273–308, 1960. 136
- [82] S. THRUN. Robotic mapping: A survey, 2002. 164
- [83] P. H. S. TORR AND A. ZISSERMAN. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, **78**:138–156, 2000. 207
- [84] H.L. VAN TREES. *Detection, Estimation and Modulation Theory - Part I*. Willey, New York, 1968. 97, 98
- [85] G.V. TRUNK. Range resolution of targets using automatic detectors. *IEEE Transactions on AES*, **14**:750–755, Sept 1978. 49, 242
- [86] R VON MISES. *Mathematical Theory of Probability and Statistics, Chapters IX(C) and IX(E)*. New York: Academic Press, 1964. 57
- [87] S. WATTS. The Performance of cell-averaging CFAR systems in sea clutter. In *The Record of the IEEE 2000 International Radar Conference*, May 2000. 250

REFERENCES

- [88] M. WEISS. Analysis of some modified cell-averaging CFAR processors in multiple-target situations. *IEEE Transactions on AES*, **18**:102–113, Jan 1982. 242, 256
- [89] W. S. WIJESOMA, L. D. L. PERERA, AND M.D. ADAMS. Towards multidimensional assignment data association in robot localization and mapping. *IEEE Transactions on Robotics*, 2006. 166
- [90] W.S. WIJESOMA, L.D.L. PERERA, M.D. ADAMS, AND S. CHALLA. An analysis of the bias correction problem in simultaneous localization and mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, (IROS)*., 2005. 165
- [91] RAND R WILCOX. *Fundamentals of Modern Statistical Methods*. Springer, 2001. 11, 23