

Minimal Infrastructure Radio Frequency Home Localisation Systems

Damian Kelly



NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Department of Electronic Engineering
National University of Ireland, Maynooth
Co. Kildare
Ireland

February 26th, 2010

Supervisor: Dr. Seán McLoone


embarkinitiative
Investing in People and Ideas



The ability to track the location of a subject in their home allows the provision of a number of location based services, such as remote activity monitoring, context sensitive prompts and detection of safety critical situations such as falls. Such pervasive monitoring functionality offers the potential for elders to live at home for longer periods of their lives with minimal human supervision.

The focus of this thesis is on the investigation and development of a home room-level localisation technique which can be readily deployed in a realistic home environment with minimal hardware requirements. A conveniently deployed Bluetooth[®] localisation platform is designed and experimentally validated throughout the thesis. The platform adopts the convenience of a mobile phone and the processing power of a remote location calculation computer. The use of Bluetooth[®] also ensures the extensibility of the platform to other home health supervision scenarios such as wireless body sensor monitoring.

Central contributions of this work include the comparison of probabilistic and non-probabilistic classifiers for location prediction accuracy and the extension of probabilistic classifiers to a Hidden Markov Model Bayesian filtering framework. New location prediction performance metrics are developed and significant performance improvements are demonstrated with the novel extension of Hidden Markov Models to higher-order Markov movement models. With the simple probabilistic classifiers, location is correctly predicted 80% of the time. This increases to 86% with the application of the Hidden Markov Models and 88% when high-order Hidden Markov Models are employed.

Further novelty is exhibited in the derivation of a real-time Hidden Markov Model Viterbi decoding algorithm which presents all the advantages of the original algorithm, while producing location estimates in real-time. Significant contributions are also made to the field of human gait-recognition by applying Bayesian filtering to the task of motion detection from accelerometers which are already present in many mobile phones. Bayesian

filtering is demonstrated to enable a 35% improvement in motion recognition rate and even enables a floor recognition rate of 68% using only accelerometers. The unique application of time-varying Hidden Markov Models demonstrates the effect of integrating these freely available motion predictions on long-term location predictions.

Acknowledgments

First and foremost, I would like to express my sincerest gratitude to my PhD supervisor, Seán McLoone. Without Seán's uniquely subtle, but persistently involved, approach to supervision I would certainly not have grown into the researcher I am today. I can also confidently say that a great deal of this thesis would have not been possible without the numerous interesting discussions we have had over the past three and a half years.

Next I would like to thank all my friends for their support, and even tolerance, of my continual sacrifice made in the pursuit of this PhD. I would especially like to thank Lorcan Walsh and Shane Lynn who have worked with me since the first days of the PhD. Lorcan has always motivated me to better myself and Shane's practical approach to problem solving has been an invaluable aid in the many issues I have encountered over the years.

A significant contributor to my enjoyment of the past few years has been the great community spirit in the NUI Maynooth Electronic Engineering Department. I cannot recall any instances when a member of staff was too engrossed in their own work to speak with a student in need of help and the positive attitude of everyone in the department has made this one of the best environments I have ever worked in. I would especially like to thank John Maloco, Denis Buckley and Joanne Bredin for ensuring any practical obstructions to completing this PhD were easily alleviated. I have also had much support and mentoring from Ronan Farrell and Rudi Villing, which has been invaluable in my professional and personal development.

I am also extremely appreciative all of all the invaluable domain expertise I have benefited from in Intel's Digital Health group. I have received an unquantifiable amount of input and support from Terry O'Shea, Adrian Burns, John Delaney, Mick McGrath, Julie Behan, Barry Greene and Beth Logan.

I also acknowledge the generous investments of the Irish Research Council in Science, Engineering and Technology and the Institute of Microelectronics and Wireless Systems

in my research, without which this endeavour would have been untenable.

And most importantly, I would like to thank my parents, Hilary and Alan, my brother, Colin, and my sister, Elaine, for their unrelenting love and encouragement. Without their belief in my abilities and their constant support I would not have had the conviction necessary to see my PhD through to its conclusion. I only hope I can continue to make them proud in this and whatever else I choose to do in my life.

1	Introduction	1
1.1	Smart Home Health Monitoring	3
1.2	Home Localisation Techniques	4
1.3	Aims and Scope of Thesis	6
1.4	Contributions of this Thesis	7
1.5	Thesis Organisation	9
2	Literature Survey	11
2.1	Radio Wave Propagation	12
2.1.1	Propagation Mechanisms	13
2.1.2	Radio Propagation Models	25
2.2	Positioning Theory	31
2.2.1	Direct Techniques	32
2.2.2	Sequential Bayes' Filters	39
2.3	Existing Localisation Systems	44
2.3.1	WLAN	44
2.3.2	Cellular Network	45
2.3.3	Bluetooth	48
2.3.4	IEEE 802.15.4/ZigBee [®]	50
2.3.5	Ultrasonic	52
2.3.6	Infrared	52
2.3.7	RFID	55
2.3.8	Combinations of Technologies	57
2.4	Summary and Motivation	59

3	Background Theory	62
3.1	k -Nearest Neighbour	63
3.2	Support Vector Machines	65
3.3	Naive Bayes Classifier	69
3.4	Discriminant Analysis	72
3.5	Gaussian Mixture Models	74
3.6	Gaussian Processes and Informative Vector Machines	79
3.7	Relevance Vector Machines	82
3.8	Conclusions	85
4	Location Tracking Framework	87
4.1	Hardware	87
4.2	Available Signals	90
4.2.1	Bluetooth [®]	91
4.2.2	ZigBee [®]	96
4.3	Localisation in a Static Scenario	97
4.3.1	Localisation Accuracy Metric	98
4.3.2	Multiple BSC Localisation	100
4.3.3	Single BSC Localisation	101
4.4	Localisation in a Realistic Scenario	104
4.4.1	Training Data Acquisition	104
4.4.2	Pre-Processing Filter	106
4.4.3	Signal Redundancy	107
4.4.4	Accuracy Metric Suitability	109
4.5	Conclusions	109
5	Long-Term Deployment	111
5.1	Alternate Technologies	112
5.1.1	Radio Frequency Identification	113
5.1.2	Passive Infrared	114
5.2	Accuracy Metrics	115
5.3	Long-Term Tests	119
5.3.1	Single Bluetooth [®] Access Point	120
5.3.2	Single Basestation Computer with Increased Signal Diversity	125
5.3.3	Signal Diversity from Multiple Beacons	128
5.3.4	Passive Infra Red Localisation	134
5.4	Algorithm Improvements	137
5.4.1	Preprocessing Filter	137
5.4.2	Time-Lagged Preprocessing	141

5.4.3	Sparse Approximations	143
5.4.4	Cellular Signal Treatment	147
5.4.5	Incorporating Prior Knowledge	148
5.4.6	Uncertainty Rejection	149
5.4.7	Combined Accuracy Augmentations	154
5.5	Conclusions	158
6	Hidden Markov Model Constrained Localisation	161
6.1	Dynamic Performance Measures	162
6.1.1	Transition Count	162
6.1.2	Distance Travelled	164
6.1.3	Transition Delay	165
6.1.4	Overall Performance Visualisation	166
6.2	Hidden Markov Models	167
6.2.1	Markov Chains	168
6.2.2	HMM Theory	169
6.2.3	The Viterbi Algorithm	170
6.2.4	Bayesian Filtering Performance	174
6.2.5	Bayesian Filtering with Augmented Classifiers	175
6.2.6	Overall HMM Performance	179
6.3	High Order Markov Processes	180
6.3.1	2nd Order Hidden Markov Models	180
6.3.2	3rd Order Hidden Markov Models	182
6.4	Real-Time Viterbi Decoding	184
6.4.1	Fixed Window Backtracking	185
6.4.2	Stable Partial Probability Backtracking	187
6.5	Conclusions	190
7	Time-Varying Hidden Markov Models	192
7.1	Movement Detection	193
7.1.1	Binary Moving/Static State Detection	195
7.1.2	Detailed Movement Information	198
7.1.3	HMM Motion Filtering	205
7.2	Online-Varying Transition Probabilities	213
7.2.1	Binary Motion Predictions	215
7.2.2	Three-State Motion Predictions	218
7.2.3	Six-State Motion Predictions	220
7.3	Conclusions	223

8	Concluding Summary and Future Work	225
8.1	Concluding Summary	225
8.1.1	Localisation and Classification Literature Review	225
8.1.2	Localisation Hardware Platform Design	226
8.1.3	Long-term Localisation Accuracy	226
8.1.4	Hidden Markov Models	227
8.1.5	Time-Varying HMMs	227
8.1.6	Limitations	228
8.2	Future Work	230
8.2.1	Bluetooth [®] Beacon Location Optimisation	230
8.2.2	Automatic Room Labelling	231
8.2.3	Motion Sensing Redundancy	231
8.2.4	Efficient High-Order HMM Implementations	232

List of Figures

1.1	An age pyramid illustrating the change in population age distribution for males and females in 2008 and predicted for 2060, taken from Giannakouris (2008).	2
1.2	Proportion of the population over the age of 60 in 1950, 2000 and projected into 2050, taken from Department of Economic and Social Affairs, Population Division (2009).	2
1.3	Global Potential Support Ratio in 1950, 2000 and projected into 2050, taken from Department of Economic and Social Affairs, Population Division (2009).	3
2.1	Large Scale and Small Scale Fading with increasing Transmitter-Receiver separation, taken from Rappaport (2002).	12
2.2	Path Loss with increasing transmitter-receiver separation distance for $f \approx 2.4GHz$	15
2.3	Angles of Incidence, Reflection and Transmission, modified from Rappaport (2002).	17
2.4	Ground reflection model, reproduced from Rappaport (2002).	19
2.5	Fresnel zones as they interact with a transparent plane between the transmitter and receiver, reproduced from Rappaport (2002).	21
2.6	Knife-edge diffraction, modified from Rappaport (2002).	22
2.7	Knife-edge diffraction gain as a function of the Fresnel-Kirchoff diffraction parameter, from Rappaport (2002).	23
2.8	Approximating two knife-edges with a single knife-edge, reproduced from Rappaport (2002).	23
2.9	Scattering of an incident wave.	24

2.10	Triangulation localisation methods can calculate the position of circle C given the known locations of squares A and B and some other information, either distances, d_1 and d_2 or angles θ_1 and θ_2	34
2.11	Lateralation used to predict the position of C using 2 fixed Access Points (APs) and 3 fixed APs. (a) Using only 2 APs results in two possible locations of C for the measurements d_1 and d_2 . (b) The availability of 3 APs removes uncertainty.	35
2.12	Decision tree for selecting the appropriate localisation technique (and most likely technology in brackets) for a given deployment scenario.	61
3.1	Implemented location classification framework.	64
3.2	Effect of varying k parameter on classification region flexibility.	65
3.3	Impact of the slack parameter, C , on classification flexibility. A slack parameter of ∞ is equivalent to permitting no misclassifications, causing overfitting. The optimum value of C for this dataset, 3, results in smooth classification borders. The calculated support vectors are circled.	69
3.4	The left figure illustrates decision regions along with the means and 50% confidence interval ellipses for each class. The right figure illustrates the probability of class 1 given the measurements. Note that the probability of class 1 never reaches 1 because it is normalised to ensure $P(R_1 \mathbf{x})$ and $P(R_2 \mathbf{x})$ sum to 1.	71
3.5	Decision regions for LDA and QDA. The left figure indicates that the common covariance assumption is not valid for the illustrative dataset. QDA achieves more suitable decision regions by allowing a different covariance ellipse for each class.	74
3.6	Block diagram view of a GMM density estimator for \mathbf{x} in class k , modified from Reynolds and Rose (1995).	75
3.7	Comparison of GMM classifiers optimised with EM, GEM and FJ algorithms. The weights of each component are indicated in black.	78
3.8	A comparison of GP classification with IVM classification using 100, 125 and 130 active vectors. Active vectors are circled.	82
3.9	Decision regions for an RVM classifier. Active vectors are circled.	85
4.1	Hardware utilised for Bluetooth [®] deployment.	89
4.2	ZigBee [®] hardware: SHIMMER without case (left), SHIMMER with case (centre) and TelosB mote (right), compared with a 1 Euro coin (bottom) for scale.	90

4.3	Bluetooth [®] platform RSSI, LQ and CRSSI for increasing T-R separation in a free field. RSSI and LQ are unitless quantities and CRSSI is measured in dBm.	93
4.4	LQ-BER relationship for the Blueradios BR-SC30N.	94
4.5	The 4-room indoor test environment.	95
4.6	RSSI, LQ and CRSSI throughout the test environment.	95
4.7	CRSSI for the cell towers detectable in the test environment.	95
4.8	ZigBee [®] platform RSSI and LQI for increasing T-R separation in a free-field. LQI is a unitless quantity.	97
4.9	ZigBee [®] RSSI and LQI throughout the test environment.	98
4.10	Two input feature-spaces for the Bluetooth [®] and ZigBee [®] multiple BSC localisation scenarios.	99
4.11	Two input feature-spaces for the Bluetooth [®] and ZigBee [®] single BSC localisation scenarios.	101
4.12	Three input feature-spaces for the Bluetooth [®] localisation platform.	103
4.13	Three-feature-spaces for two different techniques of obtaining training data for the dynamic localisation scenario.	105
4.14	The effect of filtering on localisation accuracy.	106
4.15	The three input feature-space with a 20 sample mean filter applied.	107
4.16	Accuracy when (a)CRSSI and (b)CID signals are unavailable.	108
4.17	The effect on localisation accuracy of using an accuracy metric more appropriate to the deployment scenario.	109
5.1	The chosen realistic test environment. Red dots indicate the positions of the PIR sensors.	112
5.2	Comparison of RFID derived location estimates and 100% accurate voice annotated movements for the first 200 seconds of test data.	114
5.3	Cumulative distribution function for LDA location classification of a day's worth of test data.	116
5.4	Comparison of the equal contribution mean accuracy weights and empirically derived weights.	118
5.5	Examples of RFID tags affixed to doors at positions convenient for manual scanning. Note that the numbers are only written on the tags for installation purposes, the RFID system automatically derives these numbers from the stored tag IDs.	119
5.6	Comparison of occupancy levels with PIR-derived location and accurate RFID derived location. Notice how PIR location becomes extremely varied and uncorrelated with RFID when occupancy goes to 3 people.	120

5.7	Frequency of correct room prediction for the k -Nearest Neighbour (k NN) classifier. These values are obtained directly from the confusion matrix for the classifier.	122
5.8	Effect of increasing maximum permitted samples per room on the empirically derived accuracy measure.	124
5.9	Effect of increasing maximum permitted samples per room on EA and GA for k NN, QDA and GMM classifiers.	124
5.10	Mean number of samples per location per day for the SAP dataset.	125
5.11	The influence of access point antenna orientation and type on antenna radiation profiles, hence, environment signal profiles.	126
5.12	Examples of the deployed “dumb” access points.	129
5.13	The connection topology for the multiple AP deployment. Each arrow represents a Bluetooth [®] connection. The start of each arrow indicates the connection master and the end of each arrow represents the slave.	129
5.14	Locations of the APs, indicated by green triangles.	130
5.15	GA and EA as a function of training dataset size for the multiple AP deployment scenario.	131
5.16	The effect of the subset of available APs on the EA.	133
5.17	The effect of the subset of available APs on the GA.	133
5.18	Effect of occupancy levels on PIR GA and EA.	136
5.19	Effect of filtering on EA in the multiple Bluetooth [®] beacon deployment scenario.	137
5.20	Raw CID, CRSSI and BSC RSSI and LQ signals.	138
5.21	The effect of the combination of RSSI and LQ filtering levels on mean EA.	140
5.22	Using a tapped delay line to generate a time-lagged set of classifier input features from the original feature vector.	142
5.23	EA as a function of the number of lagged features per original feature, L	142
5.24	A comparison of the different uncertainty rejection methods on the same uncertain problem.	150
5.25	EA, GA and rejection ratios for increasing absolute and relative rejection thresholds.	152
5.26	Illustration of the increase in accuracy when uncertain samples are assigned the class label of the last certain prediction.	153
5.27	The overall system combining the accuracy improvements explored in this chapter.	155
5.28	Accuracy levels at each processing stage in the overall system.	157

5.29	Automatic RFID label acquisition technique. When the foot moves across the RFID tag strip an RFID reading occurs. Then the IDs of the read tags can be used to resolve the room-level location.	159
6.1	Localisation algorithm performance comparison for the different algorithms. The projection in (a) allows a visualisation of the best overall performing classifiers. (b) allows a comparison of the classifiers in terms of EA and MTD and the projection in (c) allow a comparison of the classifiers in terms of PDIF and EA. EA and PDIF are optimal at values of 1. MTD is optimal at a value of 0.	167
6.2	A Markov chain representing the configuration of the environment used for the controlled test environment used in Chapter 4.	168
6.3	Estimating the partial probability of all states at the next instant of time using only the information about the probabilities at the current instant in time. Moving from left to right on the graph corresponds to time progression.	171
6.4	Schematic of the HMM localisation technique applied to raw signal data. .	174
6.5	Intensity plot of the transition probability matrix for the home environment.	175
6.6	Comparison of HMM predictions which use LDA derived observation probabilities and the original LDA predictions.	176
6.7	Schematic of the HMM localisation technique which pre-processes the raw data and includes priors in the density estimation technique.	176
6.8	A comparison of the performance trade-offs for the different HMM augmentations. As in Figure 6.1, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.	177
6.9	Comparison of HMM predictions which use augmented LDA derived observation probabilities and the original augmented LDA predictions. . . .	178
6.10	Comparison of the localisation trade-offs between the different localisation algorithms. The blue and red markers represent the previously presented raw data and augmented classifiers respectively. Note that the Aug QDA HMM marker in figure (c) cannot be seen because it has identical PDIF and MTD to the un-augmented QDA HMM.	179
6.11	Comparison of performance for the first, second and third order HMMs. Density estimates are derived from the augmented classifiers. As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures. .	184

6.12	(a) Empirical localisation accuracy as a function of backtracking window length, l , and (b) a zoomed comparison of the real-time decoding algorithm with the original Viterbi decoder.	187
6.13	Comparison of the original Viterbi backtracking technique and the fixed window backtracking technique. At sample 500 a backtracking step begins, initialising to room 6, since it has the highest partial probability.	188
6.14	(a) Empirical localisation accuracy and (b) mean and maximum prediction lag as a function of stable partial probability detection window length. Solid lines in Figure (b) are mean lags and dashed lines are maximum lags. The lines for QDA are not visible because they are identical to GMM in this graph.	188
6.15	Comparison of the performance of the original Viterbi decoding algorithm with the short-time Viterbi algorithm. As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.	189
7.1	RF signal windowed standard deviation, the actual user state progression and the corresponding LDA state prediction using the windowed standard deviation features.	196
7.2	Accelerometer windowed standard deviation, the actual user state progression and the corresponding LDA state prediction using the windowed standard deviation features.	197
7.3	Accelerometer frequency components, the actual user state progression and the corresponding state prediction using the accelerometer frequency components. This example uses a FFT window length of 20 samples.	201
7.4	Mean Recognition Rate (MRR) for the four movement types; “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs” as a function of the FFT window length.	202
7.5	MRR for the three movement types; “static”, “moving” and “using-the-stairs” as a function of the FFT window length.	203
7.6	MRR for the two movement types; “using-stairs” and “all-other-motions” as a function of the FFT window length.	204
7.7	Progression of the predicted state-sequence for (a) the original LDA classifier and (b) the Viterbi decoding of the state-sequence which uses the LDA density estimation technique.	206
7.8	Comparison of the MRR when (a) accelerometer standard deviation and (b) frequency domain features are used, as a function of standard deviation and FFT window lengths respectively.	207

7.9	MRR for (a) “stairs” versus “all-other” motion classifications and (b) “static” versus “moving” versus “stairs” motion classifications, as a function of FFT window length.	207
7.10	MRR for the four-state HMM predictions.	209
7.11	The influence of HMM Bayesian-filtering on the recognition rates for the six state motion classifications.	210
7.12	Progression of the predicted and actual motion state for (a) the unconstrained predictions and (b) the HMM predictions.	212
7.13	Intra-house movement model which takes into account the RF signals, the motion state of the subject and the room transition tendencies of the subject during a given motion state.	214
7.14	Transition probability matrices for when the user is (a) static and (b) moving. 216	
7.15	Performance comparison for original raw HMM (blue), original augmented HMM (red), binary motion-based raw HMM (green) and binary motion-based augmented HMM (black).	217
7.16	Performance comparison for original raw HMM (blue), original augmented HMM (red), three-state motion-based raw HMM (green) and three-state motion-based augmented HMM (black).	218
7.17	Comparison of predicted stairs use from the three-state motion classifier and the actual stairs transitions deduced from the training data labels. . . .	219
7.18	Room transition probability matrices derived from the training data for the three states, (a) “Static”, (b) “Moving” and (c) “Stairs”.	220
7.19	Performance comparison for original raw HMM (blue), original augmented HMM (red), six-state motion-based raw HMM (green) and six-state motion-based augmented HMM (black). As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.	221
7.20	Comparison of predicted stairs use from the six-state motion classifier and the actual stairs transitions deduced from the training data labels.	221
7.21	Room transition probability matrices for the six-state motion predictor. . .	222

List of Tables

2.1	Comparison of WLAN based localisation systems, part 1.	46
2.2	Comparison of WLAN based localisation systems, part 2.	47
2.3	Comparison of cellular network based localisation systems.	49
2.4	Comparison of Bluetooth [®] based localisation systems.	51
2.5	Comparison of IEEE 802.15.4 based localisation systems.	53
2.6	Comparison of Ultrasonic based localisation systems.	54
2.7	Comparison of Infrared (IR) based localisation systems.	56
2.8	Comparison of RFID based localisation systems.	58
2.9	Comparison of combination of technologies localisation systems.	60
4.1	An example confusion matrix. The diagonal terms indicate the frequency of correct prediction of the actual class. The off-diagonal terms indicate the frequency of predicting a particular class for the actual class denoted by the column label.	99
4.2	Mean accuracy for multiple BSC localisation tests. Standard Deviations are in parentheses.	100
4.3	Single Bluetooth [®] BSC static environment localisation accuracy.	102
4.4	Single ZigBee [®] BSC static environment localisation accuracy.	102
4.5	Localisation accuracies for a single Bluetooth [®] BSC using all available signals in a static environment. Overall accuracy is the mean of BSC1 and BSC2 accuracies.	103
4.6	Localisation accuracy for different training datasets.	106
5.1	Mean General Accuracy for location predictions over two consecutive days. Standard deviations are in parentheses.	121
5.2	Mean Empirical Accuracy for location predictions over two consecutive days.	121

5.3	Confusion matrix for k NN classifications from the SAP dataset, day 1. The diagonal elements, in bold, are the correct room predictions.	122
5.4	EA over two days when cellular connectivity information is no longer available.	125
5.5	The effect of the availability of a second Bluetooth [®] AP co-located with the original Bluetooth [®] AP on EA.	127
5.6	Confusion matrix for k NN classifications from the SBS dataset, day 1.	128
5.7	Confusion matrix for k NN classifications from the MAP dataset, day 1.	132
5.8	PIR localisation technique accuracy over 6 days.	134
5.9	PIR and peak Bluetooth [®] localisation accuracies. Peak Bluetooth [®] localisation accuracy is obtained using the k NN algorithm.	136
5.10	Mean EA for SVM classifiers and corresponding training times when different quantities of training data are available.	144
5.11	Mean EA for IVM classifiers and corresponding training times when different quantities of training data are available.	145
5.12	Mean EA for RVM classifiers and corresponding training times when different quantities of training data are available.	146
5.13	The effect of different treatments of the cellular connectivity information on EA.	148
5.14	The effect of <i>a priori</i> information inclusion on GA.	149
5.15	The effect of <i>a priori</i> information inclusion on EA.	149
5.16	The effect of imposing a higher probability margin on the uncertainty rejection and the corresponding rejection ratios.	157
6.1	Mean Transition Count Error (TCE) for the various classifiers compared to the true value.	163
6.2	Mean Transition Count Increase Factor (TCIF) for the various classifiers compared to the true value.	163
6.3	Mean PDE for the various classifiers compared to the true value [metres].	165
6.4	Mean PDIF for the various classifiers compared to the true value.	165
6.5	The MTD averaged over both test days. [seconds]	165
6.6	Localisation performance comparison for the HMM classifier using different observation probability models.	175
6.7	Localisation performance for the fully augmented HMM classifier.	178
6.8	2nd order HMM algorithm using the augmented density estimation techniques to produce the observation probabilities, $b_k(o_t)$	182
6.9	3rd order HMM algorithm using the augmented density estimation techniques to produce the observation probabilities, $b_k(o_t)$	182
7.1	Motion labels for motion prediction test.	194

7.2	MRR of movement type for different feature extraction techniques and classifiers. Values indicate mean values across both datasets and parentheses indicate standard deviations.	198
7.3	MRR of movement type from the four movements; “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs”, using RF and accelerometer variability.	199
7.4	Confusion matrices for the four movements, “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs”.	200
7.5	Confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 25 samples.	202
7.6	The three-state confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 25 samples. . . .	203
7.7	The two-state confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 10 samples. . . .	204
7.8	Confusion matrices for the optimal classifiers illustrated in Figure 7.9. . . .	208
7.9	Confusion matrix for the recognition of the four motion states, “static”, “moving”, “up-the-stairs” and “down-the-stairs”, when the HMM framework is adopted.	209
7.10	Confusion matrix for the application of un-constrained classifiers to the six-state motion classification problem.	211
7.11	Confusion matrix for the application of HMM classifiers to the six-state motion classification problem.	211
7.12	Localisation performance for the binary motion-dependent HMM classifier using raw RF data density estimates.	216
7.13	Localisation performance for the binary motion-dependent HMM classifier using augmented density estimates.	216

Nomenclature

AOA Angle of Arrival. The angle from which an incoming signal arrives at a receiver.

AP Access Point. The stationery element of a localisation system. Position of a Mobile Device is usually calculated relative to an Access Point.

BER Bit-Error-Rate. The proportion of bits transmitted across a communications link which arrive in an erroneous state.

BSC Basestation Computer. The computer in an RF localisation system which is responsible for retrieving or providing location-indicative signals from or to a mobile device.

cdf cumulative distribution function. A distribution function which illustrates the probability of a random variable taking on a value less than or equal to some value of interest.

CID Cell ID. The identification number of the cellular basestation to which a mobile phone is currently connected.

CRSSI Cellular Received Signal Strength Indication. An indicator of the strength of the cellular network signal incident on a mobile phone.

DCT Discrete Cosine Transform. Represents a sequence of data points as a sum of cosine functions with different fundamental frequencies. Commonly used for compression of images or sound.

EA Empirical Accuracy. A localisation accuracy measure which weights the accuracy of each room by the length of time typically spent in that room over a significant period of time. Hence, when the empirical accuracy over a short period of time is evaluated it

will correspond to the correct room classification of the system over a significant period of time.

EKF Extended Kalman Filters. Kalman filters which use linear approximations of non-linear process and/or measurement models.

EM Electromagnetic. A self-propagating wave which has electric and magnetic components.

FAF Floor Attenuation Factor. A factor used in a path loss model to account for RF signal attenuation due to inter-floor obstructions.

FFT Fast Fourier Transform. A computationally efficient technique of calculating the discrete Fourier transform, which can be used to transform time-domain signals into the frequency-domain representation.

GA General Accuracy. Similar to mean recognition rates. A measure of long-term symbolic localisation accuracy which assumes that equal periods of time is spent in each time in a typical week.

GFSK Gaussian Frequency Shift Keying . A technique of encoding bits as a smooth change in carrier frequency. Used by the Bluetooth[®] protocol.

GMM Gaussian Mixture Model. A probabilistic classifier in which the posterior densities are represented as a linear combination of several component densities.

GP Gaussian Process. A non-parametric regression and modelling technique which derives predictions from training data with Gaussian priors over its outputs.

GPS Global Positioning System. World-scale localisation system which uses radio signals from several satellites to estimate position.

GSM Global System for Mobile Communications. A globally accepted standard for digital cellular communications.

HMMs Hidden Markov Models. Implementation of a Bayes' filter, used when the unobservable states are discrete.

IR Infrared. Electromagnetic radiation with a wavelength just above that of visible light, sometimes employed in optical communications.

IVM Informative Vector Machine. A sparse representation of Gaussian Processes. Uses what is referred to as a differential entropy score to decide on the most informative vectors.

- k*NN** *k*-Nearest Neighbour. A classification algorithm which predicts class membership based on the majority class vote of the *k* nearest training samples.
- LDA** Linear Discriminant Analysis. A probabilistic classifier which approximates the data in different classes by a Gaussian distribution, all parameterised by the same covariance matrix and different means.
- LoS** Line of Sight. Direct radio wave propagation between a transmitter and a receiver without obstructions.
- LQ** Link Quality. A measurement available from many wireless communication devices. Usually related to bit-error-rate.
- LQI** Link Quality Indicator. A measurement similar to LQ, available from many wireless communication devices. ZigBee[®]
- MAF** Moving Average Filter. A low-pass filter which produces a filtered sample from the mean of the current and previous *N* samples.
- MD** Mobile Device. The non-stationery element to be localised in a localisation system.
- MRR** Mean Recognition Rate. A measure of the performance of a classifier which uses the mean of the recognition rate of each individual class. It is identical to the average of the diagonal terms of the confusion matrix.
- MTD** Mean Transition Delay. A localisation system dynamic performance measure which quantifies the average delay between a room transition occurring and it being detected.
- NBC** Naive Bayes Classifier. A maximum likelihood Bayes classifier which assumes the input features are conditionally independent given the class.
- PAF** Partition Attenuation Factor. A factor used in a path loss model to account for the attenuation in signal due to partitions.
- PDE** Predicted Distance Error. A measure of the dynamic performance of a localisation system which indicates the prediction error distance derived from the sequence location predictions.
- PDIF** Predicted Distance Increase Factor. A measure of the dynamic performance of a localisation system which presents the predicted travel distance as a ratio of the actual travel distance.
- PIR** Passive Infrared. A sensor which measures the infrared light radiating from objects. Such sensors can be used to predict the location of a moving inhabitant in their home environment.

- PL** Path Loss. Transmission power lost over a given transmission path.
- PLF** Polarisation Loss Factor. Path loss related to the angular difference in polarisation between the transmitting and receiving antennas.
- PSR** Potential Support Ratio. The ratio of potential workers (aged 15-64 years) to people aged 65 or over in the population.
- QDA** Quadratic Discriminant Analysis. A probabilistic classifier which approximates the data in different classes by a Gaussian distribution, parameterised by the mean and covariance of each class.
- RF** Radio Frequency. The waves in the electromagnetic spectrum which correspond to the rate of transmission and reception of data through an air interface.
- RFID** Radio Frequency Identification. Uses small passive or active tags attached to an object to transmit small pieces of information about that object via radio frequencies.
- ROC** Receiver Operating Characteristic curves allow the graphical consideration of the tradeoff between true positive rates and false positive rates for a binary probabilistic classifier, for a range of threshold classification probabilities.
- RSSI** Received Signal Strength Indication. An indicator of level of the received signal strength of a radio signal relative to some reference point. Not necessarily measured in dBm, units are device specific.
- RTT** Round Trip Time. A measure of distance between a transmitter and a receiver. Round trip time is approximately twice the Time of Flight between the transmitter and receiver.
- RVM** Relevance Vector Machine. A stochastic machine learning technique which uses sparse sets of basis functions to approximate functions.
- SNR** Signal to Noise Ratio. A ratio of the information bearing signal intensity to the background noise intensity.
- SVMs** Support Vector Machines. A classification method which creates a decision hyperplane maximising the distance between the hyperplane and a subset of vectors from the training set, or “Support Vectors”
- TCE** Transition Count Error. A measure of the dynamic performance of a localisation system which simply compares the predicted number of room transitions with the actual number of room transitions for a period of time.

TCIF Transition Count Increase Factor. A measure of the dynamic performance of a localisation system which presents the number of predicted room transitions as a ratio of the actual room predictions.

TDNN Time-Delay Neural Network. The application of Neural Networks to speech recognition tasks by creating a set of feature vectors from the original feature vector which is a time-shifted combination of the previous feature samples.

TDOA Time Difference of Arrival. A method of predicting transmitter-receiver separation using the difference in travel time of two signals with different propagation speeds, such as RF and ultrasonic signals.

TOA Time of Arrival. A method of predicting transmitter-receiver separation by observing the time of arrival of a wave, knowing the time the wave was transmitted.

TOF Time of Flight. See Time of Arrival.

T-R Transmitter-Receiver

UKF Unscented Kalman Filters. Kalman filters which use linear approximations of non-linear process and/or measurement models.

WLAN Wireless Local Area Network. The standard method of creating a short range wireless connection to a network, usually based on the IEEE 802.11 wireless communication protocol.

Publications Arising from this Thesis

- **Kelly, D.**, McLoone, S. and Farrell, R. (2010). Minimal Hardware Bluetooth Tracking for Long-Term At-Home Elder Supervision. In *Proceedings of the International Conference of the IEEE EMBC 2010*, in publication.
- **Kelly, D.**, McLoone, S., and Dishongh, T. (2009). Enabling affordable and efficiently deployed location based smart home systems. *Technology and Health Care special edition entitled Smart Environments: Technology to Support healthcare*, 17(3):221–235.
- **Kelly, D.**, Behan, R., Villing, R., and McLoone, S. (2009). Computationally tractable location estimation on WiFi enabled mobile phones. In *Proceedings of the IET Irish Signals and Systems Conference 2009 (ISSC09)*.
- **Kelly, D.**, McLoone, S., Logan, B., and Dishongh, T (2008). Single access point localisation for wearable wireless sensors. In *Proceedings International Conference of the IEEE EMBC 2008*, pages 4443–4446.
- **Kelly, D.**, McLoone, S., and Dishongh, T. (2008). A bluetooth-based minimum infrastructure home localisation system. In *In Proceedings of 5th IEEE International Symposium on Wireless Communication Systems (ISWCS 2008)*, pages 638–642.
- **Kelly, D.**, McLoone, S., and Dishongh, T. (2008). Experimental evaluation of a single access point bluetooth localisation system. In *Proceedings of the IET Irish Signals and Systems Conference 2008 (ISSC08)*, pages 254–259.
- **Kelly, D.**, McLoone, S., Dishongh, T., McGrath, M., and Behan, J. (2008). Single access point location tracking for in-home health monitoring. In *Proceedings of the 5th Workshop on Positioning, Navigation and Communication, 2008. WPNC '08.*, pages 23–29.

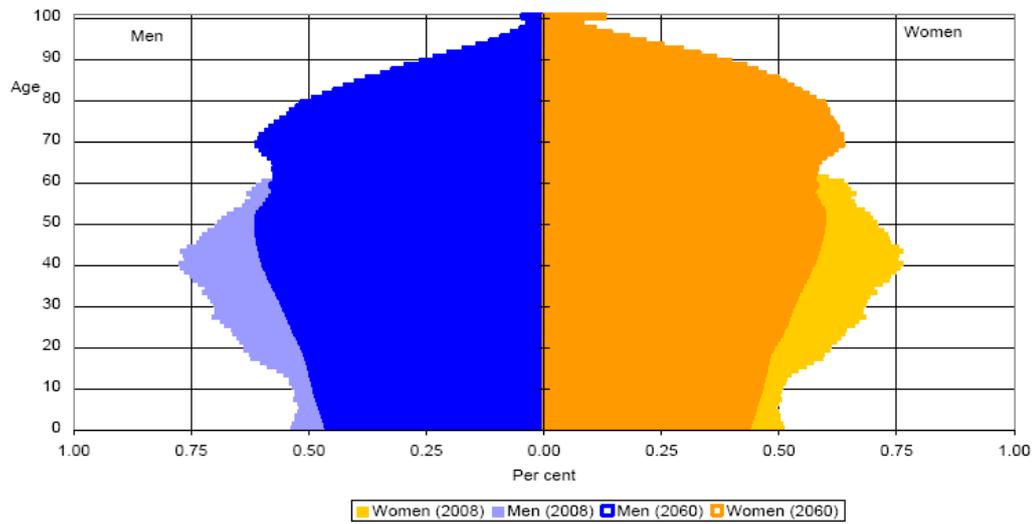
CHAPTER 1

Introduction

Much recent work highlights the growing issue of the aging population (Scanail et al., 2006, Department of Economic and Social Affairs, Population Division, 2009). Population aging refers to the rising median age of a population. Hence, along with the social and economic implications of a persistently growing global population, there are also challenges associated with the changing age distribution within the population. When considering the age pyramid of Figure 1.1, a shift in the age of the population of Europe over the next 50 years is evident (Giannakouris, 2008). The implications of this shift is that the median age of the European population will change from 40.4 years in 2008 to an estimated 47.9 years in 2060.

When viewing this effect on a global scale, Figure 1.2 illustrates how the proportion of the population over the age of 60 grew from 8% in 1950 to 11% in 2009. It also projects that the population over the age of 60 will to grow from 11% at present to 22% of the population by 2050 (Department of Economic and Social Affairs, Population Division, 2009). This will have a severe impact on the global Potential Support Ratio (PSR), which is the number of people in the age range 15-64 for every person aged 65 and older. Assuming that people in the age range 15-64 are capable of caring for an elderly person, it indicates the capacity for a population to care for the elderly. As a population ages, the PSR generally decreases (Department of Economic and Social Affairs, Population Division, 2009), leading to a lower ability for a population to care for its older members. Figure 1.3 indicates how the PSR changed from 1950 to 2009 and how it is expected to decrease by the year 2050. By the year 2050 the PSR is expected to reach a mere 4 potential carers for every person over the age of 65.

Population aging is a significant challenge facing the global population. To cope with



Source: Eurostat, EUROPOP2008 convergence scenario

Fig. 1.1: An age pyramid illustrating the change in population age distribution for males and females in 2008 and predicted for 2060, taken from Giannakouris (2008).

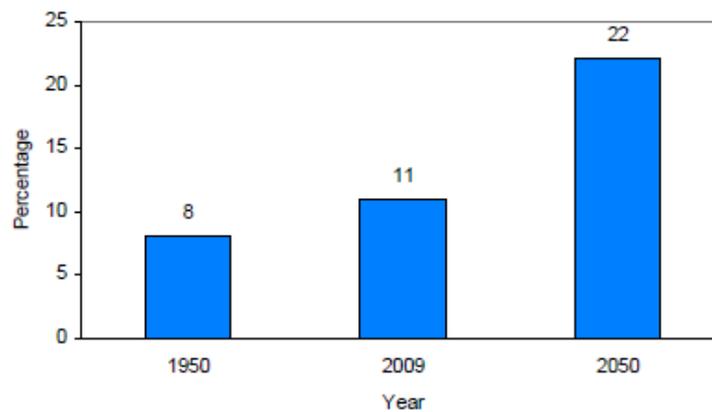


Fig. 1.2: Proportion of the population over the age of 60 in 1950, 2000 and projected into 2050, taken from Department of Economic and Social Affairs, Population Division (2009).

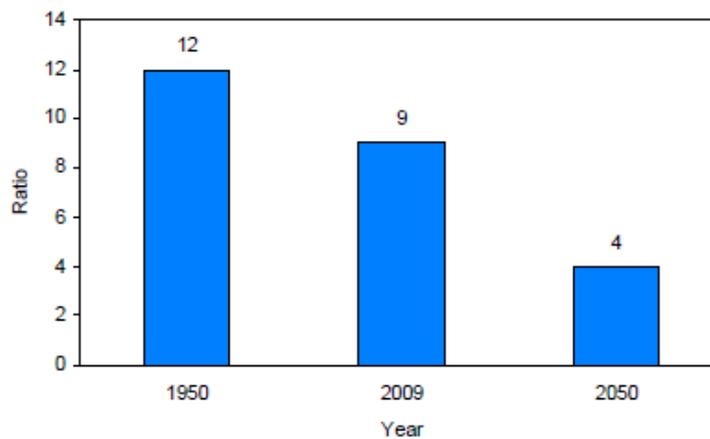


Fig. 1.3: Global Potential Support Ratio in 1950, 2000 and projected into 2050, taken from Department of Economic and Social Affairs, Population Division (2009).

the falling human resources and, as a result, growing financial shortcomings of all health care systems, a more efficient health care paradigm needs to be developed. Assistive technology has been proposed as an efficient technique for caring for the aging population without the costly need to commit elders to care facilities (Scanail et al., 2006). These technologies can enable elders to live safely in their own homes for a larger portion of their lives with monitoring and interaction facilities provided by the home environment itself. Daily activity and motion monitoring technologies are a fundamental part of these home care technologies. As such, this thesis is committed to developing a relatively inexpensive technique of unobtrusively monitoring an elder’s movements throughout their home environment. The availability of this movement information is envisioned to allow family members and/or part-time caregivers to assess and communicate with the elder on how their day-to-day behaviours are affecting their health and their ability to continue to benefit from independent living.

1.1 Smart Home Health Monitoring

Recent technological advances have spurred the emergence of semi-automated home environments, referred to as *smart homes*. The predominant benefit of smart homes is that they use technology to make the activities of everyday life more convenient for their inhabitants. Recently the importance of smart homes has been heightened by the fact that they can be used to actively provide health care services to an elder. The availability of health care services to an elder in their own home means that elders who would traditionally require attention from carers can have much of their supervision needs fulfilled by their smart home. This omnipresent monitoring facility is envisaged to allow elders to live in their own homes for longer periods of their lives before requiring a more

specialised care environment.

A wide variety of services can be provided to an elder by their smart home, including; monitoring of activity patterns (Tapia et al., 2004, QuietCare Systems, 2009), provision of activities to keep the elder proactive (Nawyn et al., 2006), detection of safety critical conditions like falls (Chen et al., 2005) and medication adherence promotion (Nugent et al., 2005, Lundell et al., 2006). As with most smart home functionalities, these technologies require some technique to detect the current context or activities of the user. To infer the context of the user a number of sensors are typically employed. These sensors can be anything from simple contact switches on furniture to RFID proximity sensors.

1.2 Home Localisation Techniques

A variety of sensing modalities are relevant to a home care smart home system. This thesis focuses on the location sensing component of smart home systems, since the context related to a particular detected activity can be influenced strongly by the location of the activity (Ofstad et al., 2008). To date a variety of elder care smart home systems have been proposed, many of which utilise location as the main form of context. One of the most obvious uses which can be made of location information is to allow monitoring of an elder's activity patterns over long periods of time. One commercially available elder monitoring system is QuietCare Systems (2009). The QuietCare system uses Passive Infra-Red (PIR) motion detectors in each room to infer the current location of the elder. Then deviations of the elder's movement and activity patterns from typical healthy patterns can be detected and a caregiver can be informed.

A context sensitive medication prompting system is presented by Lundell et al. (2006) that infers the subject's context from their room-level location, also based on PIR sensors. Based on the subject's location, different prompting devices throughout the environment are used to remind the subject to take medication. A portable wristwatch-like prompting device is used when the elder is in a location where no other form of prompting device is available. Furthermore medication prompts are sent only at times when the elder's motion patterns indicate that they would otherwise miss a dose. For example if the elder exhibits a motion pattern which, based on baseline data, indicates they may leave the house at a time close to their usual medication time, they are prompted to take their medication before they leave the house. This reduces occurrences of missed doses.

Another piece of work, which uses PIR sensors to infer location, but with resolution finer than room level is presented in Chan et al. (2002). It is achieved by placing several PIR sensors in each room, one sensor for each location of relevance within the room. That particular work is focused on assessing the subject's levels of mobility, which is indicative of motor behavioural disorders. Statistics of a patient's motion patterns over a typical 24

hours are visible from the trial data. However, it is indicated that PIR based location predictions are not reliable when a caregiver enters the environment due to PIR sensor's inability to differentiate between different people.

A system which uses an ultrasound location tracking technique is presented in Helal et al. (2003). With their high-accuracy tracking technique, the authors developed a remote monitoring application, similar to that provided by the QuietCare system. This application provides location markers on an environment map, in real-time, to interested parties with the necessary software. The authors also present an attention capture application which provides interactive displays to an elder to gauge their reactivity to certain types of prompts. The location information is integrated into the decision of which environmental display to use to engage the inhabitant, which is similar to the approach taken by Lundell et al. (2006). Finally the authors propose an indoor navigation system to assist visually impaired subjects. Navigation requires a high accuracy tracking technique such as ultrasound to allow useful directions. As such, the tracking system which we will later present is not suitable for precise indoor navigation.

One further use of indoor location is outlined by Chen et al. (2005). This paper describes a sensor for detecting falls of an elder. When a fall is detected it is necessary to be able to pinpoint the location in which the fall occurred to allow emergency personnel to quickly locate and assist the individual. Not much information is given about the localisation technique, except that it uses estimates of the portable fall sensor's transmission radio signal strength at several Basestation computers to triangulate the subject's location.

As this section highlights, there are a number of smart home systems which utilise location information. Localisation techniques of varying resolution are employed in different situations. However in a home environment room-level location is typically sufficient, which explains the ubiquity of PIR localisation techniques in many implementations to date. Many of these location-based smart home systems require an array of sensors to be installed throughout the home environment, typically at a level of one sensor per room. Hence, the location sensing element of these systems have high installation overhead, a trait which overshadows the obvious benefit of such systems.

The work presented in this thesis aims to develop and implement a location sensing technique with minimal hardware requirements to reduce the installation overhead of location-based smart home systems. To date, a vast body of research has been conducted on efficient localisation in office environments. Such techniques, however, have rarely been applied to home environments, which have significantly different topologies than office spaces. Hence this thesis conducts a summary of general localisation theory before presenting a comprehensive review of the indoor localisation systems developed to date. A major limitation of referring to prior localisation research for the development of this

home localisation technique is that localisation systems generally produce predictions with resolution inappropriate for a home environment. As will be presented later, room-level or symbolic location predictions are more relevant to human context sensing than exact spatial coordinates, as used in most prior indoor localisation techniques.

1.3 Aims and Scope of Thesis

From the discussion of the previous sections it can be understood that the location sensing elements of smart homes have not been rigorously developed in the past. The simplest approach, such as PIR sensors, or the most expensive approach, such as ultrasound, has generally been adopted. This thesis contributes to both the fields of localisation and home monitoring technologies by building a localisation system specifically for the task of room detection in a home environment with minimal hardware requirements, rather than adopting existing generic home localisation technologies for this task.

For this reason it is necessary to review all existing literature in relation to technologies suited to indoor localisation. Then the optimal technology on which to build the localisation system must be selected. However, the simple deployment and experimentation of a localisation platform is not likely to produce results entirely representative of a realistic home deployment. Hence, it is necessary to explore the performance of the localisation system over significant periods of time corresponding to that which would be encountered in a home deployment.

Assuming the availability of a reliable long-term home localisation evaluation platform, it is possible to explore further augmentations to the base localisation deployment. The most significant augmentation which can be applied is the inclusion of Bayesian filtering, which has frequently been shown to increase coordinate location predictions (Kotanen et al., 2003, Ladd et al., 2002, Rodas et al., 2008). Along with Bayesian Filtering, further improvements based on observations of the available signals and constraints within the home environment, can be considered.

The aim of thesis is to develop a room-level home localisation system. The system must be cheap in terms of hardware costs, power costs and maintenance costs for the elders, who generally do not have extensive electronic engineering training. To minimise hardware costs the system must use intelligent signal processing techniques, using all of the signals available from the equipment present in the test environment. Power costs can be minimised by utilising a hardware platform developed for low power transmissions. Using low power hardware introduces deficiencies in the quality of the signals available from the hardware. This places the onus on the processing algorithms to robustly handle the signals to enable accurate location predictions while minimising battery drain. Finally the overall system should minimise the maintenance cost to the elder by using only devices from

which they can obtain alternative functionality. For example, it is beneficial to include the localisation functionality into a mobile phone rather than a custom wrist-mounted tag. This both allows alternative functionality to be derived from the device, increasing the perceived usefulness, and reduces the assistive stigma.

The scope of the thesis extends as far as predicting location of an elder within their home during the day. This is due to the fact that detecting an elder's location outside of the home has a well defined solution in GPS and the movements which occur throughout the night are relatively low-magnitude and are better addressed with alternative technologies (Behan et al., 2008).

1.4 Contributions of this Thesis

The general focus throughout this thesis is on the development of a localisation system capable of determining the room-level position of an elder inhabitant. A general localisation platform is developed with the emphasis on cheap deployment and reliable long-term performance. It is concluded that predicting location from Bluetooth[®] signals arriving at a Basestation computer within the environment from a user's mobile phone is the most efficient and long-term reliable solution. In this context, the core novel contributions of the research presented in this thesis are as follows:

1. The deficiencies in the accuracy metrics of previous indoor localisation work is highlighted. The majority of previous indoor localisation work focuses on error distances, i.e. the discrepancy between the true position and the predicted position, in metres. This work describes techniques of assessing the ability of a system to resolve the more human-relevant symbolic location. Not only is the ability of the system to correctly predict symbolic location over long periods of time assessed, but the effect this has on activity indicators such as distance travelled and room transition times are also presented.
2. Erroneous location predictions are reduced by incorporating user movement tendencies by modelling the user's movements using a Hidden Markov Model. This also constrains the predictions to movements between rooms which are topologically possible since the user tends not to quickly transition between rooms which are not connected by doors. It is shown that all of the localisation performance measures developed in this thesis are improved for the Bluetooth[®] localisation system by utilizing a Hidden Markov Model framework. The Hidden Markov Models are also extended to second and third order models to investigate if making reference to predictions further into the past improve current predictions.
3. Typically the Viterbi HMM-decoding algorithm involves a backtracking step which

requires starting at the final state prediction and iteratively backtracking over the most likely previous states. This work proposes a short-term backtracking algorithm which has not been considered for localisation in the past. This backtracking allows globally optimal location predictions to occur in almost real-time with relatively small worst-case prediction delays.

4. To enable HMM state predictions based on motion types, it was necessary to develop a novel technique for motion prediction. This technique uses accelerometer data to predict the type of motion which the user is undertaking. While detecting motion type from accelerometer signals is not novel in itself, the application of this accelerometer frequency component technique has not been applied to realistic situations where the motion type can change arbitrarily. Furthermore, Bayesian filtering has been shown to improve accelerometer frequency component motion detection.
5. The availability of motion predictions allows the dynamic adjustment of HMM parameters, namely the transition probability matrices. This, in theory, should allow different room transition probabilities for different situations. For example, when a user is stationary, the probability of transitioning to other rooms should be zero, leading to higher localisation accuracies. In practice, however, it is found that motion predictions do not increase accuracy, since the training motion data is subject to misclassifications due to the inability to generate 100% reliable motion labels.

Besides these contributions, other less significant contributions of this thesis include:

1. A comprehensive literature review of the technologies previously utilised for indoor localisation is presented. Furthermore, a review of the classification algorithms suited to the task of symbolic location recognition is presented and their decision boundary composition is illustrated on a synthetic Gaussian mixture dataset.
2. A small home localisation scenario is considered to allow the determination of the most suitable wireless technology for our home localisation system. For power-efficiency, Bluetooth[®] and a communication protocol which forms part of the ZigBee[®] standard are considered. The optimal method of obtaining training data corresponding to human movement for such hardware is empirically confirmed. The traditional remote localisation scenario of multiple Basestation computers and our minimal approach to localisation are compared. It is shown that only slight reductions in localisation accuracy is possible while halving the quantity of deployed hardware infrastructure.

3. The novel localisation system developed in this thesis is compared to the commonly employed PIR home localisation technique. The comparison is enabled by deploying both a PIR localisation system and a novel technique of obtaining room labels which utilises an RFID reader and RFID tags on doors between rooms. It has been shown that in a realistic test environment the Bluetooth[®] platform has higher accuracy and when multiple occupancy occurs in the environment the Bluetooth[®] platform has a lower reduction in localisation accuracy.
4. Experiments investigating the benefits of increasing signal diversity on the localisation accuracy have been conducted. Higher signal diversity is achieved by using a second Bluetooth[®] beacon cohabited with the Basestation Computer, but with a different radiation profile or by placing alternative “dumb” Bluetooth[®] transceivers throughout the environment. It is shown that the availability of cheap Bluetooth[®] transceivers throughout the environment significantly contributes to higher localisation accuracy; an approach previously impossible due to the inability to retrieve Bluetooth[®] signal readings on a cheap mobile device.
5. Improvements to the deployed base localisation algorithms are presented. The additions which are considered are; signal smoothing, lagged preprocessing, sparse classifiers, integration of user location preferences and uncertainty rejection. User location preference integration and prediction uncertainty rejection has been impossible for a majority of localisation techniques in the past due to their maximum likelihood treatment of the coordinate localisation problem. Since we take a discrete probabilistic approach to localisation, integration of such relevant information is an incremental addition.

1.5 Thesis Organisation

This thesis is organised into 8 chapters as follows:

Chapter 2 provides a summary of localisation theory and a comprehensive summary of indoor localisation systems, grouped by the technology on which the systems are built on.

Chapter 3 is a second background chapter which describes the variety of direct and probabilistic classification techniques which are considered for applicability to the problem of estimating home location from RF signal input features.

Chapter 4 presents the candidate hardware platforms suitable for low-power localisation. It compares Bluetooth[®] and the ZigBee[®] physical layer hardware for outdoors and indoors signal resolution and the localisation accuracy possible with each platform.

Bluetooth[®] is formalised as the base localisation hardware platform on which to build the localisation algorithms in the following chapters.

Chapter 5 improves upon prior work by illustrating the deficiencies of traditional localisation accuracy measures. It continues by comparing this thesis' localisation system with the PIR home localisation technique by employing a novel room-label acquisition technique. It then demonstrates the effect of increasing diversity in the available signals on localisation accuracy. Finally, further augmentations to the original classification algorithms are evaluated.

Chapter 6 generates further accuracy metrics which quantify the effects of location prediction errors on estimation of the amount of motion a user exhibits and the delay in room transition predictions. It then shows how Hidden Markov Models and the Viterbi algorithm improves both the location prediction accuracy and the transient effects, such as location jitter and prediction delay. Improvements to the Viterbi algorithm are proposed such as using higher order transition models and short-time decoding of the state sequence to permit almost real-time location predictions.

Chapter 7 presents the generation of motion-type predictions from the accelerometer signals available from the mobile localisation device. These motion predictions are then utilised by the HMM to inform which transition probabilities are more appropriate for a given motion type. The effect of this modification to the HMM framework is illustrated.

Chapter 8 concludes with a summary of the work completed as part of this thesis, the contributions it made to the field and the relevant areas of work which remain to be investigated.

In the past decade, the possibility for mobile communication devices to estimate their location indoors by analysing their received Radio Frequency (RF) signals has been explored. Despite the somewhat unpredictable nature of RF propagation in an indoor environment it is still one of the most hardware-efficient methods of estimating indoor location. Location can be modelled as a function of certain traits of a received RF signal such as received intensity, time of flight and received angle. Some of these traits are easier than others to deduce, but as a result, are less dependable due to their affordable availability. For example signal intensity is a reading widely available as standard on RF receivers, but it sometimes has poor correlation with position due the low spatial resolution of the signal.

An important predictor of the success of data transmission between a transmitter and a receiver is the signal strength at the receiver. If the signal strength, hence the signal to noise ratio, of the signal at the receiver is high, then it is more likely that the information will be successfully received. Conversely, a low signal strength will mean it is less likely for the signal to be successfully received. Over the many decades since radio transmission was invented, many signal propagation models have been developed to allow the prediction of the radio signal strength for a receiver at a particular point within an environment. Since the received signal strength varies a function of the position of the receiver relative to the transmitter, it is a commonly used parameter in the calculation of the position of the receiver. Hence radio propagation models are important for many localisation systems, and as such must be understood before the localisation methods can be presented.

This chapter begins by presenting the large-scale propagation mechanisms which influence RF transmission and how distance may be modelled from the resultant received

signal intensity. Section 2.2 follows on by presenting the techniques which utilise these models to predict the position of a device based on RF signal properties. Section 2.3 then summarises the RF localisation systems developed to date and which technique each system employs.

2.1 Radio Wave Propagation

The behaviour of radio waves in a given environment strongly influences the accuracy of RF-based localisation systems. For example, the well-known Global Positioning System (GPS) is highly dependent on the interaction of the radio signals with obstacles. The presence of buildings in the Line of Sight (LoS) of the Mobile Device (MD) and the orbiting satellites can cause highly inaccurate predictions of the MD position. That is, of course, if GPS is even detectable in such conditions, which is not always the case. It is vitally important for the development and deployment of a localisation system to be able to understand and predict the behaviour of a radio signal in a specific environment.

There are two types of variations which can occur in radio waves. They are large-scale path-losses and small-scale fading. Large scale path losses refers to signal strength variations which occur over large distances (hundreds of meters). In contrast, small-scale fading refers to the variations which occur with small changes in time and distance, even of the order of wavelengths. Figure 2.1 illustrates the progression of the received signal intensity due to large-scale and small-scale effects occur over increasing transmitter-receiver separation. The large-scale signal is a low-pass-filtered version of the raw signal, which incorporates both large-scale and small-scale effects. Hence, the rapid fluctuations in the raw signal are due to small-scale effects and the gradually changing trend of the raw signal is due to large-scale path loss (or slow fading).

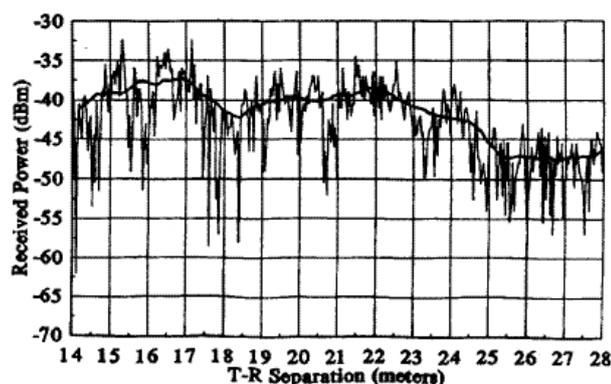


Fig. 2.1: Large Scale and Small Scale Fading with increasing Transmitter-Receiver separation, taken from Rappaport (2002).

This section will form a basis for understanding how the behaviour of a radio signal can

be predicted and subsequently utilised in the localisation techniques presented in Section 2.2. Section 2.1.1 illustrates the basic laws of radio-wave propagation which contribute to their complicated indoor behaviour. Section 2.1.2 builds on these mechanisms to develop propagation models, which can be used to predict the signal intensity of the received signal at a given position.

2.1.1 Propagation Mechanisms

There are 3 propagation mechanisms which can influence wireless signal transmission. They are Reflection, Diffraction and Scattering and will be explained in sections 2.1.1.2, 2.1.1.3 and 2.1.1.4 respectively. But before these propagation impediments are explained, the simple case when no obstructions are present must be considered.

2.1.1.1 Free Space Propagation

Free space propagation refers to the transmission of radio waves when the area around, and in the LoS of the transmitter and receiver, is free of obstructions. In this case the waves will be able to travel from the transmitter directly to the receiver with the only factor effecting the signal strength being the degradation due to distance traversed.

A free space propagation model describes how the received power decays as a function of increasing Transmitter-Receiver (T-R) separation. The *Friis free space equation* is the expression used to predict the received power $P_r(d)$ for a T-R separation distance d , given that the transmitted power is P_t ,

$$P_r(d) = \frac{P_t G_t G_r}{L} \left(\frac{\lambda}{4\pi d} \right)^2, \quad (2.1)$$

where G_t and G_r are the transmitter gain and receiver gain respectively and λ is the wavelength of the radiowave in meters. L is the system loss factor ($L \geq 1$), which applies to losses not due to propagation behaviour, such as internal hardware losses for example. $L = 1$ when it is assumed that there are no system losses, hence shall be ignored.

The antenna gains G_r and G_t are related to the effective aperture A_e . The effective aperture is derived from the apparent physical area to the front of the antenna from which it receives the energy from arriving electromagnetic waves and is expressed in units of m^2 . The antenna gain is related to the effective aperture as follows,

$$G = \frac{4\pi A_e}{\lambda^2}. \quad (2.2)$$

λ is related to the carrier frequency by;

$$\lambda = \frac{c}{f} = \frac{2\pi c}{\omega_c}, \quad (2.3)$$

where f is the carrier frequency in Hz, ω_c is the carrier frequency in radians per second and c is the speed of light [m/sec]. From equations (2.2) and (2.3) it is understandable why the lengths of antennas are different for communication systems of different frequencies; the dimensions of the antenna are chosen to maximise the gain of a received signal at a given frequency.

Another influence the antenna configuration has on the transmission power is the polarisation of the antenna. Polarisation refers to the orientation of the oscillations of the EM waves relative to their direction of travel. There are two types of polarisation; horizontal and vertical. Horizontal polarisation refers to when the EM field oscillates in the horizontal plane and vertical polarisation refers to EM field oscillations which occur in the vertical plane. Over long distances of transmission, atmospheric and environmental effects can cause distortions leading to a mixture of horizontally and vertically polarised waves arriving at a receiver.

In the ideal case of an antenna with perfectly horizontal polarisation, the maximum efficiency transmission, hence maximum received signal intensity, occurs when the receiving antenna is also horizontally polarised. Hence, if either of the antennas are rotated by $\phi = 90^\circ$ relative to each other there is an antenna polarisation mismatch; one is horizontally and the other is vertically polarised. In this theoretical scenario with perfect polarisation the transfer of power will be zero. The expression for the dependence of the amount of energy lost during transmission on the relative antenna orientation is described by the Polarisation Loss Factor (PLF),

$$PLF = \cos^2(\phi). \quad (2.4)$$

Hence, the orientation of an antenna influences received signal intensity in two ways. Firstly, changing the orientation can influence the effective aperture of the antenna, resulting in a different antenna gain. Secondly, changing the orientation of the antenna can change the polarisation of the antenna. It has been demonstrated how changing the relative polarisation angle of the transmitting and receiving antennas influences the PLF, hence influences the received signal intensity. The application of this phenomenon to this work will be demonstrated in Chapter 5.

Following on from the Friis free space equation (2.1), which predicts the RF power received at the end of the transmission path, we can define the Path Loss (PL), as the RF power lost over the transmission path, assuming PLF is negligible;

$$PL = \frac{P_t}{P_r} = \frac{L}{G_t G_r} \left(\frac{4\pi d}{\lambda} \right)^2. \quad (2.5)$$

When the antenna gains, G_t and G_r , and hardware losses, L , are assumed to be negligible and set equal to 1, path loss can be represented, in decibels, as;

$$PL(dB) = 10 \log \left[\frac{d^2 (4\pi)^2}{\lambda^2} \right], \quad (2.6a)$$

$$\text{where } P_r(d)[dBm] = P_t[dBm] - PL(d)[dB]. \quad (2.6b)$$

However, this can only be considered a valid predictor of path loss for values of d which fall within the far-field (or Fraunhofer region) of the transmitter. The Fraunhofer region of a transmitting antenna is the region beyond the Fraunhofer distance d_f . The Fraunhofer distance is defined to be;

$$d_f = \frac{2D^2}{\lambda} \quad (2.7a)$$

where D is the largest physical dimension of the antenna. There are two further conditions which the Fraunhofer distance, d_f , must satisfy. They are;

$$d_f \gg D \quad (2.7b)$$

and

$$d_f \gg \lambda. \quad (2.7c)$$

If we consider the path loss for a transmission system like Bluetooth[®] or Wireless LAN ($f \approx 2.4GHz$) we get the path loss profile illustrated in figure 2.2.

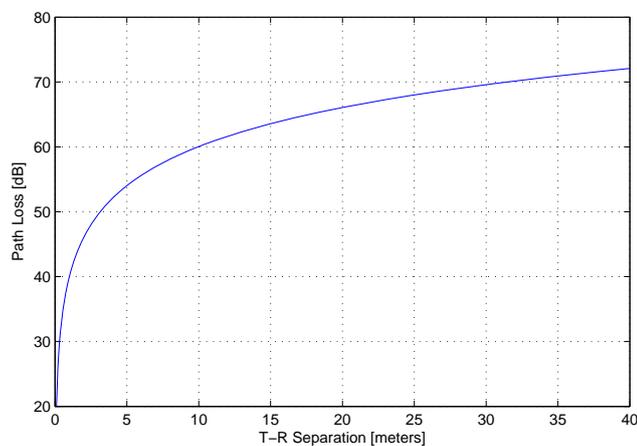


Fig. 2.2: Path Loss with increasing transmitter-receiver separation distance for $f \approx 2.4GHz$.

It is clear from Equation (2.1) that the Friis equation does not hold for $d = 0$. At $d = 0$ the received power will be ∞ . Hence, it is useful to define a reference distance d_0 , known as the received power reference distance. Now the power at the point d , $P_r(d)$, can be related to the power at the reference point d_0 , $P_r(d_0)$. The value of $P_r(d_0)$ can easily be

predicted from Equation (2.1) or it can be determined empirically from the environment by averaging the power readings taken at several different points at the distance d_0 radially from the transmitter. This makes using a reference distance even more useful since it eliminates the need for knowledge of the transmission wavelength, like in Equation (2.6a). The reference distance must be chosen so that it lies within the far-field region, $d_0 \geq d_f$. Also d_0 must be smaller than any practical distance used in the communication channel. So d_0 must satisfy the equation;

$$d \geq d_0 \geq d_f. \quad (2.8)$$

Since we ensure d is always greater than the reference distance d_0 , this equation will always hold. Also since the power decreases as the square of the distance increases (see Equation (2.1)), we can relate the power at the distance d to the power at the reference d_0 thusly;

$$P_r(d)d^2 = P_r(d_0)d_0^2. \quad (2.9)$$

This can be rearranged to give an expression for the power at a given distance, $P_r(d)$;

$$P_r(d) = P_r(d_0) \left(\frac{d_0}{d} \right)^2. \quad (2.10)$$

In a typical mobile radio communications channel, P_r can change by many orders of magnitude over the range of coverage. For this reason it is often more convenient to represent the received power in units of dBm or dBW. Converting Equation (2.10) to units of dBm (decibels with a 1 milliwatt reference level) gives;

$$P_r(d)[dBm] = 10 \log \left(\frac{P_r(d_0)}{0.001} \right) + 20 \log \left(\frac{d_0}{d} \right) \quad d \geq d_0 \geq d_f \quad (2.11)$$

It is generally accepted (Rappaport, 2002) that for a 1-2GHz mobile transmission system, d_0 is chosen to be 1m indoors and 100m or 1km outdoors. The reason for choosing values which are multiples of 10 is to make the path loss computations simple in dB units.

Now that the expected behaviour of radio waves in an un-obstructed environment have been presented, an understanding of the complicated behaviour when obstructions are present can be developed.

2.1.1.2 Reflection

Reflection is a propagation mechanism which occurs when an electromagnetic wave propagating through one transmission medium impinges upon another. Provided the interface between the mediums is flat and large compared to the electromagnetic wavelength, reflection can occur. Otherwise scattering occurs (see Section 2.1.1.4).

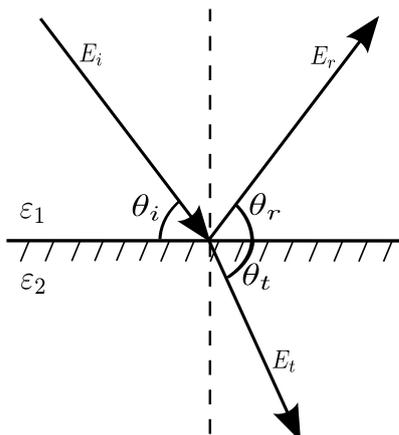


Fig. 2.3: Angles of Incidence, Reflection and Transmission, modified from Rappaport (2002).

When reflection occurs some of the incident energy is reflected back into the original medium and some is transmitted into the second medium. If the second medium is a perfect dielectric, some of the energy is reflected and some is transmitted with no energy loss occurring. Alternatively, if the second medium is a perfect conductor, all of the energy is reflected. Again no energy loss occurs.

Figure 2.3 illustrates how the signal behaves in what is known as the *plane of incidence*. The plane of incidence is a flat plane in a 3-dimensional space in which the lines of incidence, reflection and transmission all exist. The line of incidence represents the direction from which the electromagnetic waves arrive at the interface. θ_i is the angle which the incident wave makes with the interface between two mediums. Similarly, the angle of reflection (θ_r) shows the angle of the reflected energy relative to the interface, after the wave is reflected by the interface. And finally θ_t is the angle which the transmitted energy travels relative to the interface after passing through the interface. It should be clear from the diagram that $\theta_i = \theta_r$.

The electric field intensity of the reflected and transmitted waves are related to the incident waves by the *Fresnel reflection coefficient* (Γ). The electric field intensity of the reflected waves (E_r) can be determined from the field intensity of the incident waves (E_i) by using the Fresnel reflection coefficient,

$$E_r = \Gamma E_i. \quad (2.12)$$

However, the Fresnel reflection coefficient will be different depending on the polarisation of the wave. The Fresnel reflection coefficients for vertically (Γ_v) and horizontally polarised (Γ_h) waves are

$$\Gamma_v = \frac{E_r}{E_i} = \frac{\eta_2 \sin \theta_t - \eta_1 \sin \theta_i}{\eta_2 \sin \theta_t + \eta_1 \sin \theta_i} \quad (2.13)$$

$$\Gamma_h = \frac{E_r}{E_i} = \frac{\eta_2 \sin \theta_i - \eta_1 \sin \theta_t}{\eta_2 \sin \theta_i + \eta_1 \sin \theta_t} \quad (2.14)$$

where η_i is referred to as the intrinsic impedance of the i th medium. Since waves can contain both vertically and horizontally polarised components, the overall reflected energy is calculated using superposition.

It is also possible to calculate the angle of incidence which would result in no energy being reflected. *The Brewster angle* (θ_B) is the incident angle which will result in a Fresnel reflection coefficient of 0, hence no reflected energy. The Brewster angle is calculated to be;

$$\sin(\theta_B) = \sqrt{\frac{\epsilon_1}{\epsilon_1 + \epsilon_2}}. \quad (2.15)$$

ϵ_n is the permittivity of the n^{th} medium relative to the permittivity of free space (ϵ_0);

$$\epsilon_n = \frac{\epsilon_m}{\epsilon_0}, \quad (2.16)$$

where ϵ_m is the actual permittivity of the medium $m = 1, 2$. The relationship between the relative permittivity and the intrinsic impedance of a material is $\epsilon_i = \mu_i / \eta_i^2$, where μ_i is the permeability of the material. Hence, with Equations (2.15) and (2.16) the angle of incidence which results in no reflected energy can be calculated.¹

A useful application of the Fresnel reflection coefficient is that a transmission model which incorporates ground reflections can be developed, called the *Two-Ray Model*. This model can be used to predict the Electromagnetic (EM) signal intensity at the receiver for an outdoor transmission system over several kilometers. This model takes into account, not just the direct Line of Sight path to the receiver, but also a ground reflected path. If we consider a transmitter situated h_t meters above the ground, and a receiver d meters away, h_r meters above the ground (see Figure 2.4), an expression for the ground reflection model can be developed. From the principle of superposition the energy arriving at the receiver is calculated to be $|E_{TOT}| = |E_{LOS} + E_r|$.

The two transmission paths in Figure 2.4 are referred to as d_d for the direct T-R path and d_r for the total reflected T-R path. Hence the two energy components arriving at the receiver are

¹Note equations (2.15) and (2.16) only apply for vertically polarized waves

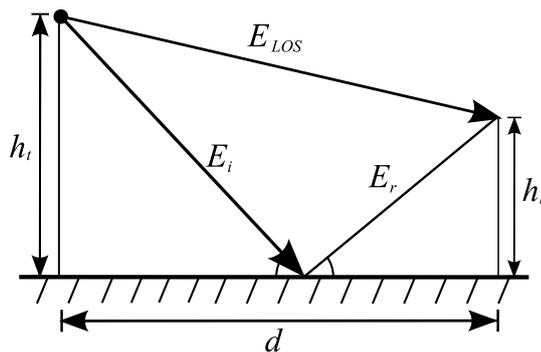


Fig. 2.4: Ground reflection model, reproduced from Rappaport (2002).

$$E_{LOS} = \frac{E_0 d_0}{d_d} \cos\left(w_c\left(t - \frac{d_d}{c}\right)\right) \quad (2.17)$$

$$E_r = \Gamma \frac{E_0 d_0}{d_r} \cos\left(w_c\left(t - \frac{d_r}{c}\right)\right) \quad (2.18)$$

where E_0 is the free space electric field at a reference distance d_0 . Assuming a small angle of incidence, due to large T-R separation distance, Rappaport (2002) shows that the reflected wave is 180° out of phase with the incident wave. Hence, assuming perfect horizontal polarisation, the Fresnel reflection coefficient is $\Gamma = -1$ and the total electric field incident on the receiver can be expressed as the sum of Equations 2.17 and 2.18,

$$E_{TOT} = \frac{E_0 d_0}{d_d} \cos\left(w_c\left(t - \frac{d_d}{c}\right)\right) + (-1) \frac{E_0 d_0}{d_r} \cos\left(w_c\left(t - \frac{d_r}{c}\right)\right). \quad (2.19)$$

Rappaport (2002) explains how the path difference is calculated from

$$\Delta = d_r - d_d \approx \frac{2h_t h_r}{d} \quad (2.20)$$

when d is very large compared with $h_t + h_r$. With some simplifying assumptions the power incident on the receiver can be calculated to be

$$P_r = P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}. \quad (2.21)$$

It can be seen from Equation (2.21) that at large distances ($d \gg \sqrt{h_t h_r}$) the power falls off with distance raised to the fourth power. This fall off in power is much more rapid than that experienced in free space. The path loss for the Two-Ray model is then defined as;

$$PL(dB) = 40 \log d - (10 \log G_t + 10 \log G_r + 20 \log h_t + 20 \log h_r) \quad (2.22)$$

Note this equation applies for large propagation distances. It starts to break down over small distances or in a cluttered environment like that which would be found indoors. The two-ray model is one example of a simple ray tracing model used to estimate the signal strength throughout the environment. Of course other, more complicated ray tracing models can be employed in real environments, such as the one presented in McKown and Hamilton (1991).

2.1.1.3 Diffraction

Diffraction allows radio waves to propagate behind obstructions, like walls, hills or even around the earth. The region behind an obstruction is often referred to as the shadowed region, and exhibits severely degraded signal intensity. The received EM field intensity decreases rapidly as the receiver moves deeper into the shadowed region. However there is often enough signal strength to retain a useful signal at the receiver.

Huygens' principle attempts to explain why this occurs. It states that every point on a wavefront can be considered as a point source for the production of wavelets. These wavelets can combine to produce a new wavefront. The diffraction phenomenon is a result of these new wavefronts propagating into the shadowed region behind the obstruction. To understand how this occurs one must consider a transmitter and a receiver separated by a transparent plane as illustrated by Figure 2.5. The concentric circles illustrate where Fresnel zones cross the plane. Fresnel zones are the paths the radio waves travel which result in constructive and destructive interference alternately. So the path difference between the direct LoS path and the diffracted path is known as the *excess path length*, denoted

$$\Delta = \frac{n\lambda}{2}, \quad (2.23)$$

where n is the Fresnel zone number.

In Figure 2.5 the first circle on the plane represents where the first fresnel zone crosses the plane. Since this is the first fresnel zone, the path difference will be $\lambda/2$, so it will cause destructive interference. Similarly the second Fresnel zone exhibits constructive interference, the third, destructive interference and so forth. The lower order Fresnel zones have the most significant influence on the net received signal strength. This means that the higher order zones have less effect on the received signal strength and can be ignored.

The radius of the loci which cross the plane can be calculated using the expression:

$$r_n = \sqrt{\frac{n\lambda d_1 d_2}{d_1 + d_2}}, \quad (2.24)$$

where r_n is the radius of the n^{th} Fresnel zone, λ is the wavelength of the transmitted

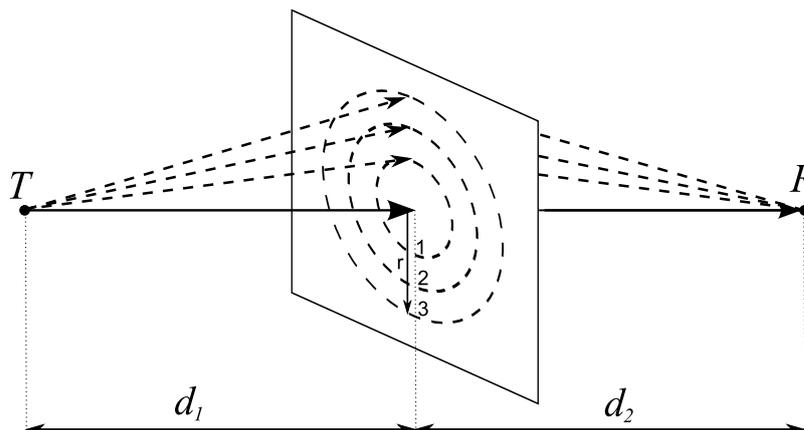


Fig. 2.5: Fresnel zones as they interact with a transparent plane between the transmitter and receiver, reproduced from Rappaport (2002).

wave, d_1 the distance between the plane and transmitter and d_2 is the distance between the plane and the receiver. Note that the radius of a locus is dependent on the distances d_1 and d_2 , and it will be at a maximum if the plane is located in the centre of the transmitter and receiver. The radius will be smaller if the plane moves away from the centre. Hence, equation (2.24) shows that the level of shadowing is dependent not only on the frequency of the transmitted wave, but also on the relative position of the obstruction.

The *knife-edge diffraction model* is used to estimate the signal attenuation due to obstructions in the direct line of sight of the transmitter and receiver. Estimating the total effect of diffraction over some terrain is a complicated and computationally intensive process, so it is generally simplified to a knife-edge obstruction between the transmitter and receiver. Figure 2.6 shows a typical knife-edge diffraction geometry.

The excess path length (Δ), which is the difference in travel distance between the direct path and the diffracted path is calculated from the geometrically derived expression;

$$\Delta \approx \frac{h^2}{2} \left(\frac{d_1 + d_2}{d_1 d_2} \right) \quad (2.25)$$

and the corresponding phase difference at the receiver is calculated to be;

$$\phi = \frac{2\pi\Delta}{\lambda} \approx \frac{2\pi}{\lambda} \frac{h^2}{2} \left(\frac{d_1 + d_2}{d_1 d_2} \right). \quad (2.26)$$

The angle between the transmitted ray and the diffracted ray (α) can be approximated to be:

$$\alpha \approx h \left(\frac{d_1 + d_2}{d_1 d_2} \right). \quad (2.27)$$

Equation (2.26) is generally normalised using the *Fresnel-Kirchoff* diffraction parameter

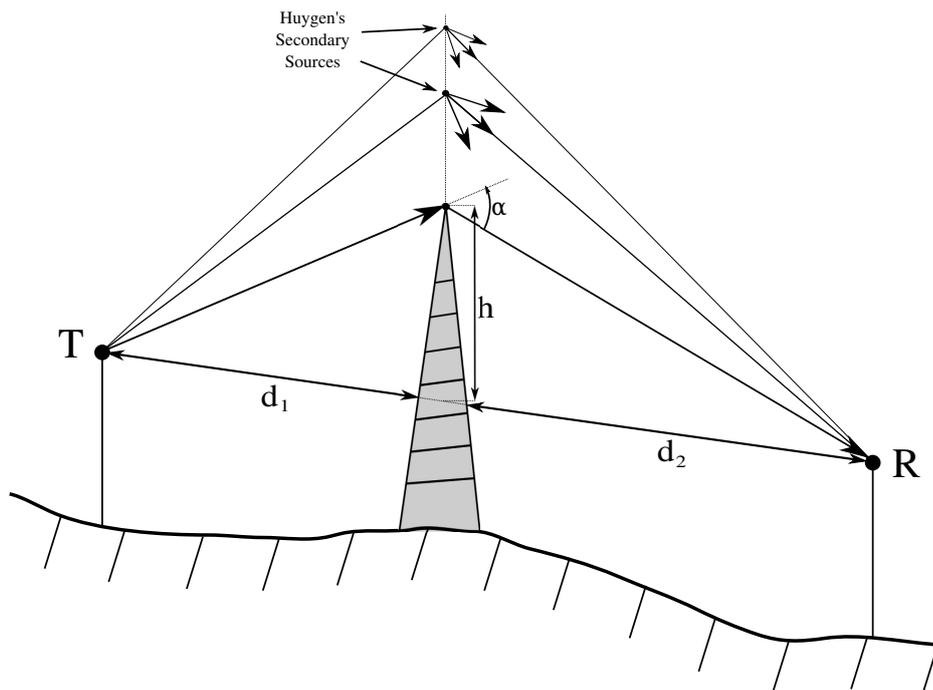


Fig. 2.6: Knife-edge diffraction, modified from Rappaport (2002).

v which is a dimensionless quantity, defined as:

$$v = h \sqrt{\frac{2(d_1 + d_2)}{\lambda d_1 d_2}} = \alpha \sqrt{\frac{2d_1 d_2}{\lambda(d_1 + d_2)}}, \quad (2.28)$$

leading to the more convenient phase difference expression of

$$\phi = \frac{\pi}{2} v^2. \quad (2.29)$$

It is clear that the received signal intensity will be a vector sum of all the waves arriving at R. The presence of a knife-edge obstruction means that the received signal intensity will be degraded as a function of the dimensions of the obstruction relative to the transmission LoS, like its height and its distance from the transmitter and receiver. The Fresnel-Kirchoff diffraction parameter accounts for these dimensions, hence the diffraction gain is related to the Fresnel-Kirchoff parameter by the function;

$$G_d(dB) = 20 \log |F(v)|. \quad (2.30)$$

In real situations, graphical models can be employed to calculate the diffraction gain described by this function. An approximate mathematical solution for $G_d(dB)$ is given in Rappaport (2002), and the graphical solution is illustrated in Figure 2.7.

This model only applies when one obstruction is present. However, it can be extended

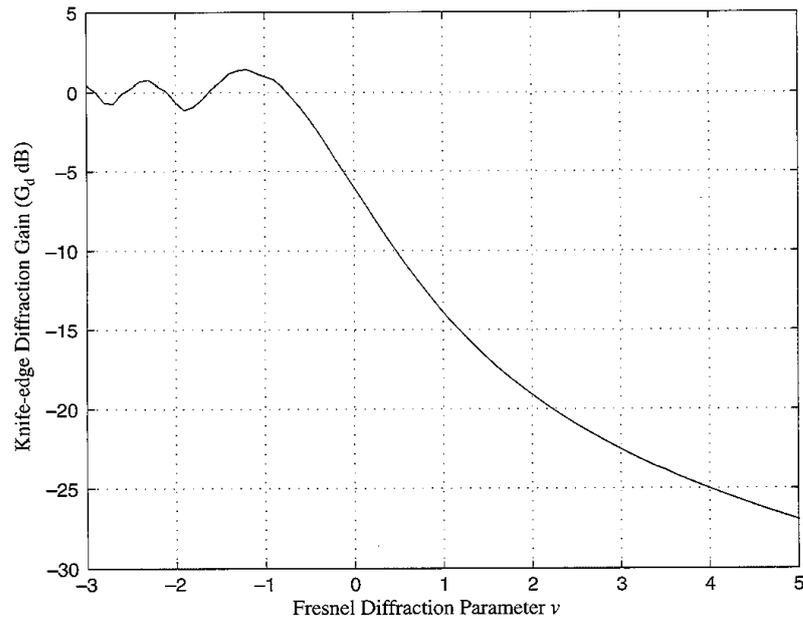


Fig. 2.7: Knife-edge diffraction gain as a function of the Fresnel-Kirchoff diffraction parameter, from Rappaport (2002).

to more complicated scenarios. For example, when the terrain corresponds to two knife-edges, they can be approximated by one dominant knife-edge, as illustrated in Figure 2.8. The overall diffraction gains can then be approximated by determining the Fresnel-Kirchoff diffraction parameter associated with the equivalent knife-edge and determining the gain from Figure 2.8. For more than 2 knife-edges this technique becomes extremely complicated and further methods must be employed, as outlined in Rappaport (2002).

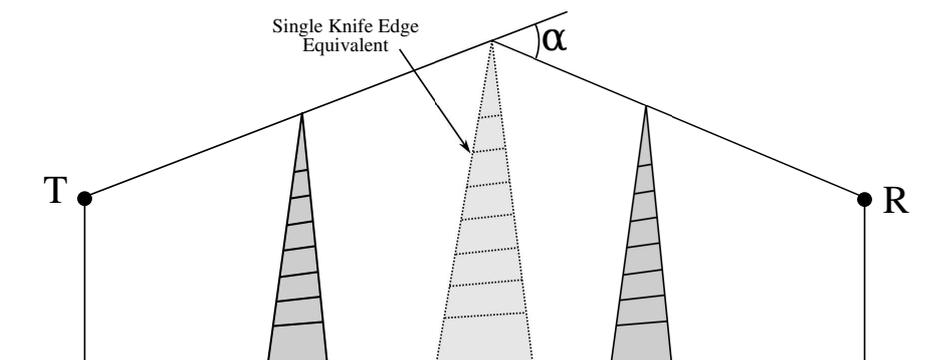


Fig. 2.8: Approximating two knife-edges with a single knife-edge, reproduced from Rappaport (2002).

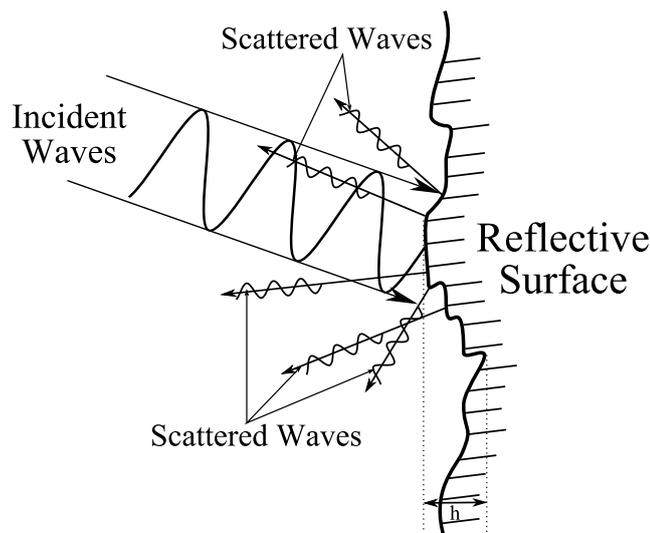


Fig. 2.9: Scattering of an incident wave.

2.1.1.4 Scattering

Another mechanism which influences the amount of energy arriving at the receiver is scattering. Scattering is similar to reflection in that the waves bounce off a reflective surface. However in scattering the reflections are more diffused. The energy is reflected out in several different directions, as illustrated in Figure 2.9.

If the incident surface has flat areas with dimensions larger than the incident wavelength, reflection will occur. Otherwise the surface is considered to be a rough surface and scattering occurs. Surface roughness is measured using the *Rayleigh criterion* which gives a maximum height for surface protrusions before the surface is considered rough and reflection becomes scattering. The maximum height of protrusions for reflection to occur is h_p ;

$$h_p = \frac{\lambda}{8 \sin \theta_i}, \quad (2.31)$$

where θ_i is the incident angle of a wave, of wavelength λ . So when the height of the actual surface protrusions is larger than h_p then the surface is considered rough and is modelled as a scatterer. Then the reflection coefficient (Γ) used in Equation (2.12) is scaled by a scattering loss factor, ρ_s , to allow for fewer waves reflected towards the receiver. ρ_s is calculated by assuming that the surface height h is a Gaussian distributed random variable with a local mean and standard deviation σ_h (Rappaport, 2002) as follows,

$$\rho_s = \exp \left[-8 \left(\frac{\pi \sigma_h \sin \theta_i}{\lambda} \right)^2 \right]. \quad (2.32)$$

So when it is determined that $h > h_p$, the reflection coefficient must be modified using the

expression,

$$\Gamma_{rough} = \rho_s \Gamma. \quad (2.33)$$

Scattering is important in understanding the behaviour of the received signal strength, not only in terms of lost reflected energy, but also in terms of extra energy arriving at a given receiver. Sometimes more energy can reach the receiver because of scattering effects. Scattered waves can bounce off a scatterer in several directions and reach a receiver, whereas a simple reflector at that point may not have delivered as much energy to the receiver. So it is important to be aware that scattering effects can cause an increase in received signal strength as well as a decrease.

2.1.2 Radio Propagation Models

Many methods have been developed to predict the amount of EM energy arriving at a receiver given the amount of transmitted energy. Since most localisation systems utilise some form of signal strength reading to perform their task, an understanding of the behaviour of the signal strength for different receiver locations is useful.

2.1.2.1 Log-Distance Path Loss

The variations of the received signal strength throughout a free-field environment can be approximated using the *Log-Distance Path Loss Model*. A free-field environment is an environment in which there are no obstructions between the transmitter and receiver for all positions of the receiver. This means that for all receiver azimuth angles the transmitter is placed at, the path loss profile over distance will be the same.

From the equation for predicting received power with reference to the received power at some distance (2.10), we can develop an expression for signal strength lost with reference to signal strength lost at some reference distance. This is referred to as the *distance-dependent path loss model* (Seidel and Rappaport, 1992);

$$\overline{PL}(d) = \overline{PL}(d_0) \left(\frac{d}{d_0} \right)^2, \quad (2.34)$$

where \overline{PL} is the mean path loss at distances d and d_0 . The exponent of 2 here is referred to as the *free-field path loss exponent*. This exponent will be different for indoor environments since the signal intensity will decay at different rates due to reflective, diffractive and scattering interactions with objects in the environment. Since the path loss exponent will be different for different environments we denote it to be the variable n . Now the distance path loss equation, in decibels, becomes

$$\overline{PL}(d)[dB] = \overline{PL}(d_0)[dB] + 10n \log \left(\frac{d}{d_0} \right). \quad (2.35)$$

Rappaport (2002) gives examples of typical values for the path loss exponent. For example, in a free-field scenario a value of $n = 2$ would be expected and in a direct Line of Sight scenario typical values would be $1.6 \leq n \leq 1.8$. Indoors, the lower n and, as a result, lower path loss can be explained by the mechanisms of reflection and scattering. The signal can reflect and scatter off the walls, floors and ceilings increasing the received signal strength, which means the signal will degrade more slowly for increasing distance, assuming direct LoS. Alternatively, for an obstructed LoS indoors, the path loss exponent is shown to be 4 to 6. The increased path loss exponent can be explained by diffraction and the attenuation of signals transmitted through walls. Section 2.1.1.3 explained how the diffracted signal received can suffer severely reduced gain.

2.1.2.2 Log Normal Shadowing

Equations (2.34) and (2.35) calculate the mean value of the path loss at a given T-R separation distance. The mean value is more significant than a single value because, due to small scale fading and measurement noise, a single measurement can vary greatly from the true mean value. Figure 2.1 illustrates how small scale signal strength can vary around the mean value. To model the instantaneous measured path loss, another term is added to the log-distance path loss model to account for the randomness of the received signal. It is assumed that at any distance d , the measured path loss $PL(d)$ is a random variable with a mean of $\overline{PL}(d)$. And it has a log-normal distribution; which means that it is a normal distribution on a logarithmic scale. Hence, a model describing a single instantaneous signal strength measurement can be generated by modifying Equation (2.35) as follows;

$$PL(d)[dB] = \overline{PL}(d)[dB] + X_\sigma \quad (2.36a)$$

$$\Rightarrow PL(d)[dB] = \overline{PL}(d_0) + 10n \log \left(\frac{d}{d_0} \right) + X_\sigma, \quad (2.36b)$$

where X_σ is a zero-mean Gaussian distributed random variable measured in dB, with a standard deviation σ , also in dB. The random variations described by X_σ are referred to as *Log-Normal Shadowing*. The parameters n and σ are usually determined using linear regression. The reference distance path loss \overline{PL} can be determined in one of 3 ways. It can be determined by measuring the path loss at the reference distance. It can also be calculated using the free space model for a distance $d = d_0$ or by using linear regression as described in Chen and Kobayashi (2002). Some typical values for n and X_σ are presented in Seidel and Rappaport (1992) and Faria (2006).

2.1.2.3 Attenuation Factor Model

The log-distance path loss model (2.35) gives better signal strength prediction accuracy than the free space path loss model (2.6a) since it utilises a parameter n which is tuned to the deployment environment. The log-normal shadowing model (2.36b) can give even better prediction of measured signal strength by statistically describing the small scale variations which impinge on the transmitted wave. However none of these models can account for the large scale attenuation of a signal due to obstructions in the environment. Partitions such as walls and doors have a significant impact on the received signal strength. A path loss exponent, n , of 4 to 6 gives reasonable overall prediction accuracy in the presence of these partitions, but its predictions are based only on T-R distance. So it will predict the same path loss for a given distance regardless of the number of partitions in the transmission LoS. Whereas in reality the presence of these partitions will significantly affect the path loss.

One type of model which utilises information about the presence of these partitions is the *Attenuation Factor Model*. This model uses a fixed value for the path loss due to a single partition, Partition Attenuation Factor (PAF). Now for multiple partitions we just need to multiply the number of partitions by the PAF to determine the total path loss due to partitions. We will obviously need a different path loss exponent, which negates the losses due to partitions, and only accounts for direct LoS losses. A simple partition attenuation factor model which accounts for partition losses, assuming all partitions cause the same level of attenuation is given by,

$$\overline{PL}[dB] = \overline{PL}(d_0)[dB] + 10n \log\left(\frac{d}{d_0}\right) + \sum PAF, \quad (2.37)$$

where $\sum PAF$ indicates that we sum the partition attenuation factors of all partitions in the direct LoS ray drawn between the transmitter and receiver. An example of how $\overline{PL}(d_0)$, n and PAF can be estimated for a given environment is provided in Chen and Kobayashi (2002). The partition attenuation factor model gives better prediction accuracy than the log-distance path loss model because the path loss exponent has its dependency on the number of partitions removed, and the $\sum PAF$ term accounts for these influences.

The case when there are multiple floors in the environment must also be considered. One method of accounting for inter-floor path loss is to use a path loss exponent which is calculated to account for the effects of multiple floors but negates the effects of partition losses. We achieve this type of path loss exponent by using regression on data from several floors, using 3D distances rather than just same floor distances;

$$\overline{PL}[dB] = \overline{PL}(d_0)[dB] + 10n_{MF} \log\left(\frac{d}{d_0}\right) + \sum PAF, \quad (2.38)$$

where n_{MF} is the multiple floor attenuation factor (Rappaport, 2002). As with the PAF we can remove the dependency of n on the number of floors by introducing a Floor Attenuation Factor (FAF);

$$\overline{PL}[dB] = \overline{PL}(d_0)[dB] + 10n_{SF} \log\left(\frac{d}{d_0}\right) + FAF, \quad (2.39)$$

where n_{SF} is the same floor path loss exponent, ignoring floor attenuation, and FAF is the attenuation due to floors for a specified number of floors separating the transmitter and receiver. Typical values of FAF for an office building is presented in Seidel and Rappaport (1992). Finally we can remove the dependency of n on both partition and floor attenuation losses by using FAF and PAF to account for these losses and using n to account for the remaining propagation variations;

$$\overline{PL}[dB] = \overline{PL}(d_0)[dB] + 10n_{LoS} \log\left(\frac{d}{d_0}\right) + FAF + \sum PAF, \quad (2.40)$$

where n_{LoS} is the path loss exponent for LoS propagation throughout the environment negating the effects of partition and floor obstructions.

2.1.2.4 Soft Partition and Concrete Wall Attenuation Factor Model

The Partition Attenuation Factor model in Equation (2.38) predicts the signal strength in the presence of partitions assuming that all partitions have the same attenuation factor. This assumption is not necessarily true in a real-life environment. Concrete partitions will attenuate the signal more than soft partitions like wooden walls, doors, or office partitions. A model presented in Seidel and Rappaport (1992) assigns separate attenuation factors to concrete partitions and soft partitions in an effort to better predict the path loss.

The *Soft Partition and Concrete Wall Attenuation Factor Model* assumes free space propagation so the authors use a model similar to that in Equation (2.6a) and do not use a path loss reference distance. Since free-space propagation is assumed, the path loss exponent is now fixed at 2. Then terms to describe the attenuation due to soft partitions and concrete partitions are added;

$$\overline{PL}(d)[dB] = 20 \log\left(\frac{4\pi d}{\lambda}\right) + p \times AF(soft)[dB] + q \times AF(concrete)[dB], \quad (2.41)$$

where p is the number of soft partitions and q is the number of concrete partitions in the T-R LoS. For the conducted experiments the T-R separation distance and resultant path loss for multiple positions are logged. At each position the number of soft and concrete partitions in the T-R LoS are also logged. Again linear regression is used to determine the soft partition attenuation and concrete partition attenuation factors. If the assumption of

free space propagation was discounted, a path loss exponent could be included to account for attenuation due to environmental propagation effects like reflection, diffraction and scattering.

2.1.2.5 Distance Dependent Path Loss Exponent

To provide more flexible representation of the path loss exponent over increasing distance, work by Cheung et al. (1998) proposed using two different path loss exponents. One path loss exponent n_1 is used for the region near the transmitter, in which propagation losses are similar to that occurring in free space. The further away region has a greater path loss exponent n_2 to allow for the increasingly noticeable influence of reflections, diffractions and scattering from obstructions. The distance at which a change in propagation loss occurs is defined as d_{bp} . Hence taking these observations into account, the new expression for path loss, modified from Cheung et al. (1998), is

$$\begin{aligned} \overline{PL}[dB] &= \overline{PL}(d_0)[dB] + 10 \log \left(\frac{d}{d_0} \right)^{n_1} U(d_{bp} - d) \\ &\quad + 10 \left[\log \left(\frac{d_{bp}}{d_0} \right)^{n_1} + \log \left(\frac{d}{d_{bp}} \right)^{n_2} \right] U(d - d_{bp}) \\ &\quad + FAF + \sum PAF, \end{aligned} \quad (2.42)$$

where $U(\cdot)$ is a unit step function. This technique of propagation modelling is shown to have signal strength prediction accuracy superior to that of Equation 2.37.

However, the downside to algorithms which involve floor and partition attenuation factors is that they require information about the number of partitions present. This information is not available online. Instead these types of models are used to obtain a signal strength map or fingerprint of the environment offline. Then fingerprinting techniques, as described in Section 2.2.1.3, are used to predict location. In contrast, expressions which do not involve attenuation factors such as Equation 2.35 allow online triangulation techniques to be employed. The propagation models presented thus far are relatively simplistic. Recently a family of more complicated propagation models have been proposed for offline generation of location fingerprints.

2.1.2.6 Simulation Models

Since the 1980s, empirical simulation models have been developed to aid in planning wireless network deployment. They capitalise on the computational power becoming increasingly available in typical computers. This allows tracing of several hundreds of paths each ray could possibly take from the transmitter, and combining the energy from each path to estimate the energy arriving at each position in the simulation environment.

One example of such a model is presented in Ikegami et al. (1991). This work focuses on modelling the mean strength of 200, 400 and 800 MHz signals in urban environments, namely downtown Kyoto and Kyoto University. It assumes reflection and diffraction to be the dominant effects. Reflection attenuation is estimated, from experimental data, to be 6dB. Diffraction effects are calculated using building heights estimated from the number of stories in the building. This technique works well in this particular test environment, however, it is stated not to work as well in a city where the building heights exceed the transmitter height, due to the way in which diffraction is modelled.

However useful this method is outdoors, it is not applicable to indoor environments. One technique which focuses on indoor signal propagation is presented in McKown and Hamilton (1991). The employed method models the signal in terms of reflections from and transmissions/diffractions through walls in a 2 dimensional map of the environment. Each ray can be traced for up to 6 reflections before termination. The chosen number of reflections permitted for each ray leads to a trade off between the execution time and accuracy.

The previous two techniques for signal strength simulation both utilise some form of theoretical propagation modelling. Another technique, which utilises mostly experimental data in the simulation, is presented in Pechac and Klepal (2001). Experimental data is used to build a “motif” for how a signal behaves upon colliding with particular types and shapes of walls. These motifs are templates for the radiation pattern probability of the signal resulting from the wave incident on the surface. Every ray traced from the transmitter has all of its transmissions, reflections, diffractions and scattering at all surfaces estimated via these motifs. A bitmap of the environment is formed and the energy at each square is proportional to the number of rays passing through that square. The benefit of using motifs is that all radio transmission phenomena are approximated by the motifs and the ability to theoretically model them is not necessary. This technique can also be easily extended to the 3D case with the availability of sufficient computational power.

These simulation models permit fine grained modelling and visualisation of the signal strength throughout a situation where it would be impractical to manually obtain signal readings throughout the entire environment. Availability of these readings throughout the environment permits the employment of fingerprinting based localisation techniques, as will be explained in Section 2.2. Bahl and Padmanabhan (2000b) illustrated that in a Wireless Local Area Network (WLAN) localisation system, a dataset generated from a simple propagation model, like that in Equation 2.37, produces inferior accuracy to that obtained using an empirical training dataset. However, in a more recent paper Widyawan et al. (2007b) demonstrated the localisation accuracy possible when data from a simulation model is utilised. It showed that the associated accuracy is competitive with

that achievable with a manually obtained dataset, and the calibration effort for a large environment is significantly lower.

So far the propagation modelling algorithms only consider large scale path loss effects. Other, less significant, propagation effects are also present in any transmission.

2.1.2.7 Small Scale Fading

Small scale fading refers to the rapid fluctuations in the received signal over short periods of time or travel distance. As alluded to in Section 2.1, small scale fading does not significantly influence the signal strength over a long period of time. Small scale fading models are mainly concerned with understanding how speed of the transmitter, receiver or objects in the environment influence received signal in terms of apparent frequency and arrival time of signal components from different directions. In a human tracking system, however, the subject's speed is insignificant compared to the speed of RF wave travel, hence relative T-R speed will not influence signal strength intensity.

Small scale fading also describes instantaneous fluctuations in the received signal intensity due to multipath effects like reflection and scattering. However small scale multipath effects are so difficult to predict and are so insignificant over long periods of time, that they are usually mitigated in localisation systems by using average signal strength values rather than instantaneous values. Small scale variations in received signal strength are usually modelled in a statistical fashion, as discussed in Section 2.1.2.2.

For radio localisation systems the consequences of small scale variations in the signals are usually ignored and the signal strength due to large scale propagation effects are used to interpret the current position. The aforementioned propagation models are commonly used for distance estimation in triangulation techniques or signal strength dataset acquisition for fingerprinting techniques, as will be described in the following section.

2.2 Positioning Theory

The field of RF localisation is concerned with determining the position of a piece of radio communication hardware by analyzing information about the radio signals sent to or received from a piece of communication hardware. The hardware to be localised is often referred to as the Mobile Device (MD). When the signals from an MD are analysed by some other piece of hardware this is known as remote positioning, since the location of the MD is determined remotely from the device being localised. If the position of the MD is calculated and stored on the MD, based on signals arriving from a stationery Access Point (AP) this is known as local positioning. When this MD is carried by a person, they can be offered location based services from either the local or remote localisation system.

There are two ways of presenting the calculated positions; relatively or absolutely.

Relative positioning presents the position of an MD in relation to some reference position within an environment. Alternatively absolute positioning presents the position of an MD in relation to some absolute frame of reference. An example of absolute positioning is the coordinate output of a GPS module, whereas relative positioning would say that a user is 20m north of some landmark. It is possible to translate from relative to absolute position and vice versa if we know the position of the landmark which the MD is relative to.

There are two main categories which a localisation technique can fall under, they are *Direct Techniques* and *Sequential Bayes Filtering Techniques*. These techniques will be the subject of the following subsections.

2.2.1 Direct Techniques

Direct localisation techniques are techniques in which the current position is derived from each sample without reference to previous or future location predictions.

There are three main categories which a direct localisation technique can fall within;

- Proximity
- Triangulation
- Scene Analysis/Fingerprinting

Each method of localisation has advantages and disadvantages in terms of installation and configuration effort, accuracy and reliability (Hightower and Borriello, 2001). There is no one localisation system ideal for every situation, each one has its ideal application. The following subsections will describe how each of these methods can be used to determine location.

2.2.1.1 Proximity

Arguably one of the simplest methods of determining the location of an object or a human is by using the *Proximity* method. This method is used when the only reading available from some sensing hardware is a binary detected/not-detected reading. Hence, the location is approximated to be the same as the position of the detecting sensor. As a result, the higher the detecting range of the sensor, the higher the uncertainty of the location prediction. Proximity localisation techniques do not lend themselves to accurate coordinate location predictions, instead symbolic location predictions are usually generated. Symbolic location refers to a location which has some meaning relevant to the actions which usually take place in that location, such as a room or area within a room.

Proximity sensing methods are well suited to the detection of room-level location when the signal to be sensed is readily attenuated by walls. Typical technologies suitable for this purpose are IR and Radio Frequency Identification (RFID). Example of IR proximity

sensing techniques can be seen in Want et al. (1992, 1995) and Abowd et al. (1997). A disadvantage of IR proximity technologies is the dependence of IR on LoS transmission. Obstructions within the room have a severe impact on the likelihood of detection for IR transmissions. RFID transmissions overcome this limitation by using RF transmissions, which are less susceptible to obstructions, but as a result are more likely to “leak” into adjacent rooms. Due to its reliability, RFID has replaced IR for symbolic localisation systems. Examples of localisation using RFID can be found in Cox et al. (2003) and Callaghan et al. (2006).

Proximity localisation using RFID technology can obtain finer resolution location predictions by placing a larger number of sensors or tags throughout the deployment environment. Koch et al. (2007) suggested fixing an array of RFID tags under a carpet throughout an entire environment, then an RFID reader fixed on a subject’s foot can resolve location with high resolution, using only proximity readings. Another piece of work by Kulyukin et al. (2008) uses similar densities of RFID tags on the floor but this time the RFID reader is attached to an elder’s stroller. The resolution of location predictions is sufficiently high to allow navigation to take place. However, this high resolution comes at the expense of high installation effort.

It is not just short range communication technologies which can be used for proximity localisation. Any technologies which can detect other devices with little or no indication of the signal intensity can be considered to be proximity technologies. For example Google Maps’ positioning system uses cell ID localisation which approximates the location of a user to be the location of the detected cell tower. In urban areas this can give accuracy within 200m but this can increase to approximately 3 km in areas with sparse cell tower deployment. Some bluetooth positioning systems also could be considered proximity localisation systems due to the lack of signal strength intensity readings on some Bluetooth[®] chips (see Anastasi et al. (2003) for example).

When a person or object is detected at multiple sensors or receivers at once, higher localisation accuracy can be obtained by using the centroid algorithm. This algorithm simply calculates the coordinate position of the object to be the geometric mean of the coordinate positions of all the sensors which detected the object (see Hightower and Borriello (2004) for example). These proximity localisation methods are employed when only binary presence readings are available or when signal strength or timing resolution is not high enough to permit T-R distance estimation. When such readings are available, higher accuracy can be obtained by using *Triangulation* or *Fingerprinting* methods, hence proximity techniques are rarely employed in recent work.

2.2.1.2 Triangulation

Triangulation refers to using the geometric properties of a triangle to determine the position of an MD relative to a number of stationery APs. Figure 2.10 illustrates an example where the location of devices A and B are known, and we want to calculate the relative position of device C. Triangulation will directly give the relative position of device C. From this and knowledge of the absolute positions of A and B the absolute position of device C can be calculated. Using triangulation methods, there are two ways the position of device C can be calculated: *Lateralation* or *Angulation*.

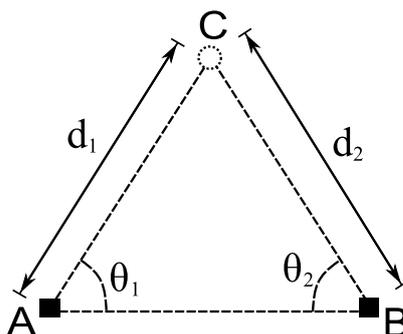


Fig. 2.10: Triangulation localisation methods can calculate the position of circle C given the known locations of squares A and B and some other information, either distances, d_1 and d_2 or angles θ_1 and θ_2 .

Lateralation (or Trilateration as it is sometimes called) is used when the distances between the MD and some APs are known. In Figure 2.10 the positions of A and B and the distances d_1 and d_2 are known. The distances d_1 and d_2 can be used to calculate the position of C relative to A and C, using simple geometry. In the situation where there are only two APs the result will be two possible locations of C. This is because two positions will satisfy separation distances d_1 and d_2 , as illustrated in Figure 2.11(a). If the distance, d_3 , to another AP, D, at a known location is available, the position of C can be deduced. Figure 2.11(b) shows that the only possible position of C can be determined using information about the third AP.

This suggests that in an unconstrained 2-dimensional space, a minimum of 3 access points will be needed for lateralation. But in a constrained space, like indoors, the environment configuration can be used to reduce the number of APs required. For example if both A and B are placed against the perimeter wall of a building and it is assumed that the MD won't be detected outside the building (for example in a multi-storey building), it can be assumed that one location of C is invalid and a confident position prediction using only 2 APs is possible.

For lateralation to successfully work in an unconstrained space the number of access points required, N, is;

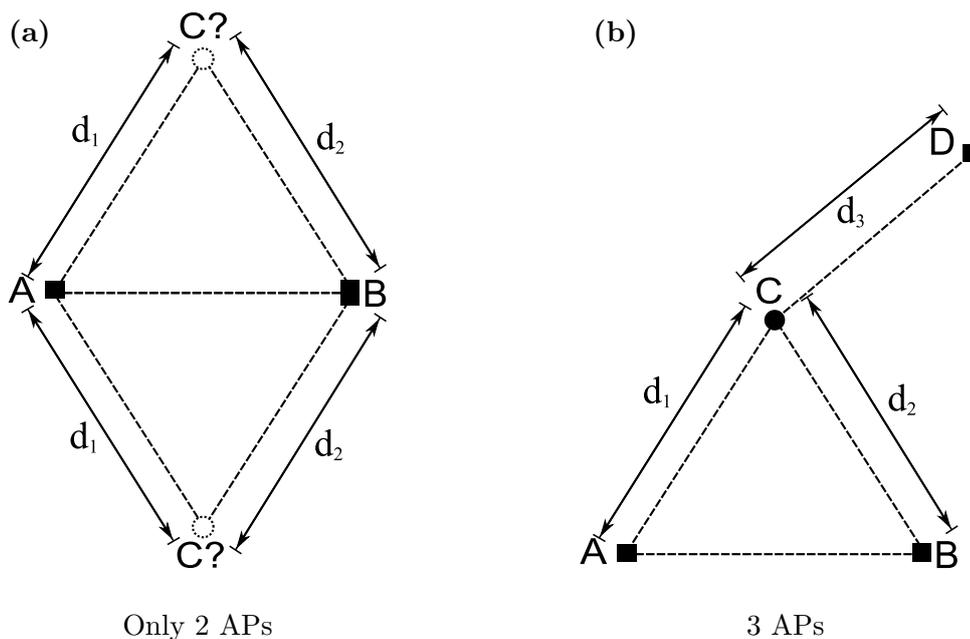


Fig. 2.11: Lateralization used to predict the position of C using 2 fixed APs and 3 fixed APs. (a) Using only 2 APs results in two possible locations of C for the measurements d_1 and d_2 . (b) The availability of 3 APs removes uncertainty.

$$N = D + 1, \quad (2.43)$$

where D is the number of dimensions in which localisation must be performed in. D is the minimum number of APs required to unambiguously predict a position, as corroborated by Khan et al. (2006). Also if more APs are available they will provide better accuracy.

The first step in lateralization is the estimation of the distances d_1 , d_2 and d_3 . Many methods of distance estimation can be employed in a lateralization-based localisation system. The most used methods are:

Signal Strength Most radio transmission devices have a Received Signal Strength Indication (RSSI) reading built-in and visible to the application layer. This RSSI value can usually be converted to a received signal intensity value, measured in dBm. With this signal intensity value a propagation model like the ones described in Section 2.1.2 can be used to determine the T-R separation distance. Examples of systems which use RSSI based lateralization are described in Hightower et al. (2001), Feldmann et al. (2003), Castano et al. (2004), Orooji and Abolhassani (2005), Sugano et al. (2006), Jin et al. (2006) and Tarrío et al. (2008). RSSI is the most commonly used distance indicator due to its common availability, even though it gives poor accuracy due to unpredictable multi-path effects. When less standard transmission hardware is used further readings are possible, resulting in higher distance estimation

accuracy.

Time of Arrival (TOA), sometimes known as Time of Flight (TOF), allows for the prediction of the T-R separation distance by observing the time a signal takes to reach the receiver after transmission. The time the signal was transmitted must also be known, usually included in the data packet sent within the signal, so the T-R propagation time can be determined. With knowledge of the propagation time, and the speed of propagation of the wave, the T-R separation distance can be calculated. TOF is generally more accurate for distance estimation than signal strength since the time it takes for the RF wave to travel is less susceptible to obstructions (Kupper, 2005) but it requires very accurate clock synchronization between the transmitter and receiver to allow calculation of the transmission time. Also statistical methods must be employed to prune out secondary signals due to multi-path effects. TOA is used in GPS (Kupper, 2005). It is also used in ultrasonic systems such as the Active Bat localisation system (Ward et al., 1997, Harter et al., 1999) and Hexamite's ultrasonic positioning system (Helal et al., 2003).

Round Trip Time (RTT) is a measure of the time it takes for a signal to reach a receiver and a response to be sent back to the original transmitter. When there are no delays present in the receiver hardware it is equal to twice the TOF. RTT is an improvement over TOF since it does not require the transmitter and receiver to have synchronised clocks. However, it is an inaccurate distance measure in the situation when there is a delay between when the receiver receives a packet and when it sends the response packet. RTT has been proposed for application to cellular network localisation (Jami et al., 1999) and WLAN localisation using auxiliary hardware (Izquierdo et al., 2006). The Active Bat localisation system may also be considered a RTT technique since it sends an RF pulse when it wants an ultrasonic pulse sent back. Then the RTT is a combination of RF transmission in one direction and ultrasonic in the other direction.

Time Difference of Arrival (TDOA) predicts distances from information about the difference in arrival time for different signals which were sent at the same time. Such signals include RF, ultrasonic, infrared or acoustic. This technique only works when the different signals have significantly different transmission speeds. Knowledge of the propagation times of each signal and the receive time difference between the two signals allows accurate resolution of transmission distance, assuming the time resolution of the readings is high enough. The Cricket location support system (Priyantha et al., 2000) uses the TDOA of RF and ultrasonic signals from a number of beacon transmitters at known locations to estimate the receiver's position. Work on a similar technique by Savvides et al. (2001) indicates that TDOA using RF

and ultrasonic is more reliable than using RF signal strength distance estimation. This increased reliability, however, comes not just at the cost of extra signals but of sophisticated hardware capable of deducing times of arrival with sufficient resolution.

Angulation is a form of triangulation in which information about the angles of the device at unknown position, **C** in Figure 2.10, relative to devices **A** and **B** are detectable. This is usually derived from the angle from which a signal arrives at the device or the Angle of Arrival (AOA). Again if the positions of **A** and **B** are known, then the position of **C** is deducible. Unlike lateration, angulation can work with only 2 APs, regardless of whether the localisation problem is 2-dimensional or 3-dimensional, provided the available angle readings include elevation as well as azimuth angles. If elevation readings are unavailable for the nodes **A** and **B** then only 2-dimensional localisation is possible.

Work by Niculescu and Nath (2003) uses an adapted form of the Cricket localisation system, called the Cricket Compass, which uses an array of ultrasonic sensors. This array of sensors allows the detection of the phase difference between a signal arriving at the different sensors. From this information the AOA, and as a result, location can be derived. Boushaba et al. (2007) use AOA to allow the nodes in a sensor network to localise themselves, however they use a Medusa node, like that used by Savvides et al. (2001), adapted to estimate AOA.

Instead of detecting a signal's angle of arrival, Khan et al. (2006) proposes an alternative technique of estimating the angle between nodes by estimating the angle at which a main node transmitted a signal successfully to a given node. A master node of known position and orientation transmits a reset beacon to all other nodes. Then it begins to transmit a narrow directional signal rotating around its axis. The slave nodes note when they hear each transmission. From these times they can work out the period of one rotation. They can also work out how long it took to detect the first transmission after the reset signal. The ratio of this initial transmission time to the period of rotation is proportional to the angle of this node relative to the master node. Hence each node knows its angle relative to the master node. This can be considered a hybrid technique since it also uses signal strength to estimate distance between master and slave. The position of each node is then estimated from one angle and one distance reading.

In theory, triangulation localisation methods are capable of resolving location with high accuracy, limited only by the accuracy of the distance or angle estimates. However, in practice the distance and angle estimates are not sufficiently reliable to allow accurate localisation. Hence another localisation method, fingerprinting, is more frequently employed in practice.

2.2.1.3 Fingerprinting

Fingerprinting is a method of predicting location which does not necessarily involve knowledge of the propagation characteristics of the environment or the location of the reference devices (usually wireless Access Points). Instead the device to be tracked is moved to coordinate or symbolic locations of interest and a training set of AP identities and corresponding signal strengths is obtained in each location. Obtaining such a set of training samples for an entire environment makes up the offline phase of fingerprinting-based localisation. In the second phase, the online phase, samples are obtained and compared to the offline training samples using a variety of methods to estimate the most likely location. These methods can generally be considered to be either direct or probabilistic methods. Examples of direct fingerprinting methods can be found in Bahl and Padmanabhan (2000b), Krumm et al. (2003), Mantoro and Johnson (2005), Varshavsky et al. (2006), di Flora and Hermersdorf (2008) and Kelly et al. (2008d) and examples of probabilistic methods can be found in Castro et al. (2001), Ladd et al. (2002), Roos et al. (2002b) and Youssef et al. (2003).

Generating a fingerprint for an environment can be performed in two ways: empirically, by obtaining real data at every position in the environment or mathematically, by using propagation models, like those outlined in Section 2.1.2. As explained in Section 2.1.2.5 propagation models capable of executing online are not as accurate as ones which can only execute offline. This is due to the ability of offline models to describe more complex phenomena such as wall and floor attenuation². Hence, generation of propagation models offline before the execution of a fingerprinting algorithm leads to better localisation accuracy than using propagation models online, for example when using triangulation. The disadvantage, however, is that fingerprinting requires more offline preprocessing and online storage of data than triangulation. Also changes in the environment such as changing AP or furniture positions requires the generation of a new training dataset.

A strong reason fingerprinting is favoured over triangulation is that it allows the use of signals in localisation which are not readily modelled in a given environment. For example Otsason et al. (2005) uses Global System for Mobile Communications (GSM) signals to estimate position indoors. GSM signals are extremely difficult to model indoors since they usually originate from cell towers over a kilometer away and they can be influenced by walls, windows and outdoor foliage as well as environmental factors such as diffraction through the atmosphere. Hence an empirical training dataset is used. Castro et al. (2001) uses an empirical dataset using Signal to Noise Ratio (SNR) instead of the typical RSSI reading. This allows the signal intensity throughout the environment to incorporate interference levels. Interference varies as a function of location so it should be incorporated

²To the author's knowledge no localisation systems exist which are capable of estimating the number of partitions in the T-R LoS online.

into the location indicative readings, something which is not possible with models due to its unpredictability. Finally, the system developed as part of this thesis uses GSM signal intensity and Bluetooth[®] Link Quality (LQ) readings along with RSSI. These signals are extremely difficult to model, hence empirical fingerprinting is used.

Fingerprinting is the most popular direct localisation technique since its prediction accuracy does not rely on propagation models which are sufficiently simple to run online. It can use sophisticated propagation models which depend on information only available offline or even mitigate understanding of propagation mechanisms by permitting the use of empirical training datasets. Regardless of these strengths, fingerprinting still has deficiencies in predicting movements which correspond to human behaviour. For this reason, more recent localisation research employs *Sequential Bayes' Filters* to provide location predictions which correspond to human movement behaviours.

2.2.2 Sequential Bayes' Filters

Direct localisation techniques perform sufficiently accurately when stable noise-free readings are available from the device to be localised. However, when only noisy readings are available location jitter can occur. This is when an individual's predicted location moves rapidly from one sample to the next. *Sequential Bayes' Filters* provide a natural mechanism to fuse information about an individual's most likely previous location and possible movements to provide sequences of location predictions which are realistic for a subject travelling with typical human behaviour.

Sequential Bayesian filtering is a recursive approach for modelling the probability density function (pdf) of some system's internal states over time. The states are estimated based on two probabilistic models:

Process Model. This model describes the relationship between the internal state of the system at one discrete instant in time and the next. In localisation this is also referred to as the motion model.

Measurement Model. This model describes an externally viewable measurement as a function of the internal state of the system. In localisation this typically relates RSSI from multiple APs to the device's location.

From a localisation perspective, the goal is to estimate the posterior pdf of the location, l_t , at discrete time t , given the available measurements, R_t , at discrete time t , that is to estimate

$$p(l_t | R_t). \tag{2.44}$$

This can be recursively estimated in two steps. The first step, before any signal reading is obtained, is to use the previous location to predict the current location using the motion model, which is derived from the Chapman-Kolmogorov equation (Ristic et al., 2004),

$$p(l_t | R_{t-1}) = \int p(l_t | l_{t-1})p(l_{t-1} | R_{t-1})dl_{t-1}. \quad (2.45)$$

This prediction is based on the first-order Markov assumption that the current state (or location) is dependent only on the previous state (Zàruba et al., 2007), ie.:

$$p(l_t | l_{t-1}, \dots, l_0, R_{t-1}, \dots, R_0) = p(l_t | l_{t-1}). \quad (2.46)$$

Next, when a new sample, R_t , is obtained, this prior probability can be combined with the motion model, $p(R_t | l_t)$, to correct the location prediction. Using Bayes' rule this correction takes the form,

$$p(l_t | R_t) = \frac{p(R_t | l_t)p(l_t | R_{t-1})}{p(R_t | R_{t-1})}, \quad (2.47)$$

where the normalising constant is

$$p(R_t | R_{t-1}) = \int p(R_t | l_t)p(l_t | R_{t-1})dl_t. \quad (2.48)$$

This recursive prediction (Equation 2.45) and correction (Equation 2.47) algorithm forms the optimal Bayesian filtering process. However, this is merely a conceptual explanation of Bayesian filtering, the integrals present in this algorithm make it computationally intractable. Instead some simplifications and approximations must be made to allow this algorithm to execute in discrete computer operations. Next some discrete approximations of this algorithm are presented.

2.2.2.1 Kalman Filters

A *Kalman filter* is capable of optimally predicting the continuous-valued multivariate state of a system by implementing a form of Bayes' filtering. Optimal prediction is possible in the case when certain assumptions are true. These assumptions are that the posterior density is unimodal Gaussian and that the prediction and motion models are linear. The unimodal Gaussian assumption is necessary to allow the posterior density to be parameterised entirely by its mean and covariance. The mean value and covariance of the state vector must be calculated for both motion model based prediction and measurement model based correction at every iteration of the algorithm. Without presenting the mathematical basis (a comprehensive explanation can be found in Welch and Bishop (1995)), the Kalman filter is implemented using matrix algebra as follows.

The mean value of the state is predicted using the motion model,

$$\hat{l}_t^- = A\hat{l}_{t-1}^- + Bu_t, \quad (2.49)$$

where A corresponds to the motion model and B specifies the contribution from an input, u_t , to the state evolution. u_t may not necessarily be known in a localisation problem and A is equivalent to the state-transition matrix in control theory. The corresponding covariance matrix, P_t^- , must also be updated based on the previous covariance matrix and the motion model noise covariance matrix, Q , as follows:

$$P_t^- = AP_{t-1}^-A^T + Q. \quad (2.50)$$

P_t^- represents the uncertainty in the location estimate l_t^- . The larger the elements of this covariance matrix, the wider the spread of the location distribution (Fox et al., 2003), hence the greater uncertainty in the location estimate, \hat{l}_t^- . When new measurements, R_t , become available this prediction can be corrected using the measurement model, which incorporates the output matrix, H . First the Kalman gain must be calculated:

$$K_t = P_t^-H^T(HP_t^-H^T + M)^{-1}, \quad (2.51)$$

where M denotes the measurement noise covariance matrix. With this Kalman gain, the state mean and covariance predictions are corrected using the expressions,

$$\hat{l}_t = \hat{l}_t^- + K_t(R_t - H\hat{l}_t^-) \quad (2.52)$$

$$P_t = (I - K_tH)P_t^- \quad (2.53)$$

Since most realistic systems, especially location estimation systems, typically do not have linear measurement models the Kalman filter is not applicable. Instead, Extended Kalman Filters (EKF) (Kotanen et al., 2003) or Unscented Kalman Filters (UKF) (Orderud, 2005) use linear approximations to satisfy the linear requirements of Kalman filtering. In these cases the underlying assumption of linear functions is no longer valid and these filters are no longer optimal Bayes' filters. Furthermore Kalman filters can only be employed when measurement models are available. As explained in Section 2.2.1.3 accurate propagation models are not always available. In such situations optimal Bayes' filtering is not possible using Kalman filters. Instead alternative discrete Bayes' filter approximations are necessary.

2.2.2.2 Particle Filters

The main disadvantage of Kalman filters is that they assume the location can be approximated by a unimodal Gaussian distribution. This is certainly not the case in

a real system in which the ambiguous signal strength readings could be the result of a number of different locations. Hence, the location is more appropriately approximated by a multi-modal distribution. As an example, if a user was equally likely to be in two positions distant from each other, a Kalman filter would estimate the most likely location to be the Euclidian mean of those two locations, hence it would be entirely wrong (Zàruba et al., 2007).

To provide a multi-modal density approximation and to overcome the mathematical intractability of the integrations in the pure Bayes' filter, a Monte Carlo sampling approach can be employed. Monte Carlo filters (also referred to as *particle filters*) use repeated random sampling to approximate the results of exact Bayes' filtering. The entire distribution is represented by a set of weighted random samples. At each step in the filtering process each sample value and weight is modified according to the motion and measurement models. Similar to the Kalman filter each iteration has two steps: prediction and correction.

At the start of the first step, prediction, a large set of samples and corresponding weights, $(l_{t-1}^{(i)}, w_{t-1}^{(i)})$, exists. Each sample is resampled randomly according to the motion model $p(l_t | l_{t-1})$. The resulting set of samples, corresponding to Equation 2.45, is denoted (Zàruba et al., 2007):

$$\left\{ (\tilde{l}_t^{(i)}, w_t^{(i)}) \mid i \in [1, N], w_t^{(i)} = 1/N \right\}. \quad (2.54)$$

The second step, update, changes the weights of each sample according to the measurement model and a new measurement, R_t . Each sample's weight is adjusted according to $\tilde{w}_t^{(i)} = p(R_t | \tilde{l}_t^{(i)})$. The weights are then normalised to ensure they sum to 1. From this set, N samples are randomly drawn according to the normalised weight distribution. Resampling is permitted to ensure it is possible to get N samples while still giving greater importance to samples with higher weights. The resulting set of samples, corresponding to the posterior distribution $p(l_t | R_t)$ in Equation 2.47, is denoted:

$$\left\{ (l_t^{(i)}, w_t^{(i)}) \mid i \in [1, N], w_t^{(i)} = 1/N \right\}. \quad (2.55)$$

This set of samples can then be used in the next iteration's prediction step and the recursive process continues in this manner. The main factor which influences accuracy in a particle filter is the number of particles chosen, N . The more particles available the higher the posterior density representation flexibility but also the higher the iteration computation time. Due to the increasing computational power available in modern computers, particle filters are commonly used for coordinate location tracking. It has been successfully applied to WLAN (Zàruba et al., 2007, Widyawan et al., 2007a), Bluetooth[®] (Rodas et al., 2008) and ZigBee[®] (Ren et al., 2007) technologies.

Along with the multi-modal capability of particle filters, other advantages lie in their ability to use empirical measurement models and their ability to easily incorporate information about the environment layout into the motion models. When the task is to track coordinate position Kalman and particle filters perform extremely well. Kalman filters perform optimal prediction under the linear assumptions and particle filters perform better with more computational power. However, neither of these techniques can optimally predict location when the location state-space is truly discrete in nature.

2.2.2.3 Discrete Approaches

Discrete approaches allow tracking by segmenting the state-space into discrete locations. One example is a *grid-based approach* which divides the localisation environment into equally sized cells (Burgard et al., 1996). Then the Bayes' filter update equations are performed on each individual cell using summations instead of integrations. The cell of highest probability at the end of each iteration is assumed to be the true location. The advantage of this method is that it can also represent arbitrarily shaped distributions. It is also assumed to be an optimal Bayes' filter when the state space is truly discrete (Ristic et al., 2004). The disadvantage of grid-based techniques is the storage requirements for the grid and the computational complexity of updating the entire grid on every iteration (Fox et al., 2003). There is a predictable tradeoff between the tracking accuracy due to grid granularity and the storage and computational complexity due to grid size.

To overcome the prohibitive computational burden of grid-based approaches *topological approaches* have been proposed. Topological approaches segment the environment into locations of meaningful significance to the user, or symbolic locations, such as different rooms or hallways. Now each cell corresponds to an entire symbolic location and the computational complexity is significantly lower than a full grid-based approach. Furthermore the motion model is easily derived from the connectivity of locations, due to doorways or adjacency for example. Topological approaches have been said to efficiently represent the localisation environment and perform sufficiently well when the sensors provide only very imprecise location information. Hence, as will be presented in Section 6 this approach is adopted to augment the accuracy of the system developed in this thesis. Examples of discrete Bayes' filter techniques can be found in Krumm and Horvitz (2004) and Kelly et al. (2008a).

Thus far this chapter has presented fundamentals of RF positioning theory and the RF propagation theory on which it builds. The next section summarises the most prominent indoor localisation systems, both commercial and academic, which have been developed to date.

2.3 Existing Localisation Systems

To date, a wide variety of position estimation techniques have been developed. Outdoors, location predictions are readily available to the consumer from global systems such as GPS and the upcoming Galileo (EU GPS), GLONASS (Russian GPS) and Compass (Chinese GPS) systems. There are two major issues with using these global scale positioning systems indoors. The first issue is connectivity. It is usually difficult for a receiver to detect signals from enough satellites to predict location, especially in multi-storey buildings. The second issue is reliability. Even if enough connections are available indoors, the signal is generally so distorted by multi-path effects that the position estimate is extremely inaccurate.

To address the challenge of indoor localisation a wide variety of alternative techniques have been investigated. Most of these techniques utilise relatively short-range radio transmissions to estimate position. These short-range radio transmission protocols, such as WLAN, Bluetooth[®] and cellular networks are particularly applicable to indoor localisation since they are typically available in indoor scenarios. Other research attempts to achieve higher accuracy or reliability using hardware more customised to the localisation problem, relying on transmissions such as Ultrasonic, RFID and IR. These different techniques will be compared in terms of accuracy, localisation technique employed, location estimate type (coordinate or symbolic), release type (commercial or academic), location availability (local or remote) and infrastructure requirements. It is necessary to note the infrastructure, or quantity of installed hardware, present when each technique was tested since it is difficult to directly compare localisation techniques deployed in different environments.

One of the most ubiquitous indoor communication and prominent localisation technologies, WLAN, shall be presented first.

2.3.1 WLAN

WLAN is a communication protocol frequently employed in indoor localisation. This is primarily due to its high deployment density in commonly inhabited areas such as offices, universities and urban homes. RADAR (Bahl and Padmanabhan, 2000b) was one of the first non-robot indoor localisation systems which relied on WLAN infrastructure already present in an office environment. Since then many authors have tried to improve upon RADAR's accuracy with varying degrees of success. Most WLAN-based localisation systems are able to calculate and store their positions locally. This is due to the fact that most WLAN positioning systems are implemented on laptops which have sufficient storage capacity and computational power to represent and analyse an entire environment's data locally. A simple summary of the most prominent WLAN-based localisation systems can be found in Tables 2.1 and 2.2.

From these tables it can be seen that until the year 2005, WLAN localisation systems typically used fingerprinting techniques to estimate location. Furthermore, triangulation is not widely employed due to the poor accuracies associated with oversimplified online propagation models. In more recent years particle filtering techniques have experienced increased adoption. These techniques are capable of achieving higher accuracy from the same training and test data using Monte Carlo Simulations of a Bayes' filter. As explained in Section 2.2.2.2, these techniques require high levels of computational power to achieve the desired levels of accuracy. As a result, particle filtering techniques are growing in popularity in line with the increasingly available laptop processing power.

Another trend observable from these tables is that proximity and Kalman filtering techniques have not been applied to any of the systems surveyed. Proximity techniques are rarely, if ever, applied to WLAN localisation systems because their relatively low accuracy is only acceptable in situations when the resolution of the signals available from the technology (such as RSSI) is extremely low or gives binary readings. Kalman filters are rarely applied to realistic indoor localisation scenarios due to the non-linear nature of indoor RF propagation. One final observation is that WLAN is such a mature wire replacement technology that it has spawned some commercial localisation systems such as Ekahau and Skyhook WPS. The same can not be said for other wire replacement technologies such as Bluetooth[®] or IEEE 802.15.4. As will be presented in the next section, cellular network localisation is also experiencing some commercial interest due to its ubiquity.

2.3.2 Cellular Network

Cellular network localisation is a technique that is now widely used due to the prevalence of mobile phones and cellular network towers. However, only low levels of position accuracy can be achieved with commonly available handsets. Enhanced performance is available to the cellular network operator using the link management readings available at the basestations³, however, these measurands are generally unavailable to the user. Typically available cellular network devices only indicate the RSSI of the currently strongest cell tower. For this reason a commonly available commercial cellular network localisation product, Google Mobile Maps, only uses the proximity localisation technique. This system has no official accuracy claims, but due to the range of cellular network basestations this technique can have position error of anything up to 3 km, assuming the position of the cell tower is correct in the database, which is not always the case. A similar technique, implemented in Intel's Place Lab (Lamarca et al., 2005) achieves its best accuracy of 107.2m in a dense urban area, namely downtown Seattle. This indicates that higher

³Examples include; RSSI from multiple basestations, basestation timing advance and time difference of arrival at multiple basestations. See Jami et al. (1999) for more examples.

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Bahl and Padmanabhan (2000b) (RADAR)	✓						C A R	R		3 APs in an office environment	2.94m median location error
Castro et al. (2001) (Nibble)	✓						S A L	L		10 APs in an office environment	97% correct location predictions
Newbury Networks Inc.	✓						C C R	R		Any available APs	No official accuracy claims
Ladd et al. (2002)	✓			✓			C A L	L		5 APs in an office environment	Less than 4m error 98% of the time
Roos et al. (2002b) (Ekahau)	✓						C C R	R		10 APs in an office environment	< 2m error on average
Youssef et al. (2003) (Horus)	✓						C A L	L		Average of 4 APs detectable at each location	Less than 4m error 98% of the time
Skyhook WPS	✓						C C L	L		Uses as many APs as detectable	No official accuracy claims
Krumm and Horvitz (2004) (LOCADIO)					✓		C A L	L		Unspecified number of APs	1.53m median error
Mantoro and Johnson (2005)	✓						S A L	L		Unknown number of WLAN APs	96% correct

Table 2.1: Comparison of WLAN based localisation systems, part 1.

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Seshadri et al. (2005)					✓		C	A	L	2 APs in an office environment	6m mean error
Lamarca et al. (2005) (Place Lab)		✓			✓		C	A	L	Any available WLAN infrastructure inside or outside	13.5m median error
Izquierdo et al. (2006)		✓					C	A	L	3 WLAN APs and a laptop with a custom WLAN receiver	Less than 2m error 90% of the time
Hermersdorf (2006)		✓					S	A	L	82 APs in a 7 storey building	88% correct floor and wing predictions
Widyawan et al. (2007b)					✓		C	A	L	5 APs in an office environment of area 1600m ²	3m mean error
Zàruba et al. (2007)					✓		C	A	R	1 AP in a home environment	2.5m mean error
Tarrio et al. (2008)		✓					C	A	L	4 APs in an office environment of area 600m ²	2.5m mean error
Kelly et al. (2009a)		✓					S	A	L	6 APs in an office environment of area 2016m ²	80% correct predictions in locations with 4 APs detectable

Table 2.2: Comparison of WLAN based localisation systems, part 2.

accuracy for such systems is obtained with higher AP density, but the error distances are prohibitively high for indoor localisation.

More accurate location prediction techniques are possible with specialised hardware or in simulations which do not account for the limited visibility of cellular signals on most mobile phones (see Table 2.3). Otsason et al. (2005) use laptops with GSM receivers or hacked mobile phones to obtain RSSI readings from the 6 strongest cellular basestations. With this specialised hardware accuracy levels similar to that of WLAN is possible. A UK-based company, Path Intelligence, is implementing a shopping centre surveying technology. This technology uses custom cellular network detectors to detect the cellular signals emanating from customers in the area. Phone identifier numbers along with the corresponding AOA's are used to triangulate each user's position and movements allowing shopping centre planners to better understand customer behaviour. However, since this technique can remotely estimate a user's location without permission, it is generating significant levels of customer concern, in spite of its anonymity preservation measures. Furthermore, these location predictions are not readily available to the user, making this technique a planning tool rather than a technique for a user to retrieve their location.

Cellular localisation has received increased attention in recent years due to the high availability of cellular devices and the presence of cellular signals in most locations. Recently some mobile phones using the Android operating system allow access to detailed cellular readings, such as the 6 strongest cell towers and their RSSI. Theoretically, this would allow more accurate techniques such as cellular fingerprinting (like Otsason et al. (2005)) or cellular triangulation (like Orooji and Abolhassani (2005)) to be performed locally on mobile phones. However, in general, the high transmission range and the low signal resolution and low access point visibility on commonly available cellular devices leads to poor resolution cellular localisation. Another commonly available communication protocol sometimes used for indoor localisation is Bluetooth[®].

2.3.3 Bluetooth

Bluetooth is a communication protocol, based on the IEEE 802.15.1 standard, which is as commonly available as cellular mobile communications due to its inclusion on most mobile phones, PDAs and laptop computers. In theory, Bluetooth[®] localisation is capable of accuracy similar to that of WLAN. In practice, however, this is not possible since Bluetooth[®] devices do not have RSSI resolution as high as WLAN. Due to Bluetooth's use of power control in an attempt to maintain consistent received signal strengths over increasing distances (Bluetooth[®] Special Interest Group, 2001), RSSI is typically a very poor indicator of distance (Bielawa, 2005). Hence, the accuracy of the triangulation techniques highlighted in Table 2.4 is very dependent on the implementation of the Bluetooth[®] specification in a particular device. In reality, high RSSI resolution is only

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Roos et al. (2002a)	✓						C	A	L	64 APs in a 30km area	279m mean simulation error
Schwaighofer et al. (2004) (GPPS)	✓		✓				C	A	L	15 DECT (small scale indoor GSM) APs in a 250m x 180m area	7.5m mean error
Orooji and Abolhassani (2005)	✓						C	A	L	7 APs in a simulated outdoor environment	Less than 61m simulation error 67% of the time
Otsason et al. (2005)			✓				C	A	L	6 strongest cell towers detectable at any indoor location	3.4m median error
Lamarca et al. (2005) (Place Lab)		✓			✓		C	A	L	Any single strongest detectable cell tower	107.2m median error
Google Mobile Maps "My Location"	✓						C	C	L	Any single strongest detectable cell tower	No official accuracy claims
Path Intelligence FootPath TM		✓					C	C	R	Custom GSM receivers throughout a shopping centre to monitor foot traffic	No official accuracy claims

Table 2.3: Comparison of cellular network based localisation systems.

possible from the higher priced Bluetooth[®] chips. In fact most mobile phone Bluetooth[®] chips do not provide any RSSI information to the application layer. Hence, to estimate position locally on a typical mobile phone prior to the findings of this thesis, proximity was the only viable technique (see Cheung et al. (2006) for an example).

Proximity localisation also overcomes another restriction of the Bluetooth[®] protocol; namely connectivity. Unlike WLAN, it is difficult to obtain RSSI readings from Bluetooth[®] devices unless a connection is formed. There is also the restriction that a device can only connect as the master to 7 slave devices and a device can only be the slave to a single master. This means that if a remote device connects as the master to a number of APs, no other remote devices will be able to connect to those APs as a master at the same time, allowing only one device's location to be processed at a time. Recent work by Rodas et al. (2008) has demonstrated success with obtaining RSSI without forming connections by using the Linux platform and a specific Bluetooth[®] adapter. However, this type of system configuration is not particularly common, meaning dedicated computers must be used for APs.

The limited connectivity of the Bluetooth[®] protocol is one reason why minimal AP localisation is the focus of this PhD thesis. Using a single AP allows the system to remotely track the location of up to 7 subjects in a particular environment without the connectivity issues associated with multiple AP Bluetooth[®] localisation. This allows the use of Bluetooth[®] to track subjects while still being more power conservative than WLAN. Another low power communication protocol which has been used for localisation in the past is the IEEE 802.15.4 protocol.

2.3.4 IEEE 802.15.4/ZigBee[®]

The IEEE 802.15.4 standard describes a wireless communication protocol for low power consumption, low data rate communications. As such, the ZigBee[®] communication protocol, which builds on the IEEE 802.15.4 specification is intended to be even more power conservative than Bluetooth[®], while still providing some of the same functionality. It also experiences less connectivity restrictions than Bluetooth[®]. However, in spite of its disadvantages, Bluetooth[®] is a protocol more commonly available in mobile devices due to its earlier inception⁴. As a result IEEE 802.15.4 and ZigBee[®] localisation generally requires dedicated devices which would not otherwise exist within a given environment.

Due to IEEE 802.15.4's high RSSI resolution, triangulation is the most commonly employed localisation technique (see Table 2.5). Even though IEEE 802.15.4 has RSSI resolution higher than WLAN devices, its use in localisation systems is not as common. The main reason for this is the availability of IEEE 802.15.4 devices. Since IEEE 802.15.4

⁴The original Bluetooth[®] specification was developed in 1994, whereas the IEEE 802.15.4 standard was completed in 2003 and the ZigBee[®] specifications were completed in 2005.

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Chen and Kobayashi (2002)	✓						C	A	R	5 APs in a 900m ² area	Best simulation error of 1.41m
Anastasi et al. (2003) (BIPS)	✓						S	A	R	One Bluetooth [®] enabled computer per symbolic location	Mobile devices will be discovered in a location within 4 seconds 95% of the time
Feldmann et al. (2003)	✓						C	A	R	3 APs in a room of area 46m ²	2.06m mean error
Kotanen et al. (2003)			✓				C	A	L	3 APs in a room of area 96m ²	3.76m mean error
Castano et al. (2004)	✓						C	A	L	3 APs in a room of area 225m ²	3m mean positioning error, increasing to 12m when there is a body in the LoS
Genco (2005)	✓						C	A	L	16 APs in a castle tour environment	Genetic Algorithm simulation yields an optimal accuracy of 37.5cm
Cheung et al. (2006)	✓						C	A	L	One Bluetooth [®] beacon (\$20) per symbolic location	Mobile device can discover location beacons in 1.5 - 6 seconds
Rodas et al. (2008)				✓			C	A	R	4 APs in a 60m ² room	Mean simulation error of 1m

Table 2.4: Comparison of Bluetooth[®] based localisation systems.

APs rarely already exist in typical deployment scenarios, dedicated IEEE 802.15.4 APs must be installed for localisation. However, due to the similarities in resolution between IEEE 802.15.4 and WLAN, and the ubiquity of WLAN APs, WLAN is usually the more practical choice. Hence all of the ZigBee[®] localisation work to date is purely academic, as evident in Table 2.5. Furthermore, due to IEEE 802.15.4's relative immaturity, there are very few commercial devices with computation and IEEE 802.15.4 communication functionality integrated into a portable form-factor, hence, remote localisation is generally employed in practice.

Half of the IEEE 802.15.4 localisation techniques surveyed in Table 2.5 use triangulation to convert distance estimates to location estimates, even though distance estimates based on RSSI generally result in poor localisation accuracy. One localisation technology which achieves extremely high localisation accuracy using triangulation techniques and dedicated hardware is Ultrasonic.

2.3.5 Ultrasonic

As presented in Section 2.2.1.2 inaccurate distance estimates caused by multi-path effects can largely be mitigated by using time-based estimates rather than signal intensity-based estimates. For this reason ultrasonic positioning techniques can achieve extremely high localisation accuracy by using triangulation (see Table 2.6). This higher accuracy comes at the cost of requiring dedicated hardware throughout the deployment environment. Because of its extremely high accuracy levels, the Active Bat system has been used in Madhavapeddy and Tse (2005) to illustrate how LQ varies throughout an indoor environment for a Bluetooth[®] chip which is equivalent to the one utilised in this research. However, the high levels of accuracy in that paper were only possible because the environment already had the Active Bat system installed. When such hardware does not already exist within the environment it is difficult to justify the cost and effort of installation for such limited tests. Hence, the tests performed later in this thesis (see Sections 4 and 5) do not use an accurate localisation technique such as ultrasonic. Instead, generation of labels from manual voice annotation and using an RFID localisation system is considered.

Ultrasonic localisation systems achieve their high resolution location predictions using expensive custom hardware. This accuracy is higher than necessary for many situations. When lower resolution location predictions are appropriate, cheaper custom localisation technologies such as Infrared or Radio Frequency Identification can be employed.

2.3.6 Infrared

Infrared (IR) localisation techniques use the reception of IR light to identify the presence of a subject in a particular location. There are two types of IR technologies used

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Chen et al. (2005)	✓						C	A	R	Unspecified home environment	Unspecified accuracy
Sugano et al. (2006)	✓						C	A	R	AP density of 0.27 nodes/m ²	Best error of 1.5m
Lorincz and Welsh (2006) (MoteTrack)		✓					C	A	R	23 APs in a 1742m ² area	Less than 1.6m error 80% of the time
Ren et al. (2007)					✓		C	A	L	10 APs in 930m ² environment	Mean simulation error of 0.6m
Kelly et al. (2008d)			✓				S	A	R	1 AP in a 4 room section of a home environment	86% correct room prediction
Tarrio et al. (2008)		✓					C	A	R	6 APs in a 4m ² area	Average error of 0.5m

Table 2.5: Comparison of IEEE 802.15.4 based localisation systems.

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Ward et al. (1997) (Active Bat)	✓						C A R	C A R		16 APs in an area of 13m ²	Less than 14cm error 95% of the time
Hexamite Ultrasonic Positioning System	✓						C C R	C C R		3 APs per room	No official accuracy claims; likely similar to Active Bat
Priyantha et al. (2000) (Cricket)	✓						C A L	C A L		No discussion of deployment	No official accuracy claims; likely similar to Active Bat
Savvides et al. (2001)	✓						C A L	C A L		9 sensor nodes in a distributed localisation scenario	No official accuracy claims
Helal et al. (2003)	✓						C A R	C A R		4 Hexamite receivers in a laboratory simulation of a home environment	Worst case error of 22cm
Hou et al. (2007) (PAS)	✓						C A L	C A L		Unknown number of APs in a home environment	No official accuracy claims

Table 2.6: Comparison of Ultrasonic based localisation systems.

for localisation; active IR and passive IR. An active IR system has two parts an IR transmitter and an IR receiver. There are two possible topologies for such systems. The local positioning topology uses multiple transmitters throughout the environment, one per location, and one receiver on the subject to deduce location (e.g. Abowd et al. (1997)). Alternatively the remote positioning topology involves one transmitter worn by the subject and several receivers throughout the environment, one per symbolic location (e.g. Want et al. (1992)). Unlike any of the previously presented localisation systems, active IR techniques require direct LoS, hence they will not function while within someone's pocket or bag. On the other hand, the advantage of such positioning techniques is the relatively unsophisticated, hence cheap, hardware necessary to merely decode the ID number embedded in an IR signal.

For an even cheaper approach, passive IR (PIR) localisation systems do not involve any IR transmission components. PIR sensors in each room detect changes in IR radiation throughout the room which are indicative of human movements. Then the location of the movements are taken to be the location of the subject (e.g. Lundell et al. (2007)). The advantage of this type of approach is that tracking can occur without the need for the subject to carry a device, and PIR sensors are extremely cheap. The disadvantage is that the system only works in the case when there is only one subject in the environment due to PIR's inability to resolve the subject's identity.

IR detectors generally do not have received signal intensity readings, only binary presence readings. Hence, the proximity localisation technique is the only option for IR localisation systems, as can be seen in Table 2.7. Another limitation of IR localisation is the dependence of the devices on direct LoS reception.

2.3.7 RFID

Radio Frequency Identification (RFID) is a technology similar to IR in that it can communicate unique identification codes between tags and readers, however it is different in that it uses RF instead of IR electromagnetic radiation. This different spectrum usage means that RFID transmissions are more suited to penetrating obstacles allowing better non-LoS transmission and hence, reliability, than IR transmissions. Due to this superior reliability, RFID localisation techniques are experiencing more research interest in recent years than IR, which is evident when comparing Table 2.8 with Table 2.7.

There are two distinctly different types of RFID communication; active and passive. In active RFID communication an RFID tag can periodically broadcast identification packets or transmit only in response to a request from an RFID reader. Due to the powered nature of active RFID tags, their transmission can be anything up to 100 meters and can penetrate obstacles such as walls and furniture. Since active RFID tags are powered they can transmit at consistent power levels and useful signal intensity readings can be

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Want et al. (1992) (Active Badge)	✓						S	A	R	One infrared receiver per symbolic location	Position updates occur every 15 seconds
Want et al. (1995) (ParcTab)	✓						S	A	R	One IR receiver per symbolic location	Position updates occur every 30 seconds
Abowd et al. (1997) (Cyberguide)	✓						C	A	L	IR beacons deployed with a density related to intended accuracy	Position error is dependent on beacon deployment density
Versus Technology IR Tracking	✓						S	C	R	IR receivers deployed with a density related to intended accuracy	Maximum detectable range and maximum error of 15m
Lundell et al. (2007)	✓						S	A	R	One PIR sensor in each room. No device carried by subject	Reliable localisation is possible if only one subject is in the environment at a time since PIR is unable to resolve identity

Table 2.7: Comparison of IR based localisation systems.

estimated at the receiver. Hence active RFID localisation can utilise techniques such as triangulation (Hightower et al., 2001) or fingerprinting (Ni et al., 2003).

Passive RFID communication does not use powered tags. Instead the tags harvest power from the transmission of the RFID receiver when it requests a tag ID. This limited power is used to charge up the tag's circuitry allowing the tag to transmit its ID back to the reader. In this way passive RFID tags do not need to be powered leading to cheaper tags with lower installation and maintenance overheads. The downside to this is that passive RFID tags suffer lower transmission ranges and readers cannot deduce consistent, if any, RSSI information. There is also a high risk of false negatives which must be taken into account in the design of the localisation system. As a result of the limited resolution of the available readings, passive RFID localisation systems must employ the proximity localisation technique. Examples of these techniques can be found in Callaghan et al. (2006), Koch et al. (2007) and Kulyukin et al. (2008). Passive RFID proximity localisation systems can be used, not just to locate people or objects within an environment but also to predict an object's location relative to a person. For example Hou et al. (2007) uses RFID readers on a subject to predict which tagged objects the subject is interacting with and infer whether they have taken their medication.

The reliability of RFID localisation is evident in its frequent use across the world for transportation tolling scenarios. Active RFID systems are generally used for toll fee collection on motorways and passive RFID systems are generally used for pedestrian toll collection such as busses, subways and trams (Oberli and Landau, 2008). These systems could also fall into the category of proximity localisation techniques.

2.3.8 Combinations of Technologies

Naturally a localisation system is not restricted to using only one sensor or communication technology at a time. Signal readings from any number of available technologies can be combined to give one location prediction. Such a location prediction is usually equally, if not more accurate than if only one of the technologies were used. Table 2.9 summarises the localisation systems which use a combination of sensors. The types of sensors used are highlighted in the right of the table. Not included, due to space constraints, is the type of localisation technique each uses. All of the techniques surveyed use either fingerprinting or particle filtering techniques. There are two reasons why predictions based on multiple technologies are beneficial;

1. It generally makes a higher density of access points available for a given area. It has been shown that accuracy is proportional to the density of access points available in a given area (Kelly et al., 2009a), hence higher accuracy is the result.
2. It has been demonstrated that using multiple sensor technologies preserves the

System Reference	Proximity	Triangulation	Fingerprinting	Kalman Filtering	Particle Filtering	Discrete Bayes	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy
Hightower et al. (2001) (SpotON)	✓						C	A	R	Unspecified number of RFID receivers in an office environment	No official accuracy claims
Cox et al. (2003) (IntelliBadge TM)	✓						S	A	R	Unknown number of RFID readers in a conference venue	No official accuracy claims
Ni et al. (2003) (LAND-MARC)		✓					C	A	R	4 RFID readers and 16 active RFID reference tags in a 40m ² area	Less than 1.5m error 90% of the time
Callaghan et al. (2006)	✓						S	A	R	1 RFID reader in each symmetric location	No indication of accuracy or reliability
Koch et al. (2007)	✓						C	A	L	4000 passive RFID tags in a 60m ² area and one foot-mounted RFID reader	Position error is within 10cm
Kulyukin et al. (2008)	✓						C	A	L	4 passive RFID tag mats installed in layers and 1 RFID reader installed on an elder's walker	Mean error of 1.1m
Lightner and Erdogmus (2008)				✓			C	A	R	2 RFID readers and an active RFID tag worn by an elder	No indication of accuracy or reliability

Table 2.8: Comparison of RFID based localisation systems.

prediction accuracy of the more accurate technology while reducing the prediction uncertainty below the level of either technology alone (Hightower and Borriello, 2004).

Point 2 is well illustrated in Lamarca et al. (2005). There it is shown that WLAN has high accuracy but low coverage and GSM has low accuracy but high coverage. Alone each technology does not have compelling performance, but combined the overall performance of the system is extremely good, surpassing any single technology alone. This shows that sensor redundancy is the key to increasing overall performance in terms of accuracy and reliability. For this reason our work makes use of all RF signals which are available without any extra hardware.

2.4 Summary and Motivation

This chapter has presented a foundation for understanding the design and analysis of the localisation system developed in this thesis. The first section outlined the principles of radio propagation which predominantly effect the travel of radio waves through an environment, hence influence RF localisation accuracy. The following section illustrated how the behaviour of radio waves in an environment can be used in the prediction of the location of a mobile device. Finally, the last section gave a broad picture of the localisation systems developed to date which utilise these theories in their location predictions. To summarise this information, Figure 2.12 outlines how the appropriate localisation technique might be chosen for a given deployment scenario.

Chapter 4 will illustrate how our localisation system builds upon this foundation. However, before the implementation details of such a system can be presented, the mathematical foundations of our location prediction algorithms will be presented in the following chapter.

System Reference	Coordinate or Symbolic	Commercial or Academic	Remote or Local	Infrastructure	Best Accuracy	WLAN	Cellular	Bluetooth [®]	IEEE 802.15.4	RFID	Infrared	Ultrasonic
Pandya et al. (2003)	C	A	L	3 Bluetooth [®] APs and 3 WLAN APs in a 827m ² area	4.7m mean accuracy	✓		✓				
Liao et al. (2003)	C	A	L	73 Versus IR receivers and 3 Cricket ultrasonic receivers	1.76m mean error						✓	✓
Lamarca et al. (2005) (Place Lab)	C	A	L	Any available WLAN and cellular APs in an urban environment	13.4m mean error	✓	✓					
Widyawan et al. (2007a)	C	A	L	5 WLAN APs and 10 IEEE 802.15.4 APs in a 1606m ² office	0.97m mean error	✓			✓			
AeroScout	C	C	R	Uses WLAN fingerprinting. When an RFID choke point is detected it is logged to the server via WLAN	No official accuracy claims	✓				✓		
Kelly et al. (2008a)	S	A	R	Bluetooth [®] computer, 3 Bluetooth [®] beacons and cellular network signals	Long-term room prediction accuracy of 88%		✓	✓				
AeroScout Ultrasonic Positioning System	C	C	R	Uses WLAN fingerprinting. When an ultrasonic emitter is detected it is logged to the server via WLAN	No official accuracy claims	✓						✓

Table 2.9: Comparison of combination of technologies localisation systems.

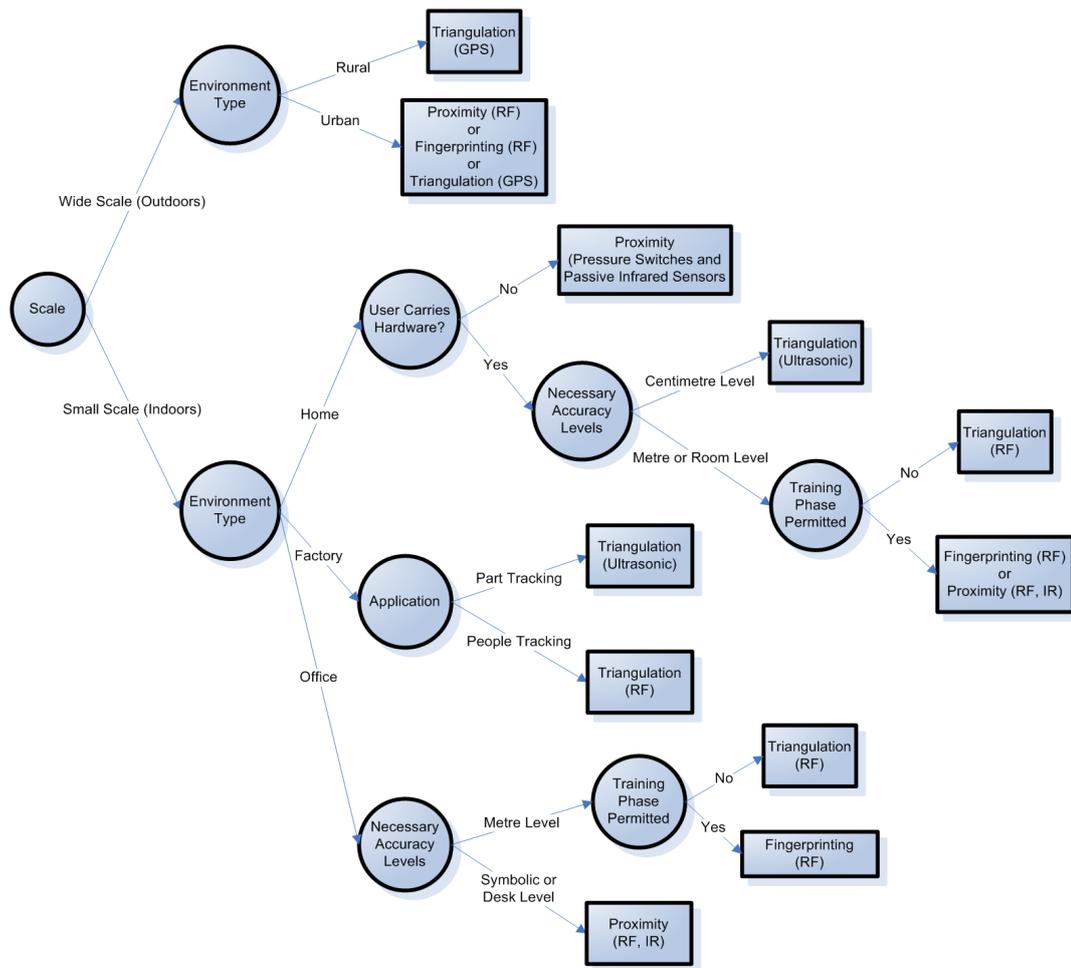


Fig. 2.12: Decision tree for selecting the appropriate localisation technique (and most likely technology in brackets) for a given deployment scenario.

Background Theory

As will be presented in Chapter 4, the aim of the localisation system developed as part of this thesis is to predict room-level location, due to the natural human interpretation of room location as opposed to coordinate location. Also, when considering Section 2.2.1.3 it is apparent that fingerprinting is the most suitable localisation technique for this system due to the difficulty inherent in mathematically modeling the available signals. Hence, it was decided to use classification techniques to translate the signals emanating from the hardware to categorical room-level location predictions.

A wide variety of pattern classification techniques have been developed to date, each with differing levels of accuracy under different modeling assumptions and computational costs. This chapter presents the background theory and implementation details of a variety of probabilistic and non-probabilistic classification techniques, considered appropriate to our particular localisation problem. Figure 3.1 outlines the implemented classification techniques. There are three sections to the classification framework. The first section is the preprocessing (or feature extraction) stage. The data to be inputted into the classifiers will be either raw data, filtered data, time-lagged¹ or a combination of filtered and time-lagged. Once the data has been pre-processed it can either be fed directly into a non-probabilistic technique such as k -Nearest Neighbour or Support Vector Machines (SVMs) or used for density estimation in a probabilistic classifier such as Naive Bayes, Discriminant Analysis, etc. If a probabilistic classifier approach is employed, the probability of each class given the measurements must be estimated using a density estimator, then these probabilities are fed into a decision function such as a Maximum Likelihood Class Decision or a Hidden Markov Model Decoder (see Chapter 6). The focus of this work is on which combination

¹Similar to context sensitive classification in speech processing, whereby each feature is expanded to include the previous f samples in that feature

of blocks, one from each column, results in the most accurate localisation. This chapter is concerned primarily with the theory and implementation of the shaded blocks in Figure 3.1. The first classification algorithm, considered due to its simplicity and relatively high accuracy, is k -Nearest Neighbour.

3.1 k -Nearest Neighbour

The k NN classifier, in its purest form, is a non-parametric, memory-based, algorithm. This means that it requires the entire training dataset, or a highly representative subset of the training dataset, to predict the class of a test sample. The training dataset consists of N sample vectors, denoted $\mathbf{x}_n = (x_n(1), x_n(2), \dots, x_n(d), \dots, x_n(D))$, where D is the number of features, or dimensions, of the dataset and $\{n \in \mathbb{R} : 1 \leq n \leq N\}$. Each training vector has a corresponding categorical class label, R_k . When a test vector, \mathbf{x}' is obtained, it is evaluated for similarity with every single vector in the training dataset. Any measure of similarity can be employed; the most popular being the Euclidean distance measure,

$$e_n = \sqrt{(x_n(1) - x'(1))^2 + \dots + (x_n(d) - x'(d))^2 + \dots + (x_n(D) - x'(D))^2}. \quad (3.1)$$

This expression must be evaluated for every sample in the training dataset. Then the list of the k most similar vectors can be populated. Class decision is based on a majority vote of the classes of these k vectors. Since k NN can use such large quantities of data for its classifications, it is a highly flexible non-linear classifier. This flexibility comes at the cost of requiring long processing times, proportional to the training dataset size (N) and dimensionality (D). Its generalisation ability and decision region flexibility are controlled by the k parameter, hence k is the most important parameter to be selected. For example, if k is low the classifier will specialise heavily on the training data, leading to misclassifications. The greater the value of k , the greater the classifier's generalisation ability. However, too much generalisation can also lead to misclassifications. Figure 3.2 illustrates how selection of k influences the classifier's generalisation ability. k must be selected carefully to optimise classification accuracy, usually using cross-validation on a second dataset.

k NN has been used in the past to generate location predictions from RF signals. Bahl and Padmanabhan (2000b) proposed approximating a device's coordinate position by the mean of the positions of the k nearest training samples. That technique simply extends the base k NN algorithm by substituting the majority class vote by a mean of the position of the k nearest training samples. Ofstad et al. (2008) used k NN classifiers to determine a person's location using data from a phone's built-in GPS and accelerometer sensors with higher accuracy than if only GPS data was available. Mantoro and Johnson (2005)

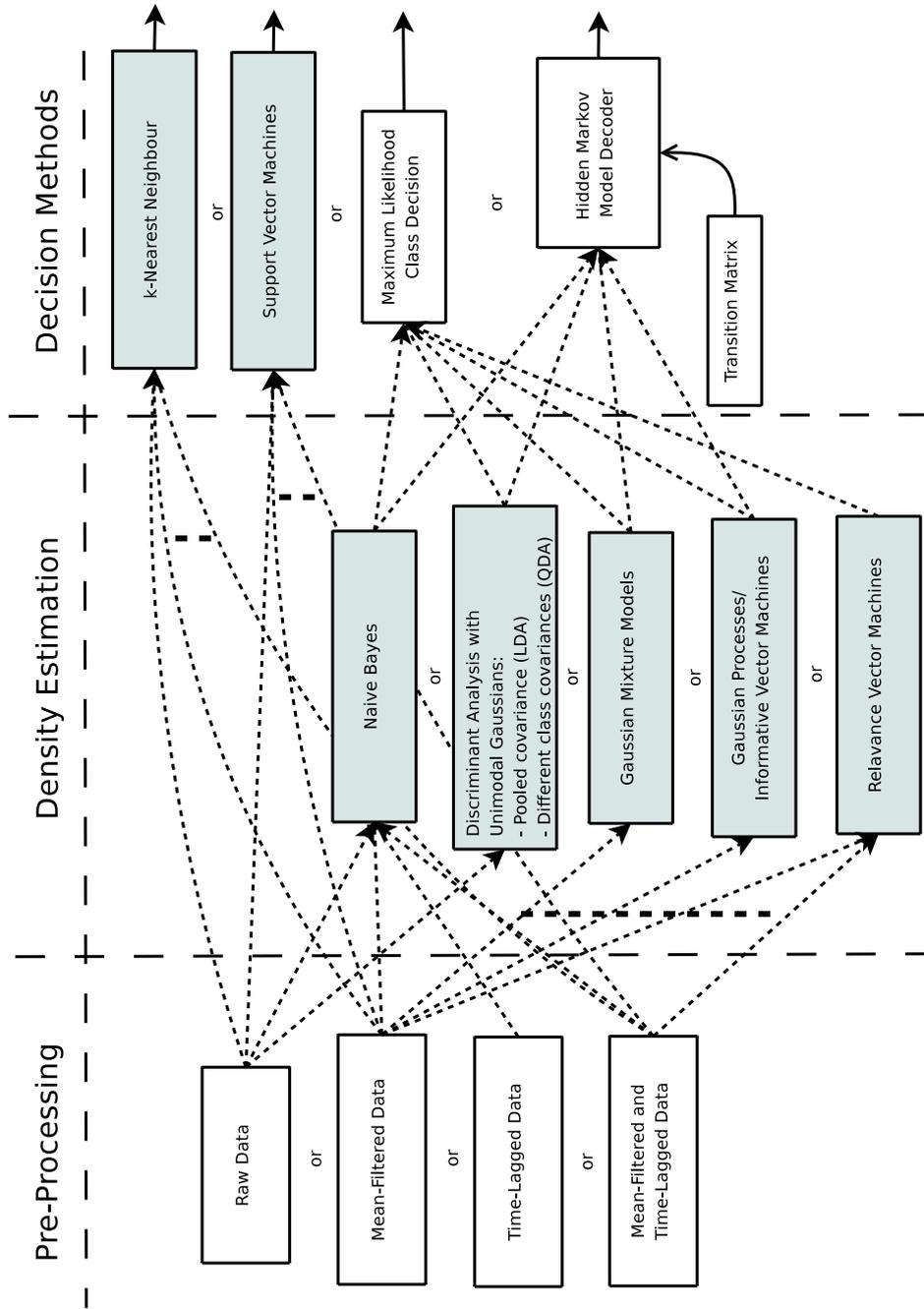


Fig. 3.1: Implemented location classification framework.

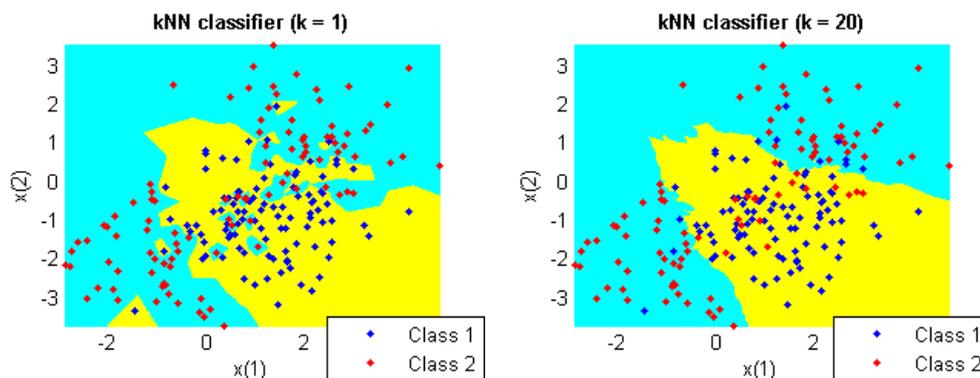


Fig. 3.2: Effect of varying k parameter on classification region flexibility.

proposed using the k NN classifier to predict symbolic location, also based on WLAN technology. That work is similar to ours in that we are interested in room-level location, which can be considered a symbolic location tracking problem. Our work is different in that we are focused on predicting the location of a subject from small quantities of test data rather than several hours of data collected from a stable unmoving device. Mantoro and Johnson (2005) found the optimal value of k to be 10 for the indoor localisation problem. A value of $k = 10$ is also employed for these experiments.

In spite of k NN's highly flexible classification regions it has some disadvantages. Firstly it is a non-probabilistic classification technique which means that it will not allow rejection of uncertain predictions or integration into a Hidden Markov Model filtering framework without modifications to the algorithm. A second disadvantage of k NN is its high computational load, particularly for large datasets. There are sparse approximations to the k NN algorithms which reduce the memory requirements (Kuncheva, 2004) and search algorithms which speed up execution time (Friedman et al., 1975). However, these techniques result in equal or lower accuracy than the original k NN algorithm. For this reason it was decided to consider other classifiers which natively operate on sparse representations of the training data.

3.2 Support Vector Machines

k NN estimates the class of a test sample from the classes of the most similar training samples. Support Vector Machines (SVMs), on the other hand, create decision boundary hyperplanes between two classes which maximise the margin between the hyperplane and the vectors perpendicularly closest to the hyperplane. These perpendicularly closest vectors are referred to as the support vectors, since they are the vectors on which the classifier is built. The hyperplane is given by the equation

$$\mathbf{w}^T \mathbf{x} + b = 0, \quad (3.2)$$

where \mathbf{w} is the normal vector and b is the hyperplane bias. In the simple binary case, the decision function will be:

$$D(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b). \quad (3.3)$$

Since we are interested in the distance between the decision hyperplane and the support vectors on either side of the hyperplane, two additional hyperplanes are defined which pass through the calculated support vectors,

$$\mathbf{w}^T \mathbf{x} + b = +1 \quad (3.4a)$$

$$\mathbf{w}^T \mathbf{x} + b = -1. \quad (3.4b)$$

If a support vector on the positive side of the hyperplane, \mathbf{x}_1 , and another on the negative side, \mathbf{x}_2 , are considered, equation (3.4b) can be subtracted from equation (3.4a) to get

$$\mathbf{w} \cdot (\mathbf{x}_1 - \mathbf{x}_2) = 2. \quad (3.5)$$

If the distance between these vectors $(\mathbf{x}_1 - \mathbf{x}_2)$ are projected onto a vector normal to the hyperplane, $\mathbf{w}/\|\mathbf{w}\|$, the result is,

$$\frac{\mathbf{w}}{\|\mathbf{w}\|} \cdot (\mathbf{x}_1 - \mathbf{x}_2). \quad (3.6)$$

Comparing this with equation (3.5) implies that the distance between the hyperplanes crossing the support vectors is given by $2/\|\mathbf{w}\|$. Hence, to obtain maximum support vector separability, this margin must be maximised. This is difficult to maximise since it depends on $1/\|\mathbf{w}\|$ which involves a square root. Maximising $1/\|\mathbf{w}\|$ is the same as minimising $\|\mathbf{w}\|^2/2$, which can be more readily solved. Since we also want to define the margin such that no points lie within the margin, the optimisation is

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.7)$$

subject to the constraint,

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \quad 1 \leq i \leq N. \quad (3.8)$$

It can be observed that the b term is ignored in this optimisation. However, it can be noted that b is compensated for implicitly by the constraints (Bishop, 2006) and does not need to be a subject of optimisation. This optimisation, subject to constraints, can be realized by introducing Lagrange multipliers. Each Lagrange multiplier, $\alpha_i \geq 0$, is multiplied by

the constraint in Equation 3.8 to obtain the Lagrangian optimisation;

$$\min_{\mathbf{w}, b, \alpha} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i \{y_i(\mathbf{w}^T \mathbf{x}_i - b) - 1\} \right\} \quad (3.9)$$

The minus term indicates that this is a minimisation with respect to \mathbf{w} and b and a maximisation with respect to α . This optimisation is subject to the constraints

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \quad (3.10)$$

$$0 = \sum_{i=1}^N \alpha_i y_i. \quad (3.11)$$

This optimisation problem can be expressed in *dual form* as

$$\max_{\alpha_i} \left\{ \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \right\}, \quad (3.12)$$

subject to

$$\alpha_i \geq 0 \quad (3.13)$$

$$\sum_{i=1}^n \alpha_i y_i = 0, \quad (3.14)$$

See Bishop (2006) for more details. This equation is easily solved using quadratic programming. The array of Lagrange multipliers indicates which vectors are in fact support vectors. All the Lagrange multipliers for which $\alpha_i > 0$ correspond to support vectors. Once the Lagrange multipliers have been found, the weight vector in Equation 3.3 can be calculated as

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i, \quad (3.15)$$

hence, binary classification can take place using the decision function,

$$D(\mathbf{x}') = \text{sign} \left[\sum_{i=1}^N \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x}' + b \right] \quad (3.16)$$

This expression, however, only produces linear decision hyperplanes. SVMs can be extended to perform nonlinear classification by using a kernel function to translate inner products $\mathbf{x}_i \cdot \mathbf{x}_j$ to a higher dimensional space. A hyperplane which is linear in this high dimensional space is equivalent to a nonlinear hyperplane in the original low-dimensional

space. The translation is defined as

$$K(\mathbf{x}_i, \mathbf{x}_j) \rightarrow \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) \quad (3.17)$$

and can be implicitly carried out using a kernel function which satisfies Mercer's condition. To satisfy Mercer's condition, the kernel function must satisfy the requirement,

$$\int K(\mathbf{x}_i, \mathbf{x}_j) \phi(\mathbf{x}_i) \phi(\mathbf{x}_j) d\mathbf{x}_i d\mathbf{x}_j \geq 0 \quad (3.18)$$

The kernel function used in this application is a Gaussian radial basis function (RBF),

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right), \quad (3.19)$$

where σ is the kernel width, one of the parameters which must be determined during the training phase.

Even with a non-linear classifier it is still possible to find classes which are not completely separable due to intermingling of samples. To permit some misclassifications and allow better generalisation performance, a slack variable is introduced. Instead of insisting on the constraint that every sample falls on or outside of the margin on the correct side of the hyperplane, the constraint in Equation 3.8 can be relaxed to include a slack parameter ε_i

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \varepsilon_i, \quad (3.20)$$

permitting some misclassifications to occur. The original minimisation can then be modified to take this slack parameter into account. This means that points which lie on the wrong side of the margin are now only softly penalised through the optimisation

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \varepsilon_i, \quad (3.21)$$

subject to the constraint in Equation 3.20.

Hence, the parameter C can be thought of as a penalty on errors. The higher it is, the greater cost an error will have on the optimisation process. The smaller it is, the more classification errors are permitted, allowing a smoother classification boundary. C is the second parameter which must be chosen during the training phase. Figure 3.3 illustrates how the C parameter permits smooth classification borders and avoids over-specialising on individual vectors.

The SVM algorithm is implemented using the simpleSVM package for Matlab[®] (Loosli, 2008). The parameter values σ and C are determined using a grid-search optimisation method to find the best validation accuracy. Validation accuracy is determined using 5-fold

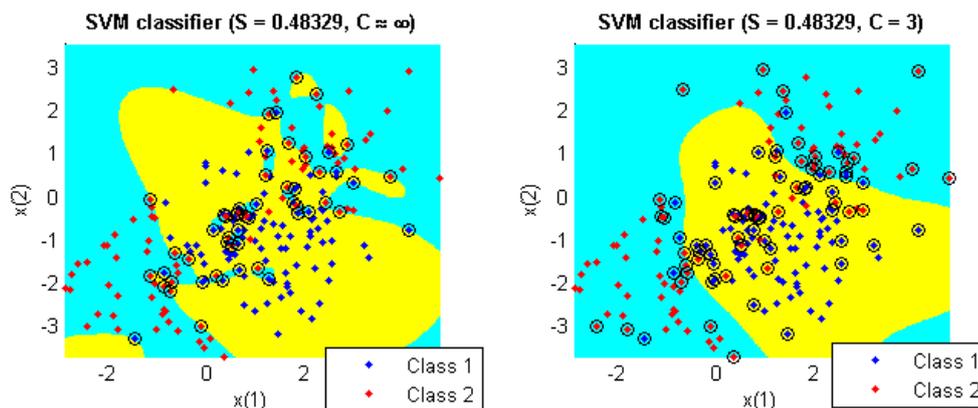


Fig. 3.3: Impact of the slack parameter, C , on classification flexibility. A slack parameter of ∞ is equivalent to permitting no misclassifications, causing overfitting. The optimum value of C for this dataset, 3, results in smooth classification borders. The calculated support vectors are circled.

cross-validation on the training data, whereby the accuracy is a mean of the classification accuracies across all 5 folds.

From Figure 3.3 it can be seen that SVMs are able to achieve a sparse representation of the training data by performing classification using only the support vectors (circled). After the selection of the optimal σ and C parameters and the calculation of the support vectors, classification is relatively fast compared to k NN. In spite of the high decision region flexibility and sparse representations possible with SVMs, they still do not permit probabilistic inferences. To allow us to exploit the advantages probabilistic classifiers have over non-probabilistic classifiers (Lawrence et al., 2003), a number of classifiers which base their class decision functions on class posteriors were considered. The simplest such classifier is a Naive Bayes Classifier.

3.3 Naive Bayes Classifier

A Naive Bayes Classifier (NBC) is a maximum likelihood classifier which uses Bayes' Theorem to estimate the likelihood of each class given some measurements. Given a single test measurement, x , the probability of the current class being R_k is denoted $P(R_k|x)$. Calculating this conditional probability becomes mathematically intractable when x can take on a large number of values. It is far easier to calculate $P(x|R_k)$ using a probability density function obtained from the training data for class R_k . Then the posterior probability of the class R_k given the observation x can be calculated using Bayes' rule;

$$P(R_k|x) = \frac{P(x|R_k)P(R_k)}{P(x)}, \quad (3.22)$$

where $P(R_k)$ is the prior probability of class R_k . In the absence of any other information $P(R_k)$ is assumed to be equal across all classes. The probability is then normalised by $P(x)$ which can be calculated over all classes as

$$P(x) = \sum_{k=1}^K P(x|R_k)P(R_k), \quad (3.23)$$

where K is the number of classes. The class decision function can then be based on the class of maximum probability,

$$D(x) = \operatorname{argmax}_k (P(R_k|x)). \quad (3.24)$$

This single input classifier can be extended into the multi-input case by calculating $P(R_k|\mathbf{x}) = P(R_k|x(1), x(2), \dots, x(d), \dots, x(D))$, where D is the number of features or the dimensionality of the input space. Similar to Equation 3.22 this can be written as

$$P(R_k|x(1), x(2), \dots, x(d), \dots, x(D)) = \frac{P(x(1), x(2), \dots, x(d), \dots, x(D)|R_k)P(R_k)}{P(\mathbf{x})}. \quad (3.25)$$

The joint probability $P(x(1), x(2), \dots, x(d), \dots, x(D)|R_k)$ requires considerable effort to calculate, especially for high dimensions. This calculation can be greatly simplified by assuming conditional independence of the measurements. If $x(1), x(2), \dots, x(d), \dots, x(D)$ are assumed to be independent given R_k , then their joint probability can simply be written as a product of their individual conditional probabilities, that is;

$$P(R_k|x(1), x(2), \dots, x(d), \dots, x(D)) = \frac{P(R_k) \prod_{d=1}^D P(x(d)|R_k)}{P(\mathbf{x})}. \quad (3.26)$$

Again $P(\mathbf{x})$ is simply a normalising term which can be calculated, using the same conditional independence assumption, as

$$P(\mathbf{x}) = \sum_{k=1}^K P(R_k) \prod_{d=1}^D P(x(d)|R_k) \quad (3.27)$$

In this work the probability $P(x(d)|R_k)$ is approximated by a unimodal Gaussian. Hence the distribution for a given feature in a given class is entirely parameterised by its mean and variance. The mean and variance for feature d in class k is calculated from training data using the expressions;

$$\mu_k(d) = \sum_{g_i=k} x_i(d)/N_k, \quad (3.28)$$

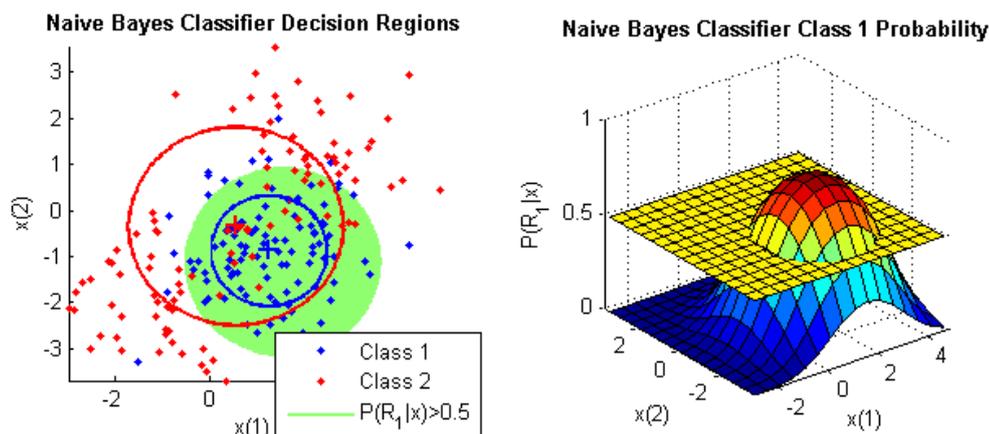


Fig. 3.4: The left figure illustrates decision regions along with the means and 50% confidence interval ellipses for each class. The right figure illustrates the probability of class 1 given the measurements. Note that the probability of class 1 never reaches 1 because it is normalised to ensure $P(R_1|\mathbf{x})$ and $P(R_2|\mathbf{x})$ sum to 1.

$$\sigma_k^2(d) = \sum_{g_i=k} (x_i(d) - \mu_k(d))(x_i(d) - \mu_k(d))^T / (N_k - 1), \quad (3.29)$$

where $g_i = k$ indicates the index i which corresponds to samples belonging to class k . With these parameters, the measurement probability of a new sample, $x'(d)$, can be calculated online from the univariate Gaussian distribution

$$P(x'(d)|R_k) = \frac{1}{\sqrt{2\pi\sigma_k^2(d)}} \exp\left(\frac{-(x'(d) - \mu_k(d))^2}{2\sigma_k^2(d)}\right). \quad (3.30)$$

Thus the probability of a set of such samples, \mathbf{x} , can be calculated from Equation 3.26. Figure 3.4 illustrates the decision regions for which $P(R_1|x(1), x(2)) > P(R_2|x(1), x(2))$ (or $P(R_1|x(1), x(2)) > 0.5$) on the same illustrative data used in Figures 3.2 and 3.3.

The assumption of independence necessary for Equation 3.26 is not entirely true, particularly in this work where a number of the signals are obtained from the same hardware and are influenced by some of the same mechanisms. However, Bayes classifiers with this naive assumption have been shown to have unexpectedly high performance in spite of this simplifying assumption, even when the inputs are not entirely independent (Domingos and Pazzani, 1996).

There are several advantages to NBCs such as the availability of probabilistic outputs and its suitability to high dimensional problems where joint density estimation is computationally intractable. The independence assumption also allows each input feature/dimension to be modeled using arbitrary probability density functions, which do not necessarily need to be unimodal Gaussian distributions. For example, in the past

nonparametric kernel density estimation (John and Langley, 1995) and spline-interpolated histograms (Kelly et al., 2008a) have been used to independently model the density of each feature. A disadvantage of NBC is that it produces covariance ellipses whose major and minor axes lie along the feature axes (i.e. $x(1)$ and $x(2)$ axes in Figure 3.4), which are not appropriate when the inputs are not conditionally independent Gaussian. In an effort to mitigate the potentially over-simplified distributions inherent in NBC, other more sophisticated probabilistic classification techniques were considered. One such classifier is Discriminant Analysis.

3.4 Discriminant Analysis

Like NBC, discriminant analysis classifiers perform maximum likelihood classifications using Bayes' Theorem. The main difference is that Equation 3.22 is modified to estimate class likelihood using a joint density estimate of the vector of measurements, \mathbf{x} , instead of the product of the individual measurement probabilities,

$$P(R_k|\mathbf{x}) = \frac{P(\mathbf{x}|R_k)P(R_k)}{\sum_{k=1}^K P(\mathbf{x}|R_k)P(R_k)}. \quad (3.31)$$

The denominator is simply a normalising term to ensure all class probabilities for a given measurement sum to 1 and can be ignored here for simplicity. The measurement density $P(\mathbf{x}|R_k)$ can be estimated in a number of ways. In discriminant analysis it is approximated by a multivariate Gaussian density of the form,

$$P(\mathbf{x}|R_k) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right), \quad (3.32)$$

where $\boldsymbol{\mu}_k$ is now a $D \times 1$ vector, Σ_k is a $D \times D$ covariance matrix and $|\Sigma_k|$ is the determinant of the covariance matrix. The denominator in Equation 3.33 is constant across all classes and can be ignored. Hence, the relative probability of class k can be expressed, using the log-likelihood, as,

$$\delta_k(\mathbf{x}) = \log\left[P(R_k) \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right)\right]. \quad (3.33)$$

By also ignoring the $2\pi^{\frac{D}{2}}$ term which is constant across all classes, this simplifies to,

$$\delta_k(\mathbf{x}) = \log(P(R_k)) - \frac{1}{2} \log(|\Sigma_k|) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k). \quad (3.34)$$

The formulation of the class covariance matrix, Σ_k , specifies whether the decision regions are linear or non-linear. For Linear Discriminant Analysis (LDA) it is assumed that all classes have a common covariance matrix (Hastie et al., 2001). Hence, by assuming

$\frac{1}{2} \log(|\Sigma_{\mathbf{k}}|)$ is constant across all classes, $\delta_k(\mathbf{x})$ can be simplified to

$$\delta_k(\mathbf{x}) = \log(P(R_k)) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_k), \quad (3.35)$$

$$\delta_k(\mathbf{x}) = \log(P(R_k)) + \mathbf{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k. \quad (3.36)$$

This function is linear in \mathbf{x} , hence the name LDA. Based on $\delta_k(\mathbf{x})$ the class with the highest probability of containing the measurement \mathbf{x} can be predicted using the discriminant function,

$$D(\mathbf{x}) = \arg \max_k (\delta_k(\mathbf{x})). \quad (3.37)$$

Similar to the Naive Bayes case, the mean of the Gaussian is calculated from the expression

$$\boldsymbol{\mu}_k = \sum_{g_i=k} \mathbf{x}_i / N_k \quad (3.38)$$

and the pooled covariance of the data across all classes is calculated from (Hastie et al., 2001);

$$\boldsymbol{\Sigma} = \sum_{k=1}^K \sum_{g_i=k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T / (N - K), \quad (3.39)$$

Due to the identical covariance ellipses across all classes, LDA has linear discriminant hyperplanes, which results in reduced discriminatory power for closely intermingled and non-Gaussian datasets (see Figure 3.5). Hence it is necessary to consider a more flexible classifier, Quadratic Discriminant Analysis (QDA). There are two ways to obtain non-linear discriminant hyperplanes for QDA. The first is to translate the inputs to a higher dimensional space using a polynomial, then perform LDA in this higher dimensional space. Another method is to use Gaussians with differing covariance matrices to represent each class. By permitting the covariance matrices in equation 3.34 to differ across classes, the simplifications in equation 3.36 do not occur. Instead the more complicated quadratic expression in Equation 3.34 must be used. This technique of obtaining quadratic discriminant regions is preferred to the polynomial technique since it does not require the optimal selection of polynomial order. It has been shown that the classification flexibility is similar for both polynomial and different-covariance Gaussian QDA (Hastie et al., 2001).

LDA is implemented in this work using the Matlab[®] statistics toolbox and QDA is implemented using Michael Kiefte's Discriminant Analysis Toolbox (Kiefte, 1999) since it safely handles singular covariance matrices. A comparison of the covariance ellipses and the corresponding decision regions can be found in Figure 3.5. It can be seen that using a

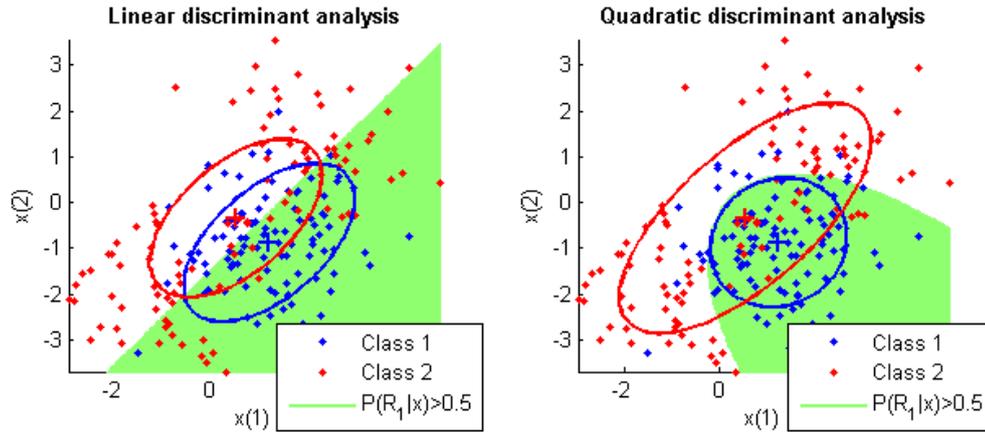


Fig. 3.5: Decision regions for LDA and QDA. The left figure indicates that the common covariance assumption is not valid for the illustrative dataset. QDA achieves more suitable decision regions by allowing a different covariance ellipse for each class.

different covariance ellipse allows higher discriminatory power. In spite of QDA's superior classification ability to that of the NBC it is also worth noting the similarity between QDA and the NBC. The QDA classification regions can be made identical to those of Naive Bayes by restricting $\Sigma_{\mathbf{k}}$ to be a diagonal matrix. Hence, inclusion of more realistic information about the covariance of the input signals contributes to classification regions more suited to the data.

A conclusion which can be drawn from the probabilistic classification algorithms presented thus far is that the classification accuracy is dependent on the accuracy of the underlying probability model. Generally LDA can achieve high accuracy by acknowledging that the input variables are jointly distributed and QDA achieves higher accuracy by acknowledging that the covariance model is different for different classes. Upon considering Figure 3.5 another deficiency of discriminant analysis can be observed. The illustrative data, like most real-life data, is not truly unimodal. Hence a multi-modal probability model should be considered. Gaussian Mixture Models use such a probability model.

3.5 Gaussian Mixture Models

As in the previous sections, the goal is to predict the class of maximum probability from the posterior probability $P(R_k|\mathbf{x})$. Again this can be computed from Equation 3.31. To allow for more complex datasets in which each class is not truly unimodal, $P(\mathbf{x}|R_k)$ can be represented by a multi-modal mixture of unimodal Gaussians. Figure 3.6 presents the graphical representation of such a Gaussian Mixture Model (GMM). From this figure it can be seen that the posterior measurement density for a given class, k , can be calculated from

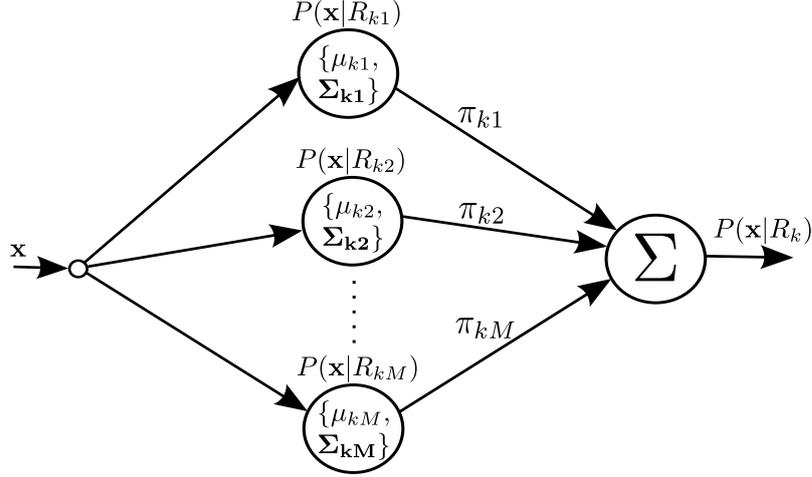


Fig. 3.6: Block diagram view of a GMM density estimator for \mathbf{x} in class k , modified from Reynolds and Rose (1995).

$$P(\mathbf{x}|R_k) = \sum_{m=1}^M \pi_{km} P(\mathbf{x}|R_{km}), \quad (3.40)$$

where M is the number of Gaussian components for a given class, R_{km} represents the m^{th} component in class k and π_{km} is the weighting of the m^{th} component in class k , such that $\sum_{m=1}^M \pi_{km} = 1$.

By choosing to have an independent covariance matrix for each GMM component, the measurement density for an individual component takes on the form

$$P(\mathbf{x}|R_{km}) = \frac{1}{(2\pi)^{\frac{D}{2}} |\boldsymbol{\Sigma}_{km}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{km})^T \boldsymbol{\Sigma}_{km}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{km})\right). \quad (3.41)$$

Hence, the Gaussian mixture density for a given class is entirely parameterised by the weights, mean vectors and covariance matrices for each component (Reynolds and Rose, 1995);

$$\lambda_k = \{\pi_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\}, m = 1, 2, \dots, m, \dots, M. \quad (3.42)$$

To fully specify the GMM for a given class, λ_k must be determined from the training data. The EM algorithm is typically used to estimate these parameters in GMMs. EM is a recursive algorithm for estimating the parameters in probabilistic models such as GMMs and Hidden Markov Models (HMMs). There are two steps to the EM algorithm; the expectation (E) step and the maximisation (M) step (Bishop, 2006). For GMMs the E-step uses the current parameter values in λ_k to estimate the responsibilities, or which Gaussian component is responsible for modelling which sample. This responsibility calculation is defined as (modified from Bishop (2006))

$$\gamma_k(z_{nm}) = \frac{\pi_{km}P(\mathbf{x}_n|R_{km})}{\sum_{j=1}^M \pi_{kj}P(\mathbf{x}_n|R_{kj})}, \quad (3.43)$$

where $\gamma_k(z_{nm})$ represents the probability $P(z_{nm} = 1|\mathbf{x}_n)$, which corresponds to the responsibility that component m takes for sample n (Bishop, 2006). With these newly assigned responsibilities of each component for each sample, the M-step can update the model parameters using the expressions

$$\mu_{km}^{new} = \frac{\sum_{n=1}^N \gamma_k(z_{nm})\mathbf{x}_n}{\sum_{n=1}^N \gamma_k(z_{nm})} \quad (3.44)$$

$$\Sigma_{km}^{new} = \frac{\sum_{n=1}^N \gamma_k(z_{nm})(\mathbf{x}_n - \mu_{km}^{new})(\mathbf{x}_n - \mu_{km}^{new})^T}{\sum_{n=1}^N \gamma_k(z_{nm})} \quad (3.45)$$

$$\pi_{km}^{new} = \frac{\sum_{n=1}^N \gamma_k(z_{nm})}{N} \quad (3.46)$$

With these updated model parameters, we return to the E-step and repeat the process. This recursive algorithm continues until the log likelihood of the training data, \mathbf{X} , being described by the model, λ_k ,

$$\ln P(\mathbf{X}|\mu, \Sigma, \pi) = \sum_{n=1}^N \ln \left[\sum_{m=1}^M \pi_{km}P(\mathbf{x}_n|R_{km}) \right], \quad (3.47)$$

no longer improves beyond some specified threshold value on each iteration. Naturally before any of the model parameters can be optimised, they must be initialised to some value. For the EM algorithm the number of Gaussian components per class must be specified and the parameters can be initialised either randomly or by using a more reliable technique. K-means clustering is generally used to initialise the mean parameters. The covariances can be initialised as the covariance of the data for each cluster and the component weights are initialised to be the ratio of the number of samples in the cluster to the total number of samples.

A disadvantage of the EM algorithm is that the number of components necessary for a given class must be known *a priori* or estimated using cross-validation. A significant level of effort must be devoted to determining the optimal number of mixtures, especially when the number of mixtures is different for each class, which is nearly always the case in real-life data. To overcome this limitation other algorithms have been proposed which allow the number of components necessary to represent the data to be determined during the optimisation process. Paalanen et al. (2006) describes two such algorithms. The first algorithm is the Figueiredo-Jain (FJ) algorithm. This algorithm starts out with a large number of components. A modified expression for the weight updates in Equation 3.44 is used, which sets weights to zero when they fall below a certain threshold. In this way the

component with a weight of zero does not contribute to the model and other components compensate by taking responsibility for describing its samples. A more detailed description of this algorithm can be found in Figueiredo and Jain (2000).

Paalanen et al. (2006) also proposes an alternative approach to EM and FJ called the Greedy EM (GEM) algorithm. The algorithm starts with a single component mixture per class. Then more components are iteratively added. The added component at each step is the one which is calculated to increase the log likelihood the most. After each component is added the model parameters are updated and the log likelihood is estimated and compared with the terminating log likelihood value. Paalanen et al. (2006) states that it is computationally easier to iteratively insert a component which increases the likelihood the most than initialising a near-optimal GMM. GMMs using all of these algorithms were implemented using the Matlab[®] package developed by Paalanen et al. (2006).

A comparison of the algorithms, applied to the illustrative dataset, can be found in Figure 3.7. The EM algorithm requires the selection of the number of mixtures per class. A value of $M = 2$ was chosen for this example. The GEM algorithm requires the selection of the maximum possible number of mixtures. A value of $M_{max} = 3$ was chosen here. The FJ algorithm did not require the selection of any parameters and gave a result similar to GEM for this data. The FJ algorithm is advantageous over the EM and GEM algorithms in that it does not require any knowledge of the underlying model used to generate the data. However, it has been empirically found that the EM and GEM algorithms result in higher accuracies on the practical home localisation data. This is due to FJ's aggressive approach to eliminating components. EM and GEM techniques achieve higher accuracy with the disadvantage that they require optimal selection of the maximum number of components. Fortunately, the determination of the optimal number of components and the parameters for each component occurs only once; during training phase. Hence, the more accurate EM and GEM algorithms shall be employed in practice, using cross-validation to estimate the most accurate technique and the number of components necessary. The GEM algorithm automatically permits differing numbers of components between classes. The EM algorithm would need to be modified to allow this, which would lead to an prohibitively large search space for each combination of component numbers for each class.

Thus far, a variety of fully parametric maximum likelihood Bayes classifiers have been presented. More recently a family of kernel methods have emerged which can produce highly representative non-parametric (or semi-parametric) probabilistic models even when the data is non-Gaussian. Next, these classifiers shall be presented and later considered for accuracy against fully parametric classifiers.

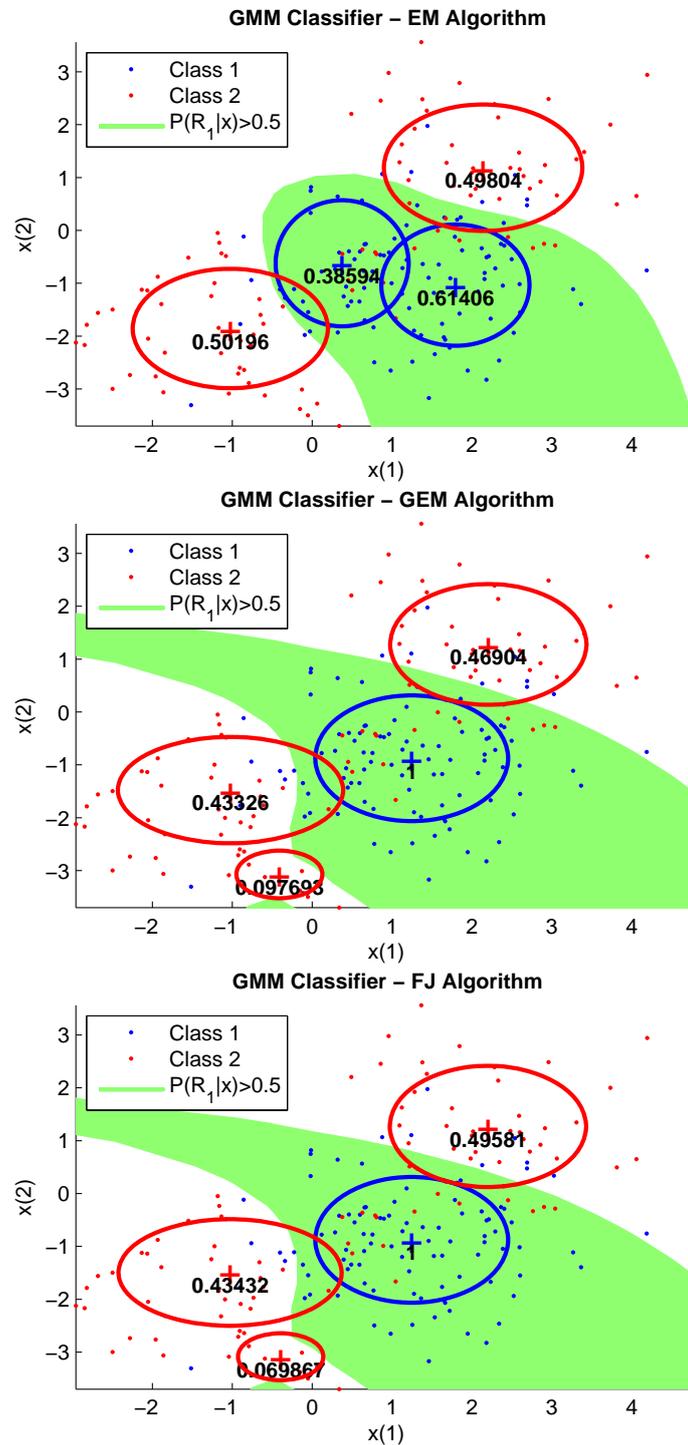


Fig. 3.7: Comparison of GMM classifiers optimised with EM, GEM and FJ algorithms. The weights of each component are indicated in black.

3.6 Gaussian Processes and Informative Vector Machines

Gaussian Process (GP) classification techniques are the result of adapting the non-parametric GP regression techniques to produce probabilistic estimates from the regression targets. With these probabilistic estimates, Bayesian classifications can take place as before. Hence, before GP classification can be considered, the theory of GP regression must first be presented.

GP regression allows the estimation of the mean and covariance of the output vectors based on the assumption of a joint multivariate Gaussian distribution on their values. In general it is assumed that the mean of a Gaussian process is zero (Rasmussen and Williams, 2005). Hence, the output vectors are related to each other by a covariance function which imposes a prior on their values;

$$\text{cov}(y_p, y_q) = \text{cov}(f(\mathbf{x}_p), f(\mathbf{x}_q)) = k(\mathbf{x}_p, \mathbf{x}_q). \quad (3.48)$$

However this covariance function does not allow for any measurement noise. To account for measurement noise a variance parameter, σ_n^2 , is introduced. The covariance function now becomes

$$\text{cov}(y_p, y_q) = k(\mathbf{x}_p, \mathbf{x}_q) + \sigma_n^2 \delta_{pq} \quad (3.49)$$

$$(3.50)$$

where δ_{pq} is the Kronecker delta which is one when $p = q$ and zero otherwise. When all the available training data is amalgamated in an $n \times D$ design matrix, X , this can be written as;

$$\text{cov}(\mathbf{y}) = K(X, X) + \sigma_n^2 I, \quad (3.51)$$

where $K = [K_{pq}]$ and $K_{pq} = k(\mathbf{x}_p, \mathbf{x}_q)$. Using this expression for covariance, the joint distribution of the training outputs, \mathbf{y} and the test outputs, \mathbf{y}_* , can be defined as

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{y}_* \end{bmatrix} \sim N \left(\mathbf{0}, \begin{bmatrix} K(X, X) + \sigma_n^2 I & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix} \right). \quad (3.52)$$

This joint distribution can be defined using any covariance function. In this work, the commonly employed Gaussian radial basis function is used (Rasmussen and Williams, 2005):

$$k(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \exp\left(-\frac{1}{2l^2}(\mathbf{x}_p - \mathbf{x}_q)^2\right) + \sigma_n^2 \delta_{pq}, \quad (3.53)$$

where σ_f^2 is the signal variance and σ_n^2 is the noise variance. l is the length scale which dictates how much of an influence observations distant from the test point will have on the prediction, i.e. the larger l is, the more of an influence distant points will have on the output for the test point (Ebden, 2008). As a result, the selection of these hyperparameters is important for achieving good model performance. If the model parameters are defined to be $\theta = \{l, \sigma_f^2, \sigma_n^2\}$ then their maximum *a posteriori* estimate occurs when $P(\theta|\mathbf{x}, \mathbf{y})$ is at its maximum. Ebden (2008) explains that this is equivalent to maximising $\log(P(\mathbf{y}|\mathbf{x}, \theta))$, given by

$$\log(P(\mathbf{y}|\mathbf{x}, \theta)) = -\frac{1}{2}\mathbf{y}^T K^{-1}\mathbf{y} - \frac{1}{2}\log|K| - \frac{n}{2}\log(2\pi) \quad (3.54)$$

using multivariate optimisation, where K refers to $cov(y_p, y_q)$ from Equation 3.51 and $|K|$ is the determinant of K .

Finally, using these parameters the covariance matrix can be populated and a prediction for a single test point, \mathbf{x}_* , can take place using the expressions

$$\bar{\mathbf{y}}_* = \mathbf{k}(\mathbf{x}_*) (K(X, X) + \sigma_n^2 I)^{-1} \mathbf{y}, \quad (3.55)$$

$$var(\mathbf{y}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}(\mathbf{x}_*)^T (K(X, X) + \sigma_n^2 I)^{-1} \mathbf{k}_*. \quad (3.56)$$

where $\mathbf{k}(\mathbf{x}_*)$ refers to the vector of covariances between the test point and the N training points (Rasmussen and Williams, 2005). Hence, it is now possible to estimate the mean and variance of a function entirely from training data and a Gaussian prior over the outputs.

As previously stated, the GP can be used to model probabilistic estimates for categorical data. This is achieved by using a GP model to generate a latent function with a Gaussian prior, which models how the likelihood of one class changes versus the other. This latent variable is then “squashed” into the range $[0, 1]$ using a logistic transfer function, $\lambda(\mathbf{f}_*) = 1/(1 + e^{-\mathbf{f}_*})$, to obtain the class probability. Hence, class probability inference takes place in two steps (Rasmussen and Williams, 2005). Given some test sample, the first step is to estimate the latent variable

$$P(\mathbf{f}_*|X, \mathbf{y}, \mathbf{x}_*) = \int P(\mathbf{y}_*|X, \mathbf{x}_*, \mathbf{f}) P(\mathbf{f}|X, \mathbf{y}) d\mathbf{y}. \quad (3.57)$$

Then this latent variable can be translated to a probabilistic prediction using the expression,

$$\lambda_* = P(\mathbf{y}_* = +1 | X, \mathbf{y}, \mathbf{x}_*) = \int \lambda(\mathbf{f}_*) P(\mathbf{f}_* | X, \mathbf{y}, \mathbf{x}_*) d\mathbf{f}_*. \quad (3.58)$$

Equation 3.57 is mathematically intractable for classification since the probability estimates now follow a Bernoulli distribution rather than a Gaussian distribution. Hence, the Laplace approximation method must be employed to perform these integrations. More details of its implementation can be found in Williams and Barber (1998) and Rasmussen and Williams (2005). Hence, binary classifications can take place based on the class of maximum probability specified by λ_* . The GP algorithm was implemented in Matlab[®] using the toolbox developed by Williams and Barber (1998). Non-binary classification was implemented using the “1 versus all” approach.

One significant issue with GPs is the large computational time and memory requirements necessary for most non-trivial datasets. For this reason a sparse GP approximation was considered. The Informative Vector Machine (IVM), developed by Lawrence et al. (2003) uses a subset of the most “informative” training data samples instead of the entire dataset. The informative vectors are chosen such that the decision regions for the subset of vectors is as similar as possible to the decision regions for the full dataset. A greedy approach is taken in which no vectors are initially active and vectors are iteratively added to the active set using what is referred to as a “differential entropy score”. This score chooses vectors whose inclusion minimises the predictive variance. More in-depth discussion of the implementation of this algorithm can be found in Lawrence et al. (2003). This sparse GP classifier was implemented in Matlab[®] using the toolbox developed by Lawrence et al. (2003).

Figure 3.8 shows the difference between full GP classification and the sparse approximations of IVM. IVM with only half of the available 200 samples active (Figure 3.8(b)) does not produce decision regions anything similar to the original GP. 125 active samples gives a rough approximation of the original and 130 active samples gives an almost perfect approximation of the original GP decision regions. This illustrates that IVMs accuracy is extremely dependent on the number of active vectors permitted. Lawrence et al. (2003) claims that IVMs can obtain accuracy similar to that of SVMs with far fewer active vectors. However, when comparing Figure 3.8(d) with Figure 3.3 it is clear that SVMs require far fewer active vectors (79) to achieve similar accuracy with this simple illustrative dataset. An advantage of IVMs, however, is that they produce probabilistic outputs.

The computation requirements of GPs makes them unsuitable for our localisation problem, which could potentially consist of several days worth of training data. IVMs provide somewhat of a solution to this limitation by using a subset of the training data. However, they will require optimal selection of the number of training samples retained, which will restrict their usefulness in a practical deployment. In response to this limitation, a sparse non-parametric probabilistic classifier which requires little or no

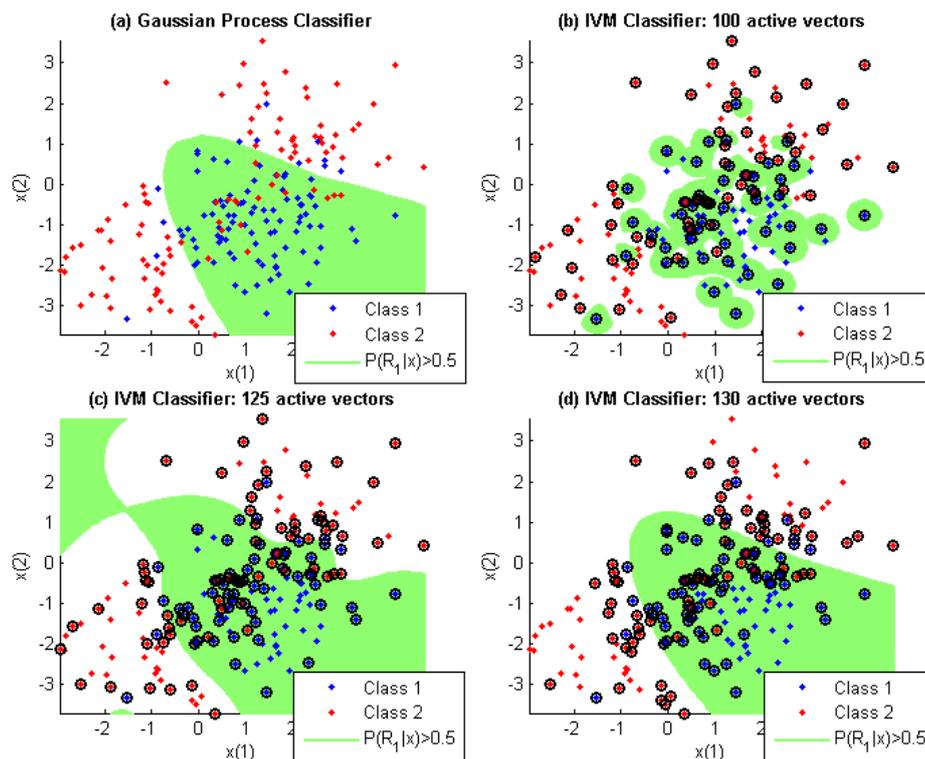


Fig. 3.8: A comparison of GP classification with IVM classification using 100, 125 and 130 active vectors. Active vectors are circled.

parameter selection is considered.

3.7 Relevance Vector Machines

The Relevance Vector Machine (RVM) is a machine learning technique which, like GPs, can produce a stochastic estimate of the target vector. As with GPs, RVMs can be used either for regression or for classification by adapting the regression outputs to produce class probabilities. Unlike GPs, RVMs produce these predictions by approximating the data by a linear combination of basis functions,

$$y(\mathbf{x}) = \sum_{n=1}^M w_n k(\mathbf{x}, \mathbf{x}_n) + b, \quad (3.59)$$

where b is a bias parameter, meaning that the predictions involve $M = N + 1$ parameters (Bishop, 2006). Hence, RVMs are similar to SVMs². RVMs use the weight vector, w_n , to encode which vectors are used in the predictive distribution; this is analogous to the Lagrange multiplier vector in SVMs. RVMs, however, impose a Gaussian prior over the weight vector. The weights of the vectors which do not conform to this assumption are

²RVMs have the advantage that the basis functions do not need to satisfy the Mercer condition

set to zero and are not included in the model. This weight prior is defined as (Tipping, 2000):

$$P(\mathbf{w}|\alpha) = \prod_{i=1}^N N(w_i|0, \alpha_i^{-1}), \quad (3.60)$$

where α_i is the precision of the weight, w_i , and $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_M)^T$. Bishop (2006) explains that when the evidence is maximised with respect to the hyperparameters, α , most of the hyperparameters go to infinity. Hence, the corresponding weights go to zero and have no contribution to the predictions, leading to an extremely sparse model.

Using Bayesian inference, Tipping (2001) explains how the posterior distribution of the weights can be estimated using the expression

$$P(\mathbf{w}|\mathbf{t}, \alpha, \sigma^2) = (2\pi)^{-\frac{M}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{w} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{w} - \boldsymbol{\mu})\right), \quad (3.61)$$

where $\mathbf{t} = (t_1, t_2, \dots, t_N)^T$ is the set of target vectors corresponding to the training inputs $X = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)^T$ and σ^2 is the noise variance. The posterior covariance and mean in this expression are defined to be

$$\boldsymbol{\Sigma} = (\sigma^{-2} \boldsymbol{\Phi}^T \boldsymbol{\Phi} + \mathbf{A}^{-1}) \quad (3.62)$$

$$\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \boldsymbol{\Phi}^T \mathbf{t}, \quad (3.63)$$

where $\mathbf{A} = \text{diag}(\alpha_1, \alpha_2, \dots, \alpha_M)$ and $\boldsymbol{\Phi}$ is the $N \times M$ design matrix with $\Phi_{nm} = k(\mathbf{x}_n, \mathbf{x}_m)$. Estimation of the necessary hyperparameters takes place by maximising the log marginal likelihood

$$\ln P(\mathbf{t}|\alpha, \sigma^2) = -\frac{1}{2}(N \ln(2\pi) + \ln |\mathbf{C}| + \mathbf{t}^T \mathbf{C}^{-1} \mathbf{t}) \quad (3.64)$$

with respect to the hyperparameters α and σ^2 , where $\mathbf{C} = \sigma^2 \mathbf{I} + \boldsymbol{\Phi} \mathbf{A}^{-1} \boldsymbol{\Phi}^T$.

There are two ways to optimise the hyperparameters (Tipping, 2000). The first way is to use EM. This is achieved by treating the model weights as the latent variables. This gives the estimate for α as

$$\alpha_m^{\text{new}} = \frac{1}{\Sigma_{mm} + \mu_i^2}. \quad (3.65)$$

The second approach is to differentiate Equation 3.64 with respect to α to give

$$\alpha_m^{\text{new}} = \frac{\gamma_m}{\mu_m^2}, \quad (3.66)$$

where $\gamma_m = 1 - \alpha_m \Sigma_{mm}$, which is a measure of how well the corresponding parameter, w_i ,

is determined by the data (Tipping, 2000). To estimate the noise variance, both methods lead to the expression

$$(\sigma^2)^{new} = \frac{\|\mathbf{t} - \Phi\boldsymbol{\mu}\|^2}{N - \sum_m \gamma_m} \quad (3.67)$$

When these update expressions are recursively evaluated, many of the hyperparameters, α_m , go to extremely large (or approximately infinite) values. Hence, using Equation 3.61 the corresponding weights, w_m , are calculated to be zero. Now prediction for a new input sample, \mathbf{x}_* , takes place using the expression (Tipping, 2001):

$$P(t_*|\mathbf{t}, \alpha_{final}, \sigma_{final}^2) = \int P(t_*|\mathbf{w}, \sigma_{final}^2)P(\mathbf{w}|\mathbf{t}, \alpha_{final}, \sigma_{final}^2)d\mathbf{w} \quad (3.68)$$

and since this is the convolution of two Gaussians, it can readily be evaluated to the form

$$P(t_*|\mathbf{t}, \alpha_{final}, \sigma_{final}^2) = N(t_*|y_*, \sigma_*^2), \quad (3.69)$$

where

$$y_* = \boldsymbol{\mu}^T \boldsymbol{\phi}(\mathbf{x}_*), \quad (3.70)$$

$$\sigma_*^2 = \sigma_{final}^2 + \boldsymbol{\phi}(\mathbf{x}_*)^T \boldsymbol{\Sigma} \boldsymbol{\phi}(\mathbf{x}_*) \quad (3.71)$$

and $\boldsymbol{\phi}(\mathbf{x}_*) = [1, k(\mathbf{x}_*, \mathbf{x}_1), k(\mathbf{x}_*, \mathbf{x}_2), \dots, k(\mathbf{x}_*, \mathbf{x}_N)]$. Bishop (2006) illustrates the difference in regression using RVMs and SVMs. It is shown that RVMs achieve similar accuracy using far fewer vectors. This means that predictions with RVMs have significantly lower computational cost. The disadvantage is that RVM training takes longer since it involves an optimisation of a non-convex function. This disadvantage is negligible for this work since we only need to train the classifier once; during the system calibration stage.

Similar to GPs, classifications can be obtained by specifying the RVM regression output, $y(\mathbf{x})$, to be a latent variable and translating the latent variable to a probabilistic prediction using the logistic function, $\lambda(y) = 1/(1 + e^{-y})$. Since the targets are now probability values, the target vectors take on a Bernoulli distribution of the form;

$$P(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^N \lambda\{y(\mathbf{x}_n)\}^{t_n} [1 - \lambda\{y(\mathbf{x}_n)\}]^{1-t_n}. \quad (3.72)$$

Since $P(\mathbf{t}|\mathbf{w})$ is no longer Gaussian the simplifying substitution of the convolution of two Gaussian with one Gaussian no longer occurs. Hence the marginal likelihood in Equation 3.64 cannot be calculated analytically. Instead a Laplace approximation must be employed to estimate the hyperparameters, more details of which can be found in Tipping (2001).

The RVM implementation described here starts with a full set of weights and works

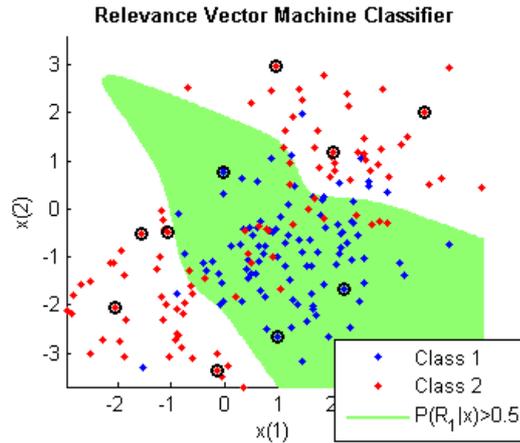


Fig. 3.9: Decision regions for an RVM classifier. Active vectors are circled.

back to a sparse solution. More recent work describes techniques of starting with an empty or minimal set of weights and iteratively adding the most relevant vectors. See “Incremental Gaussian Processes” (Quiñero Candela and Winther, 2003) and “Sequential Sparse Bayesian Learning” (Tipping and Faul, 2003) for examples. Such simplifications are not necessary for this work since training only occurs once. Hence, the RVM classifier was implemented in Matlab[®] using the “SparseBayes V1.1” toolbox (Tipping, 2002), which implements the model learning algorithm described here. Again, multiple-class classification was implemented using the “one versus all” approach.

In addition to producing probabilistic outputs RVMs have further advantages over SVMs. One main advantage is that they exhibit similar generalisation ability to that of SVMs while requiring significantly fewer support vectors. When comparing Figures 3.3 and 3.9 it is clear that RVMs require fewer training samples to approximate the dataset, albeit with slightly different decision regions. Although RVMs have longer training times than SVMs, RVMs do not require optimal parameter selection, such as the σ and C parameters in SVMs. This means that training a working RVM classifier does not require a grid search optimisation step, which is necessary for SVMs.

Although not strictly part of the Gaussian Process family (Quiñero Candela and Winther, 2003), RVMs exploit Gaussian Processes to aid in optimal weight selection. They also borrow from SVMs by using a linear weighted combination of basis functions. In this way RVMs can be considered a hybrid of SVMs and GPs.

3.8 Conclusions

This chapter has outlined the classification framework which has been implemented as part of this Thesis. A selection of probabilistic and non-probabilistic classifiers have been implemented. The array of classifiers will be assessed experimentally for suitability

to our localisation problem. Classifiers with non-probabilistic class predictions will have their outputs directly evaluated for localisation accuracy. Probabilistic classifiers, however, allow further analysis of the most likely position of the user. For example they allow rejection of uncertain estimates and Receiver Operating Characteristic (ROC) curve analysis. The availability of probability estimates also enables Hidden Markov Model analysis because the probability estimates can be used as observation probability models (Rabiner, 1989). More information about this augmentation to the classification framework can be found in Chapter 6.

The following chapters will describe the implementation details of the novel localisation technique developed. They will present the application of the classification techniques described in this chapter to generate location predictions, or more generally location probability estimates. These location probability estimates can be used for either location predictions, uncertainty rejection or Bayesian filtering.

Location Tracking Framework

In Chapter 2 an overview of the various techniques and systems in existence for predicting indoor location using RF transmissions was presented. Those prior systems share many common traits, not least the requirement of a plethora of RF Access Points (APs) throughout the environment. Having a wide array of APs spread throughout the environment provides readings that are sufficiently uncorrelated to be indicative of the location of a mobile device. This chapter describes the main contribution of this thesis; namely, an indoor localisation system which utilises absolutely minimal quantities of hardware installed in the entire environment.

Section 4.1 begins by describing the candidate technologies on which this localisation system has been implemented and evaluated experimentally. Section 4.2 assesses the suitability of the signals emanating from these candidate technologies for our particular localisation technique. Then Section 4.3 motivates the localisation technique developed and selects the optimal technology for this purpose. Finally, Section 4.4 presents experiments on a practical deployment of the technology and discusses the training data acquisition and pre-processing techniques needed to achieve maximum localisation accuracy.

4.1 Hardware

WLAN location tracking is a mature area of research, which has exhibited many compelling results, as highlighted in Section 2. However, most WLAN location tracking techniques are validated in office environments with multiple WLAN APs detectable throughout. As a result, these techniques are not as applicable to a home environment. The availability

of WLAN devices in a subject's home, especially an elder's home, is generally quite low, particularly in rural areas. Even if a WLAN AP is available, multiple APs would be necessary to provide sufficiently accurate location predictions, with location prediction accuracy increasing with the number of APs. Installing multiple WLAN APs throughout a home environment is an option if there are no cost constraints. Along with the expense of deploying several WLAN APs there is also the necessity of deploying a wired network throughout the environment to relay information back to the Basestation Computer (BSC). This work focuses on single BSC localisation to eliminate the installation and cost overheads associated with multiple BSC localisation.

As well as the installation restrictions precluding the use of a multiple WLAN AP localisation system, there also exists power consumption limitations. A mobile device to be carried by a subject must be able to last at least a day before the battery needs to be recharged. When active, WLAN devices consume high levels of power, especially when compared to an energy efficient wire-replacement technology, such as Bluetooth[®]. Bluetooth[®] devices have a lower battery drain than WLAN devices since they do not require complicated networking protocols as with WLAN. Furthermore Bluetooth is present, by default, on a variety of affordable mobile phones, whereas WLAN is available only on high-end smartphones and Personal Digital Assistants (PDAs). Hence, by developing this localisation technique for Bluetooth[®] devices, it can provide location-based services extremely cost effectively without the need for the user to carry dedicated localisation hardware. It could be argued that in the specific deployment scenario this work is focused on, namely elder monitoring, personal communication devices may not not already be present. We argue that if personal communication devices need to be provided, Bluetooth[®] is a much more cost effective and power conservative alternative to WLAN.

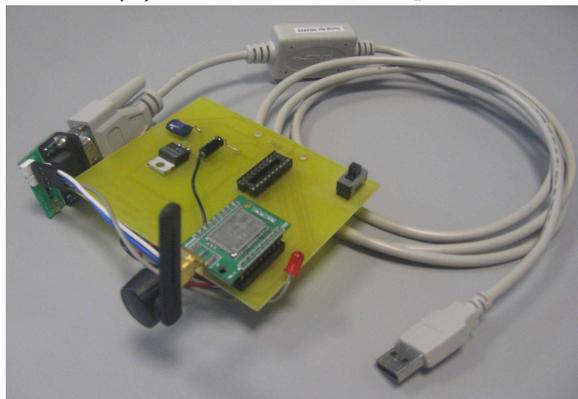
As more thoroughly discussed in Kelly et al. (2009b), the Nokia N95 (Figure 4.1(a)) was chosen as the Bluetooth[®] test device since it has a feature-set similar to that of most medium to high-end mobile phones. It has the usual cellular network connectivity along with Bluetooth[®] and WLAN connectivity. Furthermore it has other sensors such as light sensors, cameras and accelerometers. The availability of accelerometer sensors shall also be exploited in Chapter 6 to improve localisation accuracy. The Nokia N95 allows the installation of a Python programming language interpreter, which enables sophisticated handling of the sensors present in the phone with a very gentle learning curve. Python scripts were developed to retrieve cellular network and accelerometer data from the phone and to forward it to the Bluetooth[®] BSC in response to data requests. The accelerometer samples will be used in Chapter 7 to augment localisation accuracy.

The Bluetooth[®] mobile phone can estimate and provide cellular network link signal strength information to the BSC. However, mobile phones generally do not have

(a) Bluetooth® mobile phone



(b) Bluetooth® access point

**Fig. 4.1:** Hardware utilised for Bluetooth® deployment.

Bluetooth® transceivers expensive or sophisticated enough to be able to derive Bluetooth® link signal strength information. Instead, the task of estimating the signal strength information of the Bluetooth® connection is left to the connected BSC. Hence, this work implements a form of remote positioning. Bluetooth® is a peripheral wire replacement technology and as such it is not required to have signal strength readings which are meaningful to the user. Its use of power control to maintain consistent received signal strengths over increasing distances means that small changes in actual received signal strengths are not visible in the Received Signal Strength Indicator (RSSI) readings. Rather than using a standard Bluetooth AP which is available on most laptops, we chose to construct an AP using a Blueradios BR-SC30N Bluetooth transceiver (Figure 4.1(b)). This hardware gives RSSI readings with a higher resolution than that required by the Bluetooth specifications (Bluetooth® Special Interest Group, 2001). This permits higher resolution position discrimination than would be possible with typically available Bluetooth APs. Another reading which is available from this hardware when a connection is established is Link Quality (LQ). Explanations of these signals are forthcoming in the next section.

An alternative to Bluetooth® for power conservative, short-range communications is ZigBee®. ZigBee® is even more power conservative than Bluetooth® since it is designed for long-term, low data-rate communications in home, industrial and medical monitoring environments. The ZigBee® mobile device adopted for these experiments is the Intel Digital Health Group's Sensing Health with Intelligence, Modularity, Mobility, and Experimental Reusability (SHIMMER) platform (Patel et al., 2007). The SHIMMER, which can be seen in Figure 4.2, is a wireless sensing platform, developed for health monitoring applications, which is experiencing growing popularity. Its attractiveness lies not just in its small form factor but also in its ability to interface with new sensor boards. Boards such as kinematics (3-axis accelerometer, magnetometer and gyroscope),



Fig. 4.2: ZigBee[®] hardware: SHIMMER without case (left), SHIMMER with case (centre) and TelosB mote (right), compared with a 1 Euro coin (bottom) for scale.

electrocardiogram (ECG) and Electroencephalograph (EEG) sensors have already been developed, allowing it to be applied to many different healthcare projects. Along with a Bluetooth[®] chip, it possesses a Chipcon CC2420 IEEE 802.15.4 2.4GHz transceiver (Chipcon, 2006), which implements the lower level functionality of the ZigBee[®] stack¹. Hence, the SHIMMER was chosen to represent a ZigBee[®] enabled device.

The advantages of ZigBee[®] over Bluetooth[®] include the longer battery life and higher RSSI resolution. Along with high RSSI resolution, ZigBee[®] has a reading similar to Bluetooth[®]'s LQ called Link Quality Indicator (LQI). The disadvantages of ZigBee[®] include the fact that ZigBee[®] devices are not as ubiquitously available as Bluetooth[®] devices and the lack of an interaction modality on the SHIMMER such as a screen or speakers. The SHIMMER was programmed to transmit packets at a rate of 1Hz using its native TinyOS programming language. A TelosB mote (Polastre et al., 2005), which contains a CC2420 transceiver, was programmed to act as an AP for the BSC. The TelosB mote was programmed, also using the TinyOS language, to decode every packet transmitted by the SHIMMER for RSSI and LQI information.

The signals available from the localisation hardware under consideration are the vital components of a reliable localisation system. Hence, the following sections shall consider the available signals for suitability to the localisation problem and illustrate how our localisation technique will exploit the characteristics of the signals.

4.2 Available Signals

Typical RF localisation systems require signals from several APs to obtain a characteristic signature, or fingerprint, for each location. The signals typically used are RSSI or LQ/LQI,

¹The SHIMMER's communication hardware shall henceforth be referred to as ZigBee[®] for simplicity of presentation. However, it should be remembered that the CC2420 is not a full ZigBee[®] stack.

whichever is deemed more accurate for the given hardware (Pandya et al., 2003). It has generally been assumed that using only the most accurate signal is sufficient and the other signal, if available, does not provide any extra accuracy. We, however, have found that RSSI and LQ/LQI, while related, do not have identical characteristics at a given location. We therefore build our single AP localisation system on the fact that many of the signals available at a given AP are not entirely redundant. First the signals available from the Bluetooth[®] test platform are considered.

4.2.1 Bluetooth[®]

As briefly alluded to already, a Bluetooth[®] connection has 2 readings associated with it, RSSI and LQ. RSSI can coarsely be considered as a measure of the RF energy incident on the Bluetooth[®] receiver. In reality the RSSI reading actually available to the user is not the same as the incident RF energy. The Bluetooth[®] specification (Bluetooth[®] Special Interest Group, 2001) specifies a golden receive power range. This golden range is the ideal power for an RF wave incident on the Bluetooth[®] receiver, such that the transmitter is not transmitting at an excessively high power which would cause interference with more distant devices. When the incident RF energy falls below the lower threshold of this range a power increase request is sent to the transmitter. If the transmitter can, it will transmit at a higher power level. Similarly if the incident RF energy goes above the upper threshold of this golden range, a power decrease request is sent to the transmitter.

When the incident RF energy is within the golden range an RSSI value of 0 is returned. Only when the RF energy steps outside of the golden range will the RSSI value change. Furthermore, the Bluetooth[®] specification only requires the RSSI value to report whether the incident RF energy is within, above or below the golden range, with no need to indicate how far from the golden range the actual energy is. It is entirely up to the manufacturer how much the RSSI reflects the actual signal strength outside of the golden range. It is generally accepted that the variability of Bluetooth[®] RSSI throughout an environment is too low to be of use in a localisation system (Bielawa, 2005). To quantify how RSSI falls off over increasing distance in our Bluetooth[®] transceiver, free-field experiments were performed.

The free-field experiment was conducted in an open outdoor environment without any vertical obstructions within 100m of the test equipment. The transmitting phone and the receiving Bluetooth[®] AP were both placed 1m above the ground on wooden platforms and their separation distance was increased by 1m at a time. At each separation distance 200 samples were taken. Figure 4.3(a) illustrates the mean and standard deviations of RSSI for increasing distance. It can be seen that RSSI remains at 0 until about 5 metres, then it falls off to the minimum of -10 after a further 3 metres. After 22 metres the Bluetooth[®] connection becomes impossible to maintain for any significant period of time. The red line

shows an attempt to fit the path loss model in Equation 2.35 to the data. Using linear regression the path loss exponent was calculated to be $n = 0.9836$, whereas in reality the value should be $n = 2$ in a free field. The significant disagreement between theory and reality indicates that Bluetooth[®] RSSI is certainly not a reliable indicator of distance. The issue with Bluetooth[®] RSSI is that it does not exhibit resolution beyond two levels, making it a bad indicator of location by itself.

To overcome the limited resolution RSSI signal, LQ was considered. LQ is frequently used for Bluetooth[®] localisation due to its generally higher spatial resolution than RSSI (Pandya et al., 2003, Genco, 2005). Unlike RSSI, LQ is derived from the Bluetooth[®] link's Bit-Error-Rate (BER). Figure 4.4 illustrates the relationship between BER and the reported LQ value in the Blueradios BR-SC30N. Figure 4.3(b) shows how LQ varies as a function of T-R separation distance. It can be observed that, besides a minor fluctuation at 9 metres, most likely due to destructive interference from reflected and direct path differences, LQ does not significantly start to fall-off until a distance of 17 metres. The different fall-off points for RSSI and LQ can be explained by the fact that these signals are derived in different ways. RSSI is a raw analogue channel reading derived from the incident RF energy. LQ, on the other hand, is a digital channel reading which is less susceptible to interference than RSSI. This lower susceptibility is due to Bluetooth[®]'s use of Gaussian Frequency Shift Keying (GFSK) to encode bits as a change in carrier frequency. Hence, RSSI and LQ are governed by different transmission effects, as corroborated by their correlation coefficient for this dataset of $r = 0.32$. From this illustration it should be evident that using both signals gives more information about the separation distance than either signal would give in isolation. This will remain the case when these signals are applied to the task of localisation.

Besides the signals available from the phone's Bluetooth[®] link, there are also cellular network connectivity readings available from the mobile phone using the Python programming language. The first such reading is Cellular Received Signal Strength Indication (CRSSI). CRSSI is an indication of the strength of the currently connected cell tower, akin to the number of "connection bars" on the phone's screen. From 4.3(c) it can be seen that CRSSI does not change as a function of T-R separation. This is because -80 dBm is the highest value obtainable with this hardware. Since the experiment is conducted in a free-field with no obstructions, maximum strength reception is always possible. More interesting CRSSI behaviour can be observed later in the indoor scenario where obstructions significantly attenuate CRSSI. In addition to CRSSI, the connected mobile phone can also be queried for the currently connected Cell ID (CID). This CID number identifies which cell tower is currently strongest, hence, connected to the phone. Since different cell towers cover different locations, this reading also varies as a function of position. In the free-field test the CID number remained constant since there would

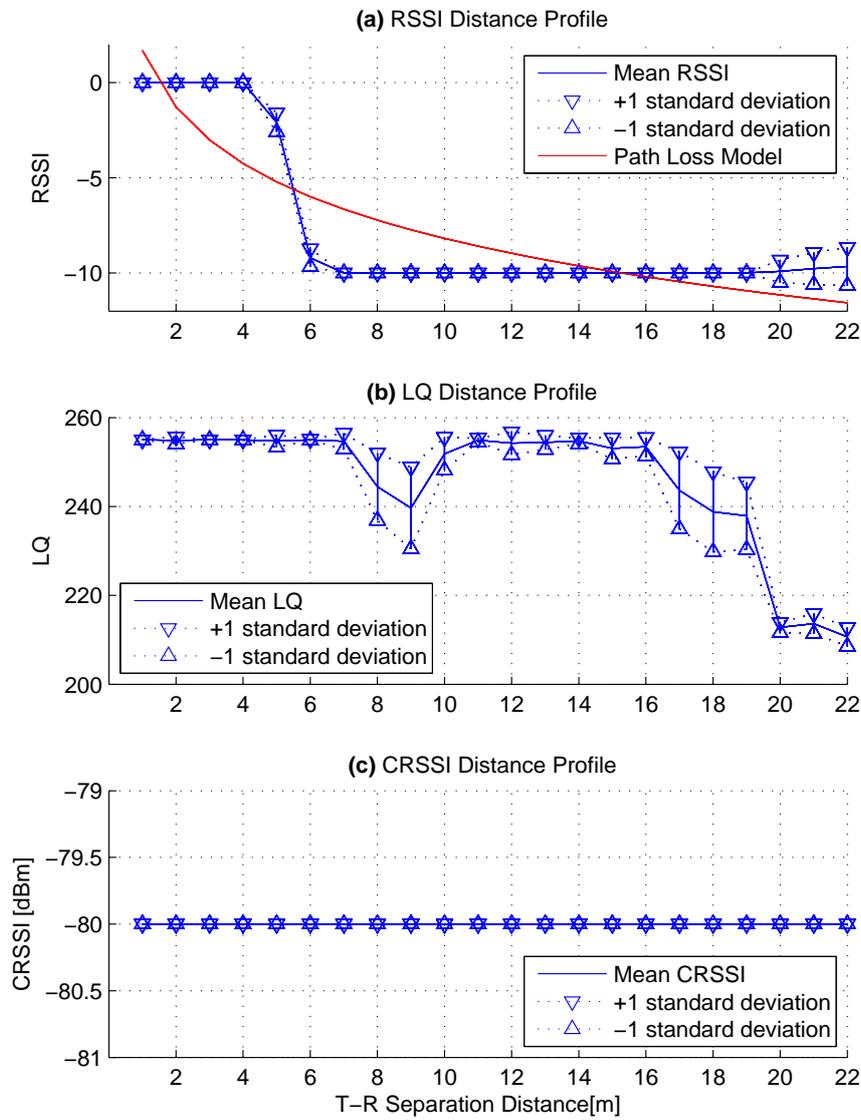


Fig. 4.3: Bluetooth[®] platform RSSI, LQ and CRSSI for increasing T-R separation in a free field. RSSI and LQ are unitless quantities and CRSSI is measured in dBm.

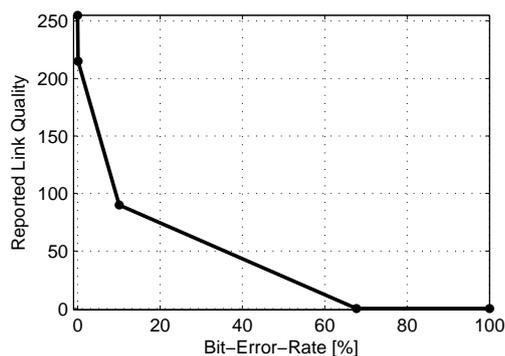


Fig. 4.4: LQ-BER relationship for the Blueradios BR-SC30N.

typically be little or no CRSSI signal variation in a 22 metre distance when compared to the outdoor transmission range of a cellular basestation. Hence the strongest cell tower, and as a result the currently connected CID, did not change. However, indoors this reading can change frequently due to the attenuation of certain cell towers by obstructions.

To appreciate how the available signals behave in an indoor environment the proposed localisation hardware was deployed in a section of a home environment. As illustrated in Figure 4.5, the test environment consists of 4 rooms, designated (1) Bedroom 1, (2) Bedroom 2, (3) Bathroom and (4) Hallway. The map shows two AP locations. Access Point 1 is used for the single BSC experiments. Access Point 2 will also be used for the multiple BSC comparison experiments. To illustrate how each signal varies throughout the environment, a signal map of the environment was obtained. To obtain this map the mobile phone is moved to a number of different fixed positions on a wooden platform 1m above the ground and sampled for a period of two minutes at each position in a static, uninhabited environment. This results in over 100 samples of each signal for every position. Data from each position is labelled with x coordinate, y coordinate and room number. The sampled x and y coordinates are selected to form an even grid of squares throughout the environment, each square covering 1m^2 . The mean of the samples at each position is assumed to approximate the real value of the signal at that position. Figure 4.6 demonstrates the behaviour of RSSI, LQ and CRSSI throughout the environment. Unsampled locations are indicated by x's. It can be seen that, while RSSI and LQ vary similarly throughout the environment, they are not entirely correlated and can be combined to give a more accurate indicator of location than either one alone.

Figure 4.6(c) illustrates the behaviour of CRSSI throughout the indoor environment. CRSSI is clearly uncorrelated with RSSI and LQ, which is intuitive when one considers that CRSSI is dependent on the connection between the phone and a basestation several kilometres away. In fact it has been found that CRSSI changes as a function of the size of window and distance from the window in each room. Also rooms with no windows have

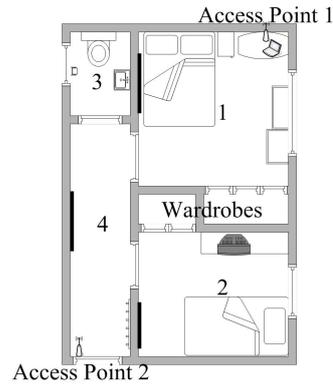


Fig. 4.5: The 4-room indoor test environment.

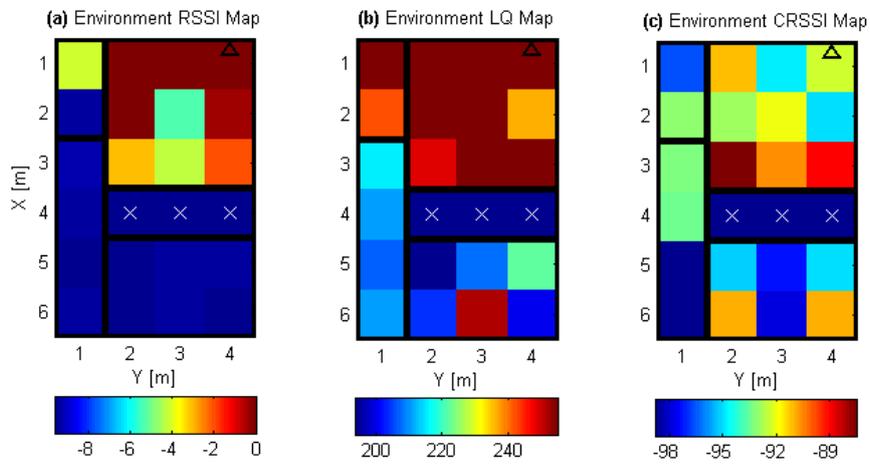


Fig. 4.6: RSSI, LQ and CRSSI throughout the test environment.

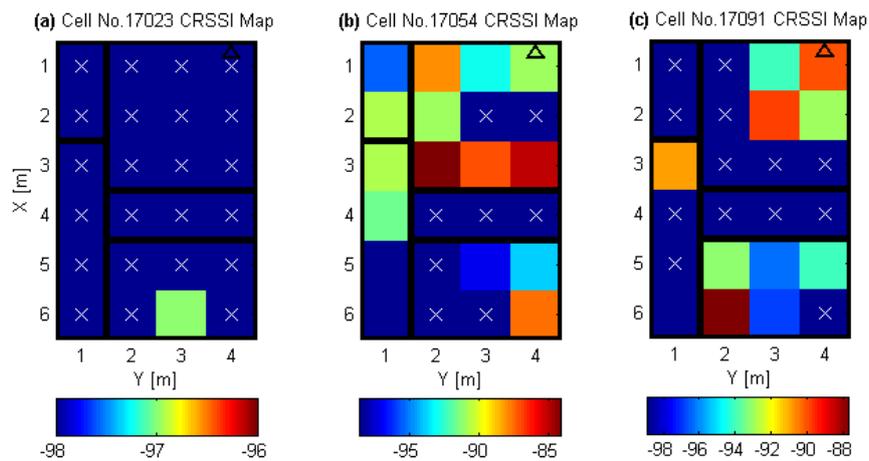


Fig. 4.7: CRSSI for the cell towers detectable in the test environment.

severely reduced CRSSI. The CRSSI map, however, does not tell the entire cellular signal story. The CRSSI reading is dependent on the connected basestation and as different cell towers are attenuated by different obstructions, the currently connected cell tower can change throughout the environment. Hence the CRSSI map can be broken down according to CID. Figure 4.7 illustrates the CRSSI readings obtained throughout the environment when different cell tower IDs are detected. In this figure x's indicate positions in which the given cell tower was not detected. From this we can see that certain cell towers are more likely to be connected at certain locations, meaning that certain cell towers are strongest at certain locations. Hence the currently connected cell tower ID is indicative of location.

The Bluetooth[®] platform has shown the availability of a number of signals useful in the task of location prediction. A limitation of this platform is the extremely low resolution of the RSSI signal, a signal typically extremely indicative of location in other technologies, such as WLAN. In anticipation of the implications this will have on localisation accuracy, another technology with higher RSSI resolution, ZigBee[®], was trialled.

4.2.2 ZigBee[®]

Since ZigBee[®] (or at least our chip's implementation of the ZigBee[®] protocol) does not implement any form of power control, the obtained RSSI readings have a highly linear relationship with the actual received signal intensity. In fact RSSI is actually measured in dBm for ZigBee[®]. The same free-field experiment was conducted as with the Bluetooth[®] platform. Figure 4.8(a) shows that ZigBee[®] RSSI fall-off follows the theoretical path loss model more reliably. The path loss exponent of $n = 2.32$ is more appropriate for a free-field.

LQI is also derived differently than in Bluetooth[®]. In ZigBee[®] LQI is a measure of the correlation between the first 8 received symbols of the physical layer header and the expected value of these symbols calculated at the transmitter before transmission. LQI is related to packet reception rate, which is significantly decorrelated with RSSI indoors (Srinivasan and Levis, 2006). Figure 4.8(b) illustrates the LQI fall-off for the free-field environment. Along with the different behaviour from that of Bluetooth[®] it can also be observed that there is significantly higher standard deviation at all separation distances. This indicates that LQI readings in ZigBee[®] will be significantly less reliable than that of Bluetooth[®].

As with Bluetooth[®] it is pertinent to consider how the signals vary in a realistic test environment. The ZigBee[®] platform was deployed in the same environment illustrated in Figure 4.5. The same sampling was conducted throughout the environment, resulting in the indoor signal maps in Figure 4.9. As in 4.6 there is lower correlation between RSSI and LQI, than expected for signals which are derived from the same hardware. This trait in both hardware platforms shall be exploited to enable location predictions from this

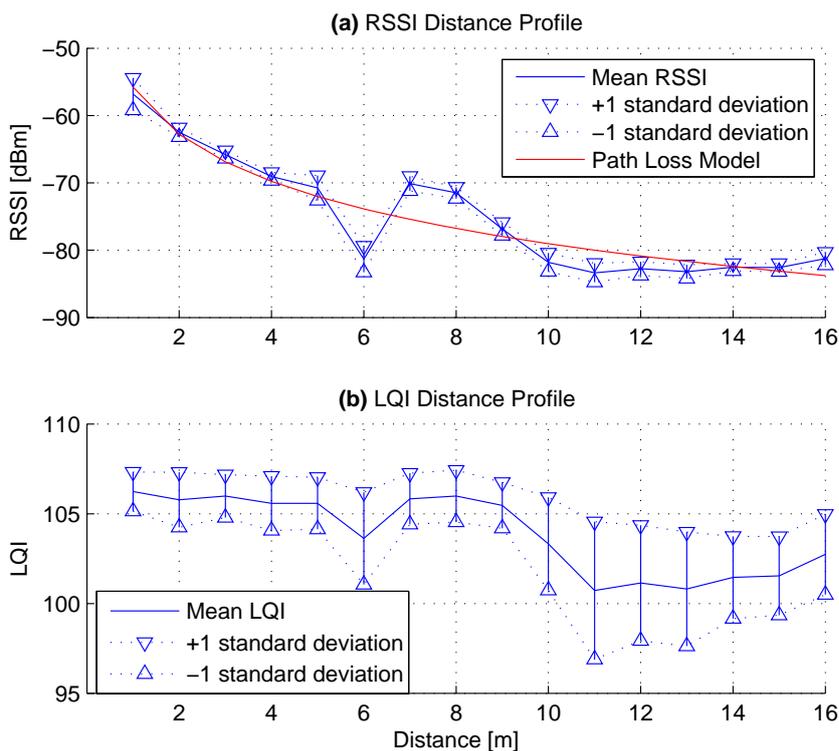


Fig. 4.8: ZigBee® platform RSSI and LQI for increasing T-R separation in a free-field. LQI is a unitless quantity.

completely minimal hardware deployment. Since it is difficult, if not impossible, to model the Bluetooth® platform's RSSI, LQI and CRSSI and the ZigBee® platform's LQI an empirical fingerprinting approach shall be taken, as the next section will present.

4.3 Localisation in a Static Scenario

Before the localisation system can be deployed in a realistically challenging localisation scenario it must be considered for viability in the simplest possible deployment scenario. A realistic deployment scenario would involve a moving mobile device and moving people and doors within the environment. Alternatively, the static scenario represents the simplest possible deployment scenario, which negates the effects of a real moving subject such as a moving mobile device and signal attenuation effects of the human body. This means obtaining samples with the mobile device in a fixed position with no people present in the environment during data acquisition. This type of data is obtained in exactly the same way as the illustrative data in the previous section. Once the accuracy of the system in this relatively trivial scenario has been considered, it can be applied to a realistic deployment with the usual environmental movement effects.

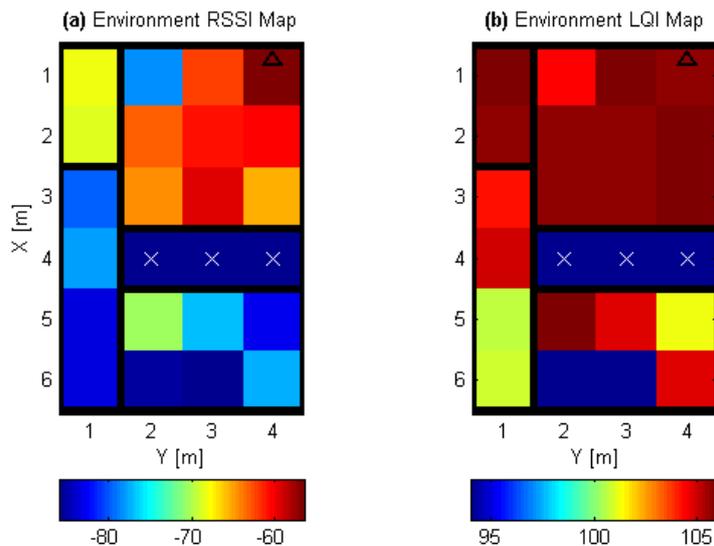


Fig. 4.9: ZigBee[®] RSSI and LQI throughout the test environment.

The primary advantage of this technique over other more traditional localisation techniques is in its low hardware requirements. For this reason it was decided to compare this technique with a more traditional localisation system deployment in the trial environment of Figure 4.5. Typically Bluetooth[®] localisation techniques require a BSC to control every Bluetooth[®] AP since stand-alone Bluetooth[®] APs are not widely available. Since other RF localisation systems typically use the highest resolution signal available from all BSCs, multiple BSC localisation was implemented using a single signal available from each installed BSC. Our novel localisation technique, however, uses all of the signals available at a single BSC. Figure 4.10 illustrates the location classification features available for a traditional localisation deployment. For each sample, the LQ from the first BSC is illustrated on the x-axis and the corresponding value from the second BSC is illustrated on the y-axis. The marker colours and shapes indicate which room resulted in the particular LQ1-LQ2 combination. For multiple Bluetooth[®] BSC localisation, the LQ signal from each computer's AP was used as the location classifier input due to its higher spatial resolution than RSSI. For the ZigBee[®] platform the RSSI readings from each BSC were used as the input features for the same reason. From our fingerprinting-classifier perspective the aim is to define the optimal classification regions for the data from each class or room. The definition of optimal in this case is the classification regions which lead to the highest localisation accuracy.

4.3.1 Localisation Accuracy Metric

At this point the method of determining localisation accuracy must be considered. Instead of the typical localisation accuracy metric of mean error distance, this work employs a

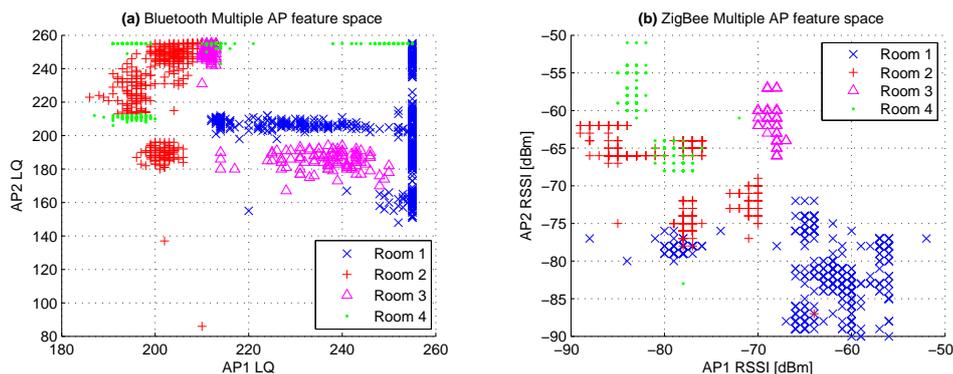


Fig. 4.10: Two input feature-spaces for the Bluetooth[®] and ZigBee[®] multiple BSC localisation scenarios.

		Actual Label		
		1	2	3
Predicted Label	1	0.5	0.25	0.25
	2	0.2	0.6	0.2
	3	0.1	0.2	0.7

Table 4.1: An example confusion matrix. The diagonal terms indicate the frequency of correct prediction of the actual class. The off-diagonal terms indicate the frequency of predicting a particular class for the actual class denoted by the column label.

classification success metric. However, overall accuracy, or room recognition rate, is not the best measure of success since large rooms with a large number of test samples will bias the overall accuracy. Instead the unweighted average of the accuracies of each individual room is used as an unbiased error metric. The accuracy of each room is calculated to be the ratio of correct predictions for a given room to the number of test samples in that room.

This unbiased accuracy can easily be calculated as the average of the diagonal terms in the confusion matrix. A confusion matrix is a square matrix in which each element corresponds to a class detection success rate (see Table 4.1). Each column corresponds to an actual class label and each row corresponds to the predicted class label for a particular actual label. Hence, the diagonal entries indicate the proportion of correct predictions for a class and the off-diagonal terms indicate the “confusion” for that class. We propose that the accuracy metric of the mean of the diagonal of the confusion matrix is more relevant to real world applications than error distance because error distances do not indicate containment of predictions within the correct room. For example a large error distance in a large room may not be as incorrect as a large error in a small room. Conversely, a small error distance near a wall may translate to an incorrect room prediction; an effect not highlighted in other localisation work. Further discussion of this accuracy metric in a long-term deployment can be found in Section 5.3.

4.3.2 Multiple BSC Localisation

Previous localisation techniques focused on coordinate location rather than room-level location. However, their underlying algorithms can be adapted to the symbolic or room-level location prediction problem. For example Bahl and Padmanabhan (2000b) and Lorincz and Welsh (2006) use a k NN algorithm and Castro et al. (2001) and Roos et al. (2002b) use unimodal probabilistic models to estimate location with high accuracy. For this reason we consider k NN and the unimodal Gaussian probabilistic classifiers NBC, LDA and QDA for the traditional localisation scenario. The more sophisticated techniques will be considered for accuracy improvement in later chapters. To calculate multiple BSC localisation accuracy two separate datasets were obtained. Separate datasets were used to reduce the influence of temporal variations in the signals on accuracy. Accuracy is then calculated across two runs. First; the localisation algorithms are trained on dataset 1 and tested on dataset 2. Second; the localisation algorithms are trained on dataset 2 and tested on dataset 1. Hence, for multiple AP localisation there were two algorithm runs, one for each dataset combination. Table 4.2 summarises the mean multiple BSC localisation accuracy across both runs.

(a) Multiple BSC Bluetooth [®]		(b) Multiple BSC ZigBee [®]	
Algorithm	Mean Accuracy	Algorithm	Mean Accuracy
k NN	0.94 (0.002)	k NN	0.60 (0.076)
NBC	0.60 (0.000)	NBC	0.63 (0.047)
LDA	0.42 (0.010)	LDA	0.62 (0.032)
QDA	0.71 (0.015)	QDA	0.60 (0.051)

Table 4.2: Mean accuracy for multiple BSC localisation tests. Standard Deviations are in parentheses.

From this table it is evident that Bluetooth[®] localisation using the k NN algorithm achieves the best accuracy. The unimodal probabilistic classifiers do not perform as well on the Bluetooth[®] platform, with LDA performing the worst. This can be explained by the fact that linear decision regions are not appropriate for the level of inter-class data mixing visible in Figure 4.10. ZigBee[®] on the other hand has similar accuracy across all classifiers. This suggests that the lower localisation accuracy is not only due to inadequate classification decision region flexibility, but also a fundamental restriction of higher inter-class overlap and lower inter-dataset repeatability. Hence, even the flexible k NN classifier is unable to fully discriminate between many of the locations.

Relatively high localisation accuracy has been shown possible with the available Bluetooth[®] and ZigBee[®] hardware installed in a traditional multiple BSC configuration. This high accuracy comes at the cost of a high level of infrastructure. Not only do multiple BSCs need to be installed, but the BSCs need to communicate with each other to route signal information to one central location. This leads to further installation effort in terms

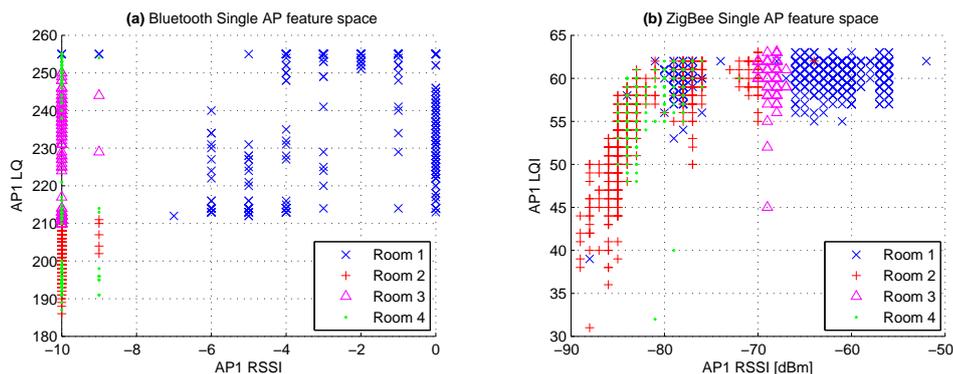


Fig. 4.11: Two input feature-spaces for the Bluetooth[®] and ZigBee[®] single BSC localisation scenarios.

of network installation and configuration along with the need to find appropriate locations throughout the environment to install BSCs. Hence, single BSC localisation reduces not only hardware costs but also installation effort.

4.3.3 Single BSC Localisation

Using the datasets already obtained, the accuracy for a single BSC localisation scenario was tested. Figure 4.11 shows the two input feature space when only data from AP 1 is available. It can be seen that while the classification regions are certainly not as distinct as those in Figure 4.10, the inclusion of both signals from the same AP allows one to uniquely discriminate between certain locations which would not be possible with any one signal alone. This is empirically corroborated in Tables 4.3 and 4.4. Again, the mean accuracy is evaluated over two runs and the standard deviation is in parentheses. From these tables it can be observed that it is possible to discern location using only the highest resolution signal, namely LQ for Bluetooth[®] and RSSI for ZigBee[®]. For Bluetooth[®], however, when the second, less resolved signal is introduced the localisation accuracy for access point 1 increases. This indicates that the decorrelated properties of RSSI and LQ contribute to more location indicative signals. The fact that there is no such improvement for access point 2 suggests that the decorrelated properties of the signals are highly dependent on the position in which the BSC is installed. Hence, the accuracy of the single BSC localisation technique is dependent on the BSC installation location. It should also be noted that the second AP used a lower gain antenna, hence its RSSI signal was saturated to -10 at most locations, leading to poorer signal, hence location, resolution. The lack of RSSI variation for AP2 explains why the inclusion of this signal does not improve localisation accuracy.

Upon considering the ZigBee[®] platform, it is evident that the second AP exhibits lower accuracy confirming that accuracy is BSC location dependent. It is also observable that there is no localisation accuracy improvement when including the second available

Algorithm	BSC1 LQ	BSC2 LQ	BSC1 RSSI+LQ	BSC2 RSSI+LQ
<i>k</i> NN	0.58 (0.040)	0.31 (0.00)	0.75 (0.01)	0.31 (0.00)
NBC	0.53 (0.004)	0.42 (0.00)	0.63 (0.04)	0.38 (0.04)
LDA	0.50 (0.002)	0.39 (0.01)	0.57 (0.00)	0.37 (0.00)
QDA	0.53 (0.004)	0.42 (0.00)	0.63 (0.03)	0.38 (0.04)

Table 4.3: Single Bluetooth[®] BSC static environment localisation accuracy.

Algorithm	BSC1 RSSI	BSC2 RSSI	BSC1 RSSI+LQI	BSC2 RSSI+LQI
<i>k</i> NN	0.56 (0.03)	0.35 (0.09)	0.49 (0.01)	0.33 (0.01)
NBC	0.61 (0.15)	0.43 (0.10)	0.60 (0.21)	0.39 (0.04)
LDA	0.71 (0.03)	0.46 (0.03)	0.69 (0.03)	0.46 (0.02)
QDA	0.62 (0.15)	0.44 (0.10)	0.62 (0.18)	0.39 (0.04)

Table 4.4: Single ZigBee[®] BSC static environment localisation accuracy.

signal. As a result ZigBee[®] has lower overall localisation performance with this raw data. It has been shown that the high ZigBee[®] measurement noise can be reduced using filtering, emphasising the decorrelated properties of the signals, hence increasing localisation accuracy (Kelly et al., 2008d). This shall however be omitted for brevity. The focus of the remainder of this work will be on a single technology, Bluetooth[®], due to its higher possible localisation accuracy. Using Bluetooth[®] allows the use of more commonly available mobile devices such as mobile phones, leading to potentially lower deployment costs. This also provides further location indicative signals, namely CRSSI and CID, which have not yet been considered for localisation accuracy.

For the Bluetooth[®] platform there are 4 location-dependent input features; RSSI, LQ, CRSSI and CID. Figure 4.12 illustrates the different classification regions for each room. Since it is difficult to visualise this 4-feature space, two separate plots are presented each with a different combination of signals. The classifiers however shall take all 4 features as the inputs. Since CID is an identification number based on network assignments it cannot be treated the same as a continuous input since it will bias the classifiers with its irregularly spaced ID assignments. Instead it was necessary to replace each CID with a number such that all CIDs were numbered contiguously from 1 up to the maximum number of unique CIDs. A conversion table was constructed with the new assignments for each CID. Then each CID in the test dataset was replaced using this table.

It is evident from the individual plots that there are many classification regions in each room which would not be uniquely identifiable using one of the illustrated subsets of signals. Hence, using all the signals available contributes to higher localisation accuracy in the static environment. Table 4.5 summarises the static localisation accuracy possible

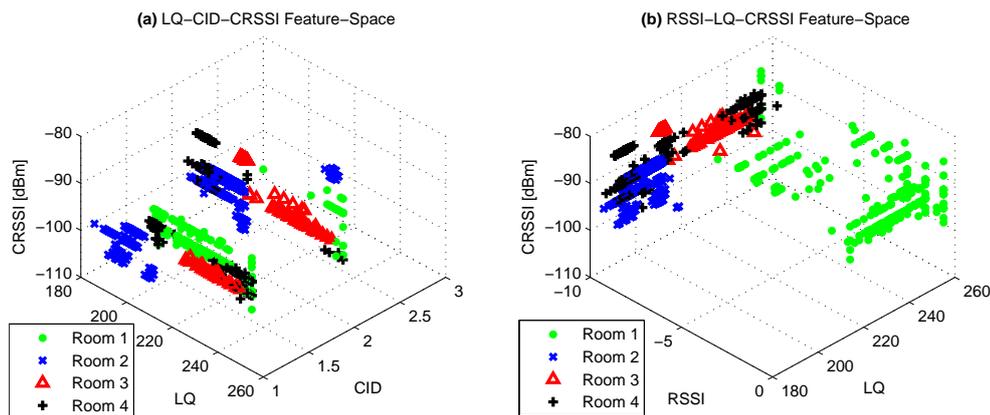


Fig. 4.12: Three input feature-spaces for the Bluetooth[®] localisation platform.

Algorithm	BSC1	BSC2	Overall Accuracy
k NN	0.90 (0.00)	0.78 (0.04)	0.85 (0.07)
NBC	0.83 (0.03)	0.63 (0.03)	0.73 (0.12)
LDA	0.76 (0.05)	0.70 (0.03)	0.73 (0.05)
QDA	0.86 (0.00)	0.71 (0.01)	0.79 (0.09)

Table 4.5: Localisation accuracies for a single Bluetooth[®] BSC using all available signals in a static environment. Overall accuracy is the mean of BSC1 and BSC2 accuracies.

using all of the signals available from the Bluetooth[®] hardware. When comparing this with tables 4.3 and 4.4 it can be seen that BSC2 localisation still suffers lower accuracy than BSC1 in spite of the use of the Bluetooth[®] AP independent signals, CRSSI and CID. This indicates that the AP type and BSC location is still an important consideration in the deployment of the system. It can now also be observed that the highest accuracy is still from k NN. For the probabilistic classifiers QDA now exhibits the highest localisation accuracy, followed by NBC and LDA. NBC performs poorly due to the assumption of feature independence and LDA performs poorly now because the assumption of a common covariance across classes becomes increasingly inappropriate with increasing dimensionality. Again k NN performs best with the downside of a higher computational cost, proportional to the training dataset size.

Our system’s highest mean single BSC accuracy of 0.85 compares favourably with the multiple BSC technique’s highest mean accuracy of 0.94, especially when one considers that it only requires half the level of installed hardware. One issue with these results is that they do not necessarily correspond to the accuracies possible during non-trivial situations when the phone does not remain stationary on a platform 1m above the ground, facing the same direction at all positions, etc. The next section shall consider the localisation accuracy possible with the localisation platform during realistic daily movements. The

Bluetooth[®] platform will be the focus of the remainder of this thesis due to the ubiquity of Bluetooth[®] enabled mobile devices, the familiar user interface and higher signal variety the cellular connectivity presents.

4.4 Localisation in a Realistic Scenario

Before the Bluetooth[®] localisation system can be deployed reliably in a realistic scenario some further details of its implementation must be considered. This section addresses the remaining implementation details impeding long-term accuracy.

4.4.1 Training Data Acquisition

As in most localisation systems this technique is implemented in two phases. The first phase, known as the calibration or training phase, is where data representative of each location is obtained. In the second phase, known as the tracking phase, the localisation algorithm trained on the training phase data is used to predict location from the current set of samples. One issue with the method of data acquisition described in Section 4.3 is that training data obtained from the static situation will not correspond to data obtained from real-life movements. For example, in the situation when a mobile phone is in a user's pocket, the signals will be drastically different from when the mobile phone is placed 1m above the ground on a platform, due to signal attenuation effects of the human body and the dynamic behaviour of the signals when the device is moving. As a result, basing location predictions for a human carrying the mobile device on static training data will not be the most accurate technique.

To address this issue it was decided to also obtain training data from a user inhabiting each room for a short period of time. This is referred to as the “One-Room-at-a-Time” (ORAT) method. For each room an experimenter carried the mobile device in their pocket while the BSC logged 200 samples of each signal, before changing to the next room. No other people were present when data was obtained. The experimenter walked to positions they would typically visit along paths they would usually take. This and the static technique are similar to the training data acquisition techniques employed for other fingerprinting-based localisation systems. Fig. 4.13(a) shows how RSSI, LQ and CRSSI vary throughout the environment for this ORAT training dataset. It is clear from the figure that there are overlapping clusters of data for Rooms 2 and 4, which highlights that 100% classification accuracy is not possible with the available signals.

This ORAT method has a major drawback. It does not allow for the acquisition of data representative of a subject moving from one room to another, hence it will have reduced accuracy when this movement is occurring. For this reason it was necessary to experiment with a third type of training dataset. This dataset is obtained by constantly

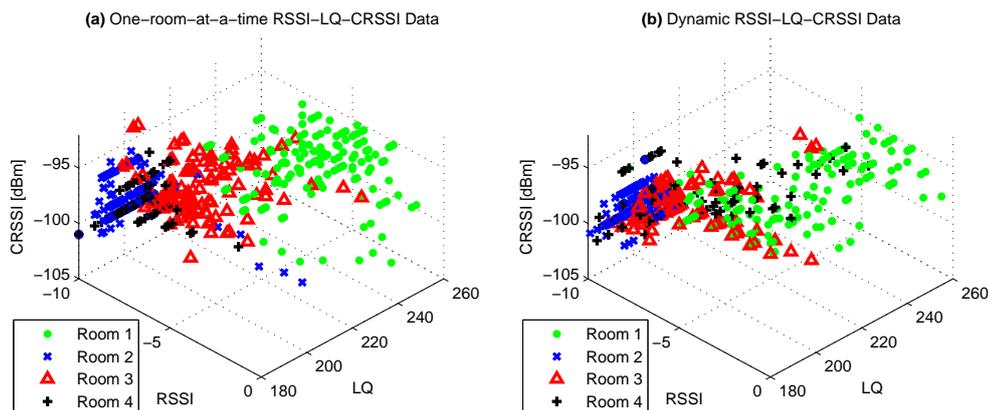


Fig. 4.13: Three-feature-spaces for two different techniques of obtaining training data for the dynamic localisation scenario.

sampling while the experimenter walks throughout the environment. The experimenter makes a voice recording during the sampling process, noting the room transitions as they occur. Using this recording the samples can be labelled offline. This leads to a training dataset which exactly matches real-life movement-based signal behaviours throughout each room. For comparison Fig. 4.13(b) illustrates how this so-called “dynamic” dataset varies throughout the environment. When comparing with Fig. 4.13(a) different coverage of the RSSI-LQ-CSQ space is observable. Furthermore, the feature-space coverage in Figures 4.13(a) and 4.13(b) is significantly different to that of Figure 4.12(b). This is due to power control’s dynamic behaviour causing RSSI to exhibit more readings in the intermediate range. Clearly this will result in different classification accuracy for these different types of training data.

To quantify the effect the training dataset acquisition method will have on localisation accuracy a test dataset was obtained. This dataset was obtained during a 15 minute walk throughout the test environment. This walk was designed to represent a user’s typical movements throughout the home environment. During the 15 minute walk data is constantly logged at the BSC. Also a voice recording of locations, much the same as that obtained during the dynamic data training phase, is obtained. Table 4.6 outlines the accuracy possible with a realistic test dataset when different types of training data are employed. It can now be observed that using the data obtained in the static scenario leads to extremely poor localisation accuracy, on par with random guessing. The highest possible tracking accuracy occurs using the dynamically obtained training data. Hence, dynamically obtained data is the training data of choice for the remainder of this chapter. The accuracy levels reported so far are for classification using the raw (noisy) data. Further accuracy improvement can be achieved by filtering the data.

Training Dataset	k NN	NBC	LDA	QDA
Static	0.27	0.30	0.27	0.25
ORAT	0.50	0.49	0.52	0.43
Dynamic	0.52	0.54	0.57	0.57

Table 4.6: Localisation accuracy for different training datasets.

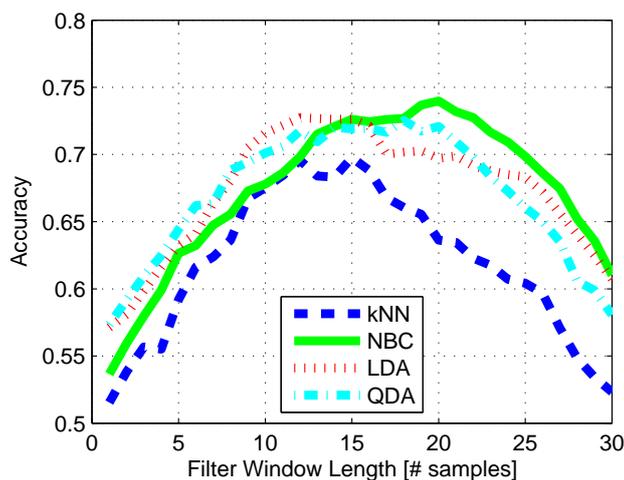


Fig. 4.14: The effect of filtering on localisation accuracy.

4.4.2 Pre-Processing Filter

Like most real-world signals the measurements quantifying radio signal quality are susceptible to noise. As a result, parts of the training data classification region overlap can be attributed to undesirable measurement noise. In an attempt to reduce the influence of measurement noise and obtain a “true” approximation of the signal a simple smoothing filter is employed. The chosen filter is one of the simplest types of low-pass filters; a Moving Average Filter (MAF). For each continuous signal; RSSI, LQ and CRSSI, filtering is performed by applying an N -sample moving window to the data and calculating the mean of the samples within each window as follows:

$$x_{filtered}(i) = \frac{\sum_{n=0}^{N-1} x(i-n)}{N}, \quad (4.1)$$

where i is the window position. Since CID is a categorical signal, it is not filtered.

Figure 4.14 illustrates the effect increased levels of filtering has on localisation accuracy when both the training and test data are preprocessed with the mean filter. It can be observed that the NBC performs best with a filtering window length of 20 samples. It can also be observed that the probabilistic classifiers outperform the completely data-based k NN algorithm at all filtering levels, indicating that probabilistic classifiers are more suited

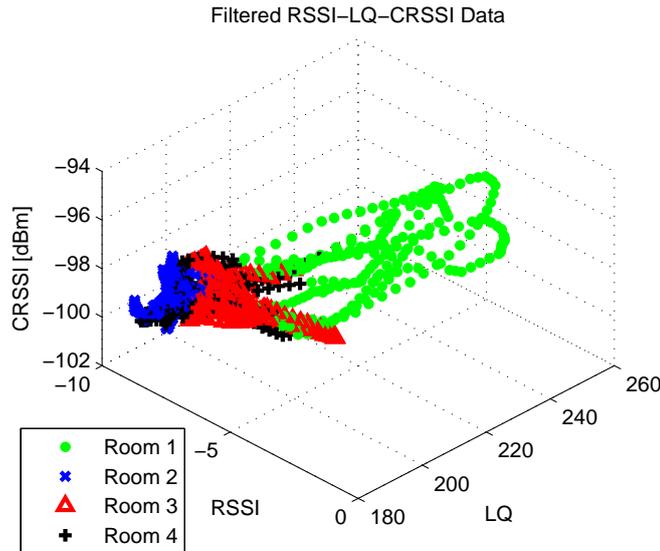


Fig. 4.15: The three input feature-space with a 20 sample mean filter applied.

to the practical localisation problem, whereas k NN is more suited to the trivial static localisation case. The improvement in accuracy can be understood by considering Figure 4.15. This filtered version of the dynamic training data has less nose-related overlap than its raw counterpart in Figure 4.13(b). There is, however, still some overlap which explains why it is impossible to achieve 100% accuracy with this data alone. Future chapters will attempt to augment this accuracy with other knowledge, such as room connectivity and user motion levels.

4.4.3 Signal Redundancy

To permit faster classifier execution and energy savings associated with obtaining less signals it is important to consider if certain signals are redundant and do not contribute to localisation accuracy. The CRSSI and CID readings were omitted and the accuracy as a function of filter pre-processing was calculated. Figures 4.16(a) and 4.16(b) illustrate the effect omitting CRSSI and CID have on localisation accuracy respectively. Most significant is the fact that ignoring the CRSSI value results in the same localisation accuracy, albeit with a different optimal filtering window size ($N = 15$). There is a slight reduction in accuracy when CRSSI is used instead of CID. This is because, although CID changes as a function of CRSSI, it is impossible to access the CRSSI value corresponding to a CID other than that of the currently strongest basestation. Otsason et al. (2005) overcame this issue by accessing the CRSSI for all cellular basestations detectable in a location and performing a form of fingerprinting with all the basestation's CRSSI. In this work, however, the test devices do not allow access to multiple CRSSI readings. Hence, it must be concluded that the available CRSSI reading is based on incomplete information and

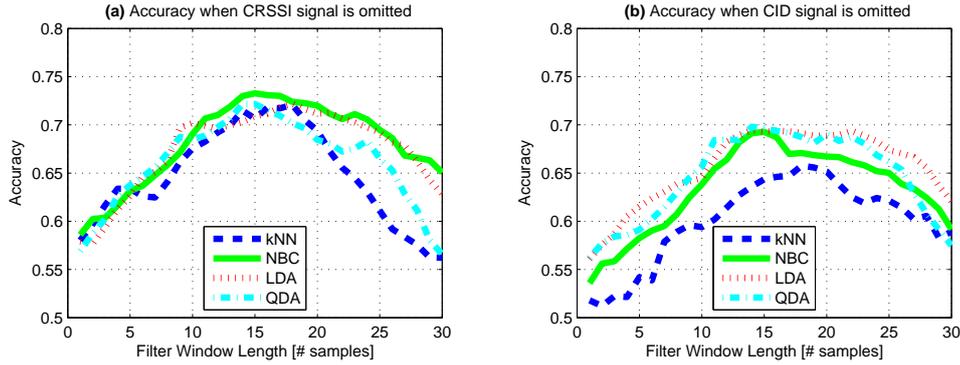


Fig. 4.16: Accuracy when (a)CRSSI and (b)CID signals are unavailable.

CID is the more informative signal.

Another interesting observation is that NBC is the superior probabilistic classification technique in Figures 4.14 and 4.16(a) which include the CID signal. But in Figure 4.16(b) all probabilistic classifiers obtain the same optimal accuracy. This can be attributed to the fact that all the signals used to obtain the accuracy levels in Figure 4.16(b) have a Gaussian covariance. However, since CID is derived from categorical data it is not Gaussian and it is certainly not covariate with the continuous signals. The NBC achieves slightly higher accuracy by realising this independence assumption, but it still treats CID as a Gaussian variable. Hence, it may be possible to obtain higher classification accuracy from the NBC classifier by modelling the CID variable as a discrete probability rather than a continuous Gaussian probability. This is only possible with NBC due to its simplifying feature independence assumption and will be investigated for the more comprehensive dataset obtained in Chapter 5.

Finally, it has been found that using only the RSSI signal results in exactly the same accuracy as using only the LQ signal (accuracy ≈ 0.65). This shows that, while RSSI gives very little location indicative information in the static case due to power control, it is as indicative as LQ in the dynamic localisation scenario. When these RSSI and LQ signals are both utilised they result in a peak localisation accuracy of 0.74, which is as accurate as if all the cellular signals were also included in the prediction. Hence, CRSSI and CID do very little to augment accuracy in this simple 4-room deployment scenario. When the localisation technique is applied to a less trivial localisation scenario RSSI and LQ signals may not be sufficiently diverse to represent location in a large number of rooms. As a result, a larger, more realistic deployment environment is considered in Chapter 5 and the importance of these cellular signals is illustrated.

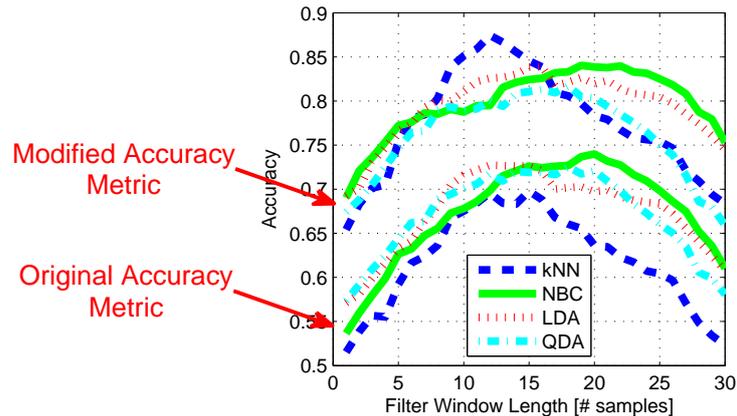


Fig. 4.17: The effect on localisation accuracy of using an accuracy metric more appropriate to the deployment scenario.

4.4.4 Accuracy Metric Suitability

As explained in Section 4.3.1, accuracies presented thus far have given an equal weighting for each room to the overall accuracy. This is still not an entirely valid assumption and can result in biased accuracy estimates. For example, Room 4 in the test environment is a room rarely inhabited during the course of a typical day since it is only a transition room between Rooms 1,2 and 3. If this room's contribution to the accuracy was ignored and accuracy was simply a mean of the accuracies for Rooms 1,2 and 3 the accuracy would look different. Figure 4.17 compares the accuracy metric when ignoring Room 4 with the original accuracy metric. It can now be seen that k NN is the more suitable classifier with a peak accuracy of 0.87.

Still it is unfair to assign an equal accuracy contribution to Rooms 1,2 and 3 while completely negating the contribution of Room 4. To better understand typical human behaviours and decide upon a more human-relevant accuracy metric, Chapter 5 will use highly accurate long-term location information generated from more costly, but more reliable, localisation hardware. This information will be used to generate accuracy metrics more relevant to a given user, hence leading to accuracy estimates more appropriate for a home localisation scenario.

4.5 Conclusions

This chapter has presented the localisation platforms considered for the purpose of single BSC home localisation. The signals available from each platform were illustrated and their suitability to the indoor localisation problem was assessed. Bluetooth[®] was chosen as the more suitable technology for its ubiquity, affordability and the wide array of sensor

readings available from a Bluetooth[®] mobile phone. Traditional multiple BSC localisation was compared to our technique for a static environment and improvements to the system for the dynamic scenario were highlighted.

Although this platform achieves relatively high performance with extremely low hardware requirements, it is still an open question whether this accuracy will scale to a large indoor environment with realistic long-term environmental variations. This chapter has addressed many of the issues related to positioning under realistic movements and has raised other questions related to large-scale, long-term reliability. It is clear that the data acquisition and labeling techniques presented so far will not scale up to such scenarios. Hence, the next chapter will use sensor redundancy and intelligent deployment techniques to develop an understanding of the ability of the system to function in a larger scale scenario both in terms of time and space.

Long-Term Deployment

As promising as the results are in the previous chapter, they cannot be assumed to reliably scale to larger environments or more lengthy deployments. For this reason, the Bluetooth[®] localisation system must be considered for long-term localisation accuracy in a realistic home environment. A realistic test environment should include typically encountered scenarios, such as large numbers of rooms, multiple storeys and multiple inhabitants in the environment. Hence, this chapter considers the deployment of the Bluetooth[®] localisation system in a realistic test environment, as illustrated in Figure 5.1. The adopted test environment consists of 13 rooms of interest labelled 1-13. The rooms include 5 bedrooms, 4 bathrooms, a kitchen, a living room, a study and two hallways.

The availability of data from a realistic long-term deployment allows a number of issues to be investigated;

1. Home location monitoring technologies commonly employed in the past such as Passive Infra Red (PIR), have accuracy issues, particularly in multiple occupancy situations where the identity of each user is unresolvable. The Bluetooth[®] localisation system is relatively immune to such multiple occupancy performance degradation since it relies on easily identifiable signals from the user.
2. The accuracy measure adopted in the previous chapter is suitable for assessing accuracy in the unrealistic scenario where the movements, and as a result, the quantity of test data in each room did not correspond to typical daily activities. Reliable long-term movement data will allow the determination of an accuracy metric which can more appropriately assess the accuracy of the system over long periods of time.

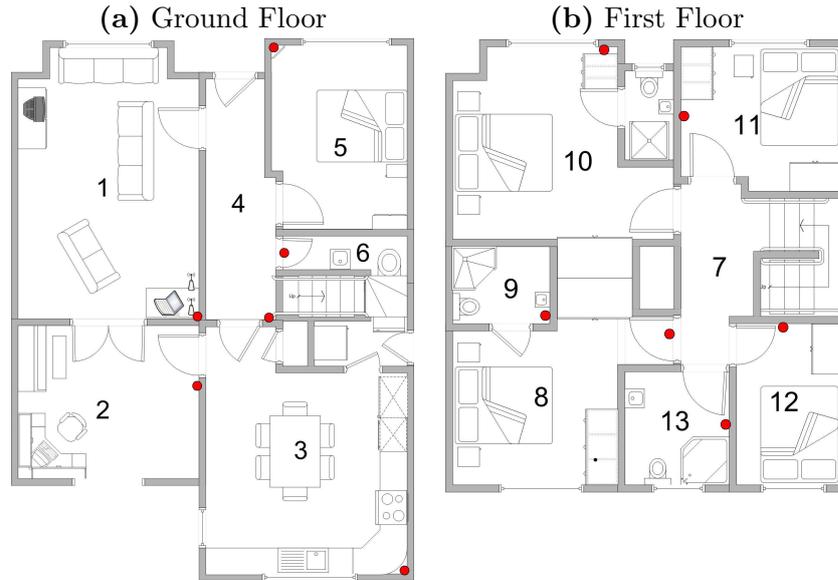


Fig. 5.1: The chosen realistic test environment. Red dots indicate the positions of the PIR sensors.

3. The signal diversity from a single AP may not be enough to allow discrimination between all the rooms in a large environment. Further signals can be obtained at the same single BSC to increase the distinctiveness of the fingerprint for each room.
4. As highlighted by using filtering in the previous chapter, directly applying the classification techniques to the data will not lead to the optimal classification accuracy. Pre-processing and post-processing of the data in a manner more specific to the deployment scenario will lead to higher long-term location prediction accuracy.

This chapter explores each of these issues. Furthermore it will highlight the maximum long-term accuracy possible with this localisation system in a realistic deployment scenario as a function of the amount of installed hardware.

5.1 Alternate Technologies

The main contribution possible from a long-term experimental localisation deployment is the validation of other affordable localisation techniques previously employed in a home monitoring scenario. Hence this section outlines the installation of and experimentation with two alternative localisation techniques; one technique typically used for home localisation and one technique custom-designed to provide 100% reliable location predictions.

5.1.1 Radio Frequency Identification

Radio Frequency Identification (RFID) is a technology which allows the retrieval of an identification code from either a powered or un-powered RFID tag. RFID allows the deduction of the proximity of an RFID tag-wearing subject to an RFID receiver, allowing estimation of the current room of a user with approximately one RFID reader per location (Lin et al., 2006). There is also the option of placing RFID tags around the environment and asking the subject to carry an RFID reader (Koch et al., 2007). This allows cheaper deployment costs since the reader is the most expensive component of an RFID localisation system, but it also results in higher long-term maintenance costs since RFID readers are extremely power intensive and require frequent charging. Hence, this work employs RFID technology to produce the accurate location predictions necessary to quantify the long-term accuracy of our system, in spite of the high long-term maintenance costs.

To minimise hardware costs, a unique approach to RFID localisation is adopted. Instead of installing one reader per location, or installing an extremely wide array of RFID tags throughout the environment, it was decided to install an RFID tag on each doorway. Then a lightweight handheld RFID reader, manufactured by Tracient Technologies[®], was carried by the experimenter and used to read the door tag as the door was passed. Upon reading a tag the reader sends the tag information to any connected Bluetooth device. To improve reliability, it was decided to use the Bluetooth[®] phone to store RFID readings, then relay them to the BSC when sending the cellular connectivity information. This prevents lost RFID readings when the BSC is out of range of the reader. This has the downside that the extra Bluetooth[®] connections cause a reduction in the Bluetooth[®] basestation signal sampling frequency to 0.5 Hz. During installation, all of the tags corresponding to each doorway were recorded. When the tag for a given doorway is detected during testing, the user's position is estimated based on their previous location and the doorway number.

To quantify the accuracy levels possible with this RFID technology 15 minutes of RFID data was obtained, with the experimenter moving constantly throughout the environment. While this data was being obtained, a voice recording noting the room transitions is made, identical to the method outlined in Section 4.4.1. The room labels derived from this voice annotated movement are assumed to be consistently accurate since they are not subject to any delays. This voice annotation data is compared to the RFID-derived location estimates in Figure 5.2. From this figure it can be seen that almost perfect location predictions are resultant from the RFID platform. The mean delay between the voice annotated data and the RFID predicted movements is 0.12 seconds and the standard deviation is 0.63 seconds. The mean error is extremely low when considering that anything up to an hour can be spent in a room at a time. The standard deviation can be explained by the fact that the RFID-derived room transitions are returned to the BSC with approximately one-

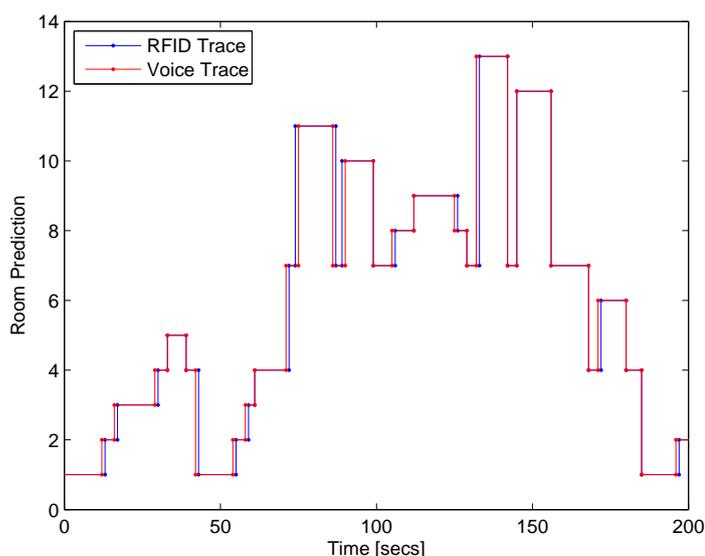


Fig. 5.2: Comparison of RFID derived location estimates and 100% accurate voice annotated movements for the first 200 seconds of test data.

second resolution. But even this variation in room-transition timing is acceptable when one considers the amount of time typically spent in a room at any one time.

This RFID localisation technique results in almost 100% localisation accuracy since it was not susceptible to environmental effects such as other humans moving and low RF signal diversity for each location. However, it cannot be considered a realistic localisation system since it requires the user to scan an RFID device every time they pass through a door. Instead this RFID localisation system is used as the baseline technology to obtain room labels for the training and test data for the Bluetooth[®] localisation technique and to quantify the accuracy of the Passive Infrared (PIR) localisation technique commonly employed in home monitoring systems. Using this reliable manual RFID localisation technique we were able to quantify the effects of human interference on the ability of our Bluetooth[®] technique and the PIR technique to resolve the location of an individual over long periods of time.

5.1.2 Passive Infrared

Passive Infra Red (PIR) is a technology commonly employed in academic (e.g. Lundell et al. (2007) and Pavel et al. (2007)) and commercial (e.g. QuietCare Systems (2009) and GrandCare Systems (2009)) elder home monitoring systems since it does not require the user to carry any mobile device. It simply requires the installation of a PIR sensor in every room, making it a completely passive monitoring solution. When the PIR in a given room detects motion it signals the BSC. When the BSC receives a signal from

a PIR sensor in a given room, it assumes that that room is the current location of the elder. The obvious downside to this technique is that when there is more than one person in the environment the identity of each person in each room cannot be resolved. Hence, this long-term deployment allows the estimation of the accuracy of the PIR localisation technique against the RFID-derived true location, over the range of occupancy levels likely to be encountered.

PIR localisation in these tests is enabled by deploying one PIR sensor in each room, in the positions illustrated by the red dots in Figure 5.1. The readings are relayed back to the BSC, in room 1, using the wireless X10 communication protocol. These readings are interpreted and logged using Intel’s BioMOBIUSTM health research platform. This can be considered an objective appraisal of the accuracy of a typical PIR localisation system since X10 is one of the communication protocols used by GrandCare’s (GrandCare Systems, 2009) commercial monitoring system, and possibly others, due to its relative affordability. The accuracy for this commonly used home localisation technique will be presented in Section 5.3.4. However, before accuracy for the PIR localisation technique and the Bluetooth[®] technique can be impartially compared, accuracy metrics more relevant to the particular deployment scenario must be considered.

5.2 Accuracy Metrics

This work utilises an accuracy metric not used in other localisation work since the focus is on the ability to predict which room the user is in, rather than a less meaningful coordinate position. Unless the application is for navigation, coordinate location would need to be converted to area or room level location to be meaningful to many applications. Navigation is not a useful service for someone in their own home, so coordinate location has little use in a home environment. Hence, the cumulative distribution function (cdf) derived accuracy measure utilised in a majority of indoor localisation work is not applicable to this work (see Youssef et al. (2003), Hightower and Borriello (2004) and Widyawan et al. (2007b) for examples). Consider the accuracy measures in Tables 2.1 and 2.2 in Chapter 2 of this thesis. Examples of the coordinate location prediction accuracy measures used are “2.94m median location error”, “less than 4m error 98% of the time” and “less than 2m error 90% of the time”. Not only do these measures fail to indicate the ability of the systems to resolve location in human-understandable terms but they are also difficult to conclusively compare with each other.

Section 4.3.1 has already outlined a novel location prediction accuracy metric, which uses an equal contribution of the prediction accuracy for each room towards the overall accuracy. This metric eliminates the bias resulting from obtaining a disproportionate number of samples in a given room during the extremely short test data acquisition period

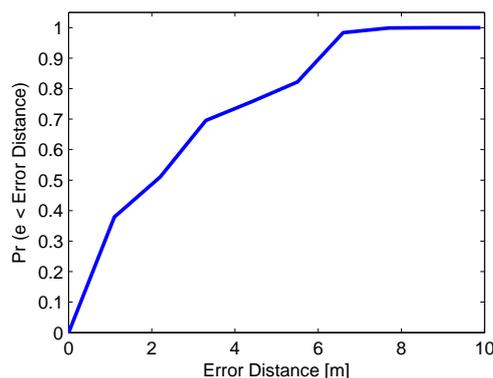


Fig. 5.3: Cumulative distribution function for LDA location classification of a day's worth of test data.

of 15 minutes. To illustrate how the commonly used cdf accuracy measure compares to our equal contribution prediction success metric, a simple test was conducted. The LDA classifier is applied to a day's worth of test data from the large test environment.¹ For each true position and predicted position combination, an error distance is estimated from the distances between the centres of the rooms using the map in Figure 5.1. Based on this entire set of error distances a cdf can be constructed indicating the probability of returning an error less than each given distance. Figure 5.3 illustrates the resultant cdf. From this cdf it can be deduced that less than 3m error can be obtained 65% of the time and that the median error distance is approximately 2m. Although this sounds competitive with other work, the same set of predictions yields an equal contribution prediction success measure of only 28.7%. Hence, previous work which uses coordinate location predictions give no indication of the ability of the technique to resolve symbolic, or room-level, location.

The equal contribution success metric is more relevant than error-distance metrics to application-focused localisation systems since it quantifies the ability of a localisation system to correctly predict each room in a user-understandable format. A disadvantage of this technique, as has been alluded to in Section 4.4.4, is that an equal contribution of each room does not give a realistic appraisal of the long-term accuracy of the localisation technique. Rooms which are rarely inhabited and as a result would have little effect on long-term accuracy, are given as much importance as frequently inhabited rooms. The availability of long-term movement data now allows the calculation of a more application-focused accuracy metric.

The accuracy metric thus far is calculated from the arithmetic mean of the accuracy for each room, defined as

¹The acquisition technique for this data is described in the next section. Knowledge of how the location predictions are obtained is not yet necessary. All that is of interest is that a set of true positions and the corresponding predicted positions is available.

$$\bar{a} = \sum_{k=1}^K a_k \cdot w_k, \quad (5.1)$$

where

$$w_k = 1/K, \quad (5.2)$$

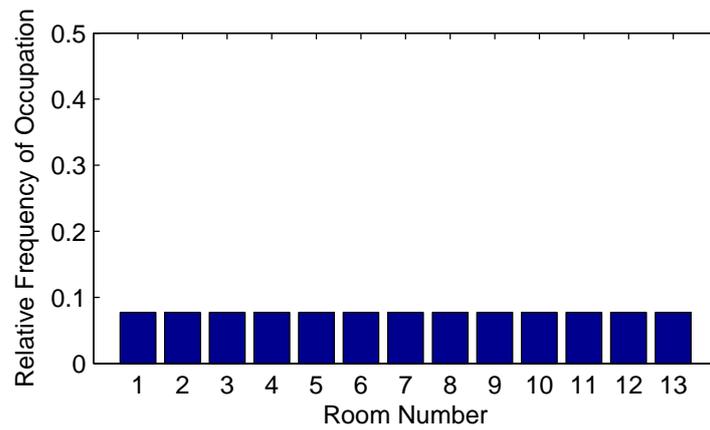
and

$$[a_1, a_2, \dots, a_k, \dots, a_K] = \text{diag}(C), \quad (5.3)$$

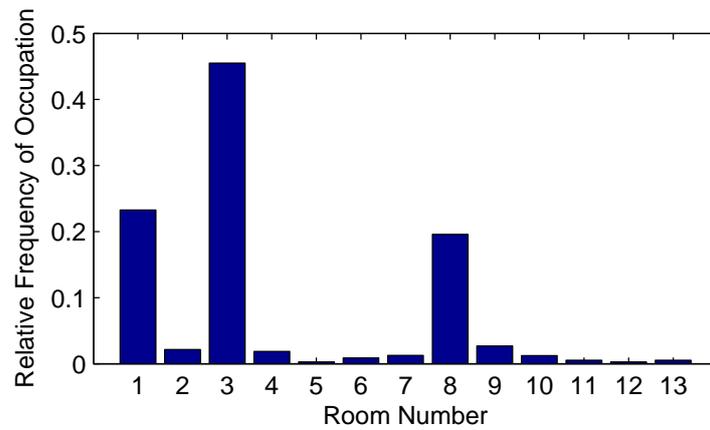
where K is the number of rooms in the environment and C is the confusion matrix for the room predictions. An equal w_k across all classes leads to an equal weighting for the accuracy of each room towards the overall mean accuracy, as illustrated in Figure 5.4(a). The availability of accurate RFID-derived room labels, however, allows the estimation of the amount of time a person spends in each room during a given period of time, in this case, a week. Over 7 consecutive days, RFID label data was obtained while the experimenter conducted their usual activities within the house. Movement data was only obtained during the day because the night-time movements are confined to one room and would bias the overall accuracy assessment of the system. Furthermore, a realistic home monitoring system would most likely utilise alternative sleep monitoring technologies (see Behan et al. (2008) for example), making a room localisation system redundant at night. Each day started at 10am and ended at 8pm to avoid discrepancies in days when the subject got out of bed early or went to bed early.

The availability of a week of accurate RFID-derived daytime movement data allows the calculation of a normalised histogram of room occupation, as illustrated in Figure 5.4(b). Instead of using an equal weighting for each room, the weighting for each room can now be set equal to the relative frequency of occupation of each room over the entire data acquisition period. Hence, this Empirical Accuracy (EA) metric can incorporate the importance of each room into the accuracy measure, based on the long-term frequency of occupation of that room. This means that the localisation accuracy evaluated using this metric, no matter how short the experimentation period is, will represent the correct room classification rate of the system over the period of an entire week.

Now that a more application-specific localisation accuracy metric has been developed, the localisation accuracy of the deployed system over long periods of time in a realistic environment can be explored.



(a) Weightings for an equal contribution from each room towards the overall accuracy



(b) Weightings for the empirically derived contribution from each room towards the overall accuracy

Fig. 5.4: Comparison of the equal contribution mean accuracy weights and empirically derived weights.



Fig. 5.5: Examples of RFID tags affixed to doors at positions convenient for manual scanning. Note that the numbers are only written on the tags for installation purposes, the RFID system automatically derives these numbers from the stored tag IDs.

5.3 Long-Term Tests

The long-term deployment scenario allows the validation of the ability of the Bluetooth[®] localisation technique to resolve the room-level location of the user over long periods of time. The most important requirement for useful long-term tests is the acquisition of reliably accurate location labels. This requirement is fulfilled by the use of the manual RFID localisation technology. To allow this technology to reliably detect room transitions, each door had an RFID tag attached to it, as illustrated in Figure 5.5. Each side of each door has an RFID tag, each one corresponding to a particular direction of movement through the door, which is converted to a room number during offline data labelling.

Since this chapter addresses the task of observing movements of a subject over significant periods of time, it was decided to focus on the performance over the period of a day. This required an entire day of testing data for each of the experiments outlined in the following sections. Because a day of test data was necessary, it was decided to also use a second day of data as the training data for each experiment. With the two sets of data available for each experiment it is possible to perform two tests, as in Section 4.3. The first test involves training each classifier on data from the first day and testing on data from the second day. Then the datasets are swapped and the test repeated. The mean accuracy of the two tests can then be calculated.

During data acquisition, the environment was mostly only inhabited by one person. However, 4 other people live in the house besides the experimenter. Since this was a realistic home environment it was unavoidable that other people periodically enter and leave the environment. Hence it was necessary to manually note how many people were present in the environment at all times. Figure 5.6 compares the PIR and RFID location estimates with the levels of occupancy over a two hour period. This manual record of

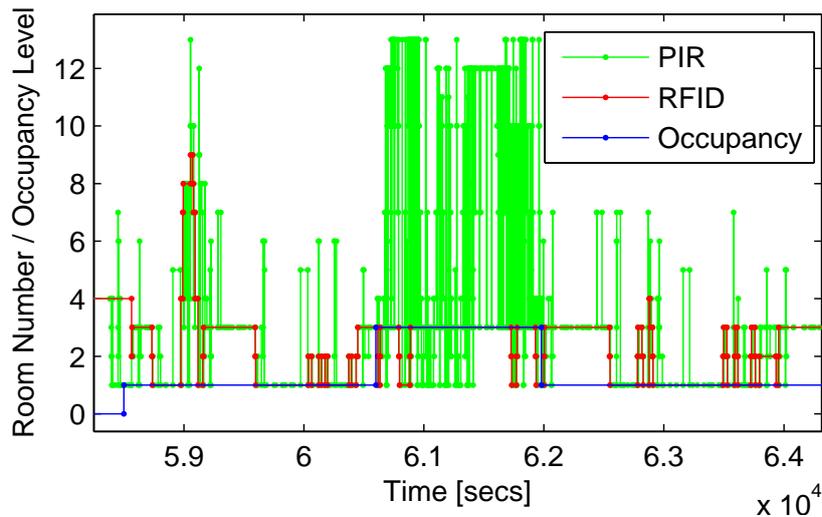


Fig. 5.6: Comparison of occupancy levels with PIR-derived location and accurate RFID derived location. Notice how PIR location becomes extremely varied and uncorrelated with RFID when occupancy goes to 3 people.

occupancy allows the available data from each day to be parsed according to the number of people present. Sections 5.3.1, 5.3.2 and 5.3.3 will consider the performance of the Bluetooth[®] localisation system at all levels of occupancy which could be encountered in a typical scenario. During these tests the number of inhabitants can change from just one person, the person being tracked, up to and including 4 other inhabitants. The occupancy levels change in an uninhibited and realistic fashion. Then Section 5.3.4 will compare our localisation technique with the PIR localisation technique for commonly encountered occupancy levels.

5.3.1 Single Bluetooth[®] Access Point

The first scenario to be considered is the absolutely minimal deployment of a BSC with a single Bluetooth[®] Access Point (AP) available. The computer with a BlueRadios AP is deployed in room 1 in the position illustrated in Figure 5.1. Then two consecutive days of data were obtained. Again, data acquisition for each day starts at 10am and ends at 8pm. These datasets are referred to as the Single AP (SAP) datasets. For the single AP scenario the signals employed for location classification are Bluetooth[®] RSSI, Bluetooth[®] LQ, phone CID and phone CRSSI, as described in Chapter 4. Since this environment has significantly more rooms than the simple test environment employed in the previous chapter, there will be higher levels of feature overlap possible with all classes. This makes sufficiently flexible density estimation difficult with the unimodal density estimation techniques considered thus far and in previous work. As such, it is necessary to also consider the more sophisticated density estimation technique; the

	kNN	Naive Bayes	LDA	QDA	GMM
Day 1	0.18	0.26	0.29	0.19	0.20
Day 2	0.22	0.25	0.33	0.19	0.19
Mean	0.20 (0.03)	0.26 (0.01)	0.31 (0.03)	0.19 (0.00)	0.19 (0.01)

Table 5.1: Mean General Accuracy for location predictions over two consecutive days. Standard deviations are in parentheses.

	kNN	Naive Bayes	LDA	QDA	GMM
Day 1	0.63	0.52	0.51	0.57	0.62
Day 2	0.70	0.53	0.54	0.71	0.59
Mean	0.67 (0.05)	0.53 (0.01)	0.52 (0.02)	0.64 (0.09)	0.60 (0.02)

Table 5.2: Mean Empirical Accuracy for location predictions over two consecutive days.

Gaussian Mixture Model (GMM).

Tables 5.1 and 5.2 highlight the mean of the accuracies from using the first and then the second day of data as the test data for a range of prediction algorithms. The prediction algorithms are k -Nearest Neighbour (k NN), Naive Bayes Classifier (NBC), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA) and Gaussian Mixture Model (GMM) as described in Chapter 3. The accuracies are calculated using both the General Accuracy (GA) metric, which refers to the accuracy metric which was originally employed in Chapter 4 and the EA metric described in Section 5.2. It can be observed that the mean GA is relatively low for all location prediction algorithms in this long-term scenario. This is due to the uneven distribution of available data for each room (reflecting the frequency of room occupancy), causing classifiers to bias their performance towards those rooms with higher quantities of data. Hence, these rooms are predicted more frequently than the less commonly inhabited rooms, which would lead to higher long-term accuracy. GA, however, does not highlight the long-term accuracy since it assumes an equal amount of time is spent in each room during a typical day.

Table 5.3 shows the confusion matrix from the k NN classifier applied to the SAP dataset for day 1. The off-diagonal elements represent the misclassifications for the room labels in the horizontally adjacent diagonal elements. It can be seen that many of the rooms with poor recognition rates are frequently misclassified as room 3. Figure 5.7 illustrates the frequency of correct room predictions for each room on day 1, which is obtained directly from the diagonal of the confusion matrix for the k NN classifier. When comparing this with Figure 5.4 it can be seen that the most frequently inhabited rooms generally have higher accuracy and the less frequently inhabited rooms generally have lower accuracy; causing the extremely low GA. By using the EA measure, these individual room accuracies can be translated to a long-term estimate of accuracy. As a result, Table 5.2

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.82	0.04	0.12	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
2	0.44	0.12	0.39	0.01	0.00	0.00	0.03	0.00	0.01	0.01	0.00	0.00	0.00
3	0.03	0.00	0.90	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00
4	0.28	0.02	0.65	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.04	0.91	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.05	0.00	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.06	0.01	0.47	0.03	0.00	0.00	0.08	0.15	0.20	0.00	0.00	0.01	0.00
8	0.01	0.00	0.43	0.01	0.00	0.00	0.09	0.12	0.25	0.00	0.00	0.01	0.08
9	0.16	0.00	0.63	0.00	0.00	0.00	0.00	0.14	0.06	0.00	0.00	0.00	0.00
10	0.01	0.00	0.35	0.00	0.00	0.00	0.05	0.06	0.51	0.00	0.00	0.00	0.01
11	0.01	0.00	0.93	0.01	0.00	0.00	0.02	0.00	0.00	0.00	0.01	0.00	0.01
12	0.00	0.00	0.29	0.00	0.00	0.00	0.07	0.29	0.00	0.00	0.00	0.21	0.14
13	0.00	0.00	0.62	0.00	0.00	0.00	0.03	0.13	0.14	0.00	0.00	0.06	0.02

Table 5.3: Confusion matrix for k NN classifications from the SAP dataset, day 1. The diagonal elements, in bold, are the correct room predictions.

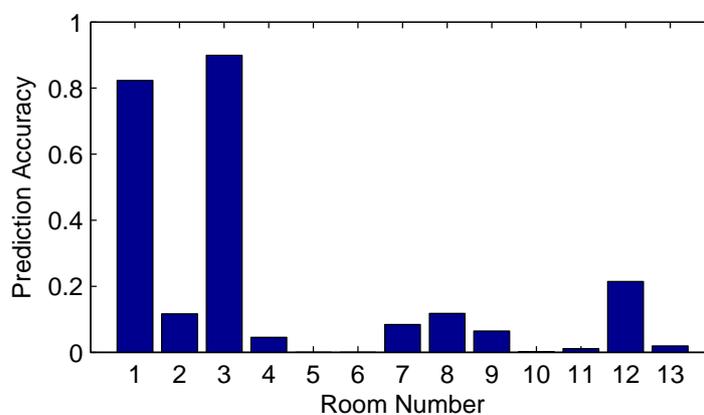


Fig. 5.7: Frequency of correct room prediction for the k NN classifier. These values are obtained directly from the confusion matrix for the classifier.

better illustrates how this system will generally perform over long periods of time.

When considering Table 5.2 it can be established that the completely data-based k NN classifier performs the best. It can also be observed that the probabilistic classifiers, QDA and GMM, achieve high accuracy, approaching that of k NN. They also have the advantage over k NN in that their online execution is significantly more efficient since they are parametric classifiers, instead of an entirely data-based classifier like k NN.

To achieve this accuracy it was necessary to spend a day obtaining the training dataset. Obtaining the training dataset is time-consuming and would most likely be performed by a professional rather than the subject themselves to ensure persistent scanning of RFID tags and, as a result, training dataset reliability. Hence it is necessary to investigate if similar levels of accuracy can be achieved with smaller quantities of training data in each room. To this end, the number of data points available per class was modified by deleting all but the first N training data points in each class. Figure 5.8 highlights the effect of

increasing the maximum number of training data points available per room on the EA. In general, increasing the maximum number of samples available for each room increases the EA. GMM exhibits the highest accuracy at approximately 3000 samples per room.

However, it should be noted that only the three most commonly inhabited rooms have such a high number of training samples, as illustrated in Figure 5.10. When a high number of samples per room is permitted for training, approximately one third of the rooms still only have less than 100 training samples available, since those rooms were rarely inhabited during the training data acquisition phase. This training dataset imbalance, however, leads to higher EA as the permitted number of training samples per location increases since it biases the classifiers towards producing higher accuracies for the more frequently inhabited rooms. Hence, using a quantity of data for each room more representative of the time spent in a room during a typical day leads to higher localisation accuracy.

To understand why similar accuracy is not possible when using the GA measure it is important to consider the effect a greater number of samples per room has on the GA. Figure 5.9 compares the k NN, QDA and GMM algorithms for accuracy with increasing numbers of samples per room for both the GA and EA measures. It is evident that, while the EA increases with higher numbers of samples permitted per room, GA shows no improvement. This indicates that higher accuracy for the commonly inhabited rooms occurs with more samples available per room. Instead of focusing on the more important, commonly inhabited rooms, the GA measure masks the effect of higher accuracy in important rooms with lower accuracy in the less important rooms. Hence, to obtain maximum long-term accuracy it is necessary to have a sufficiently imbalanced training dataset and validate the results with an accuracy measure befitting the desired long-term accuracy.

Section 4.4.3 indicates that the cellular signals, CRSSI and CID, do not contribute to localisation accuracy in the smaller, more trivial, localisation scenario. It also suggests, however, that a less trivial localisation scenario, such as the one under consideration in this chapter, will benefit from the inclusion of such cellular signals. Using all of the signals available at a single Bluetooth[®] AP, RSSI, LQ, CRSSI and CID, results in a peak empirically-derived accuracy measure of approximately 70%. Table 5.4 highlights the localisation accuracy possible when the cellular connectivity readings are unavailable to the location prediction algorithms. It can be seen that the accuracies for all except the GMM algorithm are significantly reduced from those in Table 5.2. Hence, the inclusion of the cellular signals, which are available at no extra cost, leads to higher localisation accuracy in a realistically large home environment. In an effort to obtain further location-indicative signals at little extra cost, the next section considers the effect of using another Bluetooth[®] AP at the same single basestation computer to increase signal diversity throughout the environment.

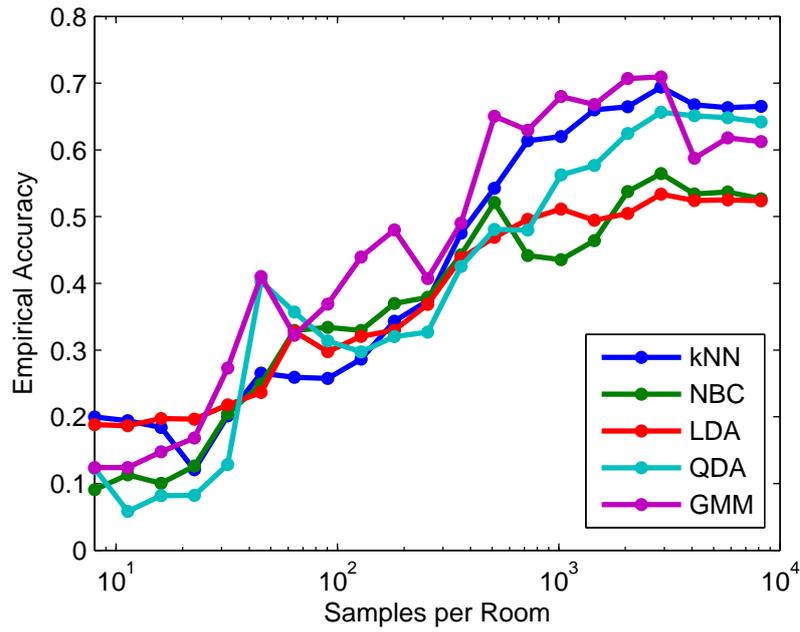


Fig. 5.8: Effect of increasing maximum permitted samples per room on the empirically derived accuracy measure.

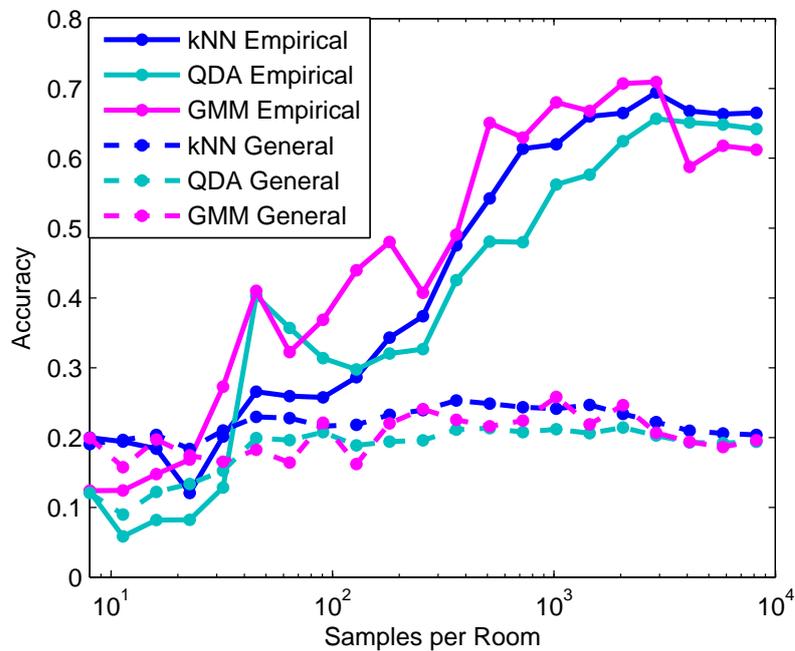


Fig. 5.9: Effect of increasing maximum permitted samples per room on EA and GA for *k*NN, QDA and GMM classifiers.

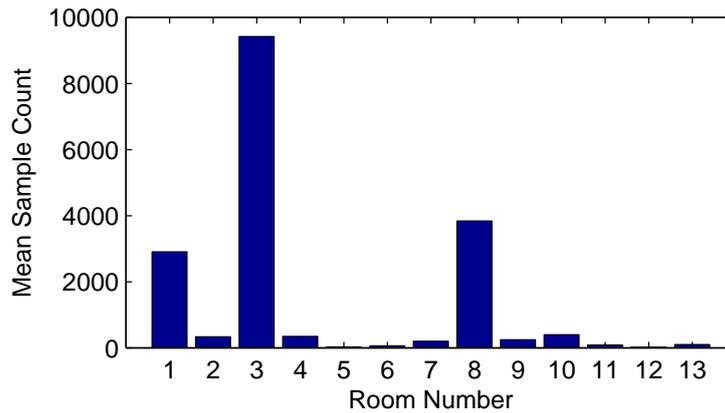


Fig. 5.10: Mean number of samples per location per day for the SAP dataset.

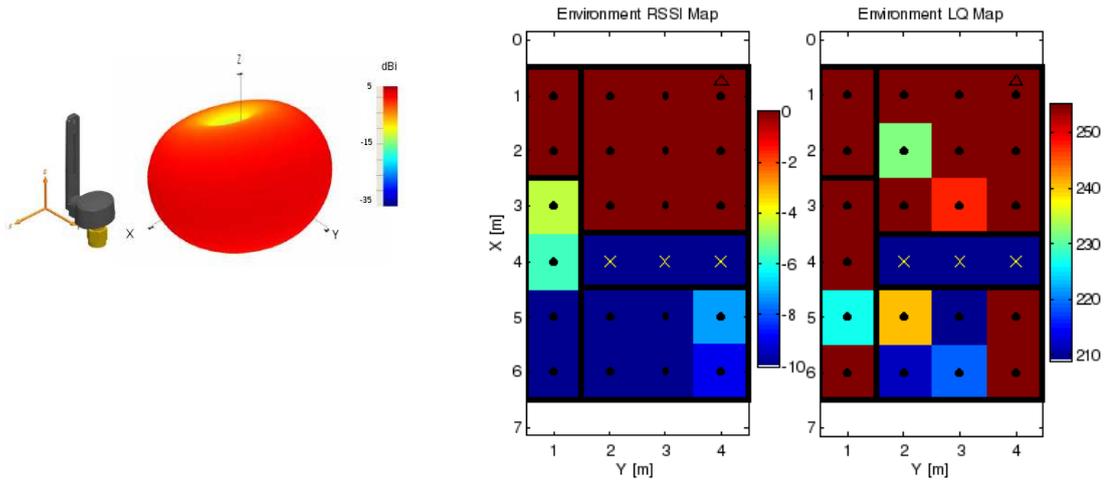
	kNN	Naive Bayes	LDA	QDA	GMM
Day 1	0.53	0.47	0.51	0.60	0.63
Day 2	0.52	0.28	0.36	0.64	0.70
Mean	0.53 (0.01)	0.37 (0.13)	0.43 (0.11)	0.62 (0.03)	0.66 (0.05)

Table 5.4: EA over two days when cellular connectivity information is no longer available.

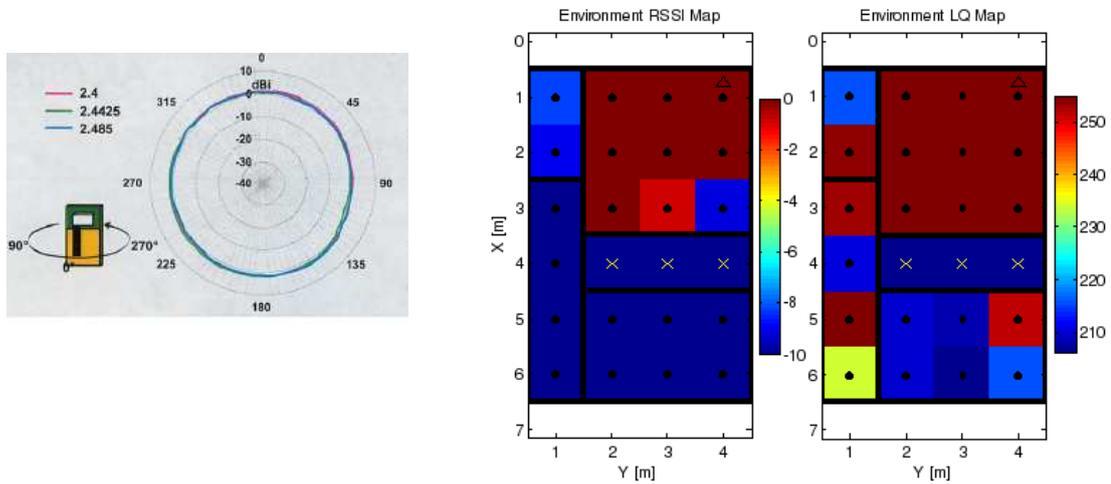
5.3.2 Single Basestation Computer with Increased Signal Diversity

Remote localisation systems typically increase signal diversity for each location by deploying several basestations throughout the environment. This dramatically increases the installation cost of a localisation system if the hardware does not already exist within the environment. To increase signal diversity while retaining minimal installation overhead it was decided to use a second Bluetooth[®] access point at the original BSC. To illustrate the influence different types and orientations of access point antennas have on the RSSI and LQ signals throughout an environment, the controlled environment presented in Chapter 4 is considered. Figure 5.11 illustrates the effect on changing antenna type and orientation on the antenna radiation profile, hence environment RSSI and LQ profiles. The antenna position remains constant throughout all tests. Figure 5.11(a) shows the antenna used for all the single AP experiments in this Thesis. The direction of the antenna is indicated by the bode plots in each figure. This profile is similar to that in Figure 4.6 since it was obtained using the same antenna type and orientation.

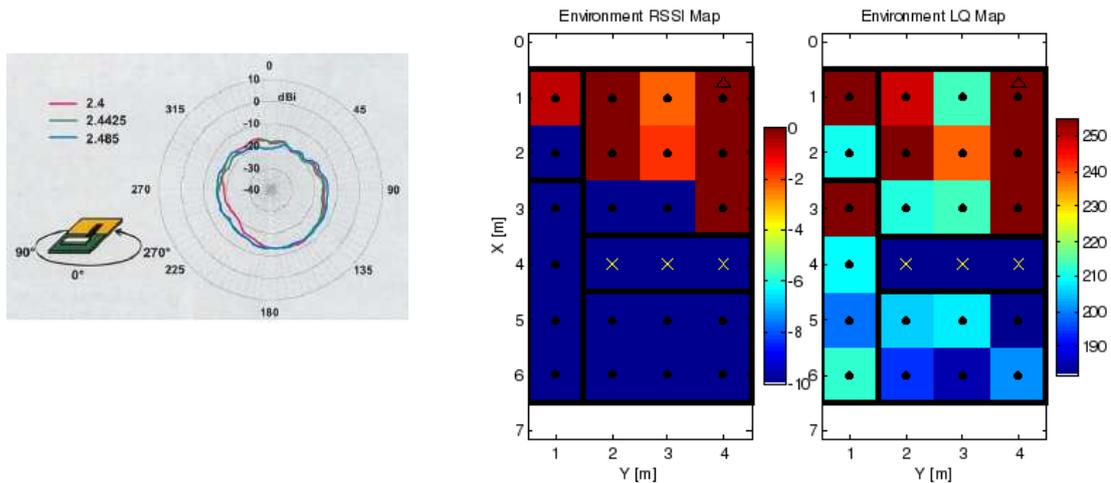
A second Blueradios Bluetooth[®] chip was considered, one with a ceramic antenna instead of a monopole antenna. When orientated as illustrated in Figure 5.11(b) it exhibits a radiation profile with slightly lower intensity than the monopole antenna. This is a result of changing the effective aperture (A_e) and polarisation (PLF) which influence the received signal intensity as described in Section 2.1.1.1. Hence, the indoor signal profile



(a) Antenova antenna (Antenova Titanis Product Specification, 2009)



(b) Blueradios Ceramic Antenna (Ceramic Chip Antenna Data Sheet, 2003)



(c) Alternative Orientation (Ceramic Chip Antenna Data Sheet, 2003)

Fig. 5.11: The influence of access point antenna orientation and type on antenna radiation profiles, hence, environment signal profiles.

differs from that given in Figure 5.11(a). Intuitively the availability of signals from both APs leads to better ability to discriminate between different rooms. This is certainly the case for the antenna orientation in Figure 5.11(c) which results in a severely attenuated radiation profile, hence, different environment signal profiles.

This feature of antenna-related signal diversity can be exploited in the large test environment. This is achieved by connecting two access points to the BSC, one orientated as illustrated in Figure 5.11(a) and one as illustrated in Figure 5.11(c). Since the Bluetooth[®] chip used by the mobile phone is unable to derive the link RSSI or LQ information, this has to be derived by the AP Bluetooth[®] chips. There is also the significant limitation that it is impossible to connect two APs to one Bluetooth[®] device as a master: each Bluetooth[®] device can have only one master and up to 7 slave devices (Bluetooth[®] Special Interest Group, 2001). Hence, the mobile phone must be programmed to establish a connection to one of the APs before sampling takes place. Then the BSC can establish the second Bluetooth[®] connection to the phone and proceed to retrieve RSSI and LQ samples from both APs at regular intervals. This results in 6 signals available at the single BSC, the original RSSI, LQ, CID and CRSSI signals along with the second AP RSSI and LQ signals.

	kNN	Naive Bayes	LDA	QDA	GMM
1 AP	0.67 (0.05)	0.53 (0.01)	0.52 (0.02)	0.64 (0.09)	0.64 (0.04)
2 APs	0.74 (0.10)	0.38 (0.14)	0.55 (0.03)	0.67 (0.13)	0.71 (0.04)

Table 5.5: The effect of the availability of a second Bluetooth[®] AP co-located with the original Bluetooth[®] AP on EA.

Again, two days of test data was obtained with this configuration, referred to as the Single Basestation (SBS) dataset, and the effect on the maximum number of samples available per room on mean localisation accuracy was investigated. Table 5.5 shows how the extra signal diversity allows slightly higher EA accuracy for all but the Naive Bayes technique. The increased accuracies are only slight since the second AP is orientated to obtain a radiation profile as different as possible from that of the first AP. As a result, the second antenna is orientated as illustrated in Figure 5.11(c). However, as illustrated in Figure 5.11(c), with this orientation the signals fall off to the minimum value closer to the basestation. Hence, higher signal diversity with the second AP is only obtainable in areas very close to the BSC. This is corroborated by comparing the confusion matrix in Table 5.6 with the SAP confusion matrix in Table 5.3. It can be seen that the SBS deployment has marginally higher accuracy in room 1 where the BSC is deployed, in the horizontally adjacent rooms 2,3 and 4 and the vertically adjacent rooms 7, 8 and 9. All other rooms have accuracy similar to the single AP scenario. This means that higher location discrimination with a second AP is only possible in locations very close to the

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.84	0.03	0.12	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.15	0.31	0.47	0.04	0.00	0.00	0.03	0.00	0.00	0.01	0.00	0.00	0.00
3	0.04	0.01	0.94	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.17	0.06	0.65	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.00	0.00	0.91	0.04	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.03	0.03	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.02	0.03	0.49	0.03	0.00	0.00	0.10	0.13	0.17	0.00	0.00	0.01	0.02
8	0.10	0.01	0.31	0.00	0.00	0.01	0.10	0.16	0.23	0.00	0.00	0.00	0.07
9	0.05	0.01	0.75	0.00	0.00	0.00	0.01	0.03	0.14	0.00	0.01	0.00	0.00
10	0.03	0.01	0.30	0.00	0.00	0.00	0.14	0.04	0.45	0.00	0.00	0.00	0.01
11	0.07	0.01	0.81	0.01	0.00	0.01	0.01	0.00	0.03	0.00	0.05	0.00	0.00
12	0.00	0.07	0.07	0.00	0.00	0.00	0.14	0.07	0.36	0.00	0.21	0.07	0.00
13	0.00	0.01	0.61	0.00	0.00	0.00	0.08	0.14	0.13	0.00	0.00	0.02	0.02

Table 5.6: Confusion matrix for k NN classifications from the SBS dataset, day 1.

BSC, such as rooms 1, 2, 3, 4, 7, 8 and 9.

This confirms that increased accuracy in certain locations is possible by increasing the signal diversity in that location. By deploying a second AP at the original basestation computer it is possible to marginally increase accuracy with little extra deployment cost. This has not significantly increased the overall empirically-derived localisation accuracy since it was impossible to sufficiently increase signal diversity in many of the rooms in the environment. This is a limitation of the transmission range of low-power Bluetooth[®] chips such as those found in mobile phones. The next section shall investigate the effect of increasing signal diversity in all rooms by deploying cheap Bluetooth[®] beacons in the environment.

5.3.3 Signal Diversity from Multiple Beacons

The previous two sections highlighted the accuracy possible with a single BSC installed within the environment. The availability of higher signal diversity, due to different APs with different antenna radiation profiles, led to slightly higher accuracy with no extra installation effort. To further increase signal diversity throughout the deployment environment it is necessary to be able to install Bluetooth[®] APs in different locations throughout the environment. However, as with previous work it is extremely cost-prohibitive to install BSCs to manage each of these APs. Instead, a novel approach is employed, whereby only “dumb” APs are installed and powered up without any connection to a controlling BSC, as illustrated in Figure 5.12. Then the Bluetooth[®] phone connects to the “dumb” AP or beacon. Since the Bluetooth[®] chip in the mobile phone is unable to calculate RSSI or LQ information, it must place the AP in command mode and remotely perform operations on the AP to calculate these readings.

This has the result that the Bluetooth[®] phone can retrieve the remote RSSI and LQ readings for connections to all beacons within the environment. These readings can

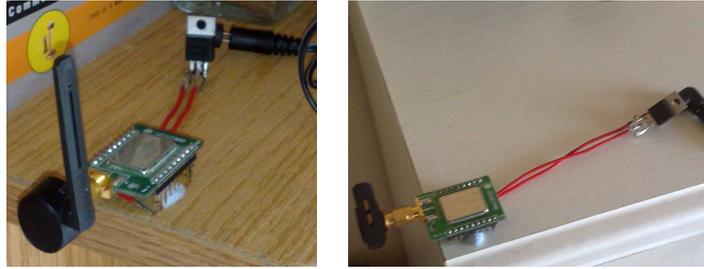


Fig. 5.12: Examples of the deployed “dumb” access points.

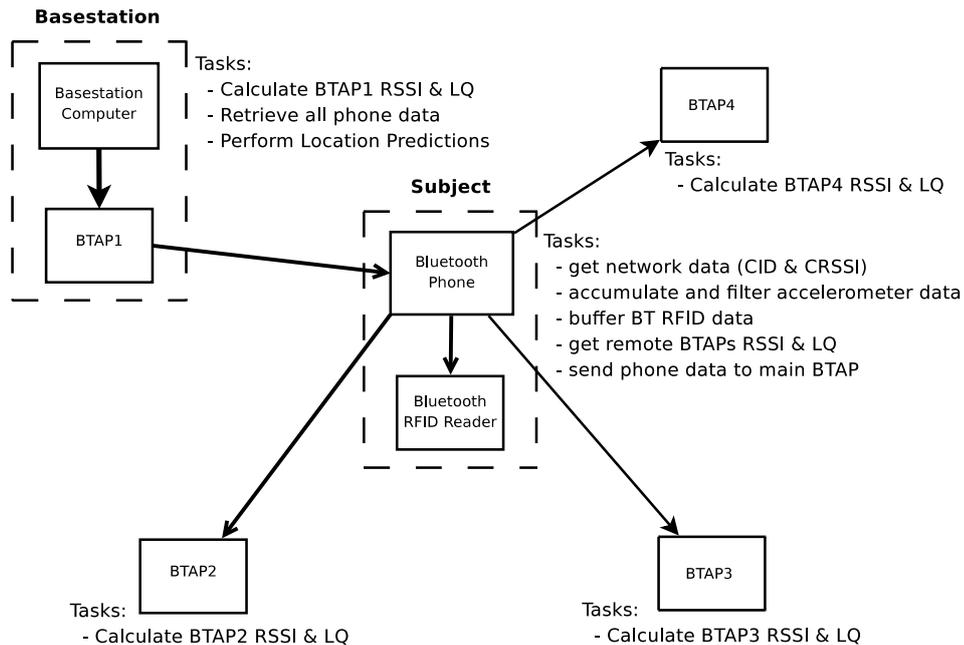


Fig. 5.13: The connection topology for the multiple AP deployment. Each arrow represents a Bluetooth[®] connection. The start of each arrow indicates the connection master and the end of each arrow represents the slave.

then be relayed to the BSC via its Bluetooth[®] connection, as presented in Figure 5.13. Throughout a typical day it is possible for connections to temporarily drop, due to T-R distance and attenuation from obstructions. In this case the phone must re-establish the connection and continue to obtain RSSI and LQ readings. Even if a connection is dropped, default values indicating a dropped connection must be stored, since the lack of a connection from a given AP is characteristic of certain locations. When the BSC connection is dropped, the readings from the other APs cannot be retrieved. For this reason the phone must store all remote BT AP measurements and RFID readings and return them to the BSC when it is reconnected. To minimise occurrences of dropped BSC connections, it was decided to install the BSC in the centre of the environment.

Prior to this thesis it has not been possible to obtain RSSI or LQ readings on a

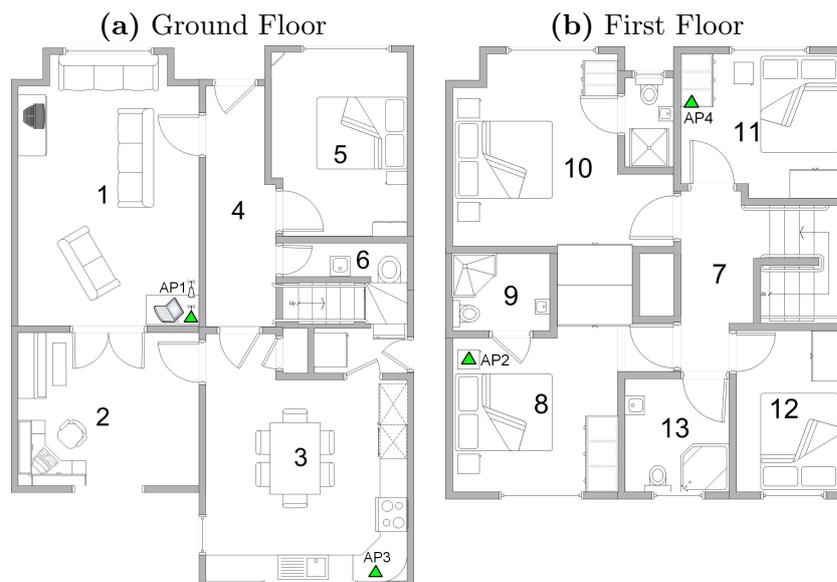
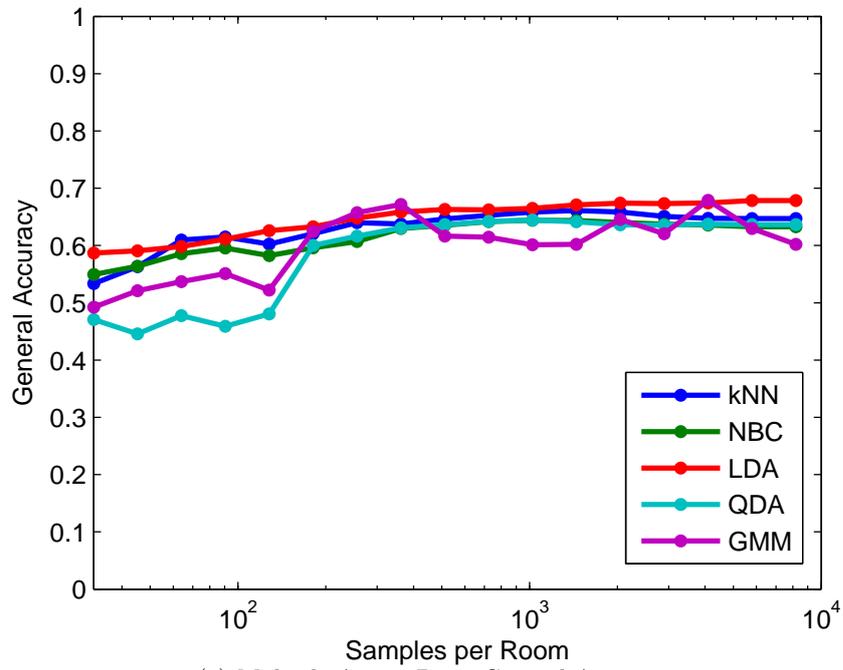


Fig. 5.14: Locations of the APs, indicated by green triangles.

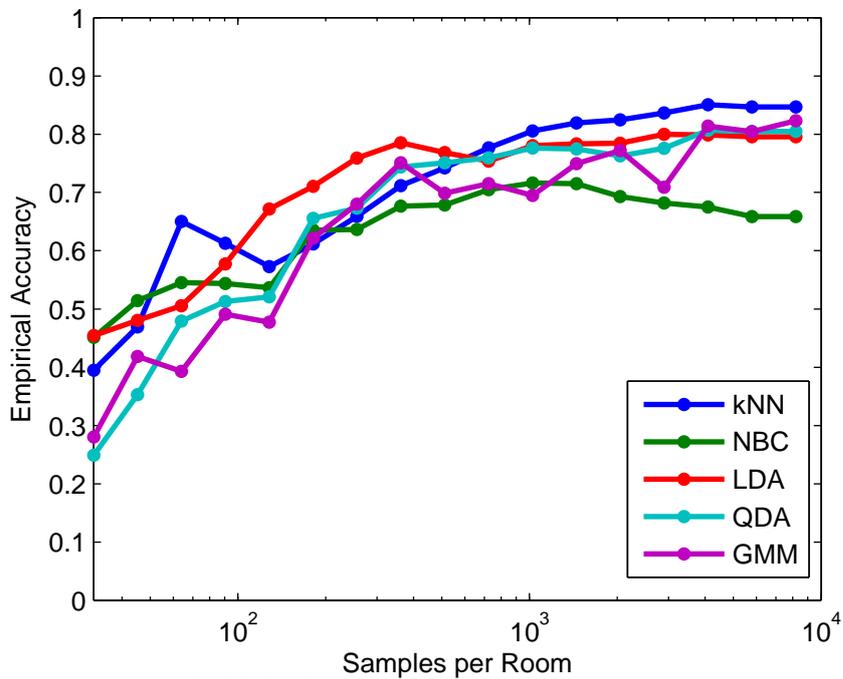
Bluetooth[®] phone and the acquisition of these high resolution signals with such cheap hardware is a direct result of the novel connection topology employed. Hence, the position estimates can be produced at the BSC as before, using the extra available signals. Naturally, the availability of a majority of the signals locally on the mobile phone would allow the location predictions to take place on the mobile phone itself, but the calculations remain on the BSC to save energy and prolong battery life of the mobile phone.

The locations of the deployed Bluetooth[®] beacons are illustrated in Figure 5.14. The positions are chosen to maximise the separation of the beacons in three dimensions while favouring positions which are in proximity to plug sockets, for ease of installation. Again, two days of data was obtained for this experiment. These datasets will henceforth be referred to as the Multiple AP (MAP) datasets. As before, an investigation into the effect of the quantity of available data on the localisation accuracy was conducted. Figure 5.15 illustrates the mean GA and EA for such a deployment. The mean GA in Figure 5.15(a) is almost 60% with minimal quantities of data available in each room. When the maximum number of samples is available per room, the mean GA increases to 68%.

The EA in Figure 5.15(b) is less than 50% for all classifiers when there is only minimal data available for each room. As in the single AP scenario, the multiple AP EA significantly increases when all the data for each room is used as training data. The maximum EA is 85% using the k NN method. Table 5.7 allows an understanding of the source of the increased accuracy over the SAP and SBS methods; higher recognition rates occur for nearly all of the rooms due to the higher signal diversity in all rooms. After the k NN method, the probabilistic classifiers which use covariance information in their



(a) Multiple Access Point General Accuracy



(b) Multiple Access Point Empirical Accuracy

Fig. 5.15: GA and EA as a function of training dataset size for the multiple AP deployment scenario.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.95	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
2	0.30	0.61	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.19	0.01	0.78	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.28	0.02	0.04	0.44	0.10	0.09	0.03	0.00	0.00	0.00	0.00	0.00	0.00
5	0.03	0.00	0.01	0.10	0.76	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.10	0.00	0.01	0.29	0.14	0.46	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.03	0.01	0.11	0.05	0.01	0.00	0.59	0.02	0.05	0.05	0.02	0.01	0.08
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.01	0.00	0.00	0.00	0.00
9	0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.54	0.14	0.00	0.00	0.00
10	0.29	0.00	0.01	0.00	0.00	0.00	0.04	0.00	0.18	0.46	0.00	0.01	0.00
11	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.92	0.00	0.00
12	0.00	0.01	0.30	0.00	0.00	0.00	0.24	0.00	0.00	0.01	0.00	0.26	0.18
13	0.00	0.00	0.11	0.00	0.00	0.00	0.01	0.01	0.06	0.01	0.00	0.13	0.68

Table 5.7: Confusion matrix for k NN classifications from the MAP dataset, day 1.

predictions (i.e. LDA, QDA, and GMM) have relatively high EA of over 80%, without the computational overhead of the k NN method.

To understand which APs have the greatest influence on localisation accuracy and potentially save on installation effort, the localisation accuracies were re-calculated using a subset of the available APs. The effect of the availability of signals from AP2, AP3 and AP4 are investigated. Since AP1 is the main AP its signals must always be available during location predictions. Figure 5.16 illustrates the EA for all classifiers for all combinations of APs. Each combination of APs is denoted by the binary string corresponding to [AP1 AP2 AP3 AP4], where a ‘1’ represents available, and a ‘0’ represents not available. Intuitively, the highest accuracy is possible when all APs are utilised. However, when AP3 is not included in the location predictions, the accuracy is approximately similar to when all APs are available for the k NN, LDA, QDA and NBC classifiers. Surprisingly GMM accuracy is slightly higher with less APs, which can be attributed to the randomness of the initialisation stage of the GEM training algorithm. However, when AP4 is unavailable instead of AP3, the location predictions are lower for all classifiers. This indicates that EA is highly dependent on which subset of APs are employed. Hence, EA is sensitive to AP installation locations.

When the same test is considered for GA in Figure 5.17 it can be observed that GA is not as sensitive to AP location. Instead GA is sensitive to the number of APs present since the tests with the same number of APs available have strikingly similar accuracies. It is also noteworthy that LDA has the highest accuracy for all GA tests because of the equal emphasis placed on each class by the common covariance matrix assumption. Conversely, the k NN algorithm performs the best for the EA measure since its prediction ability is dependent on the number of samples available in every location. Its disadvantage lies in the significantly high computational effort for a location prediction. After k NN, QDA and GMM have the highest empirical accuracies. Hence, QDA and GMM are the algorithms of choice for efficient, high long-term accuracy, location predictions.

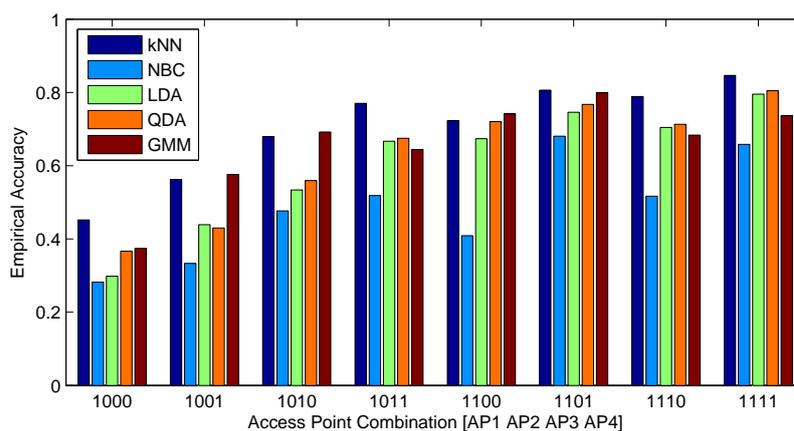


Fig. 5.16: The effect of the subset of available APs on the EA.

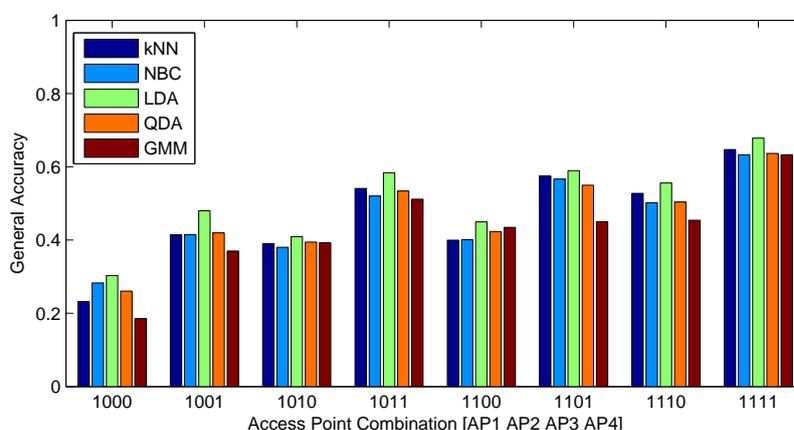


Fig. 5.17: The effect of the subset of available APs on the GA.

It has been illustrated that more available Bluetooth[®] APs lead to higher GA and EA. It has also been shown that EA is sensitive to the location of the installed APs. Previous work has used genetic algorithms to decide on the subset of most informative basestation locations (Genco et al., 2005). However, the decision of which APs can be removed from the environment cannot be made without installing and practically evaluating which combination of APs leads to the highest accuracy. Since our APs are so affordable it is not cost-prohibitive to leave them in the environment, even if they are not found to significantly increase EA. Keeping APs which do not contribute to higher EA will allow better detection of the less-commonly inhabited rooms, as corroborated by the higher GA resulting from more APs.

Even though it may appear that this multiple AP localisation technique is approaching the typical high installation requirement Bluetooth[®] localisation scenario, this work has one major difference. Unlike work where either several computers were required to

	General Accuracy	Empirical Accuracy
PIR Localisation	0.56	0.73

Table 5.8: PIR localisation technique accuracy over 6 days.

calculate remote position (Agostaro et al., 2004, Genco et al., 2005, Rodas et al., 2008) or the user needed to carry a computer or PDA to calculate their position (Kotanen et al., 2003, Feldmann et al., 2003, Castano et al., 2004), this work requires only a single BSC and one mobile phone. Instead of installing multiple BSCs or networked APs, affordable “dumb” Bluetooth[®] beacons are installed to improve location prediction accuracy by increasing signal diversity throughout the environment. This leads to a novel system which can reliably predict the location of the user over long periods of time while only requiring the user to carry a compact device which has functionality besides that of a simple monitoring device. It can also be used as a location-based interaction device and a personal communication device. Hence, the user has more motivation to carry the device.

In the past, such affordable indoor location predictions have commonly been generated by using PIR sensors throughout the home environment. To understand how the Bluetooth[®] localisation technique compares to the previously adopted techniques, a long-term comparison was conducted.

5.3.4 Passive Infra Red Localisation

The previous section highlighted that the Bluetooth[®] localisation system is capable of producing an EA of approximately 85% using the k NN algorithm and approximately 80% using the efficient unimodal probabilistic classifiers. In comparison, Table 5.8 highlights the localisation accuracy for the PIR localisation hardware. The PIR hardware was present during all of the previous tests; making 6 days of PIR data available. With this data it is possible to reliably estimate the accuracy possible with the PIR localisation technique in a realistic long-term deployment. It can be seen that the PIR localisation technique has a GA of 56% and an EA of 73%. Hence the Bluetooth[®] localisation technique, which employs 3 Bluetooth[®] beacons is able to achieve 16% higher EA than the PIR localisation technique which requires a sensor in every room.

To understand what might be impeding high PIR localisation accuracy, one must consider how susceptible to environmental interference the motion detectors are. They can be falsely triggered when curtains or doors move, or when they are directed towards windows in direct line of sight of where people occasionally move. Although care has been taken to install the sensors in positions which are not pointing out windows, as illustrated in Figure 5.1, there is still the significant effect of the sensors continuing to fire a short time after someone leaves a room, since the door in the room can still be moving. Besides the

environmental interference, there are also transmission issues. To increase transmission reliability the employed X10 communication protocol must retransmit the sensor firing packet several times. Hence, a packet from a sensor in a previous room may be received after packets from a sensor in the current room have been received, leading to location confusion.

At this point it should be noted that the extra occupants were permitted to carry mobile phones and utilise wireless laptops within the test environment to simulate a realistic deployment scenario. These wireless protocols do not interfere with the experiments since X10 transmits in a licensed frequency band which is different from that of GSM, Bluetooth[®] and WiFi. Furthermore, the background WiFi transmissions in this environment have been found not to interfere with Bluetooth[®] since Bluetooth[®] is designed to use frequency hopping to transmit on relatively noise-free channels. Naturally a more reliable communication protocol such as Bluetooth[®] or ZigBee[®] could be used to relay the PIR sensor firings back to the BSC without spurious retransmissions, but X10 is significantly less expensive, hence, more commonly used in PIR localisation systems, making this a more appropriate technology comparison.

Even if more reliable communication protocols were used for PIR sensors, there is still the issue of the inability of the sensor to resolve a user's identity. To quantify the effect of the number of inhabitants in the environment on the ability of the system to predict the location of the subject of interest, the data from the 6 days is parsed according to the occupancy levels. Over the 6 days single occupancy occurred 78% of the time. Double occupancy occurred approximately 15% of the time and triple occupancy occurred approximately 7% of the time. Higher levels of occupancy occurred so infrequently that it was impossible to calculate a confusion matrix, and as a result; accuracy, since some rooms were not inhabited by the moving test subject during those levels of occupancy. Figure 5.18 illustrates the influence of occupancy levels on the GA and EA for the PIR system. It can be seen that location predictions are sufficiently accurate for single occupancy localisation. However, when occupancy increases past one person, accuracy significantly decreases.

To appreciate how the issue of multiple occupancy is influencing the accuracy measures in Table 5.8, the location predictions are generated again; this time based on single and multiple occupancy. Table 5.9 shows the peak GA and EA for the PIR localisation system and the Bluetooth[®] localisation system. It can be seen that EA for the PIR system is relatively high, especially when considering the user does not need to carry a mobile device. However in the multiple occupancy case, where there is more than the one person in the environment, the accuracy is significantly reduced. The Bluetooth[®] localisation system, on the other hand, has almost 90% EA in the single occupancy case which reduces to 66% in the multiple occupancy case. The PIR localisation system has a 32% reduction in EA when multiple inhabitants are present in the environment, whereas the Bluetooth[®] localisation

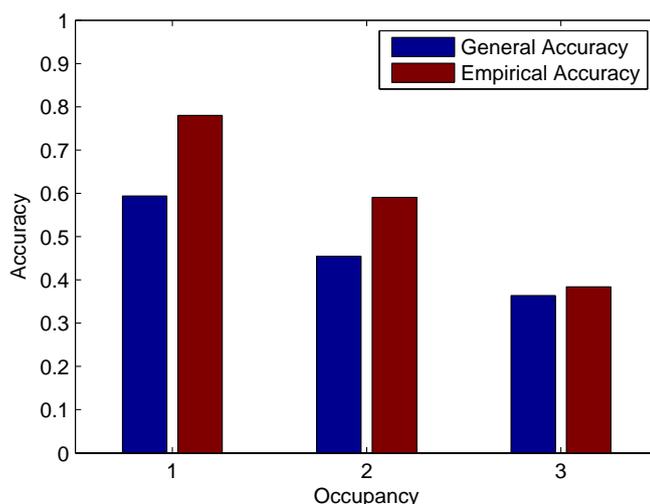


Fig. 5.18: Effect of occupancy levels on PIR GA and EA.

	General Accuracy		Empirical Accuracy	
	Single Occ	Multiple Occ	Single Occ	Multiple Occ
PIR	0.59	0.42	0.78	0.53
Bluetooth [®]	0.62	0.54	0.89	0.66

Table 5.9: PIR and peak Bluetooth[®] localisation accuracies. Peak Bluetooth[®] localisation accuracy is obtained using the k NN algorithm.

system has only a 25% reduction in EA. The reduction in Bluetooth[®] performance can be attributed to the interference of the extra occupants on the Line of Sight (LoS) between the mobile phone and BSC's AP. The reduction in PIR performance, on the other hand, can be attributed to the inability of PIR to resolve the identity of the subject being tracked. Hence, it can be concluded that, not only does the Bluetooth[®] localisation technique have higher accuracy in the single occupancy case, but it also has a lower susceptibility to environmental interference from people, doors, lighting, etc.

This section has outlined the accuracy of the Bluetooth[®] localisation system in a realistic home environment with varying levels of installed hardware. The unweighted accuracy (GA) and application specific accuracy (EA) of the Bluetooth[®] system has been compared with the PIR localisation system which has been used by commercial entities such as GrandCare Systems and QuietCare Systems in the past, under varying levels of realistic occupancy. It has been found that the Bluetooth[®] localisation system, which uses Bluetooth[®] beacons in the environment, has higher accuracy in both the single and multiple occupancy scenarios than a PIR localisation system.

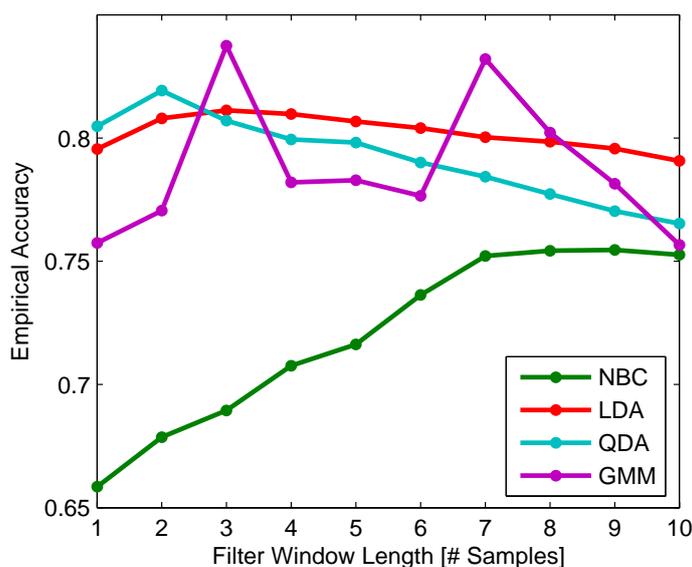


Fig. 5.19: Effect of filtering on EA in the multiple Bluetooth[®] beacon deployment scenario.

5.4 Algorithm Improvements

The previous section has presented the accuracies available from the localisation system when applying classifiers directly to the test features available from the hardware. In Chapter 4 preprocessing, in the form of smoothing or filtering, allowed higher accuracy location predictions with the same available data. This section investigates how knowledge of this application and the behaviour of the signals in this particular application can be exploited to increase the frequency of correct location predictions. The first augmentation, which has already been considered in the trivial home environment, is filtering.

5.4.1 Preprocessing Filter

Preprocessing the data with a Moving Average Filter (MAF) significantly improved the localisation accuracy in the small test environment considered in Chapter 4. Since the realistic long-term test environment has been shown to exhibit different characteristics than the small test environment, the application of filtering must be reconsidered. Similar filtering tests to those illustrated in Chapter 4 are conducted on the multiple Bluetooth[®] beacon data, whereby the localisation accuracy as a function of filtering window length is determined. k NN is not considered since its computational load per location estimate is prohibitively high for real-time location predictions.

Figure 5.19 illustrates the effect of MAF window length, N , on the mean EA for the data for the two days. This figure suggests that less filtering is necessary to

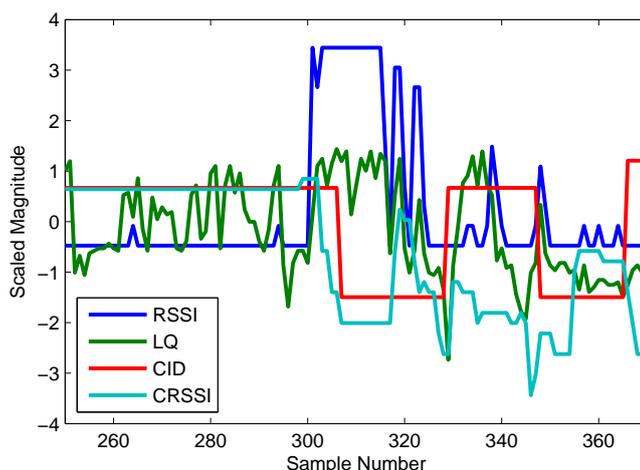


Fig. 5.20: Raw CID, CRSSI and BSC RSSI and LQ signals.

acquire maximum classification accuracy than previously estimated. The peak filtering window lengths with the highest accuracies are $N = 2$ and $N = 3$ for QDA and GMM respectively, which is significantly lower than the optimal of $N = 20$ in Chapter 4. This can be attributed to the lower hardware sampling rates resulting from connecting several Bluetooth[®] devices to the mobile phone. In Chapter 4 the test Bluetooth[®] platform was able to acquire samples at a rate of 1Hz, whereas the extra connections necessary for the plurality of devices connected in the reliable long-term tests lead to a sampling rate of less than 0.5Hz. Hence, a lower level of filtering is necessary to obtain the same reduction in noise-related signal variations.

Furthermore, this realistic test scenario generates test data with less movement-related signal variations, since the subject and mobile phone are typically stationary for long periods of time, which leads to less transient signal variations. The small test environment, on the other hand, required the user to move almost constantly to maximise coverage of the environment during the 15-minute test period. Hence, more filtering was necessary to correct for signal fluctuations during movements due to the use of power control in Bluetooth[®]. From Figure 5.19 it can also be observed that GMM accuracy is at a maximum at $N = 3$, but does not monotonically decrease as would be expected. Instead it again increases at $N = 7$. This can be explained by the fact that the EM and GEM algorithms are initialisation dependent algorithms which means that their “goodness-of-fit”, hence accuracy, is highly dependent on the initially selected mixture parameters, which in turn are dependent on the specific post-filtering dataset. Since QDA uses only a single Gaussian to represent each class, it can always find the globally optimal set of densities and as a result, has a monotonic decrease in accuracy.

In an attempt to find a more appropriate filtering technique for the available data,

Figure 5.20 was considered. This figure compares RSSI and LQ from the BSC AP with the cellular readings CID and CRSSI. The magnitudes of the signals are scaled for comparison. It can be seen that the CID signal has no noise-related variations, which is why it is generally not filtered. If CID is filtered differently from the other signals due to fewer noise fluctuations, the other signals should also be filtered by different amounts, proportional to their apparent measurement noise. Upon considering Figure 5.20 it is evident that LQ has persistent fluctuations, even when RSSI and CRSSI are not changing. Hence, LQ should have higher levels of filtering applied to reduce the higher measurement noise. To experiment with the optimal combination of filtering levels for each signal, an extra filtering parameter was introduced solely for the LQ signals. Now there is a filtering window length parameter for the CRSSI and all RSSI signals. There is also a second filtering parameter for all LQ signals.

Figure 5.21 illustrates the effect of different levels of RSSI filtering (Bluetooth[®] RSSI and CRSSI) and LQ filtering on the mean EA. For brevity, only QDA and GMM are illustrated here since they had the highest peak accuracies for filtering with a common window length. Squares with darker shades of red indicate combinations of RSSI and LQ filtering values which result in higher EA. The image matrix is upper triangular since combinations where LQ filtering windows were smaller than RSSI filtering windows were not considered, hence the blue regions do not have any accuracies associated with them. The QDA classifier accuracy plot in Figure 5.21(a) confirms that highest accuracy is possible with extremely low levels of filtering. The highest accuracy of 82.1% occurs at the window lengths RSSI = 1, LQ = 2. The second highest accuracy of 81.9% occurs at window lengths of RSSI = 2, LQ = 2, which is identical to the common filtering window length of $N = 2$ in Figure 5.19. Hence, there is little improvement in using differing RSSI and LQ filtering levels for the QDA algorithm.

As for the GMM algorithm in Figure 5.21(b), it can be observed that the intensity is not as smooth as that for the QDA algorithm. This is similar to the common filtering window length in Figure 5.19 in which GMM accuracy was less smooth than QDA due to the sensitivity of GMM to the randomness of the mixtures' parameter initialisation. It can also be observed that the peak accuracy of 84.1% occurs at the filtering window lengths of RSSI = 3, LQ = 3, which is identical to the common filtering window length of $N = 3$. Hence, using different RSSI and LQ filtering window lengths provides no improvement in GMM EA.

These results suggest that filtering provides very little increase in EA for QDA. It does allow a significant increase in GMM EA at particular filtering levels. However, this is not a reliable indicator of filtered GMM accuracy on other datasets since the unpredictable increase in accuracy at certain filtering levels is due to the initialisation of the GMM parameters for the particular post-filtering dataset. Furthermore, the practical estimation

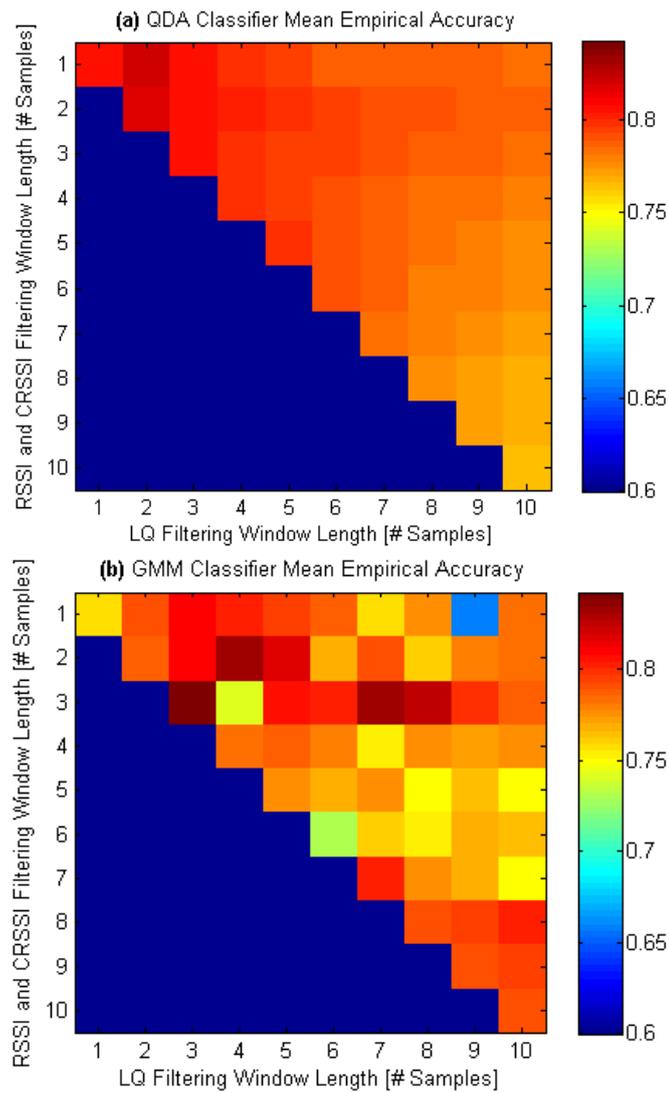


Fig. 5.21: The effect of the combination of RSSI and LQ filtering levels on mean EA.

of optimal filtering values for a given deployment would require both a training and a cross-validation dataset. Since the EA measure requires a full day of data for reliable accuracy estimates (see Section 5.3.3) this will require a day of training data and a day of validation data. Hence, filtering produces a significant overhead in classifier training which is not guaranteed to significantly increase accuracy in a realistic deployment. Filtering, however, can be seen to allow a minor increase in accuracy across all classifiers up to a value of $N = 2$. Accordingly, only a small fixed level of filtering will be considered for long-term accuracy improvement in a realistic deployment scenario.

5.4.2 Time-Lagged Preprocessing

One thing which the classification techniques have not considered thus far, is the temporal evolution of the signals. For example, the signals may change as a subject moves through a certain location in a house, in a way which is distinct for that particular location. On a sample-by-sample basis each one of these samples could be classified as being part of other locations. However, when considering the sequence of signals as a whole, it may be possible to deduce that this distinct sequence of samples is the result of one unique location. This can loosely be considered a form of time-sequence classification, which has applications in speech recognition. An efficient solution to generating predictions based on a sequence of input data is the Time-Delay Neural Network (TDNN) (Chou and Juang, 2003).

The TDNN uses a tapped delay line to generate multiple input features from a single input feature. Borrowing from this technique of time-sequence recognition, a similar tapped delay line can be used to generate new features for the classifiers under consideration in this work. The general form of a classifier with the time-lagged preprocessing can be seen in Figure 5.22. The number of samples per original feature is $L = 3$ in this case. Mathematically the expression for generating the new classifier input features from the original input features is,

$$\mathbf{x}_n = (x_n(1), x_{n-1}(1), \dots, x_{n-L+1}(1), \dots, x_n(D), x_{n-1}(D), \dots, x_{n-L+1}(D)). \quad (5.4)$$

To quantify the improvement of localisation accuracy possible with this technique of temporal pattern matching, the data was preprocessed with the time-lagged preprocessing expression before applying the classifiers. Figure 5.23 illustrates the impact of using more lagged samples for each feature on the EA. GMM is not considered because when the number of features increases above the original 10, it starts to suffer from the problem of insufficient training data. It is impossible to estimate the mixture parameters for certain rooms, one of which has as few as 80 samples; which is insufficient when the dimensionality of the problem is increased via the inclusion of time-lagged samples. QDA has the same

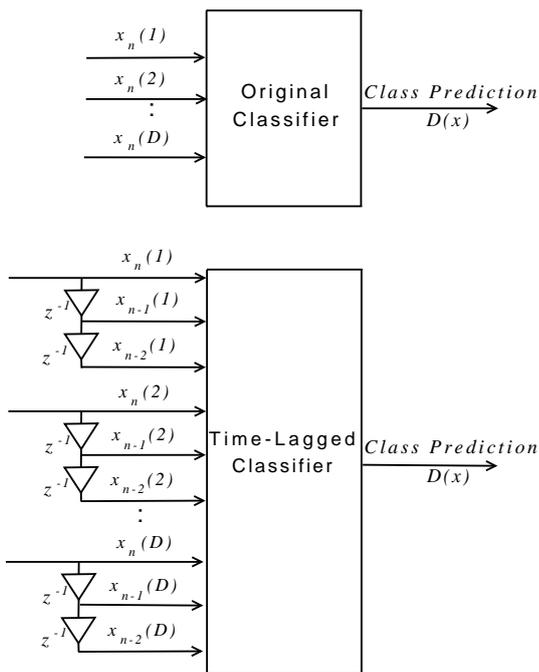


Fig. 5.22: Using a tapped delay line to generate a time-lagged set of classifier input features from the original feature vector.

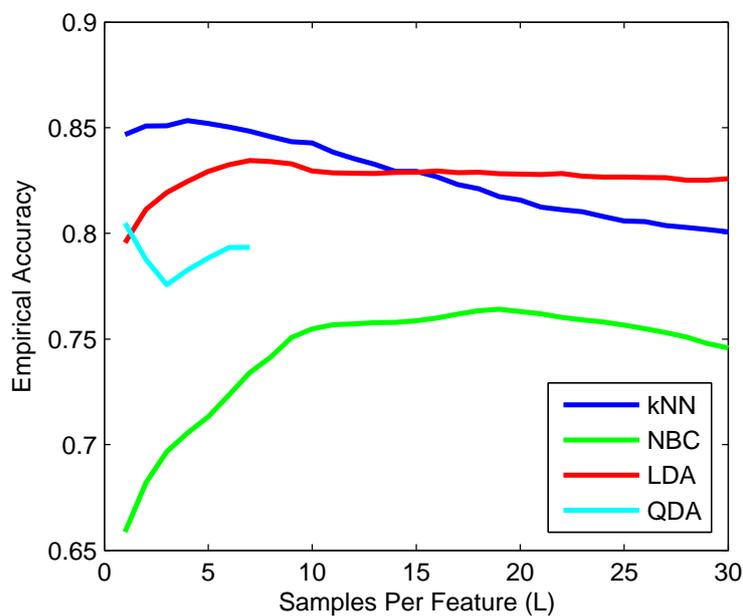


Fig. 5.23: EA as a function of the number of lagged features per original feature, L .

problem when the number of time-lagged samples, L , reaches 8. Generation of a QDA model with 80 features and only 80 training samples is impossible because the problem is ill-posed (Kuncheva, 2004). Hence, it is only possible to consider time-lagged QDA up to

$L = 7$.

It can be seen from Figure 5.23 that the k NN classifier is unable to achieve higher classification accuracy by employing time-lagged samples since it is a highly specialising classifier. It is unable to generalise on the particular temporal patterns for each location and in fact the inclusion of more time-lagged samples causes more misclassifications and lower EA. It is difficult to visualise how QDA could optimally perform with the time-lagging technique since there is insufficient training data to consider the case when $L > 7$. For the available training data, at least, it does not appear to increase accuracy. LDA does have a slight increase in accuracy when time-lagged preprocessing is applied, with the peak EA of 83.4% occurring at $L = 7$. An even more significant increase in accuracy can be seen with the NBC technique. Accuracy for the NBC increases from 65.8% for raw data to 76.4% at $L = 19$. NBC has a higher ability to model time-lagged samples since it does not place any constraints on the covariance between the samples. On the other hand, the common covariance matrix restriction of LDA gives it reduced ability to accurately represent the time-evolution of the signals.

Hence, it is possible to increase the accuracy of the probabilistic classifiers by using time-lagged preprocessing to generate more features to describe the evolution of the signals over time in a particular location. It has been found that avoiding a common covariance constraint between all features in all rooms permits a higher increase in time-lagged classification accuracy, hence NBC has the highest accuracy increase in the presence of time-lagged preprocessing. The disadvantage of using this time-lagged preprocessing is that it drastically increases the number of features necessary for each classifier, with the peak time-lagged accuracy for NBC occurring when there are 190 input features. And even with this computational overhead, the NBC classifier still can only approximate unimodal Gaussian distributions. Hence, it is necessary to consider if higher accuracy and more efficient location estimates can be produced by using classifiers which use sparse representations of the data and which are capable of representing arbitrary probability densities, such as Relevance Vector Machines, Informative Vector Machines and Support Vector Machines.

5.4.3 Sparse Approximations

Kernel machines are a family of pattern recognition techniques which utilise kernel functions to transform data to some feature space. The pattern recognition algorithms can then operate in these higher dimension feature spaces permitting flexible classifications with relatively few classification parameters. This has the potential to enable extremely sparse and efficient representations of data.

Max Samples per Room	100	200	300	400	800
Mean Empirical Accuracy	0.40	0.42	0.54	0.64	0.67
Mean Training Time [days]	3.6	8.35	15.03	21.98	51.4

Table 5.10: Mean EA for SVM classifiers and corresponding training times when different quantities of training data are available.

5.4.3.1 Support Vector Machines

As presented in Chapter 3, Support Vector Machines (SVMs) are non-probabilistic classifiers which generate optimal decision boundaries using constrained optimisation. This is a computationally intensive process which is further exacerbated by the cross-validation stage necessary to establish the optimal radial basis function (RBF) width and slack parameter. It has been shown that optimal SVMs can be trained for small datasets obtained over short periods of time in the test environment considered in Chapter 4 (Kelly et al., 2008c,b,d). When these optimisation techniques, however, are extended to datasets such as the MAP datasets, which have high sample counts with high dimensionality, these optimisations become impractical. It was found that performing the 5-fold cross validation grid-search optimisation of 20 RBF width and 20 slack parameters took far in excess of a month on a dedicated high-processing-power computation server for the entire dataset.

The long optimisation time can be partially attributed to using the simpleSVM toolkit for Matlab[®] which is not optimised for fast execution. Implementing an SVM training algorithm in a faster language, such as C for example, would result in faster training and optimisation times. However, even if an order of magnitude decrease in optimisation and training times were possible, the SVM optimisation would still take at least 3 days on a dedicated high-processing-power computer. This is not a permissible time overhead for the installation of this system in a home environment. For this reason it was necessary to consider the suitability of smaller training datasets for generating SVMs for long-term localisation. As in Section 5.3, the training datasets are modified by deleting all but the first N samples in each class before training occurs.

Table 5.10 highlights the mean EA obtainable with the SVM classifiers when different maximum levels of training data is permitted for each room. Upon considering the corresponding results in Figure 5.15(b) it can be seen that the probabilistic classifiers perform better than SVMs for all quantities of available samples per room. This is likely due to SVMs high ability to flexibly model the data in each class. The optimised SVM classifiers provide a faithful representation of the training dataset. The optimal parameters, however, are also produced using cross validation on the same dataset. Hence, any minor trends in data from one day to the next will result in misclassifications. The probabilistic classifiers, on the other hand, automatically generalise on the data and are less susceptible to daily trends in the signals.

Max Samples per Room	100	200	300	400
Mean Empirical Accuracy	0.04	0.29	0.29	0.40
Mean Training Time [hours]	0.5525	1.591944	2.988333	4.483472

Table 5.11: Mean EA for IVM classifiers and corresponding training times when different quantities of training data are available.

Furthermore, the training times for SVMs are prohibitively high, even when small quantities of training samples are available for each room. Conversely, the probabilistic classifier models can be generated almost instantly, leading to more convenient and robust installation. Even in terms of sparsity, the SVMs require greater numbers of parameters to parameterise a single class. For example, when a maximum of 100 samples per room is used, an average of 424 support vectors are necessary for each class. Hence, even GMMs can represent the same dataset with significantly fewer parameters. Another significant issue is that SVMs do not have probabilistic outputs. As already presented, probabilistic outputs enable further analysis, such as ROC curve analysis, uncertainty rejection and HMM framework implementation. Hence, sparse kernel machines which produce probabilistic outputs shall be considered.

5.4.3.2 Informative Vector Machines

Gaussian Process regression has been used in the past to model coordinate wide-scale outdoor location as a function of cellular signal strength (Schwaighofer et al., 2004). This work is different in that we are interested in using classifiers to determine symbolic location from Bluetooth[®] signals. This can be implemented using a Gaussian Process classifier, which has the advantage over SVMs of producing probabilistic outputs. As explained in Chapter 3, an Informative Vector Machine (IVM) is a sparse approximation of a Gaussian Process classifier. Hence, a classifier with similar classification performance to a full Gaussian Process classifier can be obtained by using a greedy training approach to select an optimal subset of training vectors. For the long-term localisation datasets the employed IVM algorithm (Lawrence et al., 2003) exceeded the 8 Gb of memory available on the processing computer. Hence, a subset of training data was considered, as in the previous subsection. Since Chapter 3 suggested that classification boundaries similar to a Gaussian Process classifier is achievable by using 65% of the training samples as the informative vectors. Hence, it was decided to permit 75% of the available samples to be used as informative vectors.

Table 5.11 indicates that using this technique on the same subsets of training data as used for the SVM, results in significantly poorer performance. This is likely due to the sensitivity of IVMs to the number of permitted informative vectors, as illustrated in Figure 3.8. Hence, for a practical deployment, a cross-validation stage would be required

Max Samples per Room	100	200	300	400
Mean Empirical Accuracy	0.33	0.39	0.48	0.53
Mean Training Time [hours]	0.067222	0.220139	0.464861	2.021944

Table 5.12: Mean EA for RVM classifiers and corresponding training times when different quantities of training data are available.

to determine the optimal number of informative vectors for each class, leading to even significantly higher training times. However, given the poor performance of the IVM classifiers with 75% active vectors, it is evident that the IVMs will not produce classifiers more sparse or accurate than the SVM classifiers, let alone the simple probabilistic classifiers considered thus far. Even though IVMs can represent data with arbitrary densities, their representation power is highly dependent on the quantity of informative vectors permitted. Hence, IVMs are not the solution for efficient localisation.

5.4.3.3 Relevance Vector Machines

In the regression case, Relevance Vector Machines (RVMs) can approximate data using a linear combination of basis functions. In the case of classification the regression output can be translated to a probabilistic output using a logistic function. Hence, the RVM can also represent arbitrary probability distributions. Chapter 3 explains that the training of an RVM takes longer than a SVM. However, RVMs do not require the optimal selection of the kernel width or slack parameters. Hence, RVMs have significantly lower overall cross-validation and training times than SVMs. Table 5.12 allows a comparison between the total training times for RVMs and SVMs in Table 5.10. It should be emphasised that the SVM cross-validation and training times are noted in days and the RVM times are noted in hours. It should also be noted that the RVM is not considered for 800 samples per room since memory usage exceeded the available memory for the test computer. This is because the RVM starts with a full set of weights and iteratively works towards a sparse solution.

From Tables 5.10 and 5.12 it can be observed that RVMs require significantly shorter time to train a classifier. However, it can also be observed that RVMs produce lower accuracies than the SVM classifiers. Furthermore, when comparing Table 5.12 with the equivalent sample quantities in Figure 5.15(b) even lower accuracies than the simple probabilistic classifiers are evident. Hence, it can be concluded that the sparse probabilistic classifiers do little to improve localisation accuracy. This is likely due to the “one versus all” approach taken to training the IVM and RVM classifiers. Since these are probabilistic classifiers it would be more appropriate to model the densities for all rooms dependently of each other with a single model, rather than one model per room. Refer to Rasmussen and Williams (2005) for example. However, in spite of the implications of the name,

these sparse classifiers are unable to model the densities of the data in each room more sparsely than the simple probabilistic classifiers. Furthermore, for this localisation problem the sparse classifiers have shown to have lower localisation accuracy than the simple probabilistic classifiers, indicating that highly generalising classifiers are more appropriate to the localisation problem under consideration in this thesis.

5.4.4 Cellular Signal Treatment

Section 4.4.3 highlighted the signal redundancy in the CID and CRSSI signals. It also suggested that these signals may not be as redundant in a realistic home environment which consists of more rooms with varying levels of cellular network coverage. It also demonstrated that NBC performs competitively with the other classifiers when the discrete valued CID signal is present and less competitively when the CID signal is unavailable. Hence it was suggested that the relationship of the CID signal with location could be further emphasised by treating CID as a discrete signal, independent from the other signals. This can be achieved by modelling CID with a discrete probability model and combining it with the other signals using the Naive Bayes assumption of independence.

Mathematically, this can be expressed as a modification to Equation 3.26:

$$P(R_k|\mathbf{x}) = \frac{P(R_k)P(x(CID)|R_k) \prod_{d=1}^{D-1} P(x(d)|R_k)}{P(\mathbf{x})}, \quad (5.5)$$

where

$$P(x(CID)|R_k) = \frac{\# \text{ samples of a particular CID in room } R_k}{\text{total } \# \text{ samples taken in room } R_k}. \quad (5.6)$$

The use of this discrete representation of $P(x(CID)|R_k)$ is not restricted to NBC. It can also be used for LDA, QDA and even GMM classifiers by excluding the continuous CID feature from the signal density estimation and post-multiplying the partial probability $P(\mathbf{x}|R_k)$ by $P(x(CID)|R_k)$. This, however, is not as valid for LDA, QDA and GMM as it is for NBC since the discrete signal probability does not model the dependencies of CID on the other signals. Table 5.13 compares the original treatment of the CID signal as a continuous variable with the method of including CID as a discrete probabilistic variable. This confirms that treating CID as an independent discrete variable decreases accuracy for the probabilistic classifiers which model the dependence of signals on each other using the covariance matrix. For the Naive Bayes Classifier it only shows a marginal increase in EA. However, the variability of the accuracy is lower for the discrete CID NBC treatment, as indicated by a lower standard deviation.

Unexpectedly, omitting the CID variable leads to similar EA as including it for the NBC classifier. Omitting it for the covariance sensitive probabilistic classifiers leads to lower accuracy. Conversely, eliminating the CRSSI reading leads to lower NBC EA but

	NBC	LDA	QDA	GMM
Original	0.658(0.12)	0.796 (0.00)	0.783 (0.04)	0.784 (0.04)
Discrete CID	0.660(0.05)	0.777 (0.02)	0.685 (0.11)	0.685 (0.12)
No CID	0.661(0.11)	0.746 (0.05)	0.773 (0.06)	0.773 (0.06)
No CRSSI	0.622 (0.16)	0.792(0.00)	0.806(0.00)	0.814(0.00)
No Cellular	0.617 (0.16)	0.732 (0.07)	0.750 (0.07)	0.753 (0.07)

Table 5.13: The effect of different treatments of the cellular connectivity information on EA.

higher EA for the covariance sensitive classifiers, QDA and GMM. Hence, omitting the CRSSI signal leads to the highest accuracy. This confirms the suggestion in Chapter 4, Section 4.4.3 that CRSSI is an uninformative signal and CID is more indicative of location. Finally, completely omitting all cellular signals leads to significantly lower localisation accuracy.

This section has indicated that the discrete treatment of the CID signal does not contribute to higher localisation accuracy. It has, however, found that highest localisation accuracy is possible with QDA and GMM by removing the CRSSI feature from location predictions, since it is a signal which is not indicative of location.

5.4.5 Incorporating Prior Knowledge

All of the probabilistic classifiers considered allow the incorporation of prior information into the probability estimates. Without any information to the contrary, it can be assumed that this prior probability, $P(R_k)$, is equal across all rooms, k . However, this chapter has outlined the use of a week of training data to estimate the room-level location preference of the user. This user movement profile was then used to estimate a more application specific accuracy measure: the EA. This movement preference information can also be more directly considered to be the *a priori* probability that a user is in a particular location before any location-indicative measurements have become available.

Castro et al. (2001) suggests that movement histories of users can be incorporated into the employed Bayesian network classifier. It was proposed that the location preferences can be modelled via the *a priori* distribution, which corresponds to $P(R_k)$. In this work, long-term location preferences can be empirically derived from the relative frequency of occupation illustrated in Figure 5.4(b). Now the probabilistic classifiers can be executed as before, except that the $P(R_k)$ term in Bayes' rule in equation 3.22 no longer consists of equal values. Instead they take on values proportional to each room's relative frequency of occupation and all the values of $P(R_k)$ are normalised so that they sum to 1. Tables 5.14 and 5.15 outline the accuracies when prior probabilities are ignored as before and when they are included in the location density estimates. The highest accuracy for each

	NBC	LDA	QDA	GMM
No Priors	0.63(0.04)	0.68(0.02)	0.64(0.02)	0.57 (0.18)
Priors included	0.62 (0.02)	0.61 (0.03)	0.59 (0.01)	0.58(0.10)

Table 5.14: The effect of *a priori* information inclusion on GA.

	NBC	LDA	QDA	GMM
No Priors	0.66 (0.12)	0.80 (0.00)	0.80 (0.00)	0.78 (0.04)
Priors included	0.80(0.01)	0.82(0.01)	0.81(0.01)	0.79(0.03)

Table 5.15: The effect of *a priori* information inclusion on EA.

type of classifier is highlighted in bold.

These summary tables indicate that GA suffers a decrease for most of the classifiers when prior location probabilities are incorporated. This is intuitive when one considers that priors have the effect of increasing the likelihood of detecting certain rooms, while decreasing the likelihood of other rooms. Conversely, this effect of priors is actually advantageous when the more relevant EA is considered. An increase in EA can be seen across all classifiers when prior location probabilities are included. This is because the priors can be thought to place emphasis on classes with higher prior probabilities. Since our priors are derived from the metric which indicates the relevance of each room to the EA, rooms which are more commonly inhabited will be more “detectable” over time, increasing the long-term localisation accuracy.

Hence, incorporating prior location information into the classification algorithms has the effect of increasing long-term accuracy; most significantly for the NBC which has a 21% increase in EA. Application-specific handling of the input probabilities improves the performance of the localisation system. More appropriate handling of the posterior probabilities can also increase accuracy.

5.4.6 Uncertainty Rejection

So far, the class decision block in Figure 3.1 has decided on class membership based on class posterior probabilities. However, the availability of probabilistic class membership indicators, rather than direct class predictions, allows an understanding of the confidence in a given prediction. The availability of information about the probability of a sample belonging to a given class allows minimisation of the misclassification rate by refusing to classify samples which are uncertain. An example of a health-critical application of uncertainty rejection can be found in Antal et al. (2003) which suggests leaving uncertain samples for human consideration. Similarly, falsely predicting the position of a user could also have safety-critical implications since it can cause misinterpretation of the movement

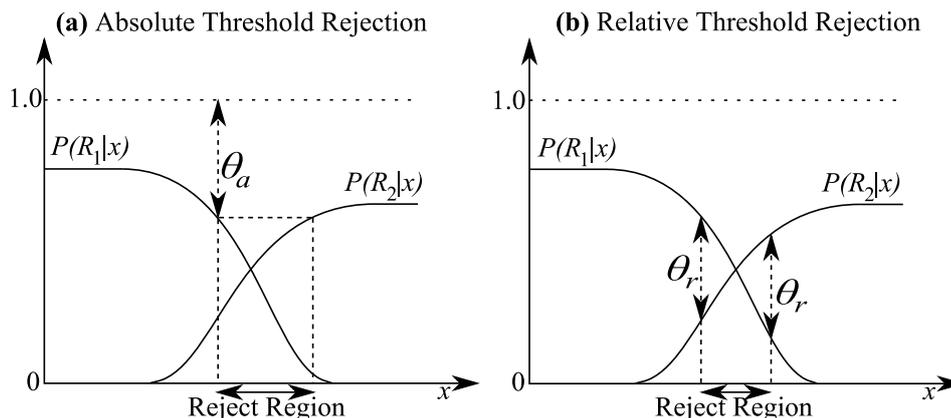


Fig. 5.24: A comparison of the different uncertainty rejection methods on the same uncertain problem.

habits of a user which can lead to interventions at inappropriate times or no intervention at all. Instead of producing location predictions regardless of their confidence, there should be the facility to refrain from making a location prediction when there is insufficient data to confidently make such a prediction. Then any user or applications wishing to retrieve location can then be informed that the available signals are currently insufficient to reliably predict location.

According to Bishop (2006), probabilistic classification errors generally occur in one of two situations,

1. The largest posterior probability is significantly less than unity
2. More than one class has the highest, or approximately highest, posterior probability.

Hence, two approaches to uncertainty rejection have been employed. The first approach is to reject predictions for which the highest posterior class probability is outside a selected threshold of unity, as illustrated in Figure 5.24(a). This is referred to as the absolute threshold and is denoted θ_a . The alternative approach is to reject predictions for which the highest class posterior is “near” to the second highest class posterior. The term “near” is realised in the specified relative threshold, denoted θ_r . Hence, any predictions for which the posterior of the most probable class is within θ_r of any other class, it is rejected, as illustrated in Figure 5.24(b).

Figure 5.25 highlights the effect of uncertainty rejection with increasing rejection thresholds on EA and GA. It also indicates what proportion of samples are rejected to achieve such accuracies, referred to as the rejection ratio. It can be seen that the proportion of uncertain samples increases with higher rejection thresholds and that higher levels of accuracy are possible when more uncertain samples are rejected. A rejection threshold of 1 is never considered since this would result in all samples being rejected.

The primary difference between the absolute threshold and relative threshold rejection approaches is the rate at which the accuracies and the rejection ratio increases. For the absolute threshold the accuracies and rejection ratios generally do not change for a value of $\theta_a < 0.5$. The increases of accuracies and rejection ratio for the absolute threshold method in the range $0.5 \leq \theta_a \leq 1$ are identical to those for the relative threshold method in the range $0 \leq \theta_r \leq 1$. Hence, the relative threshold method can be thought of as having a more gradual change in accuracy and rejection ratio than the absolute threshold method. This means that the relative threshold method is a more appropriate method to use in practice since it shows variation in the metrics of interest, such as accuracy and rejection ratio, for the entire range of rejection threshold values. On the other hand, the absolute threshold method only shows variation for half of the rejection threshold range of interest.

Although it is difficult to quantify from the figures, a definite increase in accuracy is possible with all classifiers when uncertainty rejection is employed. NBC exhibits the highest increase in accuracy, at a 25% increase from the original EA. QDA and GMM experience an accuracy increase of 7% and 9% respectively at the maximum rejection threshold value. Interestingly LDA experiences a 6% increase in accuracy which starts to deteriorate as the maximum rejection threshold is approached. This is likely due to the inability of LDA to independently model the densities of individual classes due to the common covariance assumption. Hence, with LDA incorrect classes generally have lower posteriors and incorrect classes have higher posteriors than if they were modelled independently and samples are rejected even though their highest posterior is the correct class.

A major issue with uncertainty rejection is that it leads to times when no location predictions can be produced. For example, the technique which produces the highest accuracy, QDA at 86%, achieves this by refusing to make predictions 25% of the time. This 7% increase in accuracy may not justify being unable to make predictions so frequently. Instead it may be more appropriate to make a “best guess” when the sample is rejected. Previously a best guess would have been the class of highest probability, but high measurement noise and movements slightly different from training data movements could cause the class of slightly higher probability to be that of an incorrect room. Hence, another “best guess” mechanism is proposed. This mechanism uses uncertainty rejection to find uncertain classes. Then it is assumed that the cause of the uncertainty is due to noisy or as yet unseen signals and not a room change. Then the uncertain prediction is replaced by the last certain prediction. An illustration of the accuracies for such an algorithm is illustrated in Figure 5.26. This time the rejection ratio is not illustrated because it is always equal to zero, since predictions can now be made at all times.

With this rejection-with-replacement (RWR) algorithm, accuracy increases are not as

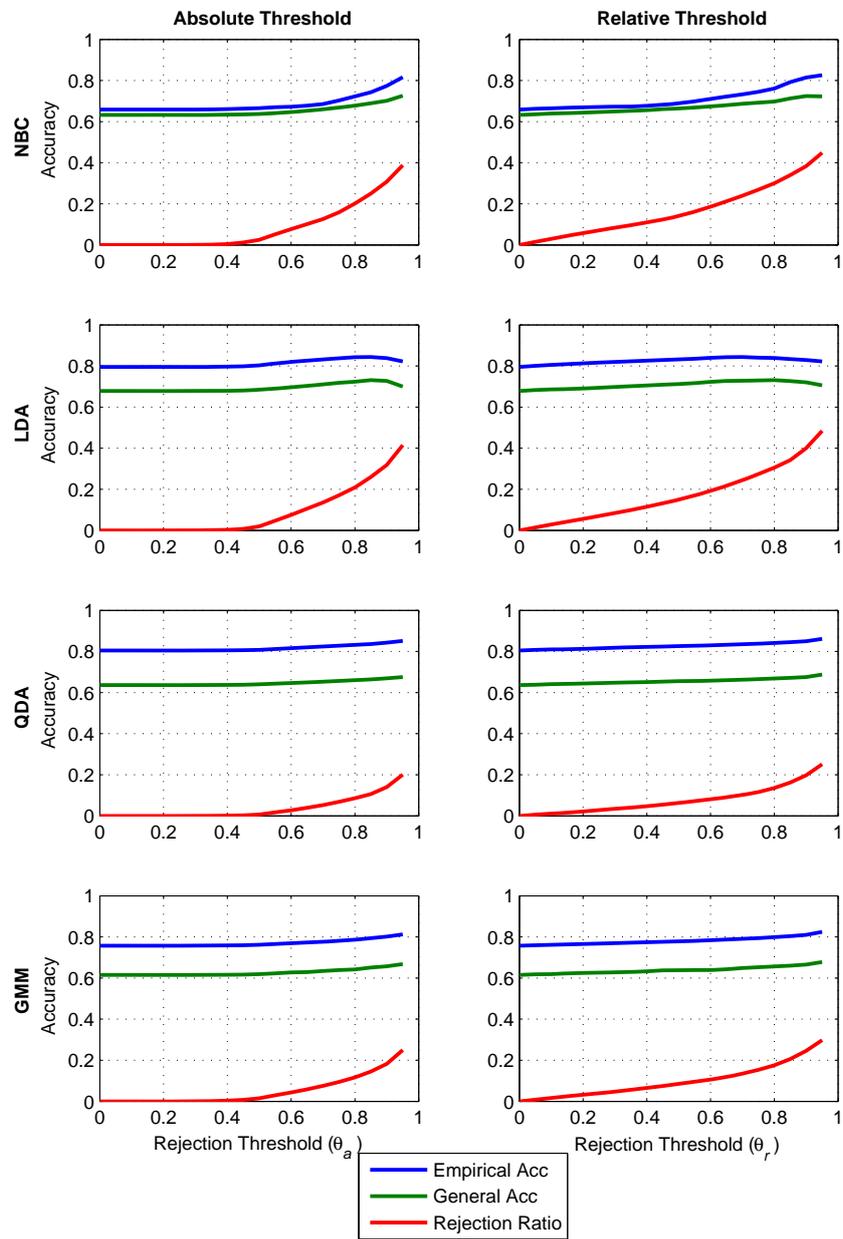


Fig. 5.25: EA, GA and rejection ratios for increasing absolute and relative rejection thresholds.

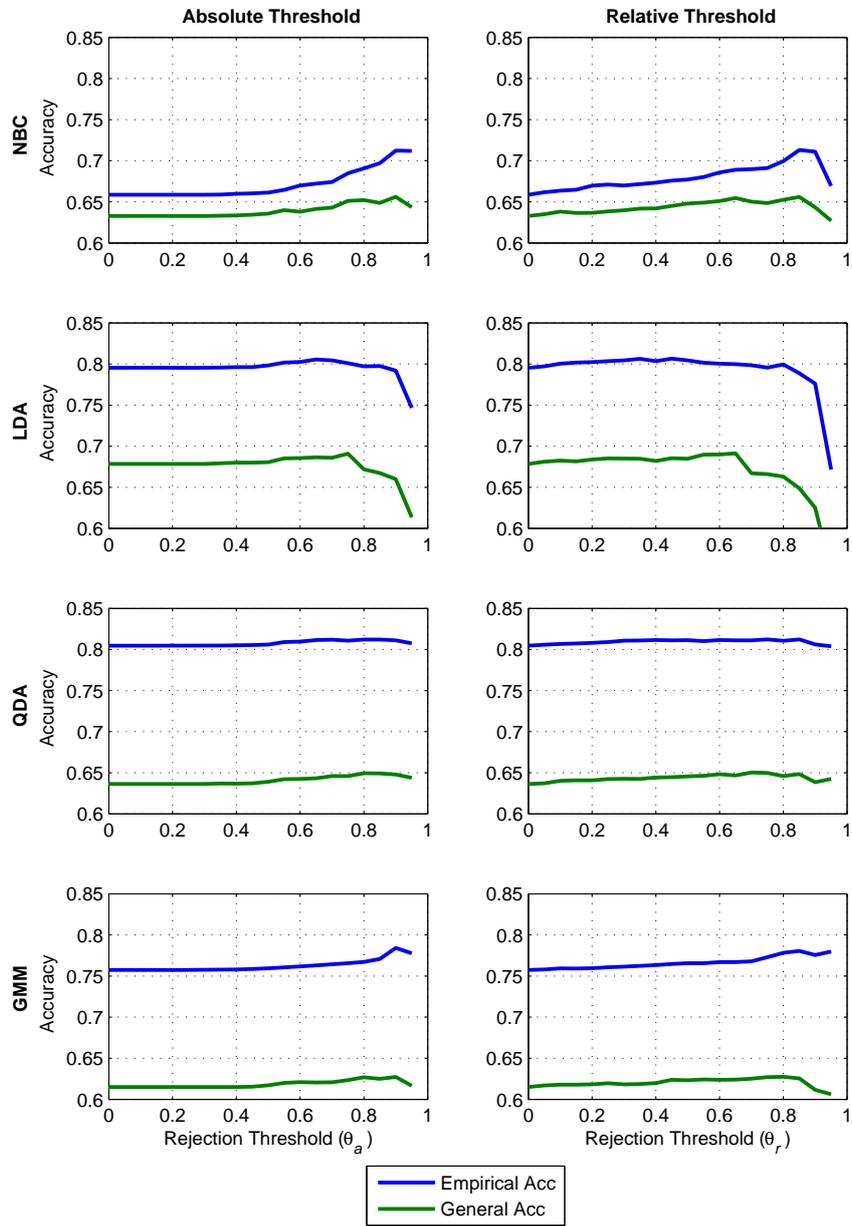


Fig. 5.26: Illustration of the increase in accuracy when uncertain samples are assigned the class label of the last certain prediction.

large as when complete rejections (CR) are possible. Again, NBC has the highest EA increase, at only 8.3% this time. LDA, QDA and GMM have accuracy increases of 1.4%, 0.9% and 3% respectively. Hence, RWR has slightly higher accuracy than the original algorithms without rejection, but lower accuracy than the CR algorithms. However, the advantage of RWR over CR is that it can make predictions at all times without refusing to make a prediction. There is an issue with RWR in that it actually has reduced accuracy when the rejection threshold approaches the upper limit. This occurs because there are simply not enough good samples retained. At any one point in time it may be a long time since a correct prediction was obtained and the user is likely to have moved in that time, making the RWR “best guess” increasingly invalid. Hence, employing the RWR algorithm requires an optimisation stage to determine the most suitable threshold value, which would require another day of training data. Furthermore, if the location predictions are safety-critical, for example in an elder monitoring scenario, it would be more suitable to use the CR algorithm to obtain the highest accuracy possible and openly admit when the predictions are unreliable. Then the end-user or application using this information can decide for themselves how to deal with this information, which may include a RWR-type algorithm.

5.4.7 Combined Accuracy Augmentations

Up to this point, individual accuracy improvements have been considered in isolation. Intuitively, to achieve the most reliable and accurate localisation technique these augmentations should be cascaded together into one entire system. Figure 5.27 illustrates how the accuracy improvement algorithms can be cascaded together. The accuracy augmentation blocks can be categorised into pre-processing, density estimation and decision blocks. Hence, there is a direct correspondence between this configuration of blocks and the general classification framework presented in Figure 3.1. Each block corresponds to an algorithm presented in this section, which has been shown to contribute to the performance of the localisation system. Accordingly, the only investigated augmentation not included in this framework is the use of sparse classifiers, which do not contribute performance improvements in spite of their processing overheads.

The implementation details of each block are as follows:

- 1. Filtering** The filtering block is the first block which the raw input feature encounters. It applies a small amount of filtering to the data to reduce noise-related misclassifications. Hence, it must have memory of the last $N - 1$ samples. A filtering window length common to all features of $N=2$ is chosen because Figure 5.19 illustrated that increases in accuracy generally occur for this level of filtering for all classifiers considered.

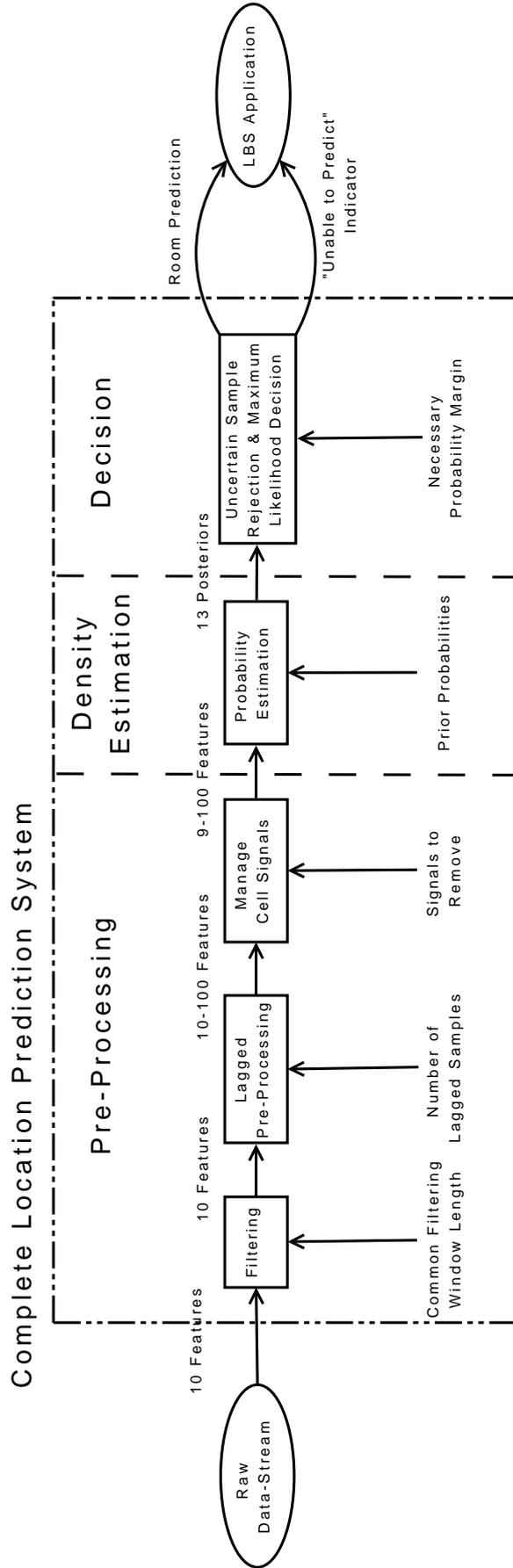


Fig. 5.27: The overall system combining the accuracy improvements explored in this chapter.

- 2. Lagged Pre-Processing** The 10 features of the filtered data are sent to the lagged preprocessing block. Like the filtering block, the lagged preprocessing block must maintain a record of the previous feature vectors to construct the new feature vector. For the QDA and GMM data no lagged preprocessing is applied, based on the findings in Section 5.4.2. As found in Section 5.4.2 data for NBC uses a lagged sample number of $L = 7$ and the data for LDA uses a lagged sample number of $L = 10$. This produces between a 10 and 100 sample feature vector, depending on the classifier for which the data is being prepared.
- 3. Cellular Signal Management** This block handles the cellular signals in the appropriate fashion. According to Section 5.4.4 it is beneficial to completely remove the CRSSI features from the data for the QDA and GMM classifiers. Removing CRSSI has no effect for the LDA classifier. For NBC, however, removing CRSSI features results in lower accuracy due to NBC's ability to independently model the CRSSI signal, hence CRSSI is retained for NBC. Hence, a feature vector with a length of between 9 and 100 features is produced for the next block.
- 4. Density Estimation (Incorporating Priors)** The preprocessing stage generates training and test data, each with 9-100 features. The training data is used to build a probability density estimation model using one of NBC, LDA, QDA or GMM. The construction of this model now incorporates prior information about the location tendencies of the user. This prior information is derived from the long-term relative frequency of occupation as explained in Section 5.4.5. When the model is applied to the test data, this block outputs a set of posterior probabilities, one for each room.
- 5. Decision with Uncertainty Rejection** As already presented, this block is a maximum likelihood decision block as illustrated in Figure 3.1. Before performing maximum likelihood predictions it detects uncertainty based on the selected probability margin. This probability margin is the relative rejection threshold, θ_r , and specifies how much the user wants the probability of the most likely class to be above the next most likely class. For these experiments a value of $\theta_r = 0.95$ was selected.

Four probabilistic classifiers are employed in this framework. Hence, different levels of processing are necessary at each block for each classifier. After each block the relevant classifier is executed and the accuracy is recorded to illustrate the contribution of each block to the overall accuracy for each probabilistic classifier. Figure 5.28 shows the accuracy when only raw data is available (step 0) and after each processing step (steps 1-5). At every stage in this classification system EA is shown to improve. The only time EA does not improve is when the features are unmodified for that stage. For example

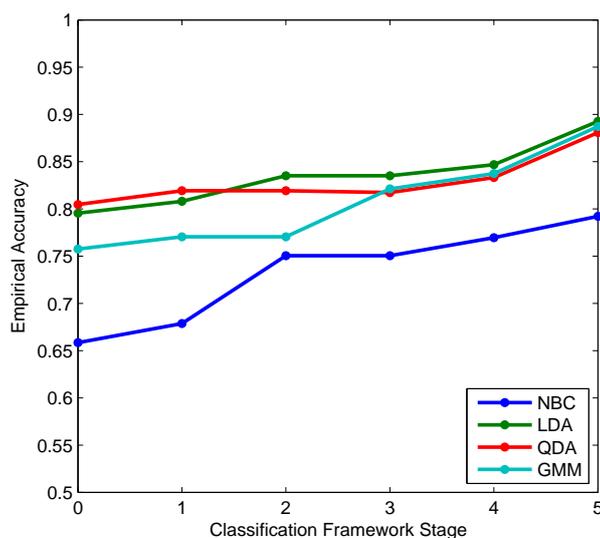


Fig. 5.28: Accuracy levels at each processing stage in the overall system.

	NBC	LDA	QDA	GMM
95% Probability Margin Acc	0.792	0.893	0.881	0.887
Rejection Ratio	0.053	0.242	0.159	0.165
99.99% Probability Margin Acc	0.818	0.897	0.933	0.944
Rejection Ratio	0.127	0.374	0.574	0.585

Table 5.16: The effect of imposing a higher probability margin on the uncertainty rejection and the corresponding rejection ratios.

between stages 1 and 2 the QDA and GMM data are unmodified, hence accuracy does not increase for these classifiers.

Table 5.16 presents the EA and the corresponding rejection ratios. Uncertainty rejection enables a peak localisation accuracy of 89.3% with a rejection probability margin of 0.95. If a more stringent probability margin was imposed, localisation accuracy would understandably increase. Table 5.16 also highlights the accuracy resulting from increasing the probability margin to 0.9999. It can be seen that the peak accuracy for the probability margin of 0.95 results in a rejection ratio of 24% for LDA. If the QDA or GMM algorithms are employed the EA is approximately the same, but the rejection ratios are much more favourable at approximately 16%. Once the probability margin is increased to 0.9999, the accuracy increases to 94%. This accuracy, however, comes at the cost of refusing to produce location estimates 59% of the time. This higher rejection ratio severely reduces the usefulness of the system. Hence, there is a tradeoff between the frequency of availability of location predictions and the accuracy of the location predictions. Instead of rejecting uncertain predictions and deciding on the appropriate accuracy versus rejection ratio

tradeoff, there is an alternative approach which uses Bayesian inference to fuse posterior probabilities with information about the home environment to infer the location of the subject. This shall be the focus of the following chapter.

This section, however, has highlighted the improvements which can be made to the location prediction accuracy by using experimental knowledge of the behaviour of the signals in the home environment. It has shown that, in the case of GMM, 17% improvement in accuracy can be achieved by filtering, removing nuisance signals such as CRSSI, incorporating user location preferences and detecting difficult-to-classify features.

5.5 Conclusions

The experimental testbed presented in this chapter enabled numerous experiments to be carried out. These experiments were aimed at quantifying the performance of the localisation system developed in this thesis over significant periods of time. The 3 major novel contributions of this chapter are:

Technology Comparison Three different localisation technologies were deployed for long-term data acquisition. Besides the localisation system developed in this thesis, a traditional home PIR localisation system was also deployed. This chapter compared the localisation accuracy of the Bluetooth[®] localisation system with the PIR localisation system. It was shown that the Bluetooth[®] localisation technique has higher accuracy in the single occupancy scenario and experiences less performance degradation in the multiple occupancy scenario.

A novel RFID localisation technique was also designed and deployed. Section 5.1, however, noted that it cannot necessarily be considered a realistic RFID localisation system since it requires users to scan an RFID device every time they pass through a door. Instead it is used to obtain training data labels over long periods of time. This means that it would require an experienced installer to obtain the training data to ensure the tags are consistently read every time a room is entered. Requiring an installation technician to spend a day walking around the house of the subject is not an ideal deployment scenario, even if the technician was able to exactly replicate the typical movements of the subject, which is unlikely to be the case.

Instead, it is proposed that the RFID reader would be mounted on the shoe of the elder as illustrated in Figure 5.29 and the training data can be obtained passively while the subject follows their typical routine. To detect the shoe crossing the room interface the RFID reader would need to scan at a high interval. To last a day with this high scan interval the RFID reader would need to be fitted with a larger battery. To last several days the battery would need to be charged or changed every day by the elder. Using the

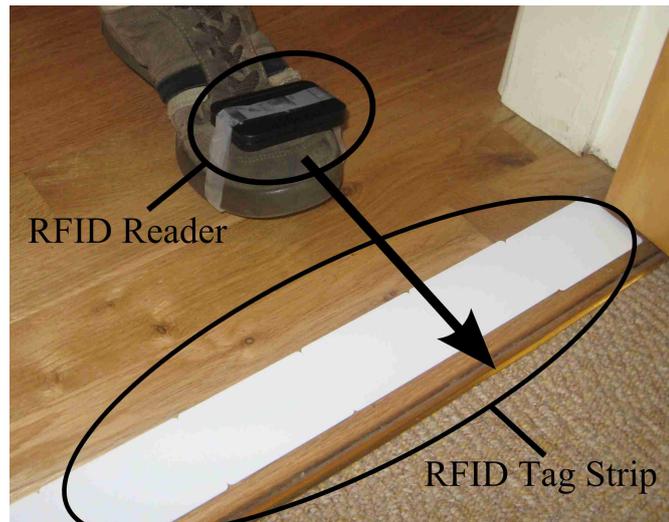


Fig. 5.29: Automatic RFID label acquisition technique. When the foot moves across the RFID tag strip an RFID reading occurs. Then the IDs of the read tags can be used to resolve the room-level location.

RFID technology for the first day allows the acquisition of reliable training data, and after the first day the more convenient mobile phone, which has alternative functionality, is the only device the elder needs to carry and care for.

The use of the RFID localisation technique allowed the inference of an accuracy metric corresponding to long-term movements and location preferences which contribute to more accurate location predictions. Hence, the availability of an RFID localisation technique is imperative to the reliable installation in a realistic environment. Furthermore, reliable long-term location labels allowed the comparison of the Bluetooth[®] localisation technique which is the subject of this thesis and the PIR localisation commonly utilised in elder monitoring scenarios.

The Importance of Signal Diversity It has been demonstrated that signal diversity is the main contributor to high localisation accuracy. It provides more location indicative features throughout the environment. The single AP test scenario had the lowest accuracy. When a second AP was added at the basestation the accuracy increased, only marginally because of the transmission power limitation of the Bluetooth[®] protocol. When several affordable beacons were deployed in the environment, using a novel connection topology, accuracy significantly increased. The availability of these beacons allowed an investigation of the effect of each beacon on the long-term accuracy. It was confirmed that accuracy is sensitive to the location of deployment of each beacon.

The use of extra Bluetooth[®] beacons means this technique begins to approach a traditional RF localisation scenario, in that multiple reference points are used. However,

it still forgoes the high cost of previously employed RF localisation topologies which used either networked APs or power-hungry and sometimes inconvenient WiFi devices such as PDAs or laptops. Previously these types of devices were necessary because conveniently carried Bluetooth[®] devices are not sophisticated enough to allow local estimation of RSSI information. This work has overcome this limitation by employing a novel technique of retrieving signal measurements to the mobile phone. Hence, signal diversity is generated throughout the environment with the installation of relatively inexpensive hardware.

Intelligent Handling of Signals Finally this chapter investigated and concluded on the most appropriate treatment of the signals in the given deployment scenario. A smoothing filter was used to reduce measurement noise, lagged-preprocessing was used to generate probabilistic models for the time-evolution of the signals and nuisance signals were removed for particular classifiers. Knowledge of user location preferences were incorporated into the density estimation techniques and analysis of certainty was used to decide on which predictions were sufficiently confident.

The result of this improved handling of signals led to higher localisation accuracy. Most notably, there is a significant trade-off in the specification of the rejection probability margin. A high margin ensures higher accuracy predictions at the expense of being more frequently unable to make predictions. Conversely, a low margin allows predictions to be made more frequently but with less accuracy. Fortunately there is another technique for handling probabilistic class predictions which produces predictions based not only on class posteriors but also knowledge of the deployment scenario. These elements can be incorporated using a Hidden Markov Model framework, which is the subject of the following chapter.

Hidden Markov Model Constrained Localisation

All of the location predictors presented in this thesis so far produce location estimates on a sample-by-sample basis. Each location prediction is based only on the current signal-derived feature set, regardless of the previous or future location predictions. Since the main focus of this work is on safety critical location monitoring and convenient location-based interactions, the location predictions must be produced as fast as possible (often referred to as real-time predictions). Such a constraint on predictions means that future predictions can be assumed to be unavailable. However, at every sample instant, information about the previous location predictions can be utilised. This thesis has already illustrated the positive effects of integrating the location preferences of a user into location predictions. With the availability of previous location predictions the movement preferences of a user can also be integrated into the location prediction framework.

As briefly presented in Section 2.2.2, Sequential Bayesian Filtering can be used to fuse current location probability predictions with previous location posterior densities to mitigate the effect of location jitter resulting from measurement noise. Generally speaking, this is performed by using a motion model to describe how a user is most likely to move. The predicted movement is then refined by the measurement model which relates the most likely location to the measurements when they become available. Since this work is focused on predicting the discrete location of a user, a discrete topological approach is employed. From a localisation perspective, the most efficient discrete topological sequential Bayes' filtering technique is the Hidden Markov Model (HMM).

Although the previously proposed location estimation algorithms produce acceptably accurate location predictions over long periods of time, they still have deficiencies in their frequency of movement prediction. Hence, the first section of this chapter motivates further

localisation system performance measures which progress beyond simple correct sample ratios. Section 6.2 proposes the HMM framework in an attempt to improve the overall performance of this localisation system. Section 6.3 attempts to improve the accuracy of traditional HMM algorithms even further by including higher levels of information about the movement habits of the user. Finally Section 6.4 demonstrates improvements to the HMM algorithms which allow HMM filtered predictions to occur as close to real-time as possible.

6.1 Dynamic Performance Measures

The accuracy metrics considered in previous chapters sufficiently describe the location detection performance of the localisation system over long deployment periods. They essentially allow the estimation of the number of samples for which the localisation system would correctly predict the location of the user in a typical day. These metrics, however, are unable to detect the difference between a classifier which incorrectly predicts user location one in every 10 samples and a classifier which is 100% correct for the first 90% of the day, then 100% incorrect for the remaining 10% of the day. Intuitively, when one reviews the movements of the user for both classifiers two different movement interpretations are possible. The first classifier would imply that the user moved between rooms on average once every 5 seconds. The second classifier would correctly detect the movements of the user for most of the day and give an unrealistic estimate for a very small portion of the day. The absolute accuracy measures considered up to this point mask such effects.

As a result, it is necessary to consider further classifier performance measures which highlight this important dynamic characteristic for the chosen classifiers. Three such performance measures are considered.

6.1.1 Transition Count

One of the simplest applications which can be provided by the localisation system is to summarise the levels of movement a user exhibits during a day. This can be used as a measure of the level of healthy exercise which a person automatically undertakes and can also be indicative of a change in their cognitive health (Lightner and Erdogmus, 2008). This room transition count can easily be derived from the number of times the user moves between locations. Each time the predicted room label changes from one sample to the next is noted as a transition. Then the transition count is simply a record of the number of such transitions in a given day. Accordingly, quantifying the ability of the system to accurately detect the number of transitions in a given day is the simplest measure of the dynamic performance of the system. This requires the definition of a new localisation

	NBC	LDA	QDA	GMM
Raw Data Classifiers TCE	546.5	358.5	300.5	441.5
Augmented Classifiers TCE	161	154.5	478	492

Table 6.1: Mean Transition Count Error (TCE) for the various classifiers compared to the true value.

	NBC	LDA	QDA	GMM
Raw Data Classifiers TCIF	2.72	2.13	1.95	2.37
Augmented Classifiers TCIF	1.51	1.48	2.52	2.56

Table 6.2: Mean Transition Count Increase Factor (TCIF) for the various classifiers compared to the true value.

performance metric, the Transition Count Error (TCE). The TCE indicates the difference between the actual number of transitions for the given test day and the detected number of transitions.

It would, however, be inappropriate to directly count the number of transitions between predicted labels, since any measurement noise which still remains after smoothing will cause the predicted room to temporarily change. This will lead to a higher transition count simply because of short-term misclassifications. This effect must be minimised before room-transition counting takes place. This is achieved by obtaining a moving window of the data. For each window the predicted sample is taken to be the most frequently occurring sample, or the mode, of this window. A relatively short window length of $n = 3$ is selected to minimise the prediction lag resulting from this operation. This processing, which we refer to as mode filtering, is performed immediately before the room transitions are counted.

Table 6.1 shows the mean TCE obtained with the the Multiple AP (MAP) data considered in the previous chapter. The TCE is considered both for the application of the classifiers directly to the available data (as in Section 5.3) and when the classification augmentations are applied (as in Section 5.4). From this data it can be seen that for the raw data the lowest transition count error occurs with the QDA classifier. When the classification augmentations outlined in the previous chapter are applied, the TCE decreases for the NBC and LDA classifiers but actually increases for the QDA and GMM classifiers. This is likely due to the fact that NBC and LDA use lagged features for the augmented classifiers, leading to more stable predictions and less spurious transitions than the augmented QDA and GMM classifiers, hence a lower TCE. Overall, LDA allows the best transition predictions using augmented classifiers.

This TCE measure does not have a great deal of meaning unless one considers the number of transitions which actually occurred. For this reason another performance

metric must be employed. This metric, the TCIF, presents the number of predicted room transitions with reference to the actual number of transitions which occurred. Hence, it indicates the factor by which the number of room transitions is overestimated. It is calculated from the expression

$$TCIF = \frac{\hat{c}}{c}, \quad (6.1)$$

where \hat{c} is the predicted number of room transitions and c is the actual number of room transitions. The average transition count for the two days is 318, which is taken into account in Table 6.2. Table 6.2 confirms that the best performing classifier for the raw data in terms of transition number estimation is QDA, which has less than twice as many predicted room transitions as the actual number. The best performing augmented classifier overestimates the number of room transitions by a factor of 1.5. Hence, the classifiers which appeared to have the worst EA of the augmented classifiers considered in Chapter 5, NBC and LDA, now have the best room transition detection performance. Accordingly, there is a tradeoff in choosing between classifiers which have high location prediction accuracy and classifiers which have the ability to detect movements between rooms. Furthermore, even though the augmented classifiers show superior movement detection ability, the best overestimates the transition counts by a factor of 1.5.

6.1.2 Distance Travelled

Transition count is a useful indicator of the number of times a user transitioned between rooms in a given period of time. However, the implications of such a reading is lost when information about the layout of the house is unavailable, i.e. a large transition count in a small home environment corresponds to less healthy exercise than the same transition count in a large home environment. Hence, to allow activity level estimation which is comparable between people in different home environments, an algorithm to estimate the distance a user travels on a particular day is developed. Now, instead of an environment-independent transition count, stakeholders in the health of the user/users can understand the user's exercise levels in a meaningful manner.

The travel distance estimator is a simple extension of the transition count estimator. The distance between the centre of one room to another room is assumed to be the average distance travelled when moving between them. Each time a transition between two rooms is detected the distance between the room centres is added to the current total distance for that day. Again a mode filter is applied to reduce spurious transitions predicted between rooms. For the MAP dataset, the actual average distance travelled was 1141.64 m. Similar to before, Table 6.3 highlights the average daily Predicted Distance Error (PDE). Again, QDA has the lowest average error for the raw data but NBC and LDA have the lowest

	NBC	LDA	QDA	GMM
Raw Data Classifiers PDE	2030.21	1477.46	1426.31	1878.16
Augmented Classifiers PDE	594.81	772.01	2015.96	2035.36

Table 6.3: Mean PDE for the various classifiers compared to the true value [metres].

	NBC	LDA	QDA	GMM
Raw Data Classifiers PDIF	2.78	2.29	2.25	2.63
Augmented Classifiers PDIF	1.52	1.67	2.77	2.79

Table 6.4: Mean PDIF for the various classifiers compared to the true value.

	NBC	LDA	QDA	GMM
Raw Data Classifiers MTD	4.42	4.78	4.68	4.92
Augmented Classifiers MTD	4.79	4.27	4.05	3.97

Table 6.5: The MTD averaged over both test days. [seconds]

errors for the augmented classifiers.

Similar to the TCE, it is beneficial to consider the factor by which the predicted distance is above the actual distance. This Predicted Distance Increase Factor (PDIF) is presented in Table 6.4. The augmented NBC classifier distance prediction is the most similar to the actual distance. However, the best-case PDIF is still 1.52 times that of the actual travelled distance. Hence, there is still the necessity for localisation algorithm improvements which allow more accurate dynamic behaviour estimation.

6.1.3 Transition Delay

The two previous performance measures allowed quantification of the ability of the system to detect the levels of movement a user exhibits in a given day. They do not, however, provide an indication of the delay between a room transition and the detection of that room transition. Accordingly, there could be a significant time difference between when a user moves between rooms and when the room transition is detected. Hence, a third dynamic localisation performance metric is proposed, the Mean Transition Delay (MTD). For this performance metric, each time a room is entered, the length of time it takes for the predictions to indicate that this room has been entered is noted as the delay. Then the mean of all of the delays for a given day is calculated. If the room remains undetected when the user transitions from the room, the timer is discarded and the next occupied room is monitored for delay.

Table 6.5 indicates that, unlike before, the augmented probabilistic classifiers which use covariance information in their density estimates produce the best performance. This is

due to the fact that augmented QDA and GMM do not have lagged feature preprocessing, which means that they do not experience delay in waiting for a particular temporal template to occur before detecting a new room. Hence, there is a tradeoff between the speed with which a room is detected, the reliability of the distance travelled predictions and the overall long-term localisation accuracy.

6.1.4 Overall Performance Visualisation

It has been illustrated that the overall performance of the localisation system involves significant tradeoffs between the long-term accuracy, transition detection ability, distance estimation ability and the speed in detecting transitions for the localisation system. Hence, a technique for comparing all of these traits for all classifiers at the same time must be developed. Since the PDE can be derived from the TCE, while giving information which is more comparable across different environments, the TCE measure shall be ignored. This leaves three performance measures to be compared; EA from the previous chapter, PDIF and MTD. Fortunately the tradeoff between three different measurements can readily be visualised in a three-dimensional plot.

Figure 6.1 summarises the tradeoffs for the NBC, LDA, QDA and GMM algorithms for both when raw data is used and when the augmented classifiers are used. Figure 6.1(a) shows a general view of the comparison, while Figures 6.1(b) and 6.1(c) allow more direct comparison between the classifier traits. For EA and PDIF, the optimal values are 1. For the MTD the optimal value is 0. From this comparison it can be gleaned that QDA is the best raw data classifier. It has the best EA and PDIF. It has higher delay at detecting transitions than NBC. This, however, is acceptable when one considers that NBC achieves faster room detection at the expense of localisation accuracy.

Figure 6.1(a) confirms that the augmented classifiers generally have superior overall performance than the raw data classifiers. In particular, QDA and GMM have the best performance in terms of accuracy and transition detection delay. This performance comes at the expense of less accurate distance estimates. LDA has higher accuracy and better travel distance estimates at the expense of having higher transition prediction delays. NBC, on the other hand, sacrifices accuracy and transition prediction timing in favour of travel distance estimation ability.

Hence, the long-term accuracy is not the only metric relevant for appraising the performance of the localisation system. Other dynamic effects such as transition prediction timing and the ability to reliably detect movement levels over time are also important. Based on this knowledge, a more complete performance comparison framework has been presented. This contribution is entirely novel in the field since other work generally only focuses on the error of coordinate position predictions. This thesis highlights system performance in terms of being able to detect user-relevant locations, user activity levels

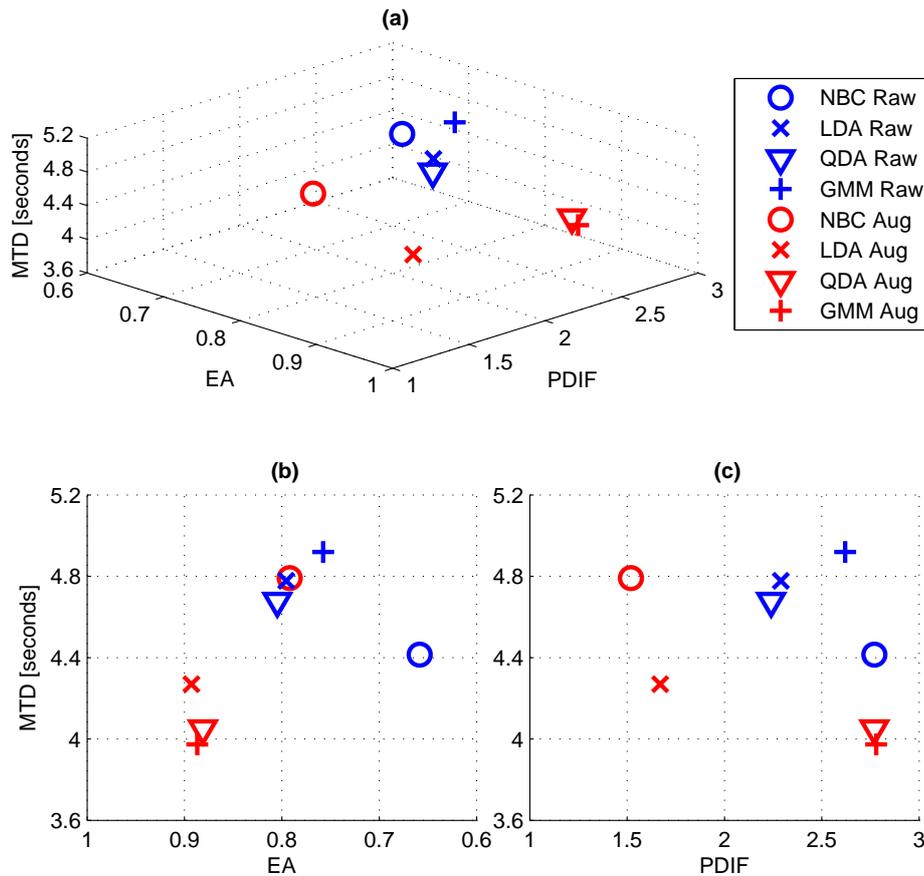


Fig. 6.1: Localisation algorithm performance comparison for the different algorithms. The projection in (a) allows a visualisation of the best overall performing classifiers. (b) allows a comparison of the classifiers in terms of EA and MTD and the projection in (c) allow a comparison of the classifiers in terms of PDIF and EA. EA and PDIF are optimal at values of 1. MTD is optimal at a value of 0.

and the speed in making such predictions. The availability of such a balanced performance measure will allow better illustration of the improvement a discrete Bayesian filtering approach will bring.

6.2 Hidden Markov Models

The primary disadvantage of all of the classification techniques considered up to this point is that they do not have any natural mechanism of incorporating information from the previous predictions into the current prediction. Each location prediction is locally optimal on an individual basis. The ability to use previous predictions to narrow down the subsequent possible predictions would clearly be beneficial in a system involving a user moving through a known configuration of rooms in an environment. Hence, HMMs have a

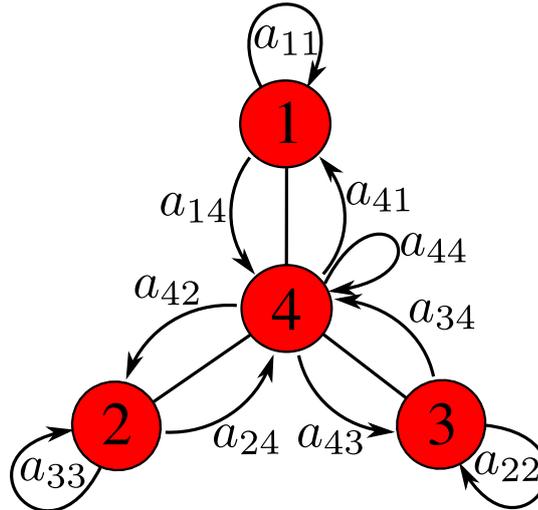


Fig. 6.2: A Markov chain representing the configuration of the environment used for the controlled test environment used in Chapter 4.

natural application to this localisation system. Before the theory behind Hidden Markov Models is presented it is necessary to consider an unsupervised method of modelling discrete states.

6.2.1 Markov Chains

A Markov chain is a discrete random process in which it is assumed that the values of future states are dependent only on the values of the present states. In a Markov chain there is a finite number of discrete internal states. Each state typically corresponds to a real-life entity. In the case of modelling the movements of a user in their home environment, each state corresponds to a symbolic location. Hence, the progression of the location of a user over time can be modelled as a discrete-time first-order Markov chain. The first order restriction ensures that the discrete state, or location, at time t , denoted q_t , is dependent only on the location at time $t-1$, denoted q_{t-1} . If there are N possible locations, denoted L_1, L_2, \dots, L_N , the probability of moving from one location $q_{t-1} = L_i$ to the next location $q_t = L_j$ is represented by the transition probability a_{ij} . Figure 6.2 illustrates a graph structure which corresponds to the simple test environment used in Chapter 4. From this figure it can be seen that rooms 1, 2 and 3 are all connected to room 4. Hence, there are certain probabilities of transitioning from each room to room 4, denoted a_{14} , a_{24} and a_{34} . There are also the probabilities of transitioning from room 4 to any other room, denoted a_{41} , a_{42} and a_{43} . Since rooms 1, 2 and 3 are not directly connected their transition probabilities a_{12} , a_{23} and a_{31} are equal to zero. Finally, the quantities a_{11} , a_{22} , a_{33} and a_{44} represent the probabilities of staying in the current location at any instant in time.

Using the a_{ij} values for every i - j combination, an $N \times N$ transition probability matrix, denoted \mathbf{A} , can then be constructed, where each row corresponds to the room being moved

from and each column corresponds to the room being moved into. The diagonal elements indicate the probability of staying in the same room. Each element in the matrix is calculated as

$$a_{ij} = P(q_t = L_j | q_{t-1} = L_i). \quad (6.2)$$

Then at every instant in time, the probability of being in location i and then transitioning to location j can be calculated using

$$P(q_{t-1} = L_i, q_t = L_j) = a_{ij}P(q_{t-1} = L_i). \quad (6.3)$$

Thus the probability of being in a particular state at time $t = T$ can be calculated recursively through the known sequence of states $\mathbf{q} = q_1, q_2, \dots, q_t$. The probability of being in the state L_i at $t = 1$ is based solely on the initial probability, π_i . For $t \geq 2$ the probabilities are obtained simply by multiplying the probability of the previous state by the transition probability from that state to the current one.

The transition probability matrix, \mathbf{A} , can be calculated from some reference data. For each element a_{ij} , the number of samples which correspond to a transition from location L_i to location L_j is divided by the total number of samples corresponding to a transition from L_i . Samples of the user staying in L_i are also included in this count as they correspond to transitions from L_i to L_i . As a result, each element of the transition matrix is calculated to be

$$a_{ij} = \frac{\# \text{ transitions from } L_i \text{ to } L_j}{\# \text{ transitions from } L_i}. \quad (6.4)$$

Hence, a Markov chain could be used to simulate the movements of a user throughout a typical day, to gain insights into where a user is likely to be at a certain time of the day. However, this functionality alone is of little use to our location prediction problem. Further information must be used in parallel with this model to enable location predictions.

6.2.2 HMM Theory

The Markov chains described in the previous section are only useful when the state sequence, \mathbf{q} , is directly observable. From the localisation perspective the sequence of locations is certainly not observable online, since it is the sequence which we are interested in predicting using the available RF signals. A HMM is a model in which the state sequence is not observable and all that is visible is the observation sequence $\mathbf{o} = o_1, o_2, o_3, \dots, o_T$. Hence, besides the transition probability matrix, a HMM also needs an observation probability model which relates each state to a probability of observing a particular reading. Such a probability model, for a given class j at time t takes the form,

$$b_j(o_t) = P(o_t|q_t = L_j). \quad (6.5)$$

The collection of measurement models for an entire HMM is denoted \mathbf{B} .

As a result a HMM is entirely parameterised by the expression

$$\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi}), \quad (6.6)$$

where \mathbf{A} is the transition probability matrix, \mathbf{B} is the observation probability model and $\boldsymbol{\pi}$ is the initial state probability vector. For such a HMM there are three problems which are typically of interest (Rabiner, 1989):

Problem 1: Given a particular observation sequence and model, how do we compute $P(\mathbf{o}|\lambda)$, the probability of getting the observation sequence given the particular parameters of λ

Problem 2: Given the observation sequence and model, how do we estimate the most likely sequence of hidden states.

Problem 3: Given the observation sequence how can the model parameters be chosen to maximise $P(\mathbf{o}|\lambda)$.

Problem 2 is particularly applicable to this work since in an RF localisation system the observation sequence is available in the form of RF readings and we are interested in estimating the most likely underlying state sequence. Hence the entire HMM, λ , is available since \mathbf{A} is derivable from the statistics of a user's room occupancy over the period of a day. The probabilistic measurement model matrix, \mathbf{B} , is also already available since the probabilistic classifiers employed in previous chapters are essentially probabilistic signal models with class decisions applied. The quantity $b_j(o_t)$ is identical to $P(\mathbf{x}|R_j)$ for the probabilistic classifiers. Hence, the measurement model matrix can be constructed from the parameters of the probabilistic models presented in previous chapters. Finally, the initial probability vector $\boldsymbol{\pi}$ can be assumed to be equal across all classes or if the starting room is known, its probability can be set to 1 and all others set to zero. The estimation of the most likely state sequence from this data is commonly estimated using the Viterbi algorithm.

6.2.3 The Viterbi Algorithm

The major issue with the previously employed classification algorithms is that they perform classifications without knowledge of the connectivity of the environment or the movement habits of the subject. When one considers the transition matrix it may become evident that some of these transitions should rarely ($a_{ij} \approx 0$), or never ($a_{ij} = 0$) occur. Hence,

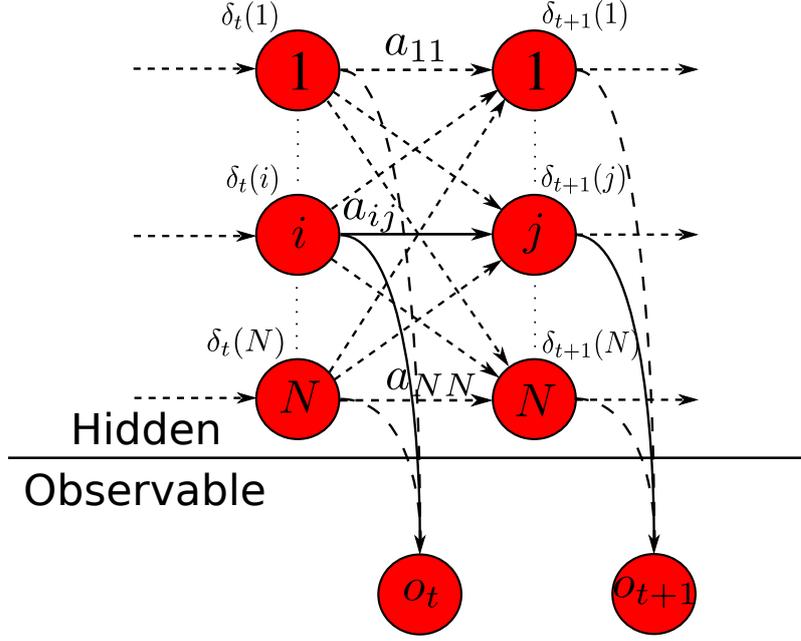


Fig. 6.3: Estimating the partial probability of all states at the next instant of time using only the information about the probabilities at the current instant in time. Moving from left to right on the graph corresponds to time progression.

instead of producing the locally optimal prediction, the Viterbi algorithm is concerned with estimating the sequence of states which is globally optimal when considering the sequence of observations. This maximisation of $P(\mathbf{q}|\mathbf{o}, \lambda)$ is equivalent to maximising $P(\mathbf{q}, \mathbf{o}|\lambda)$. For convenience we define this as the partial probability,

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1, q_2, \dots, q_t = i, o_1, o_2, \dots, o_t | \lambda), \quad (6.7)$$

which is the un-normalised probability of being in the most likely location, location i , at time t and observing the measurement sequence o_1, o_2, \dots, o_t .

The assumption of a first-order Markov process means that if the quantity $\delta_t(i)$ is available then the probabilities of earlier states need not be known for the estimation of future states. The partial probabilities of all states reached from this state, i , on the next time interval, $t + 1$, can be calculated from the expression,

$$\delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] \cdot b_j(o_{t+1}), \quad (6.8)$$

which is the product of the probability of the most likely current state and the probability of the next state given the observation o_{t+1} . Hence, the partial probability of each state at each instant in time can be maintained by a recursive algorithm which only needs memory of the states from the previous iteration, as illustrated in Figure 6.3.

It is important to note that the location of highest partial probability in each step

is not necessarily the most likely location, since it may lead to an unlikely or even an impossible location given the transition matrix and future observations. Hence, at every iteration of the recursive algorithm it is necessary to store the most likely previous location for each current possible location. This corresponds to the value of i in Equation 6.8. A matrix Ψ is constructed which keeps track of the most likely previous state for every state at each instant in time. Then the final part of the Viterbi algorithm is to start at the final most likely prediction and recursively “backtrack” over time, calculating the most likely previous state for every state. After this step, the globally most probable sequence of states is available.

Mathematically the Viterbi algorithm is executed in four steps (Rabiner, 1989):

1 Initialisation

$$\delta_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N \quad (6.9a)$$

$$\Psi_1(i) = 0. \quad (6.9b)$$

2 Recursion

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t), \quad 2 \leq t \leq T, \quad 1 \leq j \leq N \quad (6.10a)$$

$$\Psi_t(j) = \operatorname{argmax}_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \quad 2 \leq t \leq T, \quad 1 \leq j \leq N \quad (6.10b)$$

3 Termination:

$$q_T^* = \operatorname{argmax}_{1 \leq i \leq N} [\delta_T(i)]. \quad (6.11)$$

4 Backtracking:

$$q_t^* = \Psi_{t+1}(q_{t+1}^*), \quad t = (T-1), (T-2), \dots, 1. \quad (6.12)$$

This is implemented in Matlab[®] using the pseudocode described in Algorithm 1. Due to the backtracking step, the Viterbi algorithm in its current form cannot produce location estimates in real-time. The final sample for a given period of time must be processed before a globally informed location decision can be made for each state at each instant in time. Hence, improvements to the Viterbi algorithm will be considered in Section 6.4 to enable location predictions to be produced as fast as possible and the effect this has on localisation performance will be investigated.

It should be remembered that the Viterbi algorithm implements a discrete state form of the sequential Bayes’ filter. When comparing the Viterbi algorithm implementation with the general Bayes’ filter nomenclature in Section 2.2.2 similarities are evident. In particular, Equation 6.10a implements the recursive “Process Model”-“Measurement Model” update. The $\max_{1 \leq i \leq N} [\delta_{t-1} a_{ij}]$ term is the process model which relates the

```

Input: observationSequence, obProbs, transMatrix
Output: decodedLocSeq
// Initialise probabilities at  $t = 1$ :
for  $j \leftarrow 1$  to  $N$  do
    prior  $\leftarrow 1/N$ ;
    currentOb  $\leftarrow$  observationSequence(1);
    obProb  $\leftarrow$  GetObProb(currentOb,  $j$ , obProbs);
    partialProb(i) = prior  $\times$  obProb;
end
// Recursively update state probabilities for  $2 \leq t \leq T$ :
for  $t \leftarrow 2$  to  $T$  do
    currentOb  $\leftarrow$  observationSequence( $t$ );
    for  $j \leftarrow 1$  to  $N$  do
        obProb  $\leftarrow$  GetObProb(currentOb,  $j$ , obProbs);
        for  $i \leftarrow 1$  to  $N$  do
            possibleCurrentProb( $j$ )  $\leftarrow$  prevProb( $i$ )  $\times$  transMatrix( $i,j$ )  $\times$  obProb;
        end
        currentProb( $j$ )  $\leftarrow$  max(possibleCurrentProb);
        likelyPrevState( $t,j$ )  $\leftarrow$  argmax(possibleCurrentProb/obProb);
    end
    prevProb  $\leftarrow$  currentProb;
end
// Termination:
decodedLocSeq( $T$ )  $\leftarrow$  argmax(prevProb);
// Backtracking:
for  $t \leftarrow (T - 1)$  to 1 do
    decodedLocSeq( $t$ )  $\leftarrow$  likelyPrevState( $(t + 1)$ , decodedLocSeq( $t + 1$ ));
end

```

Algorithm 1: Viterbi algorithm implemented in Matlab[®]

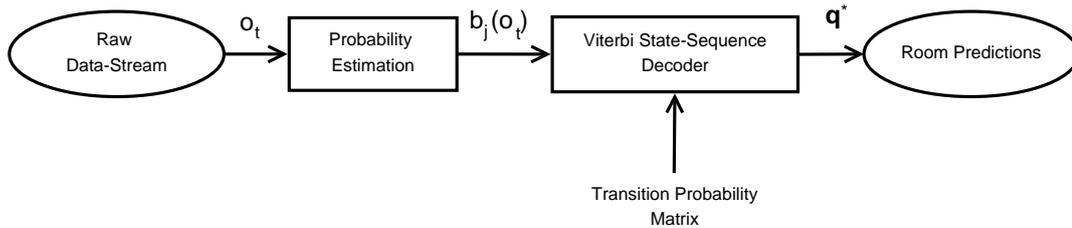


Fig. 6.4: Schematic of the HMM localisation technique applied to raw signal data.

internal location state at one instant in time to the next and the $b_j(o_t)$ term is the measurement model which refines the predicted location probabilities with information derived from the newly available measurements.

6.2.4 Bayesian Filtering Performance

To enable Bayes' filter location prediction updates, the previously described Viterbi algorithm is implemented as per Algorithm 1. Figure 6.4 illustrates the complete localisation system which uses the Viterbi algorithm in an attempt to provide improved location predictions. It can be seen that the Viterbi state-sequence decoder takes two inputs. The transition probability matrix is calculated from the training data labels, and as a result corresponds to the intra-house movement pattern of the user on the training data acquisition day. An illustration of the transition probability matrix for this environment can be seen in Figure 6.5. This matrix describes the subject movement inclinations and room connectivity of this environment. For example, to move from room 2 to room 10 a subject must travel from room 2 to room 3, then room 4, then room 7 and finally room 10. Some elements of this matrix are too faint to be seen, such as room 1 to room 2 and room 3 to room 2. This is because transitions from these rooms happen rarely when compared with the length of time spent in the rooms. The observation probability model, $b_j(o_t)$, is derived from the probability estimation block, which utilises the probabilistic signal models previously described for NBC, LDA, QDA and GMM.

Figure 6.6 compares location predictions for the original LDA classifier with the HMM classifier, which uses the observation probability model derived for LDA. It can be seen that in Figure 6.6(a), transitions between rooms occur more frequently and, in fact, behave in a nonsensical fashion. At many times, the predicted room oscillates rapidly between two different rooms. Such movements are not physically possible in reality. The HMM classifications in Figure 6.6(b), which constrains the predictions with room connectivity and typical user movement habits, produces much more sensible predictions over time.

Table 6.6 summarises the performance metrics resulting from this localisation system implementation. When comparing the HMM empirical accuracies with the raw empirical classifier accuracies in the previous chapter an increase in accuracy is evident. Furthermore

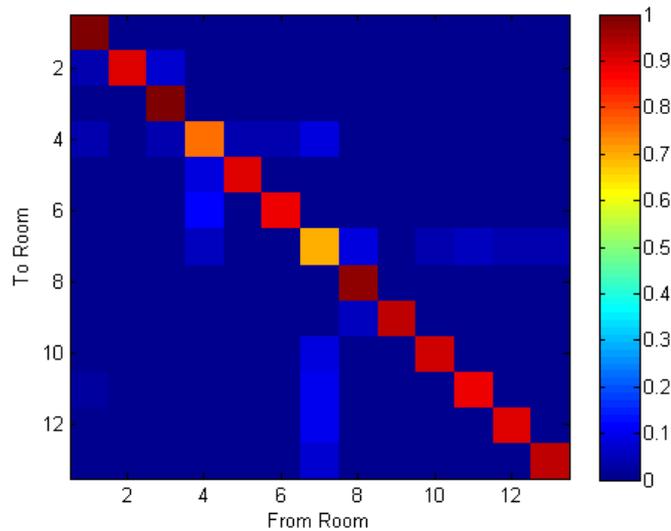


Fig. 6.5: Intensity plot of the transition probability matrix for the home environment.

	NBC	LDA	QDA	GMM
EA	0.71	0.83	0.83	0.79
PDIF	1.66	1.29	1.48	1.58
MTD	4.57	4.51	4.11	4.05

Table 6.6: Localisation performance comparison for the HMM classifier using different observation probability models.

when comparing this table with Tables 6.4 and 6.5 there is also an improvement in the lowest PDIFs but no improvement in MTDs. Even though the accuracy of the HMM classifier is higher than the raw data classifiers, it is still lower than the augmented location classifiers. The augmented location classifiers, however, achieved higher localisation accuracy by refusing to produce classifications in the presence of uncertainty. In fact 95% accuracy was possible by refusing to predict location 60% of the time. HMMs, on the other hand, produce classifications at all times even in the presence of uncertainty by integrating environment configuration and user movement tendencies into the predictions. Hence, HMM classifiers produce high localisation accuracy at all times while allowing more reliable transient event detection, such as user travel distance estimates and room transition detection timing.

6.2.5 Bayesian Filtering with Augmented Classifiers

Prior to considering HMMs for localisation system performance improvement, classifiers were augmented with pre-processing, prior room probability inclusion and uncertainty rejection. The HMM Viterbi state decoder completely negates the use of the uncertainty rejection block illustrated in Figure 5.27. There is, however, still the potential for

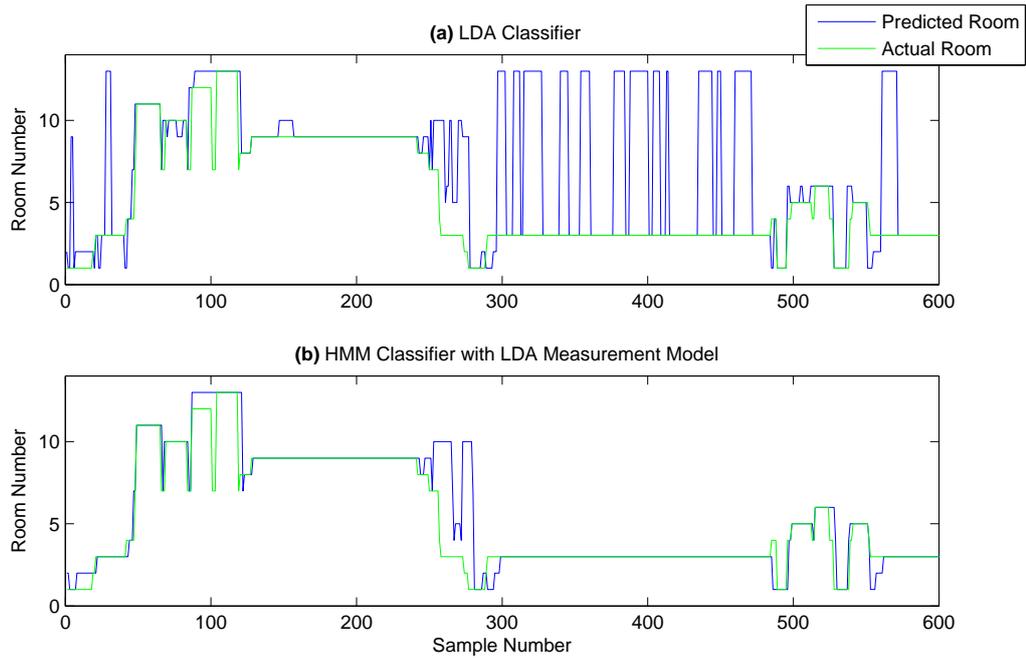


Fig. 6.6: Comparison of HMM predictions which use LDA derived observation probabilities and the original LDA predictions.

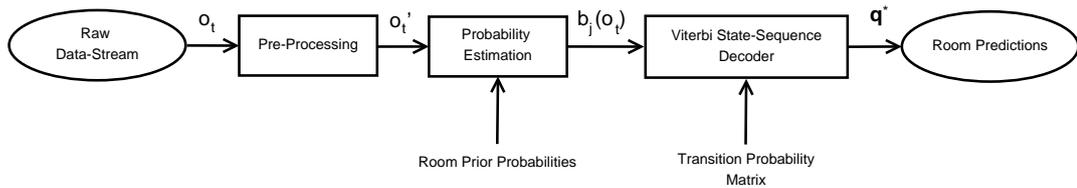


Fig. 6.7: Schematic of the HMM localisation technique which pre-processes the raw data and includes priors in the density estimation technique.

increasing performance by considering the other augmentations illustrated in Figure 5.27. That figure is similar to Figure 6.4 except that the probability estimation block in Figure 5.27 includes the prior room probabilities as explained in Section 5.4. Furthermore the observations can be pre-processed using filtering, lagging and cellular signal handling as described in Section 5.4.

Figure 6.7 illustrates the complete set of additions which can be made to the HMM framework to improve accuracy. Hence, there are three different ways in which the augmentations can be applied;

1. Only include prior probabilities in the density estimation. This means the pre-processing block is excluded from the system, making $o_t = o'_t$. This is referred to “Prior HMM” for comparison.
2. Only pre-process the observations as per Section 5.4. This means the prior probabilities are now equal across all classes, $\pi_i = 1/N$. This is referred to as

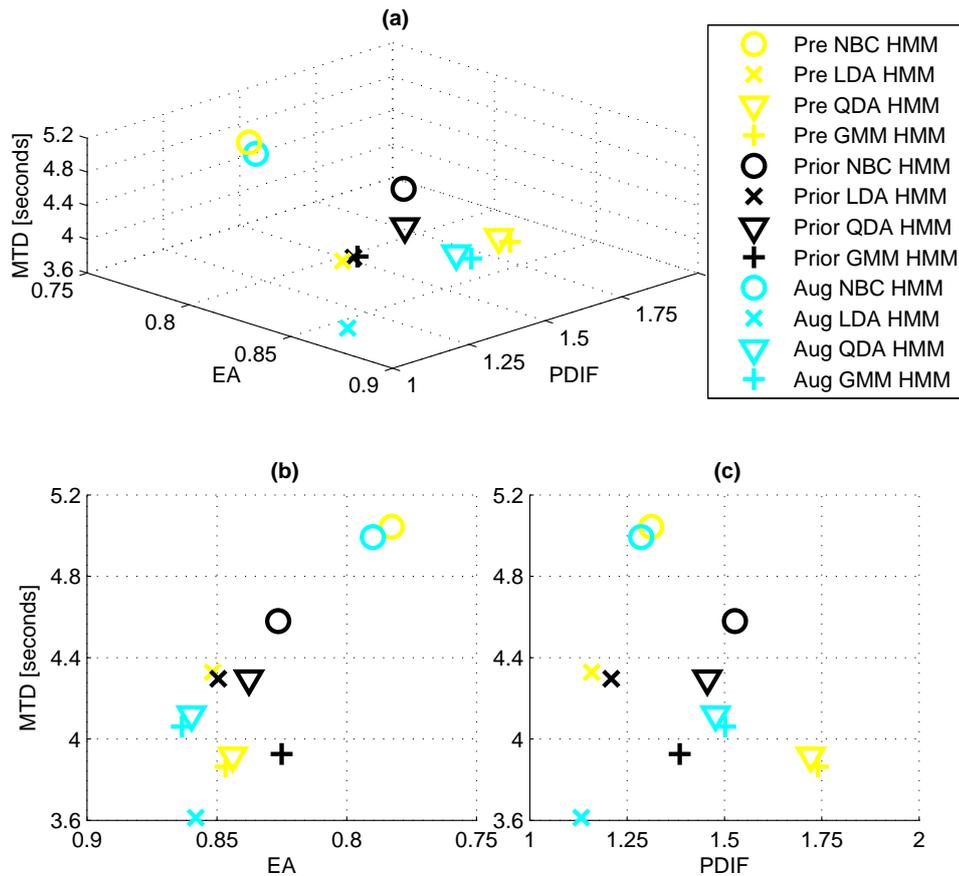


Fig. 6.8: A comparison of the performance trade-offs for the different HMM augmentations. As in Figure 6.1, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.

“Pre HMM” for comparison.

- Use both pre-processing and prior probabilities in the observation model, as illustrated in Figure 6.7. This fully augmented HMM configuration is referred to as “Aug HMM” for comparison.

Figure 6.8 illustrates the tradeoffs for these different approaches to improving HMM performance. From Figure 6.8(b) it can be seen that of the three HMM augmentation approaches the fully augmented LDA HMM produces the lowest MTD. It can also be seen from Figure 6.8(c) that using LDA density estimates produces the best PDIF for all HMM types. For overall performance it can be seen that Aug LDA HMM not only produces the highest accuracy, but also exhibits the best trade-off between PDIF and MTD.

Hence the fully augmented HMM allows the best performance of all the HMM augmentations. For comparison with Figure 6.6, Figure 6.9 illustrates the improvement the augmented LDA HMM classifier allows over the original augmented LDA classifier.

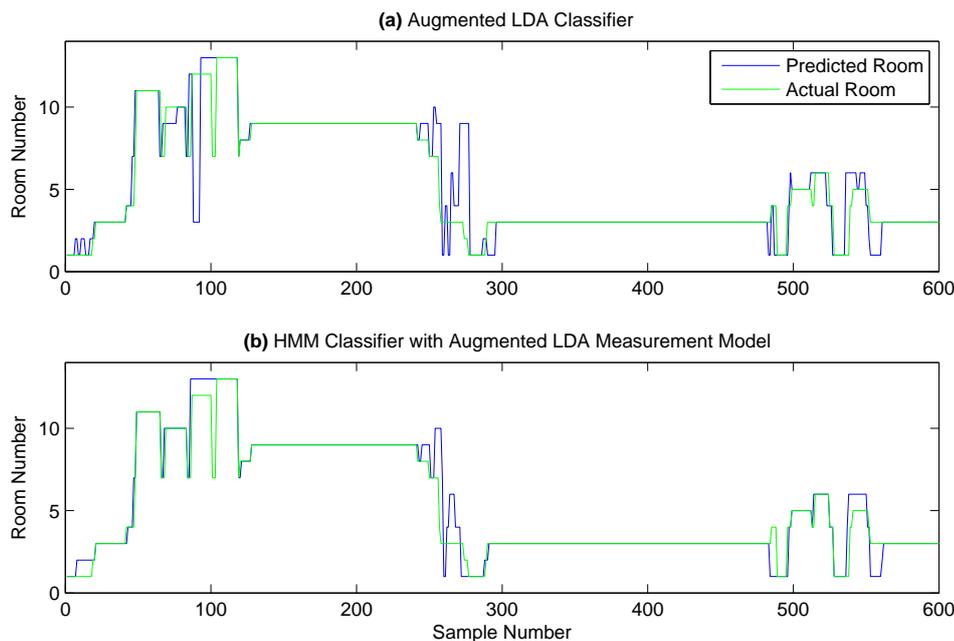


Fig. 6.9: Comparison of HMM predictions which use augmented LDA derived observation probabilities and the original augmented LDA predictions.

	NBC	LDA	QDA	GMM
EA	0.79	0.86	0.86	0.86
PDIF	1.29	1.13	1.48	1.50
MTD	4.99	3.61	4.12	4.06

Table 6.7: Localisation performance for the fully augmented HMM classifier.

It can be seen that the original augmented LDA classifier of Figure 6.9(a) has far fewer location fluctuations than the un-augmented LDA classifier in Figure 6.6(a). This is due to reduced misclassifications as a result of more appropriate preprocessing and prior probability inclusion. Furthermore, the augmented HMM classifier in Figure 6.9(b) has more realistic location predictions than the original augmented LDA classifier.

Table 6.7 summarises the overall performance of the fully augmented HMM classifiers. It is evident that the performance of the augmented classifiers is generally superior to that of the raw HMM classifier summarised in Table 6.6. Hence, augmented HMM classifiers, as with the original augmented classifiers, allow improved performance. The accuracy however, is still not as high as that which occurred with the augmented classifiers. This, however, is expected because the HMM classifiers do not need to resort to rejection in the presence of uncertainty.

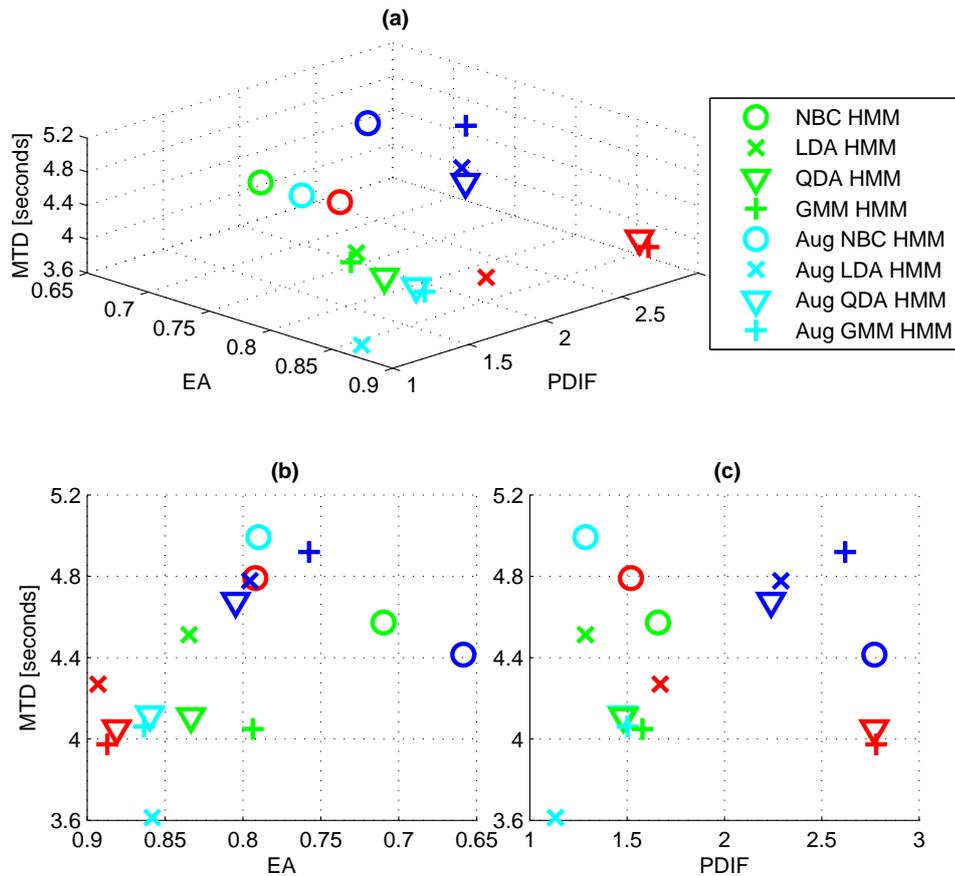


Fig. 6.10: Comparison of the localisation trade-offs between the different localisation algorithms. The blue and red markers represent the previously presented raw data and augmented classifiers respectively. Note that the Aug QDA HMM marker in figure (c) cannot be seen because it has identical PDIF and MTD to the un-augmented QDA HMM.

6.2.6 Overall HMM Performance

Now that the theory behind Hidden Markov Models has been presented and the application of the Viterbi algorithm to long-term localisation has been described, a summary of the effect of HMMs on localisation performance can be produced. Of particular note is that there are inherent trade-offs in the selection of the optimal localisation algorithm. The graphical method of comparing these trade-offs allows a direct comparison between the different localisation algorithms. Figure 6.10 compares the raw classifiers and augmented classifiers already presented in Figure 6.1 with the newly available raw data HMM classifiers and the augmented data HMM classifiers.

From the figure it can be seen that the highest absolute accuracy occurs with the original augmented classifiers. However, this accuracy comes with the caveat that the classifiers are unable to produce predictions at all times due to uncertainty rejection.

Besides that, the highest accuracy comes from the augmented HMM classifiers, which are able to produce these accurate predictions at all times. Not only are accurate predictions consistently available with the HMM classifiers, but the distance predictions and the transition detection speed are superior to the original classifiers which do not exploit HMM theories. Hence, HMMs provide improved overall localisation performance, with the best performance occurring with the augmented HMM classifiers.

6.3 High Order Markov Processes

The HMM has exhibited excellent ability to model the probability of a user occupying a discrete location over time. The main deficiency, however, of the HMM structure is that it relies on the first order Markov assumption that the current state depends only on the previous state. In an indoor localisation scenario, knowledge of states earlier in time than the previous state would clearly be beneficial. Understanding which room a person has transitioned from before the current room would allow insight into which room they may transition to next. The fact that the standard HMM allows only the most recent state probabilities to contribute to the evolution of the state probabilities in the following time instant prevents this. Hence, this section considers a modification to the HMM form presented in the previous section to allow the inclusion of further user movement information.

6.3.1 2nd Order Hidden Markov Models

To allow the inclusion of further movement information into the HMM it is necessary to forgo the first order Markov assumption. This allows modelling the probability of the current state on not just the previous state but the probabilities of any number of states prior to the current state. To investigate the applicability of such a HMM structure to this localisation problem a second order HMM was first considered. A second order Markov model extends the original HMM implementation to include state partial probabilities from both the last step in time and the second last step in time. Hence the transition probability matrix must be extended to note the probabilities of transitioning from the state q_{t-2} to state q_{t-1} and then to state q_t . One such transition probability is referred to as a_{ijk} . Hence, the transition probability matrix, \mathbf{A} , is now a $N \times N \times N$ matrix, where each element is

$$a_{ijk} = P(q_t = L_k | q_{t-1} = L_j, q_{t-2} = L_i). \quad (6.13)$$

As a result, the partial probability at each point in time is dependent on the state partial probabilities for the last time instant, i , and the second-last time instant, j . Hence,

the state partial probabilities at each time instant, $\delta_t(j, k)$, must be maintained by an $N \times N$ matrix. Similarly, the matrix to track the most likely previous state for each state at each time instant, Ψ , must also keep track of the most likely second last state based on the states in the subsequent two time instants. Hence, an $N \times N$ matrix must be maintained for each instant in time to allow backtracking of the most likely second last state, q_{t-2} , given the most likely subsequent states, q_{t-1} and q_t .

The most likely state-sequence in this HMM is decoded using the extended Viterbi algorithm (modified from He (1988));

1 Initialisation

$$\delta_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N \quad (6.14a)$$

$$\Psi_1(i) = 0. \quad (6.14b)$$

$$\delta_2(i, j) = \delta_1(i) a_{ij} b_j(o_2), \quad 1 \leq i \leq N, 1 \leq j \leq N \quad (6.15a)$$

$$\Psi_2(i, j) = 0. \quad (6.15b)$$

2 Recursion

$$\delta_t(j, k) = \max_{1 \leq i \leq N} [\delta_{t-1}(i, j) a_{ijk}] b_k(o_t), \quad 3 \leq t \leq T, 1 \leq j \leq N, 1 \leq k \leq N \quad (6.16a)$$

$$\Psi_t(j, k) = \operatorname{argmax}_{1 \leq i \leq N} [\delta_{t-1}(i, j) a_{ijk}], \quad 3 \leq t \leq T, 1 \leq j \leq N, 1 \leq k \leq N \quad (6.16b)$$

3 Termination:

$$q_T^* = \operatorname{argmax}_{1 \leq k \leq N} [\delta_T(j, k)], \quad (6.17a)$$

$$q_{T-1}^* = \operatorname{argmax}_{1 \leq j \leq N} [\delta_T(j, k)]. \quad (6.17b)$$

4 Backtracking:

$$q_t^* = \Psi_{t+1}(q_{t+1}^*, q_{t+2}^*), \quad t = (T-2), (T-3), \dots, 1. \quad (6.18)$$

This algorithm was also implemented in Matlab[®], with the transition probability matrix again derived from the training day data. Since it has already been shown that using the density estimates from the augmented classifiers leads to the best HMM performance, these will also be used to generate the observation probabilities, $b_k(o_t)$, for the second order HMMs. Table 6.8 highlights the performance for this implementation of the Viterbi algorithm. The table confirms that marginally higher long-term accuracy is possible with the higher order HMM, when compared with Table 6.7. Along with

	NBC	LDA	QDA	GMM
EA	0.79	0.87	0.87	0.87
PDIF	1.23	1.00	1.18	1.21
MTD	4.81	3.68	4.01	3.95

Table 6.8: 2nd order HMM algorithm using the augmented density estimation techniques to produce the observation probabilities, $b_k(o_t)$.

	NBC	LDA	QDA	GMM
EA	0.79	0.87	0.87	0.88
PDIF	1.13	0.90	1.05	1.08
MTD	4.81	3.69	3.94	3.85

Table 6.9: 3rd order HMM algorithm using the augmented density estimation techniques to produce the observation probabilities, $b_k(o_t)$.

higher accuracy, there is also improved ability to estimate the travelled distance in a given day, with HMMs using augmented LDA density estimates producing optimal distance predictions. The second order HMM also enables lower MTD for NBC, QDA and GMM, meaning it generally produces room predictions with lower delay.

The slight performance improvements possible with the higher order Markov model come with higher computational cost. The recursion step is the most significant part of the Viterbi algorithm since it involves performing multiplications for every combination of values for j and k . For an environment with N rooms, the second order Viterbi decoding algorithm has N times as many multiplications as the first order Viterbi decoding algorithm. Furthermore, the higher dimensionality of the previous state matrix, Ψ , causes a storage requirement increase by a factor of N . This higher memory requirement still does not exceed the memory restrictions of the test computer for the data for an entire day. Hence, the second order HMM is still computationally tractable on a desktop computer with moderate specifications.

6.3.2 3rd Order Hidden Markov Models

Since increasing the HMM order to 2 enabled higher empirical localisation accuracy and travel distance prediction accuracy with some extra computational overhead, it is necessary to investigate the implications of further HMM order increases. Hence, a third order HMM was implemented in Matlab[®]. The Viterbi algorithm presented in the previous section is extended to the third order by incorporating state probabilities for the previous three time intervals into the prediction of the current state probabilities. Hence, this requires a 4-dimensional transition probability matrix and for every discrete time interval; a 3-dimensional partial probability matrix and a 3-dimensional previous state matrix.

Table 6.9 presents the performance for the 3rd order HMM. This time it can be seen

that the increase in accuracy over the 2nd order HMM in Table 6.8 is minimal. More significantly, there is a further reduction in PDIF and a marginal reduction in MTD for QDA and GMM HMMs. Hence, if the computational implications of the selected algorithm were negligible, a higher order HMM would allow the best distance prediction and the lowest room transition prediction delay. The computational implications, however, are not negligible. The 3rd order Viterbi decoding algorithm is actually N times more computationally intensive than the second order HMM. This makes it N^2 times more computationally intensive than the first order Viterbi algorithm.

There are also extremely high memory requirements for the previous state matrix, Ψ . For the third order HMM this means that an $N \times N \times N$ matrix must be maintained for every discrete time interval. In the test scenario considered, there were up to 12,000 samples on a given day. This led to extremely high memory overheads, which actually caused the execution of the third order Viterbi algorithm to fail with a Matlab[®] memory warning on occasion. Hence, the execution of the third order HMM is extremely unreliable in its current form. Accordingly, Section 6.4 will consider a real-time augmentation to the Viterbi algorithm, which will not only reduce the location prediction lag, but also eliminate the significant memory requirements of the higher order HMMs by discarding many of the past samples when the current partial probabilities are high enough to make an informed decision.

Finally, it should be noted that Lee and J.-C.Lee (2006) illustrated that higher HMM efficacy is possible by using both a higher order transition probability model and a higher order observation probability model. This work, however, does not need to consider the higher order probabilistic observation model since the use of time-lagged input samples in the augmented density estimation framework can be considered a high-order probabilistic observation model. Hence, the order of the observation model is equivalent to the number of lagged samples, referred to as L in the previous chapter. The optimal values have been found to be $L = 7$ for the NBC, $L = 10$ for the LDA and $L = 1$ for both the QDA and GMM density estimation techniques. Hence, the overall HMM implementation in this thesis can be considered to have a high order probability transition probability model and a high order observation probability model, when appropriate.

In summary, Figure 6.11 compares the performance measures for augmented HMMs in the first order, second order and third order implementations. This confirms that the empirical localisation accuracy increases marginally with HMM order. It also shows that both PDIF and MTD generally decrease with higher model orders. Figures 6.11(b) and 6.11(c) indicate that the higher order HMMs have better overall performance than the first order HMMs. The only exception is for LDA which has marginally higher MTD for the 1st order HMM than for the higher order HMMs. Even though the second and third order HMMs have higher localisation performance, the third order HMM produces its

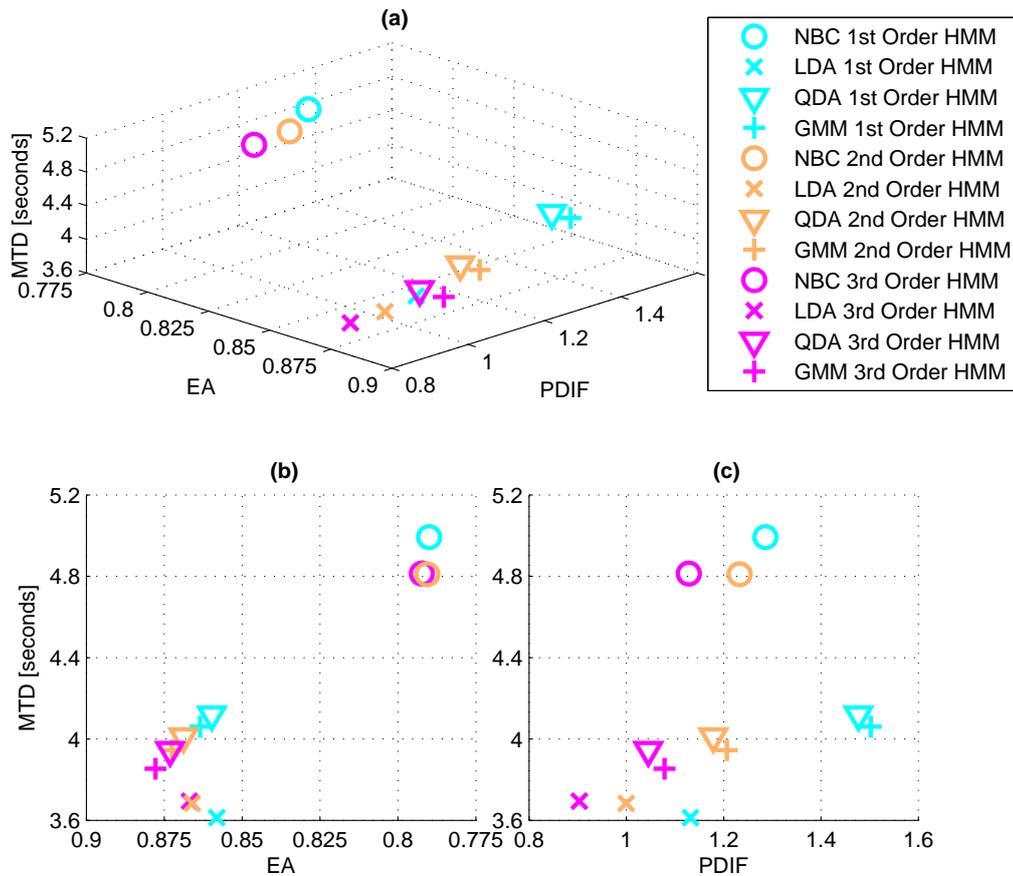


Fig. 6.11: Comparison of performance for the first, second and third order HMMs. Density estimates are derived from the augmented classifiers. As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.

performance with an almost prohibitively high memory cost. Hence, when high location prediction accuracy, high travel distance prediction accuracy and low prediction delay is important the higher order models produce better results. However, the 1st order LDA HMM produces the lowest MTD while still achieving relatively high EA and PDIF with significantly lower computational overhead than the higher order HMMs.

6.4 Real-Time Viterbi Decoding

Thus far, a number of techniques have been explored for improving localisation performance by employing a HMM framework. The original HMM formulation, along with second-order and third-order Markov models have been considered. In spite of the improvements in localisation performance possible with HMMs, there is still the significant delay inherent in requiring the entire set of past and future transition predictions before

decoding of the current state can take place. In the context of a home localisation system, this means that it is necessary to wait until the data from an entire day or week has been processed before the global optimal state sequence can be estimated using the backtracking step. This presents significant issues when the purpose of the home localisation system is to enable real-time monitoring of the activities of the elder to allow detection and response to irregular behaviour. For this reason, techniques to enable HMM-based location predictions to be produced as fast as possible must be investigated.

To date, there have been a number of attempts at real-time HMM decoding in the fields of speech and image processing. Ryyanen and Klapuri (2007) use Viterbi decoding to estimate the sequence of notes in polyphonic audio. That work observes that it is possible to periodically backtrack along the past n samples. When backtracking over a window of the past n samples is conducted, the first $n/2$ states are stored and the window is moved forward by a further $n/2$ samples. This overlapped window approach means that, while locally optimal predictions can be produced online, there is still a minimum of $n/2$ sample delay and a maximum of n sample delay for each prediction. Ardo et al. (2007), on the other hand, keeps note of multiple backtracked hypotheses until they converge to a common state. Then the local optimal path is backtracked from the most recent local optimal state. This has the issue that a number of hypotheses must be maintained over an unpredictable number of previous states. Bloit and Rodet (2008) introduces the concept of a fusion point, which exists when the initial portions of the backtracked window are identical. When this fusion point is found, the initial paths which are identical across all of the backtracked state paths are stored. This also has the issue that multiple hypotheses must be stored and updated.

This work develops real-time Viterbi decoding algorithms to allow the production of Bayesian-filtered room predictions as early as possible. To our knowledge, all previous work on HMM localisation uses the original formulation of the Viterbi decoding algorithm and is incapable of generating online location predictions. In this work, online predictions are produced using new algorithms which do not require maintaining multiple backtracked sequences and are able to produce predictions with a minimum lag of zero samples. First a fixed-windowed backtracking algorithm, which backtracks from points at regular intervals, is presented. Then the deficiencies of this algorithm are alleviated with the derivation of an algorithm which detects the optimal backtracking interval online.

6.4.1 Fixed Window Backtracking

The original formulation of the Viterbi decoding algorithm made it impossible to make a globally optimal prediction of the Bayesian-filtered location without the availability of all future and past transition predictions. Now we allow the decoding algorithm to perform the backtracking step before all samples up until termination have been processed.

Instead of performing backtracking at $t = T$, the backtracking step is performed at regular intervals of $t = sl$, where l is the non-overlapping backtracking window length and s is the backtracking iteration. Hence, for each value of s , backtracking occurs starting at $t = sl$ and ending at $t = (s - 1)l + 1$.

Since backtracking occurs online, there is no longer the final termination (stage 3) or global backtracking (stage 4) stage present in the original Viterbi algorithm. Now the recursion stage (stage 2) continues as before, and in each step if $t = sl$ the following steps take place:

$$q_t^* = \underset{1 \leq i \leq N}{\operatorname{argmax}} [\delta_{sl}(i)] \quad (6.19a)$$

$$q_t^* = \Psi_{t+1}(q_{t-1}^*), \quad t = (sl - 1), (sl - 2), \dots, ((s - 1)l + 1). \quad (6.19b)$$

There is, however, no guarantee that the state which maximised the probability of occupation in equation 6.19a is the globally optimal state. Hence, if a particular value of sl occurs at a time when the state partial probabilities have not settled to their true values, backtracking could begin at a globally sub-optimal state. As a result, the global optimality of the decoded state-sequence is sensitive to the chosen value of l .

To understand the effect of backtracking window length on localisation accuracy the original Viterbi algorithm was modified as above and the augmented localisation technique illustrated in Figure 6.7 was again implemented. Figure 6.12(a) shows the effect of the backtracking window length on EA. From the zoomed view in Figure 6.12(b) it can be seen that extremely small window lengths result in slightly reduced accuracy. When a backtracking window length of 20 is used, the accuracy is equal or greater than that of the original non-real-time Viterbi decoding algorithm. Hence, when a backtracking window of 20 samples is used this technique allows a reduction of prediction lag to a maximum of 20 samples and a minimum of 0 samples while obtaining the same localisation accuracy as the original offline Viterbi algorithm.

Even though it is possible to generate near-real-time predictions with localisation accuracy similar to the original HMM, there is still an issue of sensitivity of the algorithm to backtracking window length. The reason accuracy does not approach the original Viterbi algorithms' accuracy until approximately $l = 20$ is that, the smaller the window length is, the more likely it is that a particular value of sl will occur when the maximum partial probability has not truly settled to the most likely true state. The algorithm can be modified to allow for this defect.

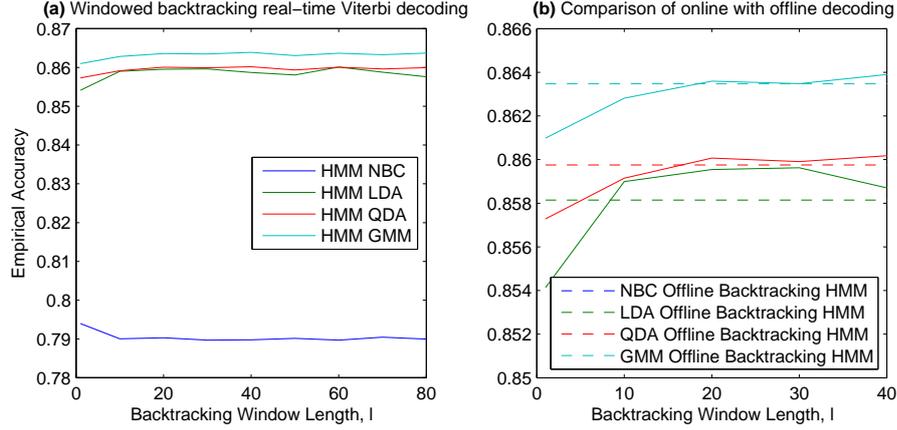


Fig. 6.12: (a) Empirical localisation accuracy as a function of backtracking window length, l , and (b) a zoomed comparison of the real-time decoding algorithm with the original Viterbi decoder.

6.4.2 Stable Partial Probability Backtracking

The main issue with the fixed window-length algorithm presented in the previous section is that it allows backtracking to start at an inappropriate time, such as when the partial probability of the incorrect room is temporarily elevated. An example of the effect this has on backtracking can be seen in Figure 6.13. Subfigure (a) illustrates the original Viterbi algorithm which performs backtracking from the final prediction of the day. Subfigure (b), on the other hand, performs fixed window backtracking with a window length of $l = 20$. At sample number 500 backtracking occurs, beginning at room 6 rather than room 5 because it does not have future information to suggest that the room of maximum partial probability at this point is not the globally most likely room. To overcome this issue it is necessary to employ a method of detecting if a given sample instant is a suitable time to commence backtracking.

By studying prediction and backtracking traces such as that in Figure 6.13(a) it is apparent that the globally optimal backtracked prediction generally occurs when the highest partial probability has remained unchanged for a number of samples. Hence, the backtracking interval can be detected online based on the variation in the value of the state of maximum partial probability over a window of the past m samples. That is, backtracking occurs at time t if, and only if,

$$\text{std}(\underset{1 \leq i \leq N}{\text{argmax}}(\delta_{[t,(t-l),(t-2)\dots(t-m+1)]}(i))) = 0. \quad (6.20)$$

Then backtracking takes place from t back to the last sample backtracking proceeded from. Hence, the maximum lag for this approach is dependent on the frequency of movement within the environment and RF measurement noise.

Figure 6.14(a) illustrates that a stable partial probability detection window length

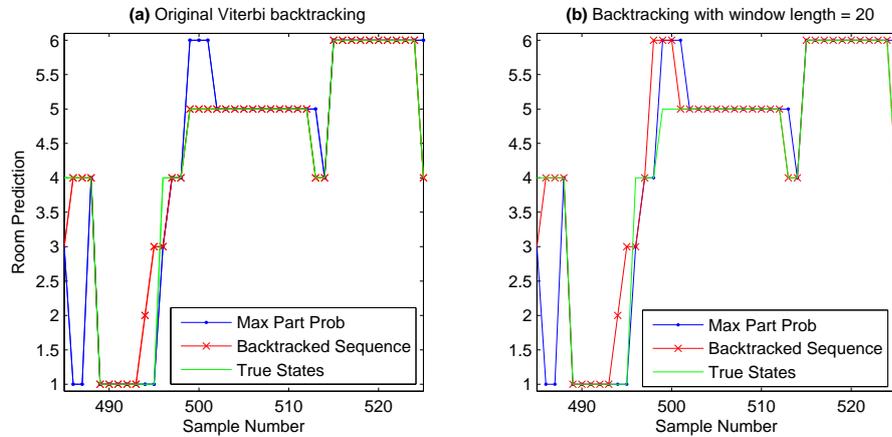


Fig. 6.13: Comparison of the original Viterbi backtracking technique and the fixed window backtracking technique. At sample 500 a backtracking step begins, initialising to room 6, since it has the highest partial probability.

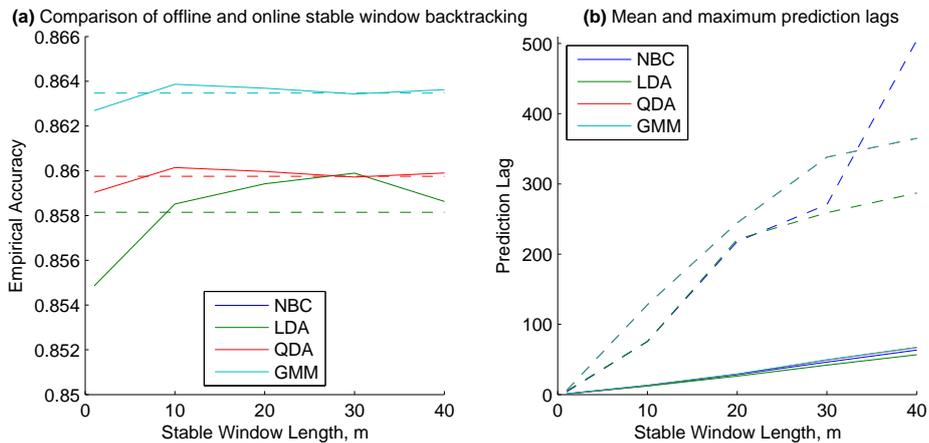


Fig. 6.14: (a) Empirical localisation accuracy and (b) mean and maximum prediction lag as a function of stable partial probability detection window length. Solid lines in Figure (b) are mean lags and dashed lines are maximum lags. The lines for QDA are not visible because they are identical to GMM in this graph.

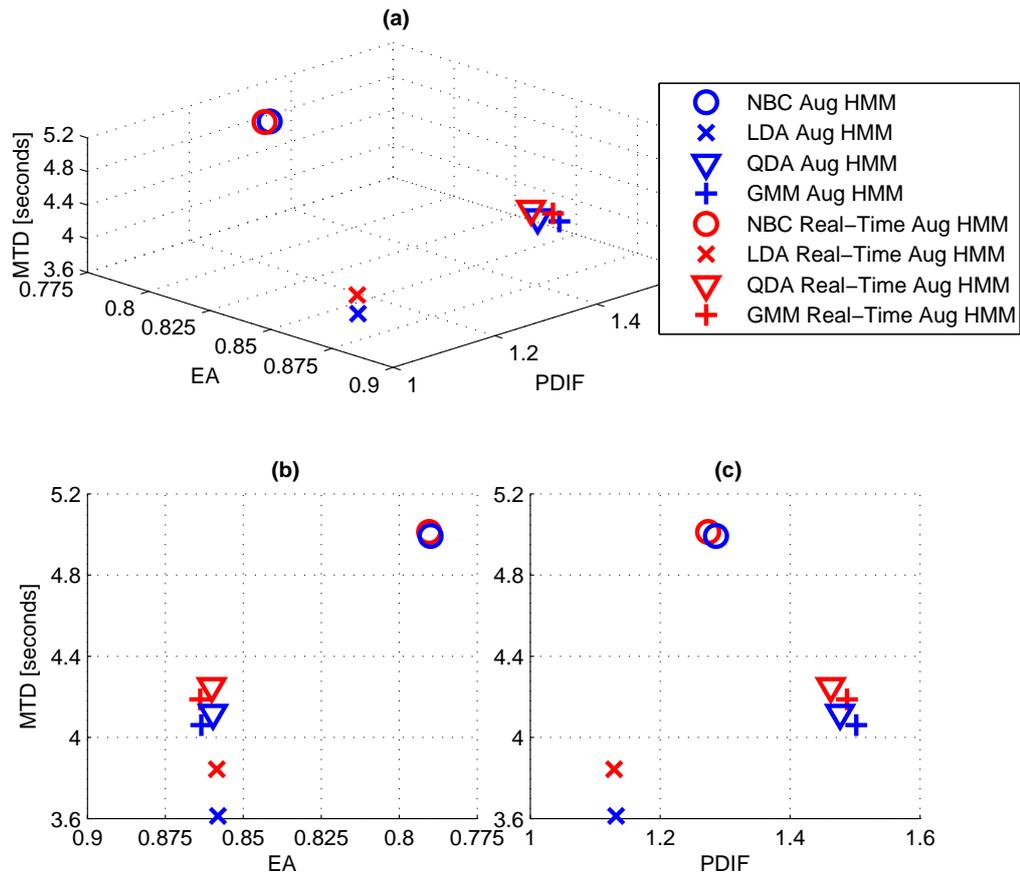


Fig. 6.15: Comparison of the performance of the original Viterbi decoding algorithm with the short-time Viterbi algorithm. As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.

of $m = 10$ leads to accuracy similar to the original Viterbi algorithm for all density estimation techniques. Figure 6.14(b) indicates that the maximum corresponding lag is 100 samples, and the mean lag is approximately 13 samples for real-time Viterbi decoding with all density estimation techniques. Hence, it can be observed that prediction lag is less predictable and mean prediction lag is actually higher than the fixed window length backtracking technique. However, it can also be observed that this less favourable prediction lag comes with the benefit of reducing instances of backtracking into invalid state sequences due to backtracking from uncertain samples. Maximum prediction delays of 100 samples (or under 3 and a half minutes, given the 0.5 Hz sampling frequency) would be a justifiable cost in all but the most time-sensitive of localisation applications. Furthermore permitting online variable prediction lags will allow more robust performance at times when long periods of frequent room transitions are occurring while permitting faster predictions when backtracking is more reliable.

To compare the overall performance of the original augmented HMM implementation with the short-term backtracking implementation, Figure 6.15 was produced. The short-term backtracking in this comparison is implemented using the stable partial probability method. The only significant difference between the original HMM implementation and the real-time implementation is the marginally higher MTD for the LDA, QDA and GMM density estimation techniques. This figure confirms that even though the real-time Viterbi algorithm allows predictions to occur with minimal lag, it still permits localisation performance on par with the original full backtracking implementation. Finally, it should be noted that, besides the low latency location predictions, real-time localisation also eliminates the need to maintain memory of the previous state for every single state at every instant of time since tracking began. With the short-time Viterbi algorithm all that needs to be stored is the most likely previous state for every state for every time instant since the last backtracking step. Hence, the memory limitations precluding higher-order HMM implementations could be alleviated. This would make the higher localisation performance associated with the 3rd order HMM possible with only high computational cost and relatively low memory requirements.

6.5 Conclusions

This chapter has explored numerous ways of improving in-home localisation performance by applying a HMM framework to the localisation problem. The simple application of discrete Bayesian filtering to the task of localisation is not a new concept. Bahl and Padmanabhan (2000a) were one of the first to propose incorporating previous location information into proceeding predictions in indoor human-tracking scenarios. That work uses what is referred to as a “Viterbi-like” algorithm on a number of k NN “best guesses” of the user’s location. That algorithm removed physically unlikely predictions in favour of smooth travel paths, hence minimising the influence of noise on predictions and increasing accuracy. Krumm et al. (2002) use a HMM to describe a user moving throughout an environment in a custom RF hardware localisation testbed. They constructed a connectivity graph of the environment and placed probabilistic constraints on the connections between each position. The transition probabilities were manually assigned between rooms. This is not entirely appropriate when one considers the assigned probability of 0.95 of remaining in a room and 0.05 of transitioning to an adjacent room in the following second. This assumes that a person moves from one room to the next once every 20 seconds during the day, which is not a valid assumption. Finally, Deasy and Scanlon (2004) propose an algorithm which borrows from HMM theory to filter predictions onto a grid, maintain a list of potential paths and eliminate paths which are infeasible considering a maximum movement speed of 2m/sec. This however is not strictly an

implementation of the Viterbi algorithm.

Based on previous work, the contributions of this work in relation to discrete Bayesian filtering are as follows;

Localisation Performance Metrics As already explained, prior localisation work generally presents accuracies in relation to an error distance cumulative distribution function (cdf). This was addressed in Chapter 5 with the development of an empirical localisation accuracy metric. This measure, however does not indicate the timing accuracy of changes between rooms or the frequency at which such changes occur. To this end further metrics were necessary to fully quantify the performance of an indoor localisation system. The complete set of metrics; Empirical Accuracy (EA), Mean Transition Delay (MTD) and Predicted Distance Increase Factor (PDIF) addressed all relevant location prediction performance measures not encapsulated by EA alone. A novel combination of original localisation performance measures and a 3-dimensional visualisation technique was presented to compare the tradeoffs between different localisation techniques and select the optimal.

HMM Implementation Describing an indoor human localisation scenario using HMMs has been presented in some previous work. The work of this thesis builds upon previous work by deriving transition matrices entirely from room labels for a typical day. This approach has not been attempted previously since it was easier to make assumptions about the movement tendencies of the user. The availability of an automatic labelling technology was a significant enabling factor for this approach. Further to utilising empirically derived transition matrices, this work also employs high-order HMMs to create higher dependencies of predictions on room connectivity, which is a novel contribution to the field of indoor localisation. Although higher localisation performance was possible, it is computationally prohibitive to progress higher than a 3rd order HMM.

Real-Time Viterbi Decoding The final investigation of this chapter was whether internal HMM state predictions could be produced in real-time or in near-real-time. It was found that it was possible to obtain maximum performance almost identical to the original Viterbi algorithm by using a short-time Viterbi decoding scheme. It was possible to produce these predictions with a mean lag of 13 samples, rather than having to wait until the end of a day to benefit from HMM predictions for the day. The maximum lag for the short-time Viterbi algorithm developed in this work was approximately 100 seconds which equates to a maximum prediction lag of just under 3.5 minutes. This is an entirely novel contribution to the field since HMM augmented localisation has always been implemented in the offline-backtracking form.

Time-Varying Hidden Markov Models

The use of HMMs to model a moving person has been shown to improve localisation performance. This occurs by incorporating information about the movement tendencies of the user into the location predictions. Higher order HMMs have demonstrated that inclusion of more information in the location predictions generally lead to improved localisation performance. Intuitively, if knowledge of the movement levels of the user at different times of the day were available, it could be incorporated into the HMM localisation framework to allow more informed movement predictions. One way of achieving this is to use different transition matrices for different times of the day. This has the downside of not being robust to changes in behaviour of the user in similar time periods of different days.

A more appropriate way of varying the transition probabilities is to use further sensors to detect the movement levels of the user at a given time. Krumm and Horvitz (2004) propose using a Viterbi algorithm to produce WiFi location predictions which incorporate the probability of a user travelling a particular distance in a particular period of time. The probability of the user moving a given distance to a particular room is coarsely estimated from the variability of the WiFi signal strength. Chapter 4 has indicated that the hardware utilised in this thesis allows the acquisition of other signals, besides RF signals. Accelerometer data is one signal which is consistently streamed back to the BSC. Accelerometers are a more reliable indicator than signal strength variability of the intensity of the movements of the MD since RF signal strength variability can be high even when the user is not moving. In fact, previous work has indicated the possibility of using accelerometers to discriminate between the different types of movement undertaken by the user (Allen et al., 2006, Ibrahim et al., 2008, Kilmartin et al., 2009, Wang et al.,

2009).

Instead of mathematically estimating the likelihood of a user transitioning a given distance (Krumm and Horvitz, 2004) we take an approach which makes better use of the available training data. This approach involves empirically estimating the transition matrix given each type of movement from the training data. Then in the location estimation phase a different transition matrix can be used for each type of detected movement. Hence, the most important element of such a time-varying HMM is the reliable detection of movements and the correct identification of the movement type. Accordingly, Section 7.1 considers the detection of the movements relevant to the localisation of a person in a typical home environment. Section 7.2 then discusses how knowledge of these movements can be integrated into the localisation framework and the effect this has on localisation system performance.

7.1 Movement Detection

To date, a number of systems have used motion estimates to predict location or improve location predictions. Widyawan et al. (2008) describes a system which utilises only inertial measurements from extremely accurate accelerometers and magnetometers mounted on the foot of the subject to predict their location indoors. Such an approach is not applicable to this work since the mobile phone under consideration does not contain magnetometers, hence, the orientation information necessary to track the movement of the user entirely from inertial measurements is unavailable. Ofstad et al. (2008), on the other hand, uses accelerometer sensors on a phone the same as the phone considered in this work to augment GPS location predictions. Estimates of whether the user is walking, sitting or standing is used to refine GPS estimates of the building the subject currently resides in. That approach worked in a large-scale urban environment where the posture and movements of the user are distinctly different in different locations, for example, if they are sitting in a restaurant instead of standing in a retail store. The system in this thesis, however, cannot rely solely on such information to refine position estimates since the user can typically undertake any arbitrary combination of postures and motions throughout the home. Furthermore, it is unreliable to assume that the phone is always similarly orientated in the pocket of the user at all times.

As already stated, Krumm and Horvitz (2004), use crude motion estimates, derived from WiFi signal strength variability, to contribute to the accuracy of a HMM-based WiFi localisation system. That approach is most applicable to the work of this thesis since it does not rely on consistently repeatable actions in each location or highly calibrated inertial sensors to predict location. The motion information is merely an aid for the RF-based location filtering. Unfortunately the work of Krumm and Horvitz (2004) does not indicate

Label	Motion Type	Label	Motion Type
1	“Downstairs-static”	2	“Downstairs-moving”
3	“Upstairs-static”	4	“Upstairs-moving”
5	“Moving-up-the-stairs”	6	“Moving-down-the-stairs”

Table 7.1: Motion labels for motion prediction test.

the localisation improvement, if any, permitted by the inclusion of motion estimates into the localisation framework. Furthermore, since our work relies on an RF communication protocol which does not provide signal intensity readings with resolution as high as WiFi, it is necessary to borrow from other work by using accelerometers to reliably estimate motion levels and even motion types.

First it is necessary to consider the localisation scenario to establish the most appropriate motion detection algorithm. The simplest piece of motion information, a binary moving/not-moving estimate, would allow more appropriate location predictions. For example, if the user is moving, there is some probability of a room transition occurring. If the user is not moving there should be zero probability of a room transition occurring. If more detailed motion information is available further constraints can be placed on the transition probabilities. For example, if certain events, such as the use of stairs, is detectable then informed decisions can be made as to which floor the user is occupying. Stairs detection alone is enough to theoretically reduce the prediction search-space by half. Such a constraint on candidate search-space can naturally be realised by the HMM framework by modifying the transition probabilities according to which floor is most likely. Hence, the HMM framework can integrate motion information by modifying the transition probability matrix according to the detected type of motion. As a result, it is necessary to consider the effect of utilising two types of motion sets available in a typical home environment; (1) simple binary “moving”/“non-moving” states and (2) more detailed motion states, such as “moving”/“non-moving”/“moving-up-stairs”/“moving-down-stairs”.

Before the integration of motion information into the HMM framework is possible, it is first necessary to quantify the ability to detect such motions. To this end it was necessary to acquire another set of data in the large-scale test home environment. This environment presented the ideal test environment since it contained stairs. It should be noted that stairs are not present in all home environments, hence any performance improvements which result from the more detailed motion information are not applicable to all deployment scenarios. To acquire this test data two 5-minute walks were conducted on two different occasions while logging the usual collection of RF and accelerometer signals from the test hardware. During the walks voice-recordings were made which noted the current motion of the user from the list of labels described in Table 7.1.

As will be presented later, these are the labels which are used for the experiments on applying HMMs to the motion predictions. These detailed labels can be easily converted to less detailed labels. For example, binary “moving”/“static” labels can be obtained by replacing all instances of label 1 and 3 with label 1 for “static” and all instances of labels 2, 4, 5 and 6 with label 2 for “moving”. Hence, with this dataset, both the simple and detailed approaches to motion detection can be explored.

7.1.1 Binary Moving/Static State Detection

The simplest form of motion detection is the binary “moving”/“static” motion detection. Krumm and Horvitz (2004) performs discrimination between such motions by analysing the standard deviation of a moving window of RSSI samples from a single signal. Maximum probability estimates are used to decide whether the detected signal standard deviation is the result of a “moving” or a “static” user based on some training data. Hence, before considering the motion recognition rates using accelerometer signals, it is necessary to investigate the motion recognition rates possible on phones which do not contain accelerometers, by utilising only RSSI variability. With the available labelled motion data an RSSI variability motion recognition technique, similar to that of Krumm and Horvitz (2004), was implemented by replacing the original labels in Table 7.1 with binary “moving”/“static” labels. As in earlier experiments, one of the available datasets were used as the training dataset and the other used as the test dataset. Then the datasets were swapped and the test repeated. Then the accuracy for each classifier is the mean accuracy for both test runs.

An algorithm similar to that in Krumm and Horvitz (2004) was implemented to determine the ability of this system to predict whether the user is moving without using accelerometers. This algorithm uses the standard deviation of the running window of samples as the classification feature for a “moving”/“static” radio signal source. A window length of 10 samples is used. Since the available communication hardware only provides RSSI signals with extremely low resolution it was decided to use further available signals, rather than one RSSI signal, as proposed in Krumm and Horvitz (2004). Depending on the number of connected APs there can be any number of signals available, however, there is always a minimum of four signals available; AP1 RSSI, AP1 LQ, CRSSI and CID. Firstly, CID would not be a reliable indicator of motion since it is a categorical signal which changes in certain locations, but not others. RSSI, LQ and CRSSI are continuous signals which should exhibit higher levels of variation when the user is moving, but also when placed in a highly changeable transmission environment, such as when doors are moving or when placed near reflective surfaces. Hence, signal variability will not be a completely reliable indicator of movement due to environmental factors.

Figure 7.1(b) illustrates the progression of the state of the user during test data

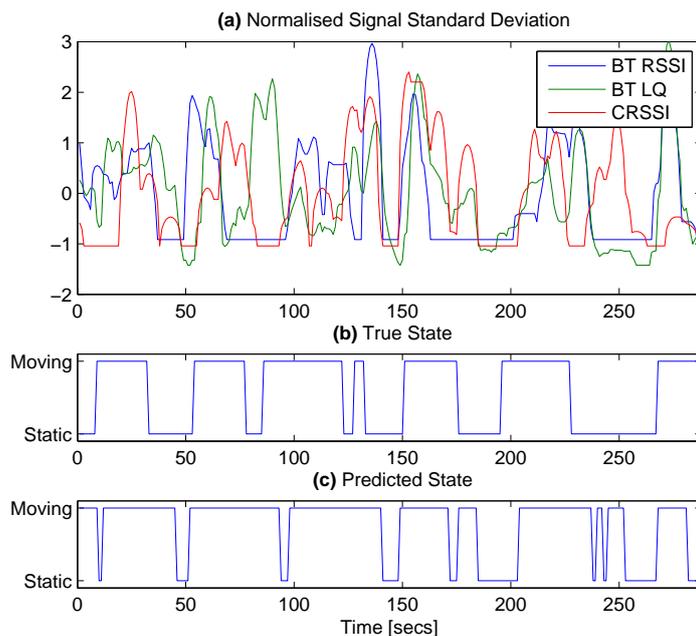


Fig. 7.1: RF signal windowed standard deviation, the actual user state progression and the corresponding LDA state prediction using the windowed standard deviation features.

acquisition and Figure 7.1(a) illustrates the corresponding signal windowed standard deviation features, normalised for illustration. It can be observed that the Bluetooth[®] RSSI signal frequently does not exhibit an elevated standard deviation feature, even though motion is occurring. This occurs because the user is far from the BSC and the signal is saturated to -10. Hence, the LQ and CRSSI signals can allow motion detection from signal variation when RSSI is saturated. It is difficult to visually infer if the combination of these features would lead to accurate motion detection. Instead it must be empirically determined if it is possible, by applying the probabilistic classifiers already considered to these test features. The reason why only probabilistic classifiers are considered will become evident when Bayesian filtering is applied to these predictions in Section 7.1.3. This algorithm improves upon previous work by utilising the windowed standard deviation feature of more than one signal to allow motion detection with hardware which has lower RSSI resolution than that available in prior work.

When considering the LDA predictions in 7.1(c) it is apparent that numerous misclassifications are possible. This is an issue with the fact that the variability of the RF signals is the result of environmental interference as well as user movement. There is also the issue that predictions, if they do occur, are delayed compared to the actual states. This is the result of producing predictions from the standard deviation of a window of the last 10 RF samples, which is equivalent to the last 20 seconds of data. To address these deficiencies of the windowed RF signal standard deviation approach it was decided

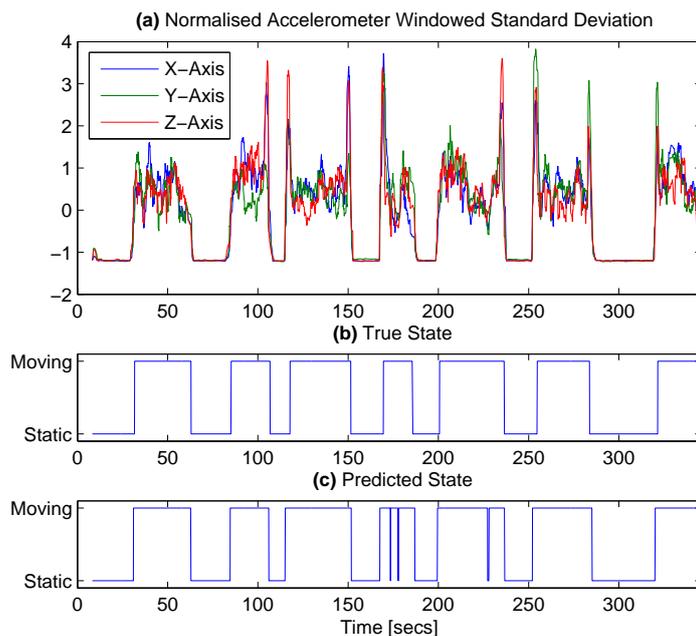


Fig. 7.2: Accelerometer windowed standard deviation, the actual user state progression and the corresponding LDA state prediction using the windowed standard deviation features.

to use the standard deviation of the accelerometer signals as the input features to the motion detection classifiers. This should be a more reliable indicator of movement since the accelerometer signals are not susceptible to environmental interference; only phone movements. Furthermore, it will reduce prediction lag since the accelerometer samples are available at the BSC at a much higher rate. Every accelerometer channel is filtered and decimated by the mobile phone before being sent back to the BSC to reduce the amount of data sent on each query. Hence, each accelerometer channel produces samples at a rate of approximately 10Hz. As a result, producing predictions from the windowed standard deviation of the accelerometer signals will reduce prediction lag.

Figures 7.2(a) and 7.2(b) illustrate the windowed standard deviation of the accelerometer channels over a window of 20 samples and the actual state evolution respectively. It can easily be observed that the accelerometer standard deviation is at a minimum when the user is static and high when the user is moving. Hence, the LDA classifications in Figure 7.2(c) illustrate that almost perfect motion detection is theoretically possible with the accelerometer standard deviation features.

Table 7.2 allows a direct comparison between motion detection using RF signals and accelerometer signals. The accuracy metric employed this time is the Mean Recognition Rate (MRR) for each state. This is analogous to the General Accuracy (GA) metric considered in Chapter 5. This accuracy metric is employed because the recognition of a particular state will switch how the HMM handles predictions, hence, the ability to detect

	NBC	LDA	QDA	GMM
RF Signal Std	0.613 (0.02)	0.607 (0.00)	0.642 (0.06)	0.640 (0.05)
Accelerometer Std	0.920 (0.03)	0.907 (0.05)	0.906 (0.04)	0.930 (0.01)

Table 7.2: MRR of movement type for different feature extraction techniques and classifiers. Values indicate mean values across both datasets and parentheses indicate standard deviations.

each state is equally important. This accuracy metric also negates the effect of a large sample size for a particular state, which leads to an accuracy measure more robust to the quantity of data available. From the table it can be seen that using the accelerometer signals allows significantly higher motion detection for all types of classifiers. The highest accuracy occurs with the GMM classifier, which now uses 3 components for each class and is trained using GEM.

Previous work by Krumm and Horvitz (2004) obtains correct predictions 84.5% of the time when using only WiFi signal strength variability, which increases to correct predictions 87.4% of the time with the use of Bayesian filtering. Experiments with our hardware, however, were only able to obtain maximum accuracy of 64.2% with signal strength standard deviation. This is for two reasons; firstly the signals do not have the resolution and range of variation necessary to allow signal variations to occur in all locations in the house and secondly the the RF signals can vary as a function of environmental factors as well as user motion. This work also uses a more challenging accuracy metric, the Mean Recognition Rate (MRR). Krumm and Horvitz (2004) use a classification success rate which reduces the impact of rarely occurring classes on the estimated accuracy. Hence, classifiers which specialise on predicting the class which has more data will exhibit higher accuracy while actually being less able to predict infrequently occurring classes. This work finds that when using the more relevant, but challenging, MRR, higher recognition rates are possible using accelerometer data.

7.1.2 Detailed Movement Information

Simple binary motion predictions will allow HMM location predictions which incorporate knowledge of the user’s mobility. Intuitively, the availability of more detailed movement information should allow more informed location predictions. Hence, it should also be investigated if further useful movement information can be derived from the available signals. Previous work investigated the possibility of using accelerometer signals to detect whether a user is walking on flat ground, moving up or down stairs or moving up or down a slope (Ibrahim et al., 2008, Kilmartin et al., 2009). Such techniques are relevant to this work since a subject sometimes needs to traverse stairs to reach certain locations. The ability to detect such events may lead to more accurate location predictions. Hence, the

	NBC	LDA	QDA	GMM
RF Signal Std	0.344 (0.06)	0.327 (0.04)	0.313 (0.02)	0.298 (0.01)
Accelerometer Std	0.639 (0.02)	0.607 (0.02)	0.482 (0.07)	0.494 (0.02)

Table 7.3: MRR of movement type from the four movements; “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs”, using RF and accelerometer variability.

approach to detecting stairs from accelerometer signals should be empirically evaluated for efficacy in this realistic home deployment.

Before the approach adopted by Ibrahim et al. (2008) is implemented, it should be investigated if the standard deviation feature classifiers presented in the previous section will allow the detection of stair-crossing events. To generate data with the relevant labels it was necessary to convert the original high-detail motion data to medium-detailed data by replacing certain labels. The required labels are

1. “Static”
2. “Moving”
3. “Moving-up-the-stairs”
4. “Moving-down-the-stairs”.

Hence, to acquire such labelled data it was necessary to replace labels 3 and 4 with labels 1 and 2 respectively and replace labels 5 and 6 with labels 3 and 4 respectively in the original datasets.

Table 7.3 highlights the MRR for the four classes outlined. It can be observed that the recognition rates are significantly lower than those in Table 7.2 for all classifiers and feature extraction techniques. As before, using features extracted from the RF signal windowed standard deviation results in significantly lower recognition rates than when using the accelerometer windowed standard deviation features. Furthermore, NBC exhibits the highest recognition rates and the more sophisticated density estimation techniques such as QDA and GMM result in lower motion recognition rates. This indicates that the windowed variability of the RF signals and the different accelerometer channels should be modelled independently.

Even though the ability to discriminate individual motion types using only accelerometer variability has been demonstrated, the recognition rates would need to be extremely high to enable an improvement to the already relatively accurate RF location predictions. Table 7.4 shows the confusion matrices for the NBC algorithm based on both RF signal and accelerometer standard deviation. It can be observed from Table 7.4(a) that using RF features leads to a high bias towards the “moving-up-the-stairs” class, resulting in the

poor recognition rates highlighted above. Table 7.4(b), on the other hand, confirms that using accelerometer features allows excellent ability to discriminate between a “static” state and all other states. There is, however, a great deal of confusion between “moving”, “moving-up-the-stairs” and “moving-down-the-stairs”. This is because, even though there is more variation in the accelerometer’s z-axis when stairs are traversed, there is still not sufficient variation to allow unique identification of such events. Hence, the variation of the accelerometer signals are approximately equal for all states which involve movement.

(a) RF Signal Standard Deviation					(b) Accelerometer Standard Deviation				
	Static	Moving	Moving-Up	Moving-Down		Static	Moving	Moving-Up	Moving-Down
Static	0.38	0.13	0.42	0.07	Static	0.89	0.10	0.00	0.01
Moving	0.16	0.17	0.60	0.07	Moving	0.01	0.49	0.22	0.28
Moving-Up	0.00	0.00	1.00	0.00	Moving-Up	0.00	0.32	0.66	0.02
Moving-Down	0.12	0.06	0.82	0.00	Moving-Down	0.00	0.44	0.09	0.46

Table 7.4: Confusion matrices for the four movements, “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs”.

To address this issue it was decided to adopt an alternative approach which describes the accelerometer signal resulting from a subject’s gait at every instant of time by its frequency components, rather than just the variation levels. Ibrahim et al. (2008) use the Fast Fourier Transform (FFT) to estimate the frequency components of the accelerometer signal over a window of the previous 8 seconds. Then a Discrete Cosine Transform (DCT) is applied to the logarithm of the resultant frequency components to represent the result of the FFT more efficiently, i.e. with less features. The outputs of the DCT are taken as the input features for a GMM classifier. In this work the DCT step is not necessary since there are fewer FFT components due to the lower accelerometer sampling rate. Ibrahim et al. (2008) perform classifications on data compiled from focused walking tests. As a result they can use a long window length to maximise the recognition rates. This work, however, cannot use such long FFT windows since it is focused on recognising motion type in the presence of persistently changing motion. When motion type can frequently change, lower recognition rates are the result of higher lag due to long window lengths. Hence, for realistic motion type recognition, only short FFT windows are used and the DCT stage is not necessary.

Figure 7.3(a) illustrates the frequency spectrum features extracted from the training dataset for an FFT window length of 20 samples for all accelerometer channels. Since the FFT spectrum is symmetrical we only need 10 samples to represent the features for each channel. The darker the shade of red in the figure, the larger the logarithm of the FFT component. Figure 7.3(b) illustrates the progression of the state of the user during

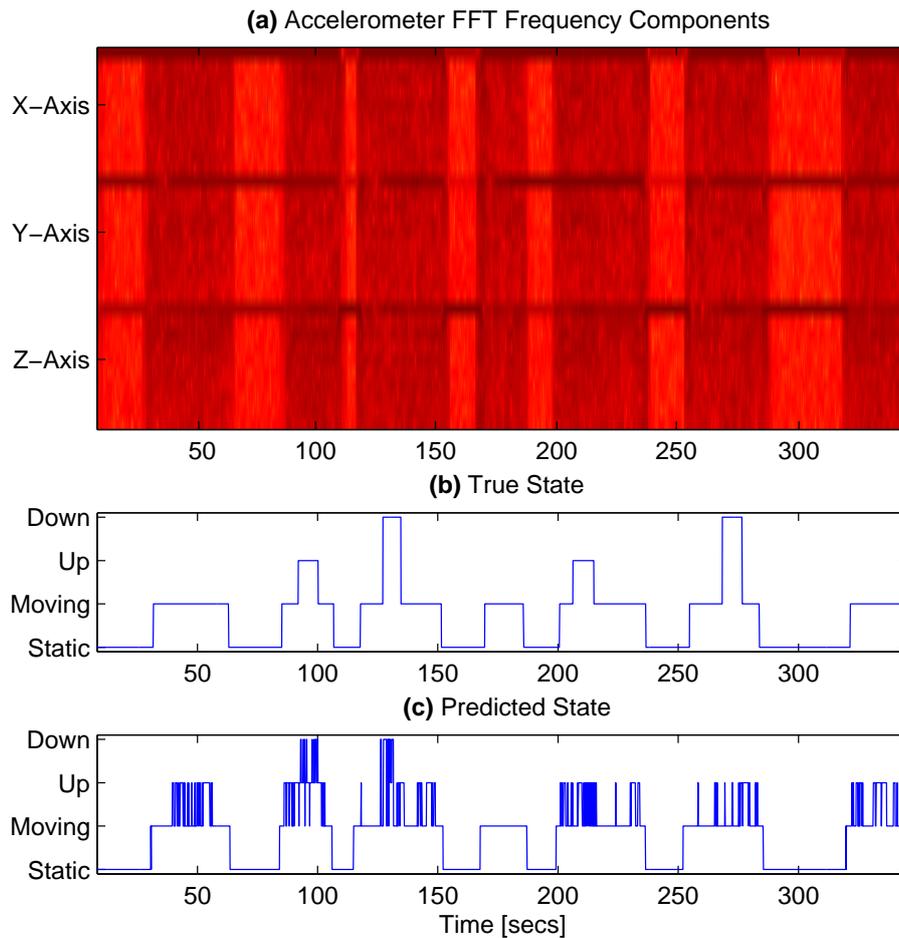


Fig. 7.3: Accelerometer frequency components, the actual user state progression and the corresponding state prediction using the accelerometer frequency components. This example uses a FFT window length of 20 samples.

the acquisition of this data. It is difficult to visually infer if the combination of the 30 features allow unique representation of a given class. Figure 7.3(c), however, illustrates the predictions possible with a NBC classifier. It can be observed that the “moving” and “static” classes are easily detected and the going up and down the stairs classes cause misclassifications.

Since the recognition rates are dependent on FFT window length, as a result of frequency detection ability and prediction lag, it is necessary to consider the influence of FFT window length on recognition rates for each classifier. The number of samples in the FFT window was varied and the MRR for the probabilistic classifiers was noted. Figure 7.4 illustrates the variation of MRR as a function of FFT window length. The optimal FFT window length is 25 samples, followed closely by a window length of 15 samples. A window length of 25 samples corresponds to just over 2 seconds of accelerometer data.

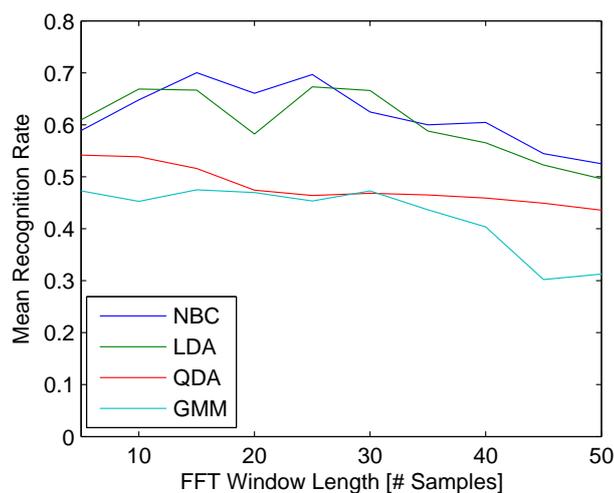


Fig. 7.4: MRR for the four movement types; “static”, “moving”, “moving-up-the-stairs” and “moving-down-the-stairs” as a function of the FFT window length.

	Static	Moving	Moving-Up	Moving-Down
Static	0.87	0.13	0.00	0.00
Moving	0.01	0.81	0.14	0.03
Moving-Up	0.00	0.25	0.72	0.03
Moving-Down	0.00	0.51	0.11	0.38

Table 7.5: Confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 25 samples.

Table 7.5 shows the confusion matrix for this optimal algorithm. When comparing this with the accelerometer variance algorithm results in Table 7.4(b) it can be seen that there are higher recognition rates for “moving” and “moving-up-the-stairs” classes. There is, however, more confusion for the “moving-down-the-stairs” class, which is frequently misclassified as either “moving” or “moving-down-the-stairs”.

The low ability to discriminate between “moving-down-the-stairs” and other moving classes, suggests that it might be possible to improve recognition rates by grouping the “moving-up-the-stairs” and “moving-down-the-stairs” classes into one class, namely “using-the-stairs”. This will reduce the effect of confusion between “moving-down-the-stairs” and “moving-up-the-stairs”. Then crossing the stairs could be used to predict which floor the user is on, which can be used by the HMM. It is, however, unreliable to assume which floor the subject resides on, because an erroneous detection of the “using-the-stairs” class will change which floor the person is on. Instead information about whether someone is crossing the stairs can be used to indicate that the subject is likely to transition to a room close to the stairs, which will reduce location prediction errors

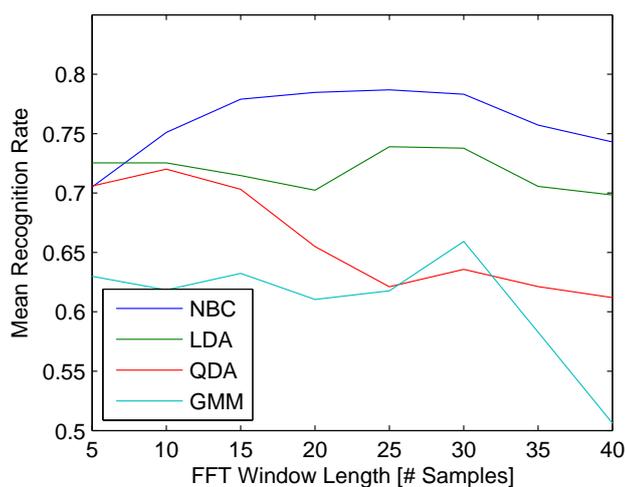


Fig. 7.5: MRR for the three movement types; “static”, “moving” and “using-the-stairs” as a function of the FFT window length.

	Static	Moving	Using-Stairs
Static	0.87	0.13	0.00
Moving	0.01	0.81	0.18
Using-Stairs	0.00	0.32	0.68

Table 7.6: The three-state confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 25 samples.

without introducing an excessively high sensitivity to erroneous predictions of movement type. The available datasets were re-labelled to produce a dataset with the labels:

1. “Static”
2. “Moving”
3. “Using-the-stairs”

Figure 7.5 shows that, for the available datasets, the peak recognition rates occur again for the NBC with a 25 sample FFT window length. The confusion matrix for this dataset can be found in Table 7.6. The MRR for this motion recognition technique is now 79% which is an improvement over the 70% for the four-movement discriminator. Since reducing the number of classes increases the overall recognition rate it is necessary to investigate if it is possible to obtain higher ability to detect stairs-use by reducing the number of alternative states. To this end the states of the available datasets was reduced to the states:

1. “Using-the-stairs”

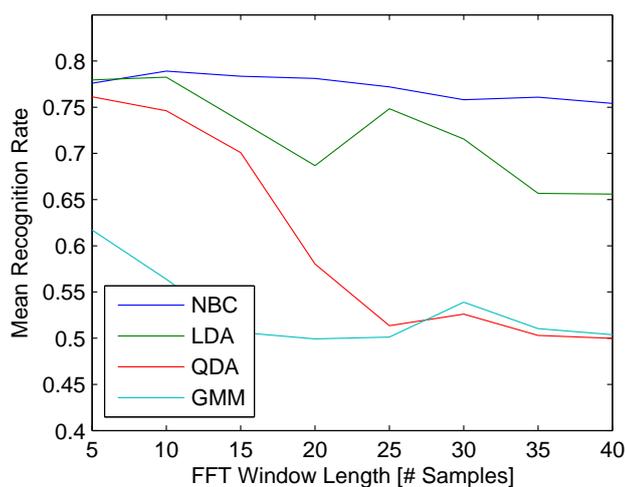


Fig. 7.6: MRR for the two movement types; “using-stairs” and “all-other-motions” as a function of the FFT window length.

	Using-Stairs	All-other-motions
Using-Stairs	0.99	0.01
All-other-motions	0.41	0.59

Table 7.7: The two-state confusion matrix for the optimal frequency component classification method; NBC with an FFT window length of 10 samples.

2. “All-other-motions”

When considering Figure 7.6 and Table 7.7 it can be seen that the optimal MRR is 79%. This technique almost never fails to detect when the stairs are being used. This, however occurs with the disadvantage that it frequently misclassifies the other motions as stairs crossings. This is a result of the fact that stair crossing events occur with some of the same frequency components as when walking is occurring. The result is that this technique has the same MRR as the three-motion-state classifier, even though it has fewer states and should actually have higher accuracy. Hence, the three-motion-state classifier has better performance when one considers that it has the task of discriminating between three distinct classes rather than 2 somewhat similar classes.

It has been illustrated that the use of accelerometer standard deviation levels are an extremely effective indicator of whether a user is moving when compared to RF signal standard deviation. When more detailed motion information is necessary, however, the use of accelerometer standard deviation information is not sufficient. Previous work on gait recognition from accelerometer signals has been applied to the task of recognising the type

of motion the user is undertaking, with some success. Unlike previous work, this work is focused on recognising motion type in the realistic situation in which the movement type can change at any time, rather than with a dataset in which the user performs a specified action for a prolonged period of time. Furthermore, this work suggests that an independent feature modelling technique such as a Naive Bayes Classifier performs better than a conditionally dependent feature probabilistic modelling technique employed in the past, such as GMM. Further work, such as Kilmartin et al. (2009), uses artificial neural networks and SVMs on similar datasets for motion recognition, which implicitly ignore dependencies between features. We, however, do not consider such approaches since the probabilistic classifiers lend themselves to HMM implementation.

7.1.3 HMM Motion Filtering

HMMs allow the production of sequences of predictions which are globally more likely than direct sample-by-sample predictions. As a result, the predicted sequence of internal HMM states are usually more accurate than a sequence of direct probabilistic classifications. Similar to location predictions in previous sections, HMMs can also be used to apply Bayesian filtering to user motion predictions, leading to more sensible state changes and more accurate state predictions as a result. The gait recognition work considered thus far has been unable to exploit Bayesian filtering, since the datasets are not sequentially arranged; they merely produce motion estimates from the mean accuracies of individual detection tests on partitioned data. To our knowledge, the best example of applying HMMs to activity recognition can be found in Shi et al. (2009). That work uses FFT feature extraction for both accelerometer and gyroscope sensors and indicates that HMMs produce higher accuracies than the direct classifiers; cascaded neural networks and SVMs. That work, however, also uses partitioned motion datasets in which motions do not change in a realistic sequence. No information about how HMMs are applied to the data or how they produce such high accuracies on partitioned datasets is presented.

This work attempts to improve the motion recognition rates by applying Bayesian filtering, in the form of the already presented HMM, to the motion datasets already considered. As before, the Viterbi algorithm is used to decode the sequence of hidden motion states which resulted in the sequence of sensor observations. Now the HMM model $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$ is derived from the available motion data. The transition probability matrix, \mathbf{A} , is derived from the probability of transitioning from one motion type to another. For example, it should be impossible to change from the state “static” to the state “moving-up-the-stairs” without first transitioning to the state “moving”. The observation probability model, \mathbf{B} , describes the relationship between the frequency component or standard deviation features and the corresponding probability of occupation of a given class, which is available from the probabilistic classifiers considered already. \mathbf{A} and \mathbf{B} are

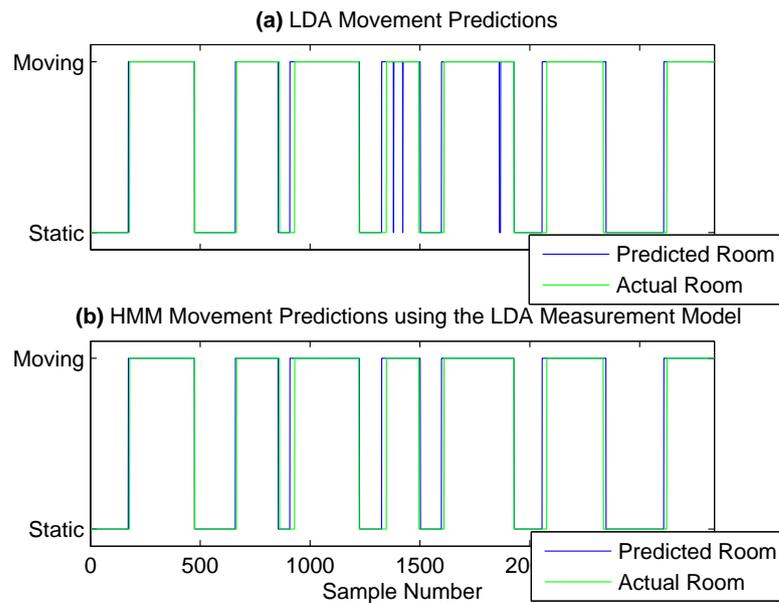


Fig. 7.7: Progression of the predicted state-sequence for (a) the original LDA classifier and (b) the Viterbi decoding of the state-sequence which uses the LDA density estimation technique.

both derived from the training dataset. The prior probability, $\boldsymbol{\pi}$, is simply assumed to be equal across all states.

To illustrate the effect of Bayesian filtering on movement predictions, Figure 7.7 compares the un-filtered predictions from Figure 7.2 with the predictions resulting from Viterbi decoding of the state sequence. It can be observed that the noisy transitions present in Figure 7.7(a) do not occur in the Bayesian-filtered version in Figure 7.7(b). Hence, there will be less erroneous predictions in the Bayes' filtered predictions. At this point it should be noted that it is equally valid to estimate the binary “static”/“moving” state from the accelerometer frequency components as it is from the accelerometer standard deviation. Figure 7.8 illustrates that similar, if not slightly higher, recognition rates are possible when applying the HMM classifier to the frequency component features rather than the standard deviation features. It should also be noted that GMM is the optimal density estimation technique when standard deviation features are employed, but are extremely unreliable when frequency component features are used. This suggests that the standard deviation of the three accelerometer axes are covariate features which are best modelled by a specialising multi-modal density estimation technique. The frequency component features, on the other hand, are better modelled by more generalising density estimation techniques.

Now, by applying HMMs to the binary “static”/“moving” classification task using accelerometer standard deviation features, the peak recognition rate increases from 93% in Table 7.2 to 94.3% using GMMs. By using the frequency components of the accelerometer

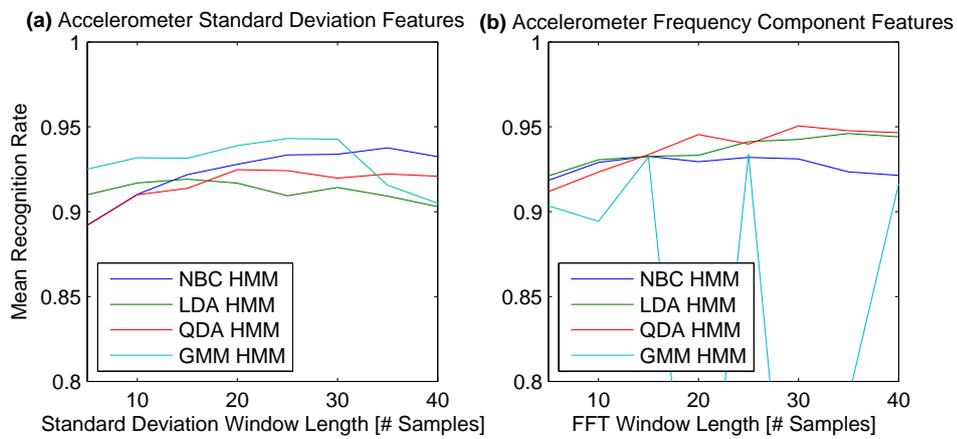


Fig. 7.8: Comparison of the MRR when (a) accelerometer standard deviation and (b) frequency domain features are used, as a function of standard deviation and FFT window lengths respectively.

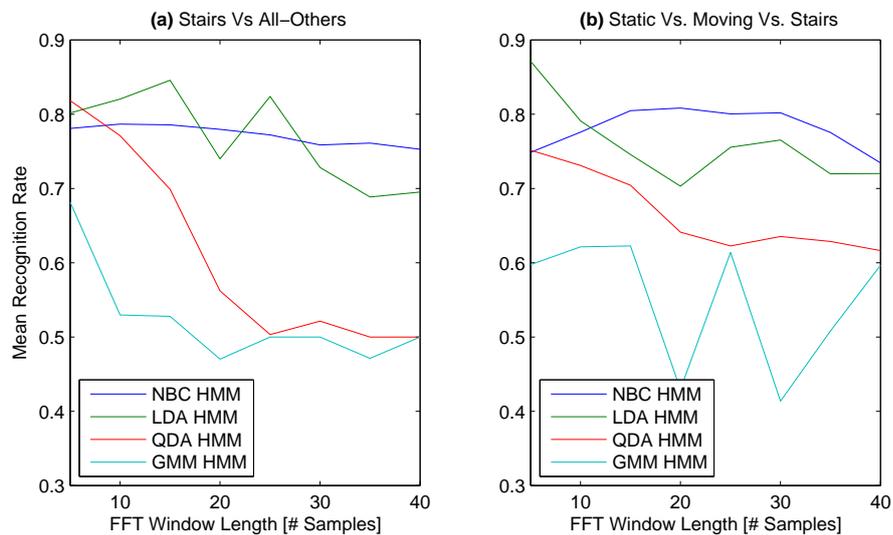


Fig. 7.9: MRR for (a) “stairs” versus “all-other” motion classifications and (b) “static” versus “moving” versus “stairs” motion classifications, as a function of FFT window length.

(a) Stairs Vs. All-Others			(b) Static Vs. Moving Vs. Stairs			
	Using-Stairs	All-other-motions		Static	Moving	Using-Stairs
Using-Stairs	0.88	0.12	Static	0.88	0.12	0.00
All-other-motions	0.19	0.81	Moving	0.04	0.84	0.12
			Using-Stairs	0.00	0.10	0.90

Table 7.8: Confusion matrices for the optimal classifiers illustrated in Figure 7.9.

signals the peak recognition rate rises to 95% for HMMs using the QDA density estimation technique. Hence, in the case of binary movement type, HMMs allow an improvement in motion recognition rate by constraining the state sequence. Next the effect of HMM-constrained classifications on datasets with more varied and numerous motion types shall be considered. Previously, the “using-the-stairs” versus “all-other-motions” classifier was capable of detecting the use of stairs extremely reliably (see Table 7.7). This was, however, achieved with a reduced ability to detect the “all-other-motions” class. By using HMMs it is possible to reduce the effect of over-specialising on the “using-the-stairs” class by imposing transition probabilities which recognise the relatively small time spent in that state and as a result the small probability of remaining in that state at any time. Figure 7.9(a) illustrates the peak recognition rate possible with the classification of these motions by using HMMs. It can be seen that the peak recognition rate is now 85% as opposed to the 79% resulting from the un-filtered classifier in Figure 7.6 and Table 7.7.

85% recognition rate is still not particularly high for a two-class problem such as this. As before, the limited discrimination ability between the two classes can be attributed to the similarity between the frequency components of the “using-the-stairs” class and some of the motions which make up the “all-other-motions” class. This “using-the-stairs” versus “all-other-motions” approach was previously adopted due to the inability to detect the use of stairs in the three-state motion detection technique, which attempted to discriminate between “moving”, “static” and “using-stairs”. By using the HMM framework we can reduce the confusion between “using-stairs” and “moving” classes evident in Table 7.6 by imposing relevant transition probabilities. Figure 7.9(b) and Table 7.8(b) show the peak three-state recognition rates and corresponding confusion matrix respectively. It is evident that a more evenly distributed confusion matrix occurs, with less confusion between “using-stairs” and “moving” than in Table 7.6. Hence, by using the HMM framework it is possible to detect the use of stairs along with a moving and a stationary subject without needing to use a dedicated stairs detection technique. Hence, it is now possible to achieve a MRR of 87% when classifying these three states, which is even higher than the recognition rate for the two states, “using-the-stairs” and “all-other-motions”.

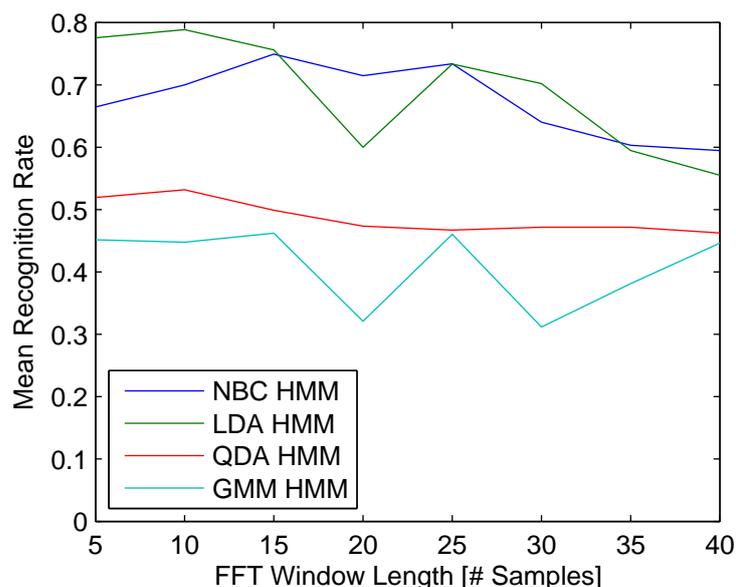


Fig. 7.10: MRR for the four-state HMM predictions.

	Static	Moving	Moving-Up	Moving-Down
Static	0.88	0.11	0.01	0.00
Moving	0.03	0.80	0.12	0.05
Moving-Up	0.00	0.24	0.76	0.00
Moving-Down	0.00	0.28	0.01	0.71

Table 7.9: Confusion matrix for the recognition of the four motion states, “static”, “moving”, “up-the-stairs” and “down-the-stairs”, when the HMM framework is adopted.

The ability to detect the use of stairs allows the reduction of the room prediction possibilities by emphasising rooms adjacent to the stairs with the transition probability matrix. The ability to detect the direction in which the stairs are being traversed allows further modification to the transition probability matrix, according to which floor the subject is occupying. Hence, it is again necessary to consider the accuracies of the four-state, “static”, versus “moving” versus “up-the-stairs” versus “down-the-stairs”, classifier proposed in the previous section. By using the HMM approach it is possible to achieve an increase in the peak recognition rate to 79% in Figure 7.10 over the un-filtered recognition rate of 70% in Figure 7.4. The confusion matrix in Table 7.9 demonstrates a better ability than that demonstrated in Table 7.5 to detect whether the movement is up the stairs rather than down the stairs, by using HMMs.

A significant issue with using this technique of detecting which floor the user resides upon is that it is extremely sensitive to prediction errors. For example, if the most recent detection of a stairs crossing erroneously indicated that the subject was walking down the

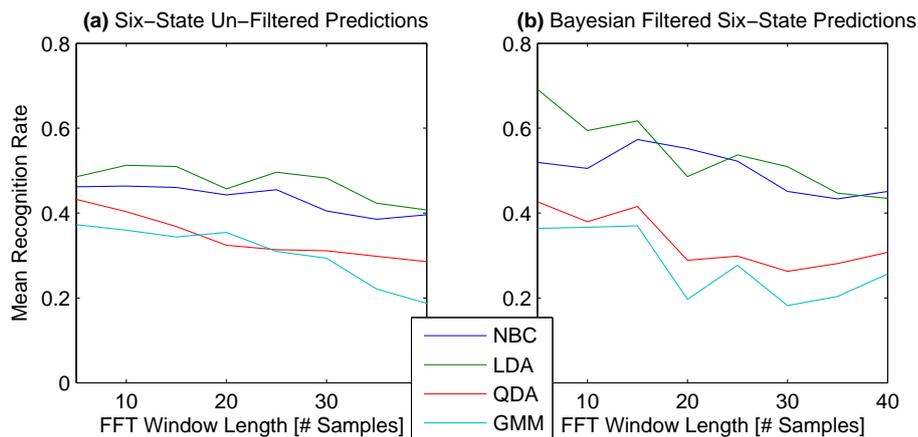


Fig. 7.11: The influence of HMM Bayesian-filtering on the recognition rates for the six state motion classifications.

stairs instead of up the stairs, it would suggest that the user was downstairs instead of upstairs. This would mean that when using an “improved” HMM which utilises this floor information, all location predictions since the last stairs prediction will be undoubtedly incorrect. This inability to reliably infer the correct floor can be addressed by creating new states. These states contain not just motion information, which is directly derivable from accelerometer frequency components (such as “moving”, “static” or “using-the-stairs”) but also information which cannot be derived from accelerometer signals alone, such as which floor the user is inhabiting. As already noted, the original motion states in Table 7.1 already contain this information.

However, with only accelerometer signals it would be impossible to discriminate between certain states. For example, it is impossible to determine if a subject is in the state “downstairs-moving” instead of “upstairs-moving” without further information. Fortunately, by incorporating memory of the previous predictions it is possible to reduce the current possible states based on the previous states. For example, it is impossible for the subject to be “upstairs-moving” without first being either in the state “upstairs-static” or “moving-up-the-stairs”. Similarly the subject cannot be in the state “downstairs-moving” without first inhabiting either the state “downstairs-static” or the state “moving-down-the-stairs”. Conveniently, this relationship between the different states can be encoded in the transition probability matrix, \mathbf{A} , of a HMM. The probability of getting to any of the motion states from any other motion state can be empirically derived from the training data. Hence, the use of HMMs allows the prediction of motion type and general location states which would be otherwise extremely unreliable on a sample-by-sample basis. This is achieved by introducing classes which explicitly contain this general location information and realising their relationship with the transition matrix.

Figure 7.11 highlights the peak accuracies possible for the detection of these six motion

	Static-Downstairs	Moving-Downstairs	Static-Upstairs	Moving-Upstairs	Moving-Up-the-Stairs	Moving-Down-the-Stairs
Static-Downstairs	0.69	0.03	0.20	0.04	0.03	0.01
Moving-Downstairs	0.01	0.37	0.01	0.29	0.24	0.09
Static-Upstairs	0.26	0.07	0.60	0.05	0.02	0.01
Moving-Upstairs	0.02	0.41	0.04	0.28	0.17	0.08
Moving-Up-the-Stairs	0.00	0.30	0.00	0.10	0.54	0.06
Moving-Down-the-Stairs	0.00	0.22	0.00	0.09	0.12	0.57

Table 7.10: Confusion matrix for the application of un-constrained classifiers to the six-state motion classification problem.

	Static-Downstairs	Moving-Downstairs	Static-Upstairs	Moving-Upstairs	Moving-Up-the-Stairs	Moving-Down-the-Stairs
Static-Downstairs	0.73	0.08	0.15	0.03	0.01	0.00
Moving-Downstairs	0.02	0.63	0.01	0.24	0.08	0.03
Static-Upstairs	0.20	0.04	0.63	0.12	0.00	0.00
Moving-Upstairs	0.03	0.09	0.04	0.71	0.07	0.06
Moving-Up-the-Stairs	0.00	0.05	0.00	0.05	0.57	0.33
Moving-Down-the-Stairs	0.00	0.06	0.00	0.00	0.07	0.88

Table 7.11: Confusion matrix for the application of HMM classifiers to the six-state motion classification problem.

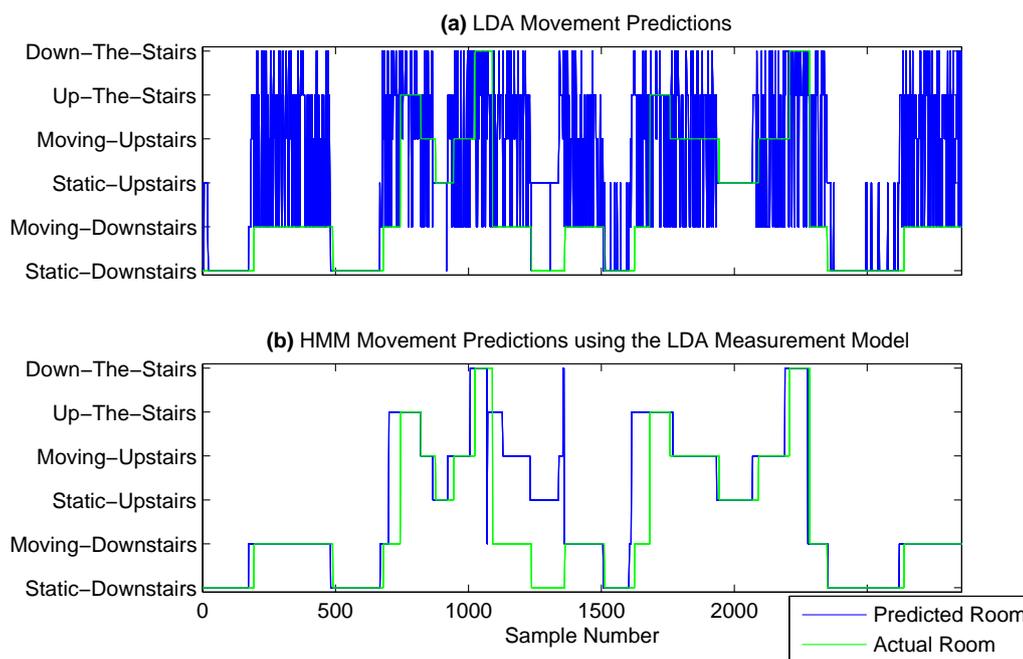


Fig. 7.12: Progression of the predicted and actual motion state for (a) the un-constrained predictions and (b) the HMM predictions.

states when using both (a) classifiers applied directly to the data and (b) HMM-based classifiers applied to the data. It can be seen that 69% peak recognition rate is possible with the use of HMMs, as opposed to only 51% with the direct application of the classifiers to the accelerometer frequency component features. This is a 35% improvement in motion recognition rate. To understand why such low recognition rates are the result of the un-constrained classifiers, it is necessary to consider Table 7.10. This confusion matrix shows how there is a great deal of confusion between the states “static-downstairs” and “static-upstairs” and the states “moving-downstairs” and “moving-upstairs”. This is because accelerometers are not indicative of which floor the user currently resides upon. When HMMs are used to constrain the state transitions the result can be seen in the confusion matrix in Table 7.11. There is now less confusion between all states.

The progression of state predictions compared to the actual state progression for both the un-constrained and HMM constrained classifications is illustrated in Figure 7.12. The fluctuations in the state predictions in Figure 7.12(a) is the result of not having enough distinct signal information in each class to reliably predict a single class. Conversely, Figure 7.12(b) uses temporal, as well as signal information to produce visually more sensible predictions. Hence, by adopting this approach, the floor recognition rates increase to 68% by using HMMs instead of 49% without using HMMs.¹ The main limitation to the accuracy of the HMM technique is the ability to discriminate between “moving-up-

¹The floor recognition rates can be derived from the mean of the individual recognition rates for the states; “static-downstairs”, “moving-downstairs”, “static-upstairs” and “moving-upstairs”.

the-stairs”, “moving-down-the-stairs” and all other classes. Erroneously high observation probability for either class results in the Viterbi decoded state sequence progressing down an incorrect path. It could be argued that using higher-order HMMs would reduce instances of instantaneous transitions from stairs to a “moving” state and back to the stairs again, such as that evident at sample 1050 in Figure 7.12(b). However, the higher-order HMMs presented in Section 6.2 have been evaluated on this data, but with no improvement evident over first-order HMMs. By using the third order HMM, the instantaneous transition at sample 1050 was merely delayed rather than avoided. In fact, to avoid such an error indefinitely, the employed HMMs would need to have an extremely high order to keep a memory of the likelihood of making such a sequence of transitions, which is computationally intractable, especially since accelerometers have such a high sampling rate (10 discrete observations per second).

7.2 Online-Varying Transition Probabilities

Now that predictions with respect to the motions of the user have been made available, the HMM framework can be modified to include this information. Krumm and Horvitz (2004) used binary motion classifications to inform the transition matrix in the HMM. The transition matrix was constructed from the set of locations the user is likely to be able to reach in a given period of time based on the speed of the user. That approach had the downside that it based the transition matrix entirely on assumptions about the speed of travel of the user. It was assumed that, when the user is in the “moving” state, he or she would move at a probabilistically derived speed, with a maximum speed of 10.22 meters/second. It did not account for the fact that the user could move within their current room. In this work, on the other hand, we avoid the necessity of such assumptions. Instead of assuming what speed the user moves at given a particular state, we take the approach of empirically deriving the room transition matrix for each motion state.

The empirical transition matrix derivation is conducted using the training data room labels and the states predicted from the accelerometer data obtained during the training phase. The movement states for the training and test data are not available *a priori*; instead they must be estimated, using all of the accelerometer data acquired in the previous section as HMM classifier training data. For each detectable motion state in the training data, a different room transition probability matrix is derived from the room transitions which occurred during all instances of that motion state. Once a room transition probability matrix has been derived for each motion state, a complete intra-house movement model can be constructed as illustrated in Figure 7.13. This model has two phases; firstly, the motion state predictions must be produced using the techniques described in the previous section. Secondly, for each RF signal sample the

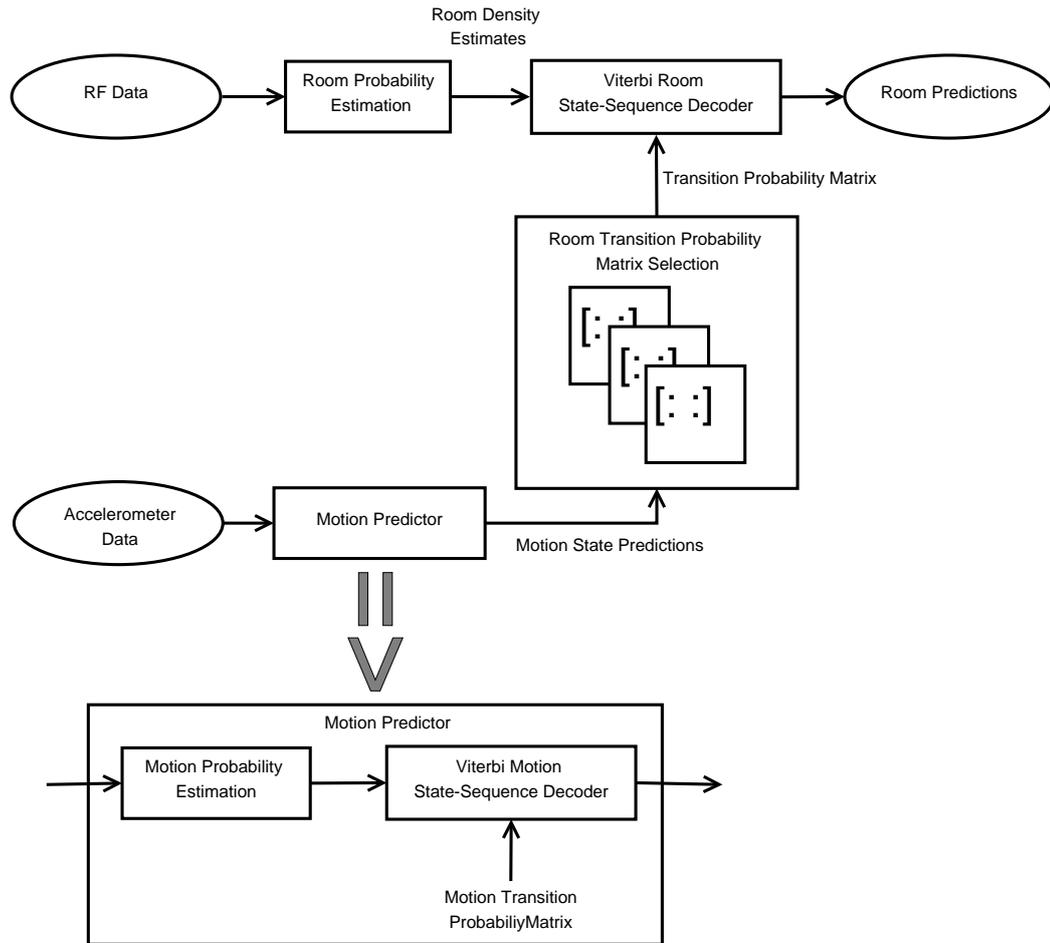


Fig. 7.13: Intra-house movement model which takes into account the RF signals, the motion state of the subject and the room transition tendencies of the subject during a given motion state.

room prediction is executed by the Viterbi Room State-Sequence Decoder which uses a transition probability matrix which is selected based on the type of detected motion at that instant of time.

It is important to note that the accelerometer data and the RF data are produced at different rates. RF data is produced at a rate of 0.5 Hz and the accelerometer data is produced at a rate of approximately 10 Hz. Since the motion predictions are also modelled with an HMM, it is inappropriate to produce motion predictions only when needed by the Viterbi Room State-Sequence Decoder. Instead it is necessary to produce motion predictions for every available accelerometer sample. Then, approximately 1 out of every 20 motion predictions are used for the selection of room transition matrix, which is used in the production of each new room prediction. Another important issue with this technique is the effect of motion prediction accuracy on the room prediction accuracy. If an incorrect motion prediction occurs, it will almost certainly result in incorrect room prediction. For example, if a “static” state is erroneously predicted it will prevent the room prediction

from changing, causing incorrect room predictions. Hence, HMM localisation techniques which attempt to increase localisation accuracy by discriminating between more motions could actually cause more incorrect room predictions due to their lower mean motion recognition rates. To determine the benefit motion detection has on room predictions, three of the motion prediction techniques presented in the previous section are considered; binary motion (“static” versus “moving”), three-state (“static” versus “moving” versus “stairs”) and six-state (“static-downstairs” versus “moving-downstairs” versus “static-upstairs” versus “moving-upstairs” versus “moving-up-the-stairs” versus “moving-down-the-stairs”). The stairs versus all and four-state methods are not considered due to their relatively low recognition rates.

7.2.1 Binary Motion Predictions

The use of binary motion predictions in the production of Viterbi decoded room sequences necessitates derivation of two motion-dependent transition matrices from the training data. Figure 7.14 compares (a) the static and (b) the dynamic transition matrices for the binary motion HMM room predictor. The static room transition matrix generally has higher magnitude diagonal elements. The only exceptions are for rooms 4 and 7. The high probabilities of transitioning away from rooms 4 and 7 during a “static” motion state is due to the fact that rooms 4 and 7 are intermediary rooms between other rooms and are very rarely occupied in a “static” state. Hence there is very little training data for a static subject in rooms 4 and 7, and any transitions from rooms 4 and 7 in the “static” state are unrepresentative of real movements. The unexpectedly high probabilities of transitioning away from certain rooms in the static transition probability matrix is due to the incorrect classification of motion states for the training data.

The static subject room transition matrix is the most similar to the original transition matrix in Figure 6.5 since the subject was static for the majority of the time spent in the environment. The moving subject room transition matrix, on the other hand, has lower intensity diagonal elements, which confirms that the subject is less likely to remain in a particular room if he/she is moving. This approach, however, still permits the subject to remain in a given room with a probability of approximately 0.5 to 0.7, which improves on previous work which insisted on a transition when motion was detected.

The HMM configuration illustrated in Figure 7.13 is first implemented using the binary motion predictor and the transition matrices illustrated in Figure 7.14. The Viterbi room sequence decoder is implemented using both the raw and the augmented room probability estimates, as described in Section 6.2. The motion-based localisation performance for the raw density estimation can be seen in Table 7.12 and for the augmented density estimation in Table 7.13. Figure 7.15 provides a comparison of the performance of the motion-based HMM classifier with the performance of the classifier which does not use

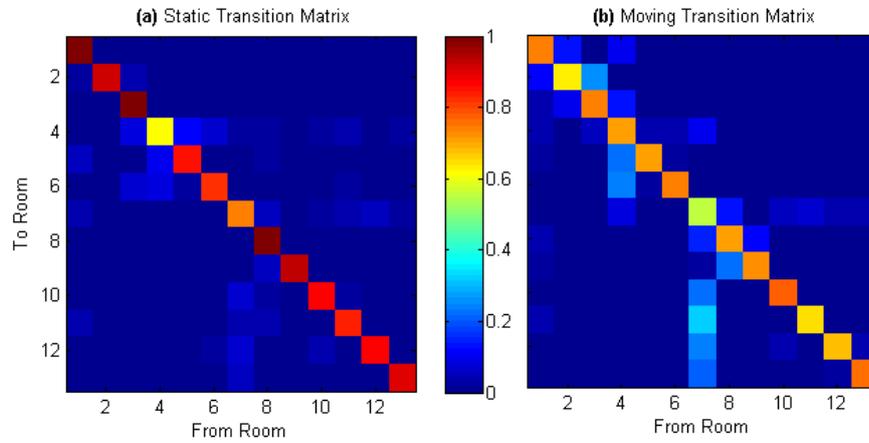


Fig. 7.14: Transition probability matrices for when the user is (a) static and (b) moving.

	NBC	LDA	QDA	GMM
EA	0.71	0.83	0.83	0.80
PDIF	1.75	1.36	1.57	1.67
MTD	4.55	4.61	3.78	3.99

Table 7.12: Localisation performance for the binary motion-dependent HMM classifier using raw RF data density estimates.

	NBC	LDA	QDA	GMM
EA	0.79	0.86	0.86	0.86
PDIF	1.31	1.20	1.55	1.57
MTD	4.58	3.73	4.27	4.18

Table 7.13: Localisation performance for the binary motion-dependent HMM classifier using augmented density estimates.

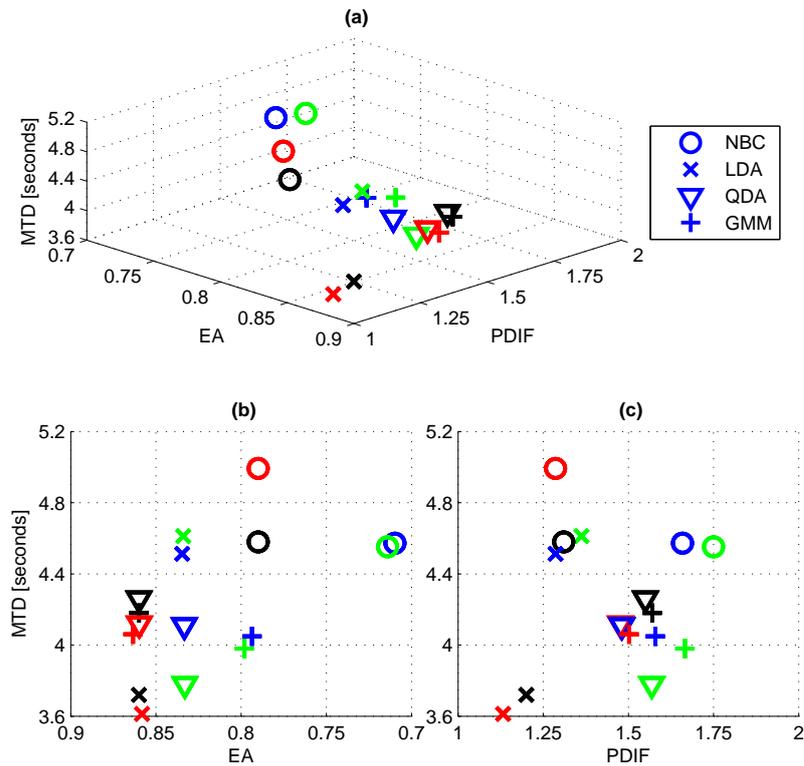


Fig. 7.15: Performance comparison for original raw HMM (blue), original augmented HMM (red), binary motion-based raw HMM (green) and binary motion-based augmented HMM (black).

motion information. In this figure, the blue and green markers indicate the original raw HMM and the motion-based raw HMM classifiers respectively. The red and black markers indicate the performance of the original augmented and the motion-based augmented HMM classifiers respectively. As would be expected, the augmented motion-based HMM classifiers perform better than the raw motion-based HMM classifiers. However, the only cases in which including motion information increases performance is for the raw QDA HMMs and augmented NBC HMMs. For these classifiers, the availability of motion information reduces the Mean Transition Delay (MTD). For all other classifiers, the motion-based online-varying HMMs have minimal effects on performance.

This relatively low localisation performance increase indicates that the binary motion predictions contribute very little to the localisation framework. The binary motion information available from the accelerometer signals offer little extra location indicative information than the available RF signals. Since the RF-based location predictions are now dependent on motion predictions, they are very sensitive to incorrect motion predictions. To investigate if the ability to detect the use of stairs contributes to localisation performance, the use of the three-state motion predictor as an input to the

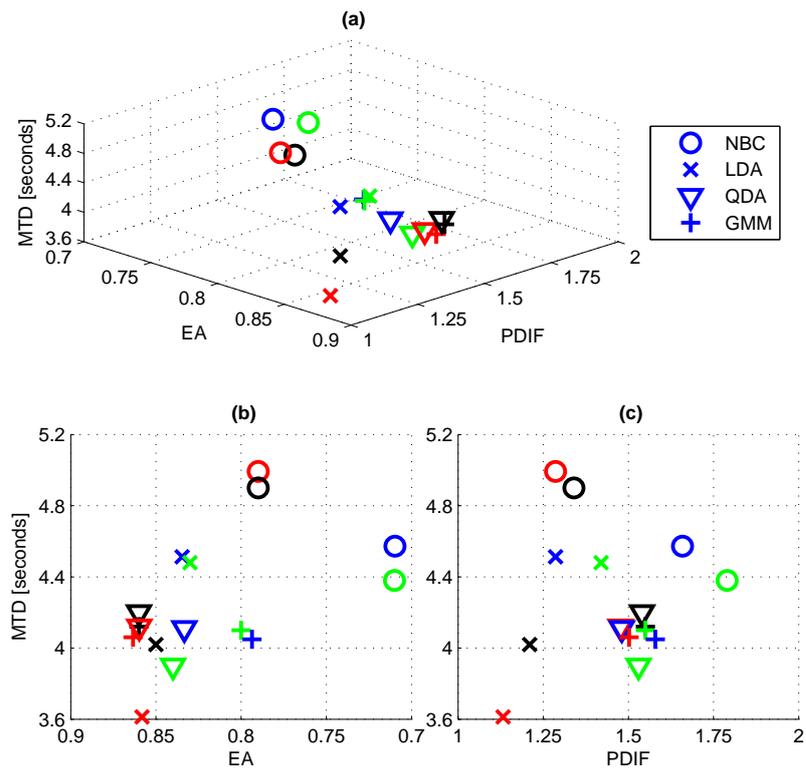


Fig. 7.16: Performance comparison for original raw HMM (blue), original augmented HMM (red), three-state motion-based raw HMM (green) and three-state motion-based augmented HMM (black).

transition matrix selection block is considered.

7.2.2 Three-State Motion Predictions

The motion-based HMM classifier in Figure 7.13 is now implemented using a three-state motion detector. Hence, the training data is now used to derive the room transition matrices for three motion states; “static”, “moving” and “stairs”. As before, the scatter plot in Figure 7.16 compares the performance of the three-state motion-based HMM localisation technique with the non-motion-based technique for both raw and augmented room probability density estimates. This figure indicates that the use of three-state motion predictions does not allow localisation performance higher than when binary motion predictions are used. In fact, there are reductions in performance for the raw QDA HMMs, augmented NBC HMMs and augmented LDA HMMs. Hence, the ability to recognise the extra motion state does not permit higher localisation accuracy.

The most likely reason the availability of motion predictions does not contribute to localisation performance is the misclassifications of motion states for both the training and test data. It is impossible to get reliable motion labels for the training and test data since it

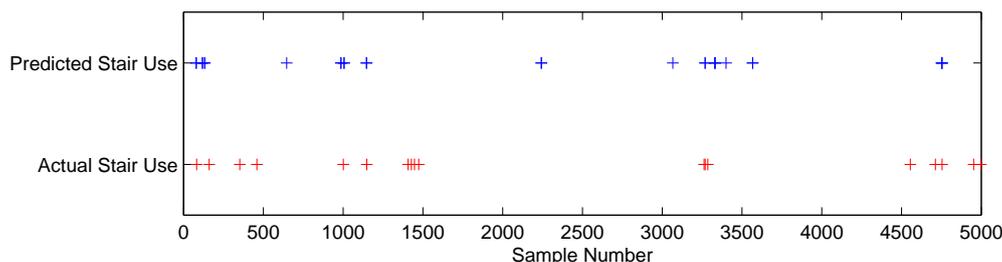


Fig. 7.17: Comparison of predicted stairs use from the three-state motion classifier and the actual stairs transitions deduced from the training data labels.

is impractical to require the user to voice annotate every single motion change throughout a given day. Room transitions were easily detectable for the long-term deployment due to the availability of the RFID labelling technique, however, there is no automatic technique for labelling motion states. Instead the motion labels for both the training and test RF localisation data have to be generated using the motion classifiers described in the previous section, using all of the available motion data as the motion training data. Hence, both the training and test RF data are prone to motion state errors. As a result, the transition probability matrices are not exactly as would be expected for a given state.

Since there are no motion labels available for the long-term localisation data it is difficult to quantify the frequency of incorrect motion classifications. It is, however, possible to detect the instances of stairs-use by processing the labels of the training and test data. For example, a transition from room 4 to room 7 corresponds to walking up the stairs and transitioning from room 7 to room 4 corresponds to walking down the stairs. Figure 7.17 compares the instances of actual stairs use, as derived from the first 5000 samples of the training data, and the stairs detected from the accelerometer data using the HMM framework as described in the previous section. It can be observed that stairs detections are frequently incorrect. In fact only 36% of stairs detections are actually correct. Hence, generating transition matrices from motion predictions which are so frequently incorrect will lead to transition matrices which have little correspondence with their intended function. For example, transition matrices for stairs detection should have high probability of transitioning to either rooms 4 or 7.

If the stairs state is detected at incorrect times for the training data, the transition probability matrices will not constrain the predictions to rooms 4 or 7 when stairs are detected. This effect can be confirmed when viewing Figure 7.18(c) which, amongst other things, has a relatively high probability of transitioning from room 7 to room 1 when stairs are detected. It does, however, correctly identify that there should be a high probability of transitioning from room 4 to room 7 when stairs are detected, suggesting that there are high detection rates for ascending the stairs. Regardless, the fact that there are high probabilities for transitions between any rooms besides 4 and 7 when stairs are detected

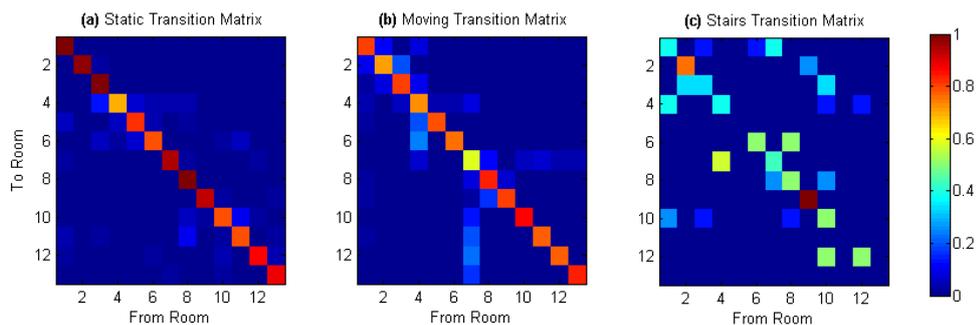


Fig. 7.18: Room transition probability matrices derived from the training data for the three states, (a) “Static”, (b) “Moving” and (c) “Stairs”.

confirms that the false detection of stairs is extremely frequent and will lead to incorrect room transition constraints during Viterbi decoding. Hence, the detection of stairs also does not positively contribute to room recognition accuracy.

7.2.3 Six-State Motion Predictions

The final consideration for the efficacy of motion-based HMM classifications is if the availability of floor predictions from the accelerometers contributes to localisation performance. As before, the motion-based HMM classifier illustrated in Figure 7.13 is implemented, but with the six-state motion predictor and 6 corresponding transition matrices. Again, the performance of this motion based technique can be viewed in Figure 7.19. It can be seen that, even though augmented LDA HMMs which utilise 6 state motion information have increased performance over the 3 state technique in Figure 7.16, neither it or any other classifiers have performance improvement beyond the classifiers which do not incorporate motion information.

Upon considering the ability of the six-state motion predictor to recognise stairs in Figure 7.20, it is understandable why an increase in accuracy is not possible with this technique. Compared to the three-state predictor in Figure 7.17, this technique over-predicts stairs, while still only producing correct stairs predictions 36% of the time. Hence, this technique is unable to reliably predict which floor the user is on. This unreliability is evident in Figure 7.21(c) which should have high probabilities of remaining in rooms 8, 9, 10, 11, 12 and 13, but instead has zero probability of remaining in rooms 10 and 12. This indicates that the floor predictions were frequently incorrect for the training data. Furthermore, it is expected that for the “moving-up-the-stairs” state there should only be a high probability of transitioning from room 4 to room 7 and for the “moving-down-the-stairs” state there should only be a high probability of transitioning from room 7 to room 4. This is not the case due to frequently incorrect stairs detections.

This section has presented the integration of accelerometer-derived motion information into the Viterbi decoded state-sequences. Accelerometer frequency components were used

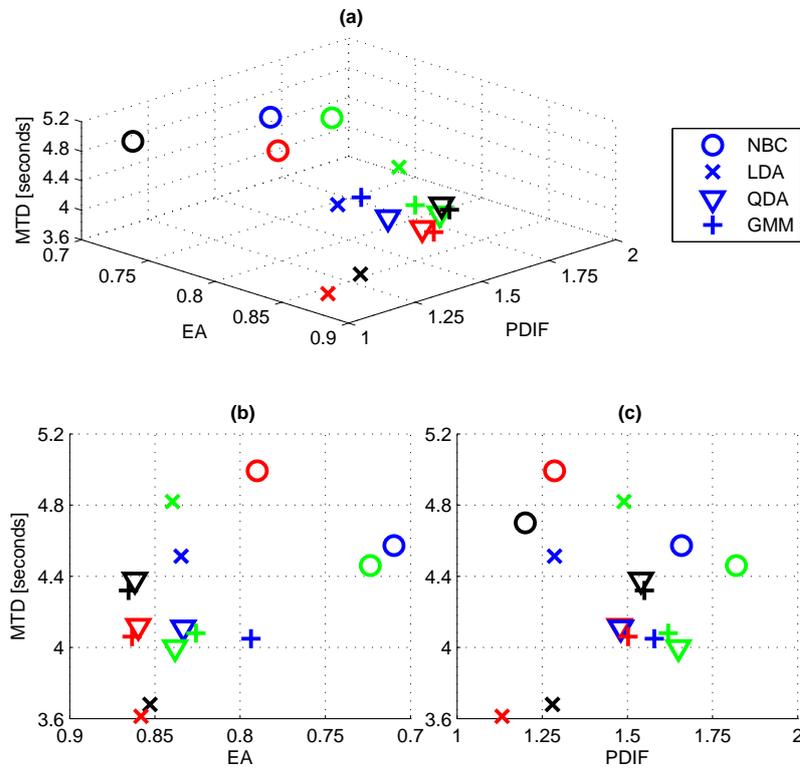


Fig. 7.19: Performance comparison for original raw HMM (blue), original augmented HMM (red), six-state motion-based raw HMM (green) and six-state motion-based augmented HMM (black). As before, (a), (b) and (c) show different projections of the same graph to allow a direct comparison of the classifiers under the different accuracy measures.

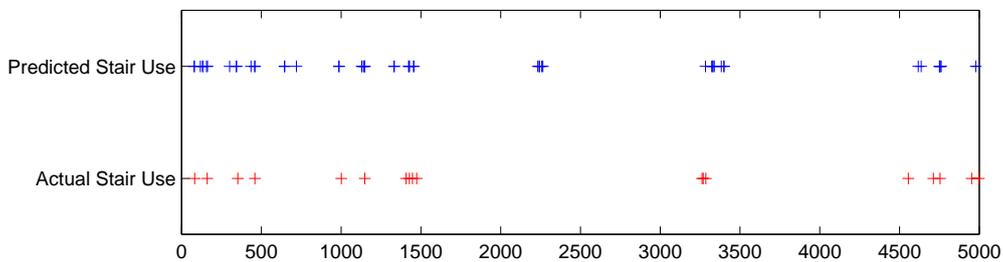


Fig. 7.20: Comparison of predicted stairs use from the six-state motion classifier and the actual stairs transitions deduced from the training data labels.

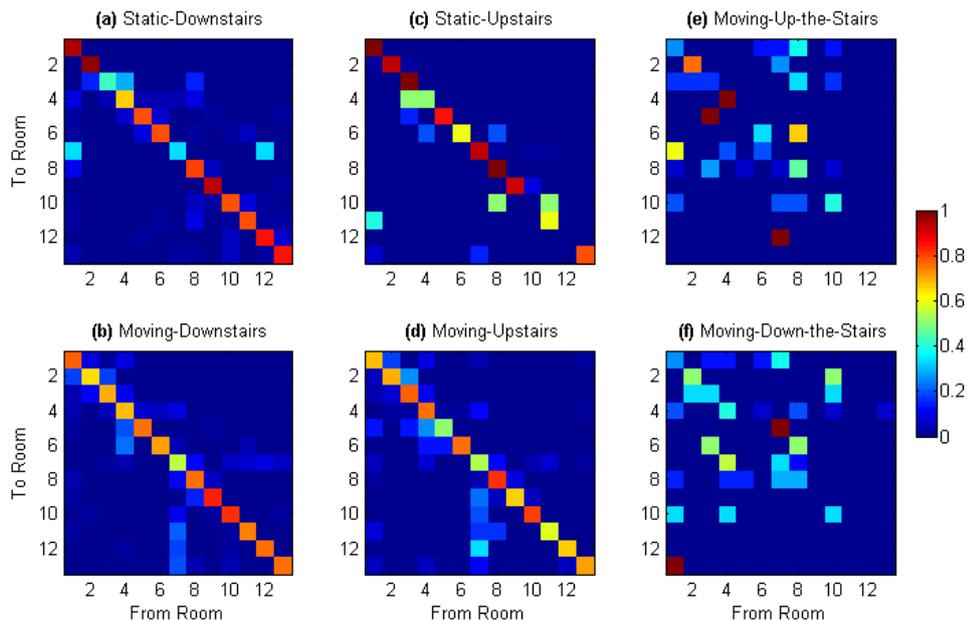


Fig. 7.21: Room transition probability matrices for the six-state motion predictor.

instead of RF variability and accelerometer variability as motion classification features due to the higher recognition rates they permit. Phone orientation, derived from accelerometer magnitudes (Ofstad et al., 2008), is not used to detect if the user is sitting in a “static” state since it is unreliable to assume the phone is consistently orientated in the subject’s pocket. It has been found that even though the accelerometer frequency components allow detection of a range of motion states with varying degrees of accuracy, they do not contribute to the localisation performance. This is likely due to the fact that the available RF signals already contain sufficient location indicative information, and motion information, in its current level of reliability, does not permit any additional localisation performance.

The finding of this work, that motion information does not contribute to localisation accuracy, goes against the implication of Krumm and Horvitz (2004) that using motion information allows higher HMM localisation accuracy. Even though this work uses more accurate accelerometer frequency components rather than RF variability components and does not make assumptions about the speed of the subject, it actually shows either a negligible improvement or a marginal disimprovement in localisation performance when including motion information. The slight reduction in localisation performance is due to the inability to automatically label the training data with 100% accurate motion labels as evident in the transition probability matrices which are not as would be expected for the detected motion types. Hence, the investigated technique of incorporating motion predictions into Viterbi state sequence decoding is not a viable approach to increasing localisation performance, unless a reliable motion detection technique similar to the RFID

location labelling technique is available to generate training data.

Even though the detectable motion levels, in their current levels of accuracy, do not contribute to long-term localisation performance, they can still be used for healthy activity indicators. The binary static versus moving predictor or the detection of stairs crossing can be used to appraise the levels of healthy activity a user undertakes in a day. This would actually be more reliable than estimating the distance travelled from room transitions since a user could be performing a high level of healthy activity within a room with no room transitions being detected. Hence, a majority of the motion detection techniques developed in this chapter can be applied in an elder physical and cognitive health monitoring scenarios.

7.3 Conclusions

Prior to this work Krumm and Horvitz (2004) suggested using the Viterbi algorithm for a WiFi localisation platform. That work used WiFi signal variability as the dependent variables for the HMM transition matrices. Transition probability matrices were estimated based on assumptions about the travel distances of the user given detected motion. The work of this thesis, however, showed that basing a HMM localisation framework on motion information does not contribute to higher localisation accuracy. This finding is based on more reliable accelerometer motion predictions rather than the RF variability method, which has been shown to be inferior. The main contributions of this chapter are as follows;

Movement Classification This chapter derived effective methods for detecting motions of interest to a motion-dependent localisation system. It was shown that binary moving and static classifications could be conducted using the windowed accelerometer variability. By employing frequency component feature extraction further motion types could be discriminated between, such as “moving”, “static”, “moving-up-the-stairs” and “moving-down-the-stairs”. The true novelty in this approach is in applying Bayesian filtering to produce more realistically distributed predictions, hence higher mean motion recognition rates. Finally, the availability of Bayesian filtering allowed the prediction of which floor the user is inhabiting from only accelerometer signals with 68% MRR. Such an approach has not been attempted in literature prior to this work.

Time-Varying HMMs Varying the probabilities of making certain transitions based on the detected motion type should, in theory, produce higher localisation accuracies. Even though it was possible to detect many motions with varying degrees of success, it was found that basing the location predictions on these motion predictions did not permit higher accuracies. The main issue was that the training data did not have labels for the

motion data. Hence, the motion labels had to be produced from classifiers which were not 100% reliable. Then training a time-varying HMM using these potentially mis-labelled classes was likely to result in degraded localisation performance. Unfortunately, it is impractical to deploy technologies to automatically obtain motion labels and unrealistic to expect an elder subject to produce these labels manually in a realistic deployment. Finding resolutions to these issues shall be the focus of future work.

Concluding Summary and Future Work

8.1 Concluding Summary

This work contributes to an affordable and relatively unobtrusive home localisation system with the aim of making aging in place a more attractive alternative to dedicated elder care facilities. The developed system achieves its affordability by utilising absolutely minimal levels of cheap Bluetooth[®] hardware within the home environment. Location predictions with such minimal hardware are generated using as many location and activity indicative signals as possible, such as Bluetooth[®] signals, cellular signals, phone accelerometer signals and even RFID readings for redundancy. A variety of improvements to the localisation system are explored and compared for efficacy in a typical home environment. The findings of this exploratory research have led to the development of an optimal home localisation system which is capable of generating location predictions in near-real-time by using a modified discrete Bayesian filtering framework.

The core contributions of this thesis can be summarised as follows.

8.1.1 Localisation and Classification Literature Review

Chapter 2 presented a review of the fundamental theories and concepts of localisation theory, including Radio Frequency (RF) signal propagation models and RF localisation techniques which depend on such signal propagation mechanisms. Following that, a comprehensive review of the existing indoor localisation work is presented. Fingerprinting is the most commonly utilised localisation technique since it does not rely on simplified propagation models to inform location predictions. For this reason Chapter 3 was dedicated to presenting the candidate pattern recognition techniques which were to be

employed throughout this thesis.

8.1.2 Localisation Hardware Platform Design

Chapter 4 enabled the selection of the optimal home localisation hardware by experimentally comparing the low-power wireless technologies Bluetooth[®] and ZigBee[®]. Higher redundancy, hence location prediction accuracy, was illustrated by employing all available transmission channel readings from each wireless protocol. It was found that the signal resolution present in the chosen Bluetooth[®] implementation enabled higher localisation accuracy. Furthermore Bluetooth[®] devices allow the acquisition of cellular signals and Bluetooth[®] devices are more commonly available, leading to cheaper and more useful mobile localisation devices. It is later found that the use of the standard Bluetooth[®] protocol allows the easy integration of an RFID localisation technique. Finally, the selected Bluetooth[®] localisation platform's localisation potential was demonstrated in a 4-room environment with absolutely minimal deployed infrastructure, a single Bluetooth[®] enabled computer.

8.1.3 Long-term Localisation Accuracy

The localisation technology is all but useless unless it can be validated in a realistic home environment. For this reason Chapter 5 presented results from a realistic day-to-day application of the localisation technology. Alternative localisation techniques, such as Radio Frequency Identification (RFID) and Passive Infrared (PIR), were installed to enable a number of experiments. The availability of the accurate RFID localisation technique enabled the generation of a novel localisation accuracy metric, the Empirical Accuracy (EA), which allows the estimation of the accuracy of the system over long periods of time from only short periods of experimentation. The availability of reliable location labels also allowed a comparison between the localisation system developed in this thesis and the PIR localisation technique frequently employed in home room-level localisation scenarios. It is demonstrated that the Bluetooth[®] localisation system outperforms the PIR localisation system, especially as the environment occupancy increases.

This chapter also highlighted the importance of signal diversity throughout the deployment environment. The single access point deployment topology which showed promise in Chapter 4 was unable to resolve the high number of locations in a typical home environment, in this case 13 rooms. A novel connection topology was utilised to enable signal resolution previously impossible with cheap Bluetooth[®] mobile devices and inexpensive “dumb” Bluetooth[®] beacons. An exploration of the optimal subset of access points indicated that long-term localisation accuracy is sensitive to the AP deployment locations. With this multiple AP data it was possible to explore further improvements to the classification framework to maximise localisation accuracy. These improvements

include smoothing, dynamic modelling, room prior probability inclusion and uncertainty rejection. The optimal parameters for each improvement was explored and the implications of increasing accuracy using uncertainty rejection was explained.

8.1.4 Hidden Markov Models

Chapter 6 developed further measures of localisation system performance, which have been ignored in prior localisation work. As well as the long-term room detection ability, it was necessary to examine the effect of spurious transitions between rooms and the delay in detecting transitions between rooms. The ability to visualise these performance metrics enabled the selection of the classifiers which exhibit the best overall performance. However, there were still deficiencies in the performance of the original classifiers, even the augmented implementations. Hence, a Hidden Markov Model (HMM) framework was applied to the localisation problem to allow the integration of room connectivity and user movement tendencies into the location predictions. It was demonstrated that applying augmented HMMs enabled higher overall localisation performance. This performance was shown to improve with the use of higher order Markov models, which have not previously been considered for localisation. This performance improvement came with significantly higher computation and memory overheads.

To combat the high order Markov model memory overheads and to achieve near-real-time computation of room predictions, a modified implementation of the Viterbi decoding algorithm was developed. Instead of producing globally optimal room sequence estimates after the final observation has been obtained, the Viterbi algorithm was modified to produce state sequence estimates as soon as the room partial probabilities have stabilized to a single room. This approach is also unique to the field of localisation and is achieved with approaches far simpler than those necessary for real-time speech recognition. This is because in a localisation application there are far fewer states and the partial probability is likely to converge on a stable maximum probability state for longer.

8.1.5 Time-Varying HMMs

In an attempt to improve localisation performance, Chapter 7 develops a HMM framework which permits the modification of the room transition probabilities based on auxiliary knowledge about the motions of the user. A previous approach of analysing signal strength variability is considered but it is found that the frequency components of the accelerometer signals permit better motion discrimination, particularly when higher numbers of motion classes are considered. This work contributes to other motion classification work by modelling motion as the hidden states of a HMM. This has been shown to permit higher motion recognition rates for all combinations of motions. Even the possibility of being able to predict which floor the user is inhabiting from accelerometer signals has been

demonstrated. This is achieved by using a unique combination of HMMs and labels which contain information beyond that which can be inferred from accelerometer signals alone. A floor recognition rate of almost 70% has been shown which improves upon the 50% resulting from accelerometer floor classification without HMMs.

The availability of these motion predictions enables the modification of HMM location transition constraints depending on the current motion. For example the probability of remaining in the current room is at a maximum if the user is “static”, and the probability of transitioning to other rooms goes up when a “moving” state is detected. Even though relatively high motion detection accuracies were possible they did not consistently enable significantly higher localisation performance. However, for the accurate binary motion predictions there were two instances of the room prediction delay decreasing when utilising accelerometer motion predictions, which indicates that there is potential for decreasing room transition detection lag with this technique. The main limiting factor is the quality of the motion labels for the long-term localisation dataset. It was prohibitively difficult to obtain actual motion labels for long periods of time so the labels had to be generated from the relatively small motion prediction test dataset. Hence, the room transition matrices are based on some erroneously labelled motions and as a result do not accurately represent all of the room transitions typical of certain motions.

8.1.6 Limitations

In spite of the positive findings of this work, it is important to highlight the caveats which underpin these findings. The initial localisation tests for a static scenario in Chapter 4, Figure 4.5, were obtained in an unrealistic test scenario. This scenario required the mobile phone to be moved to positions throughout the test environment and held 1m above the ground on a wooden platform. Positions which fell on beds required a smaller platform to be used to ensure the total height of the platform and bed was 1m. This smaller platform was made of cardboard which may have slightly different RF characteristics than the wooden platform. This could have a minor effect on signal repeatability for locations which are similar, but have different types of platforms underneath the mobile phone. These static tests also use the experimental simplification that the phone’s orientation does not change throughout the test environment. This is not meant to be representative of a realistic scenario; rather merely suggest the viability of the proceeding experiments.

Next, dynamic tests were conducted to alleviate these simplifications. The dynamic test results obtained in Chapter 4 are obtained using training and test datasets which are 15 minutes in length. Such short periods of data acquisition are conducted because the tests in the small test environment use the accurate, but lengthy process of using voice-annotation to produce the room labels. Again, these dynamic small environment experiments were merely to explore the viability of more rigorous tests which are conducted in Chapter 5.

Chapter 5 utilises the RFID labelling technique to enable the acquisition of labelled data for “long-term” experiments.

The term “long-term” should also be clarified. Long-term experiments could be presumed to take place over several weeks or months. In this work, however, long-term refers to experiments which use one day of training data and one day of test data. Hence, these tests are “long-term” relative to the tests in Chapter 4. Furthermore, they are significantly longer than tests which have generally been employed in the field of RF localisation to date.

Even though the tests in Chapter 5 quantified the localisation accuracy over the significant period of a day, the results would have benefited from evaluation over a greater number of days. It was prohibitively difficult to obtain numerous days of training and test data since it required the experimenter to stay at home and refrain from leaving the house as much as possible during the test period. The tests also needed to be conducted during weekdays to reduce instances of multiple occupancy. Hence, it was only possible to obtain two days of data for the SAP tests, two days for the SBS tests and two days for MAP tests. The difficulty in obtaining greater quantities of data was addressed by employing the Empirical Accuracy (EA) measure. This measure was derived from the relative frequency of room occupation for a week. Hence, EA was employed for these experiments to suggest the accuracy which would occur with this deployment for a week of experimentation.

There were also implications of allowing the phone to move freely within the experimenter’s pocket. Even though the phone was placed in the experimenter’s pocket in a consistent orientation, it was able to freely move during experiments. This could have implications on the consistency of the detected signal readings. In a realistic deployment, the phone could be placed in the user’s pocket in any arbitrary orientation. The affect this could have on signal repeatability, due to changing PLF, and as a result, accuracy, was not explored in this work. There is also the issue in a realistic deployment that the user could leave the phone down for periods of time. This would suggest that the user is static while he/she is actually moving around. Since accelerometer readings are consistently relayed to the basestation computer, implementing a technique to detect when the phone is placed down for a significant period of time would be a trivial addition to the system which was not implemented in this work. In reality, such a technique would refrain from tracking the user when the magnitudes of the accelerometer axes suggest that the phone is lying flat and prompt the user to carry the phone when it is left down for several minutes.

This work could have also benefited from a more reliable BSC configuration. Due to the transmission range limitations of the Bluetooth[®] protocol, the BSC had to be placed central to the test environment to ensure a connection was consistently maintained. Even still, it was possible for the Bluetooth[®] connection to momentarily fail. When the connection is re-established the data which was stored by the phone since the connection

failure is downloaded to the computer. In a real deployment, this would make location predictions impossible when the user is outside the range of the BSC. If this became a serious issue, for example in houses with concrete walls, the connection topology of the phone presented in Chapter 5 could be modified to remove the necessity of a BSC. Then the calculations could be performed on the phone or the readings could be downloaded to a remote server via GSM for location calculation. However, the current topology ensures the greatest power savings while providing relatively consistent coverage of the home environment. Furthermore, using Intel's BioMOBIUS research platform to decode the X10 PIR packets at the BSC may have led to some erroneous room predictions. More accurate PIR tracking may be possible by decoding the X10 packets using custom software.

Finally, significant issues were present with the motion labels for the experiments in Chapter 7. Room labels were easily obtained using the RFID labelling technique. There is, however, no such technique to automatically generate labels for user motion type. Previous work focused on accelerometer motion predictions from data obtained in unrealistic sequences of motion (e.g. Allen et al. (2006), Ibrahim et al. (2008)). In the realistic scenario we are investigating, it would take considerable sensor redundancy within the environment to accurately label the motions of the user. Instead of deploying a vast array of sensors, it was decided to automatically generate the motion labels for the long-term test data using classifiers trained on a small amount of voice-labelled accelerometer data. This means that the labels for the long-term data used for the Time-Varying HMM experiments are susceptible to labelling errors even before the location predictions take place. To ensure accurate labels for the long-term Time-Varying HMM experiments it is necessary to either produce voice-labels for the entire long-term localisation experiment or equip the home environment with sensors which can detect these motions.

8.2 Future Work

This thesis addresses many of the central issues surrounding the deployment of a cheap, accurate and reliable indoor localisation system. However, there are still issues remaining, which, if addressed correctly, could lead to a localisation system with lower deployment costs and higher long-term accuracy. Some of the issues for future consideration are as follows.

8.2.1 Bluetooth[®] Beacon Location Optimisation

It has been demonstrated in Chapter 5 that the positions of the deployed access points strongly influences the long-term localisation accuracy. It has also been proposed that the establishment of which access points are most beneficial to localisation performance can only be empirically evaluated when APs are installed in all of the candidate deployment

locations. Furthermore, when APs have been deployed there is little benefit to removing the least useful APs since they can still occasionally be of use to the localisation system. Hence, to minimise deployment costs, it would be beneficial to generate patterns for the best deployment methodologies from experimental evidence from several deployment environments.

This would allow the establishment of a set of best practices for AP installation locations without requiring a comprehensive deployment, experimentation and then removal of the most redundant APs. For example, deploying APs only in locations where the user spends the most time during the day may generally lead to the highest accuracy. Hence, the best deployment locations could be easily established with a simple survey of the person to be localised when the system is deployed.

8.2.2 Automatic Room Labelling

An RFID technology was employed to generate labels for accuracy metric derivation, training and testing of the Bluetooth[®] localisation system. For a realistic scenario, however, it would be inconvenient for an elder to manually scan each doorway they pass through for the first day of deployment, which would inevitably lead to inaccurate room labels. Hence, Chapter 5 concluded by proposing a foot-mounted RFID reader and RFID tags across doorways to enable automatic room label acquisition. This would allow at least a day of labelled data to be automatically obtained before the RFID reader could be discarded in favour of the mobile phone alone. Using the mobile phone alone is favourable since the RFID reader would require the elder to ensure the device is placed on whatever footwear they choose for a given day and ensure the battery is charged every day.

The automatic RFID labelling system, however, still requires some development. The battery capacity of the reader is at present insufficient for the constant scanning cycle necessary to successfully detect all door crossings for an entire day. There is much scope for the development of the automatic RFID labelling system. For example the phone can turn off the reader when a “static” state is detected to save battery, or the detection of stairs can be used to increase confidence in detected transitions to certain floors. Furthermore, the antenna of the foot-mounted RFID reader could be redesigned and repositioned to reside on the sole of the shoe, to minimise the effect of attenuation of the human foot on tag detection rates.

8.2.3 Motion Sensing Redundancy

Chapter 7 concluded that the motion labels for an entire day were insufficiently accurate to base motion-dependent predictions on. This could be addressed in future work by increasing sensor redundancy throughout the test environment to allow the accurate long-term detection of all motion states. For example, pressure pads could be placed on seats

to detect when the subject is sitting in an unmoving state, break-beam sensors could be placed on stairways to detect when stairs are used and more accurate accelerometers can be foot-mounted to increase motion detection rates. Independently of motion estimation, higher accuracy motion predictions are envisaged to permit more informative summaries of the levels of exercise an elderly person undertakes on particular days. In terms of location estimation, improved motion detection capability will enable higher accuracy location predictions with even lower quantities of APs within the environment. Most significantly, higher accuracy motion predictions will enable more reliable detection of which floor an elder resides upon, using only accelerometer signals.

8.2.4 Efficient High-Order HMM Implementations

It was demonstrated in this thesis that improvements in localisation performance were possible using higher order Hidden Markov Models to further constrain the state sequence to probable paths. These improvements, however, came with significant computational and memory requirement increases for each model order increase. The real-time Viterbi algorithm had the attractive property that short-term backtracking required significantly less memory to maintain a list of states since the previous backtracking interval. Even though memory requirements were significantly reduced, there were still identical computational requirements.

Future extensions to this work can reduce the computational requirements of decoding higher order HMMs by borrowing from prior work in the field. For example, early work by du Preez (1997) proposes a technique of translating arbitrarily high-order HMMs to a first-order equivalent. This is equivalent to translating arbitrarily high-dimension transition matrices and partial probability matrices to the 2-dimension equivalents. This approach has the attractive property that unused combinations of state sequences have negligible computational overheads. Hence, higher localisation performance can be obtained without requiring the full computational burden of high order HMM decoding. Future work will explore the optimal tradeoff between localisation performance and computational burden such that maximum performance can be accomplished within the computational constraints of affordable Basestation computers.

References

- Abowd, G.D., Atkeson, C. G., Hong, J., Long, S., Kooper, R., and Pinkerton, M. (1997). Cyberguide: a mobile context-aware tour guide. *Wireless Networks*, 3(5):421–433. ISSN 1022-0038.
- Agostaro, F, Genco, A, and Sorce, S (2004). A fuzzy approach to bluetooth positioning. In *8th WSEAS International Conference on COMPUTERS*, pages 393 – 397.
- Allen, Felicity R., Ambikairajah, Eliathamby, Lovell, Nigel H., and Celler, Branko G. (2006). Classification of a known sequence of motions and postures from accelerometry data using adapted gaussian mixture models. *Physiological Measurement*, 27(10):935 – 951. ISSN 0967-3334.
- Anastasi, G., Bandelloni, R., Conti, M., Delmastro, F., Gregori, E., and Mainetto, G. (2003). Experimenting an indoor bluetooth-based positioning service. In *Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on*, pages 480–483.
- Antal, P., Fannes, G., Timmerman, D., Moreau, Y., and De Moor, B. (2003). Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection. *Artificial Intelligence Med*, 29(1-2):39–60. ISSN 0933-3657.
- Antenova Titanis Product Specification (2009). Antenova - Integrated Antenna Solutions. [Internet]. Available From: www.antenova.com [Accessed 11th October 2009].
- Ardo, H., Astrom, K., and Berthilsson, R. (2007). Real time viterbi optimization of hidden markov models for multi target tracking. In *IEEE Workshop on Motion and Video Computing, 2007. WMVC '07.*, pages 2–2.
- Bahl, P. and Padmanabhan, V.N. (2000a). Enhancements to the RADAR user location and tracking system. Technical Report MSR-TR-2000-12, Microsoft Research.

-
- Bahl, P. and Padmanabhan, V.N. (2000b). Radar: an in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784 vol.2.
- Behan, J., Prendergast, D., Walsh, L., and Quigley, B. (2008). Social rhythms and nocturnal routines in community dwelling older adults. In *ICOST'08*, pages 73–80.
- Bielawa, T.M. (2005). *Position Location of Remote Bluetooth Devices*. Master's thesis, Virginia Polytechnic Institute and State University.
- Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. Springer. ISBN 0387310738.
- Bloit, J. and Rodet, X. (2008). Short-time viterbi for online hmm decoding: Evaluation on a real-time phone recognition task. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 2121–2124. ISSN 1520-6149.
- Bluetooth[®] Special Interest Group (2001). Specification of the bluetooth system: Version 1.1.
- Boushaba, M., Benslimane, A., and Hafid, A. (2007). A2l: Angle to landmarks based method positioning for wireless sensor networks. In *IEEE International Conference on Communications, 2007. ICC '07.*, pages 3301–3306.
- Burgard, W., Fox, D., Hennig, D., and Schmidt, T. (1996). Estimating the absolute position of a mobile robot using position probability grids. In *AAAI/IAAI, Vol. 2*, pages 896–901.
- Callaghan, MJ., Gormley, P., McBride, M., Harkin, J., and McGinnity, TM. (2006). Internal location based services using wireless sensor networks and rfid technology. *International Journal of Computer Science and Network Security (IJCSNS)*, 6(4):108–113.
- Castano, J.G., Svensson, M., and Ekstrom, M. (2004). Local positioning for wireless sensors based on bluetooth. In *Radio and Wireless Conference, 2004 IEEE*, pages 195–198.
- Castro, P., Chiu, P., Kremenek, T., and Muntz, R. (2001). A probabilistic room location service for wireless networked environments. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, volume 2201, pages 18–34. Springer-Verlag, Atlanta, Georgia, USA.

-
- Ceramic Chip Antenna Data Sheet (2003). Bluetooth wireless 2.4GHz embedded data radios, wireless modems, modules, RS232 voice and LAN Access Point repeaters. [Internet]. Available From: www.blueradios.com [Accessed 11th October 2009].
- Chan, M., Campo, E., Laval, E., and Esteve, D. (2002). Validation of a remote monitoring system for the elderly: Application to mobility measurements. *Technology and Health Care*, 10:391 – 399.
- Chen, J., Kwong, K., Chang, D., Luk, J., and Bajcsy, R. (2005). Wearable sensors for reliable fall detection. In *27th Annual International Conference of the Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005.*, pages 3551–3554.
- Chen, Y. and Kobayashi, H. (2002). Signal strength based indoor geolocation. In *Communications, 2002. ICC 2002. IEEE International Conference on*, volume 1, pages 436–439.
- Cheung, K.-W., Sau, J.H.-M., and Murch, R.D. (1998). A new empirical model for indoor propagation prediction. *Vehicular Technology, IEEE Transactions on*, 47(3):996–1001. ISSN 0018-9545.
- Cheung, K.C., Intille, S.S., and Larson, K. (2006). An inexpensive bluetooth-based indoor positioning hack. In *UbiComp 2006 Extended Abstracts (Posters Program)*.
- Chipcon (2006). CC2420 datasheet. Rev. 1.4.
- Chou, W. and Juang, B.-H., editors (2003). *Pattern recognition in speech and language processing*. CRC Press.
- Cox, D., Kindratenko, V., and Pointer, D. (2003). Intellibadge : Towards providing location-aware value-added services at academic conferences. In *UbiComp 2003: Ubiquitous Computing*, pages 264–280. Springer-Verlag, Seattle, WA, USA.
- Deasy, T.P. and Scanlon, W.G. (2004). Accuracy improvement algorithms for prediction of user location using receive signal strength indication in infrastructure w lans. In *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004.*, volume 3, pages 1757–1761.
- Department of Economic and Social Affairs, Population Division (2009). World population ageing 2009. Technical Report ESA/P/WP/212, United Nations.
- di Flora, C. and Hermersdorf, M. (2008). A practical implementation of indoor location-based services using simple wifi positioning. *Journal of Location Based Services*, 2(2):87 – 111.

-
- Domingos, P. and Pazzani, M.J. (1996). Beyond independence: Conditions for the optimality of the simple bayesian classifier. In *International Conference on Machine Learning*, pages 105–112.
- du Preez, J.A. (1997). Algorithms for high order hidden markov modelling. In *Proceedings of the 1997 South African Symposium on Communications and Signal Processing, 1997. COMSIG '97.*, pages 101 – 106.
- Ebden, M. (2008). Gaussian processes for regression: A quick introduction. Technical report, Department of Engineering Science, University of Oxford.
- Faria, D.B. (2006). Modelling signal attenuation in iee 802.11 wireless lans vol 1. Technical report, Stanford University, Research & Development.
- Feldmann, S., Kyamakya, K., Zapater, A., and Lue, Z. (2003). An indoor bluetooth-based positioning system: concept, implementation and experimental evaluation. In *International Conference on Wireless Networks*, pages 109 – 113.
- Figueiredo, M.A.T. and Jain, A.K. (2000). Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:381–396.
- Fox, V., Hightower, J., Liao, Lin, Schulz, D., and Borriello, G. (2003). Bayesian filtering for location estimation. *Pervasive Computing, IEEE*, 2(3):24–33. ISSN 1536-1268.
- Friedman, J.H., Baskett, F., and Shustek, L.J. (1975). An algorithm for finding nearest neighbors. *IEEE Transactions on Computers*, C-24(10):1000–1006. ISSN 0018-9340.
- Genco, A. (2005). Three step bluetooth positioning. In *LoCA*, pages 52–62.
- Genco, A., Sorce, S., and Scelfo, G. (2005). Bluetooth base station minimal deployment for high definition positioning. In *The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005.*, pages 454–460.
- Giannakouris, K. (2008). Ageing characterises the demographic perspectives of the european societies. Technical Report EUROSTAT Statistics in Focus, 78/2008, European Commission.
- GrandCare Systems (2009). GrandCare Systems. [Internet]. Available From: www.grandcare.com [Accessed 30th September 2009].
- Harter, A., Hopper, A., Steggles, P., Ward, A., and Webster, P. (1999). The anatomy of a context-aware application. In *Mobile Computing and Networking*, pages 59–68.

-
- Hastie, T., Tibshirani, R., and Friedman, J. H. (2001). *The Elements of Statistical Learning*. Springer-Verlag.
- He, Y. (1988). Extended viterbi algorithm for second order hidden markov process. In *9th International Conference on Pattern Recognition*, volume 2, pages 718–720.
- Helal, S., Winkler, B., Lee, Choonhwa, Kaddoura, Y., Ran, L., Giraldo, C., Kuchibhotla, S., and Mann, W. (2003). Enabling location-aware pervasive computing applications for the elderly. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)*., pages 531–536.
- Hermersdorf, M. (2006). Indoor positioning with a wlan access point list on a mobile device. In *Proceedings of WSW06 at SenSys06*.
- Hightower, J. and Borriello, G. (2001). Location sensing techniques (companion to “location systems for ubiquitous computing”). *Computer*, 34(8):57–66. ISSN 0018-9162.
- Hightower, J. and Borriello, G. (2004). Particle filters for location estimation in ubiquitous computing: A case study. In *Proceedings of the Sixth International Conference on Ubiquitous Computing (UbiComp 2004)*, volume 3205 of *Lecture Notes in Computer Science*, pages 88–106. Springer-Verlag.
- Hightower, J., Vakili, C., Borriello, G., and Want, R. (2001). Design and calibration of the spoton ad-hoc location sensing system. Technical report, University of Washington, Computer Science and Engineering.
- Hou, J.C., Wang, Q., AlShebli, B.K., Ball, L., Birge, S., Caccamo, M., Cheah, C.-F., Gilbert, E., Gunter, C.A., Gunter, E., Lee, C. G., Karahalios, K., Nam, M.-Y., Nitya, N., Rohit, C., Sha, L., Shin, W., Yu, S., Yu, Y., and Zeng, Z. (2007). Pas: A wireless-enabled, sensor-integrated personal assistance system for independent and assisted living. In *High Confidence Medical Device Software and Systems (HCMDSS '07)*. Boston, MA.
- Ibrahim, R.K., Ambikairajah, E., Celler, B., Lovell, N.H., and Kilmartin, L. (2008). Gait patterns classification using spectral features. In *Proceedings of the IET Irish Signals and Systems Conference 2008 (ISSC08)*, pages 254–259.
- Ikegami, F., Takeuchi, T., and Yoshida, S. (1991). Theoretical prediction of mean field strength for urban mobile radio. *IEEE Transactions on Antennas and Propagation*, 39(3):299–302. ISSN 0018-926X.
-

-
- Izquierdo, F., Ciurana, M., Barcelo, F., Paradells, J., and Zola, E. (2006). Performance evaluation of a toa-based trilateration method to locate terminals in wlan. In *1st International Symposium on Wireless Pervasive Computing, 2006*, pages 1 – 6.
- Jami, I., Ali, M., and Ormondroyd, R.F. (1999). Comparison of methods of locating and tracking cellular mobiles. In *IEE Colloquium on Novel Methods of Location and Tracking of Cellular Mobiles and Their System Applications*, pages 1–6.
- Jin, G.-Y., Lu, X.-Y, and Park, M.-S. (2006). An indoor localization mechanism using active rfid tag. In *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, volume 1, pages 40–43.
- John, G.H. and Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338 – 345.
- Kelly, D., Behan, R., Villing, R., and McLoone, S. (2009a). Computationally tractable location estimation on WiFi enabled mobile phones. In *Proceedings of the IET Irish Signals and Systems Conference 2009 (ISSC09)*, pages 254–259.
- Kelly, D., McLoone, S., and Dishongh, T. (2008a). A bluetooth-based minimum infrastructure home localisation system. In *In Proceedings of 5th IEEE International Symposium on Wireless Communication Systems (ISWCS 2008)*, pages 638–642.
- Kelly, D., McLoone, S., and Dishongh, T. (2008b). Experimental evaluation of a single access point bluetooth localisation system. In *Proceedings of the IET Irish Signals and Systems Conference 2008 (ISSC08)*, pages 254–259.
- Kelly, D., McLoone, S., and Dishongh, T. (2009b). Enabling affordable and efficiently deployed location based smart home systems. *Technology and Health Care special edition entitled Smart Environments: Technology to Support healthcare*, 17(3):221–235.
- Kelly, D., McLoone, S., Dishongh, T., McGrath, M., and Behan, J. (2008c). Single access point location tracking for in-home health monitoring. In *Proceedings of the 5th Workshop on Positioning, Navigation and Communication, 2008. WPNC '08.*, pages 23–29.
- Kelly, D., McLoone, S., Logan, B., and Dishongh, T (2008d). Single access point localisation for wearable wireless sensors. In *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society - EMBC2008*, pages 4443–4446.

-
- Khan, H.M., Olariu, S., and Eltoweissy, M. (2006). Efficient single-anchor localization in sensor networks. In *Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems, 2006. DSSNS 2006.*, pages 35 – 43.
- Kiefte, M. (1999). Matlab discriminant analysis toolbox ver. 0.3. [Internet]. Available From: <http://www.mathworks.com/matlabcentral/fileexchange/189-discrim> [Accessed 27th October 2009].
- Kilmartin, L., Ibrahim, R., Ambikairajah, E., and Celler, B. (2009). Optimising recognition rates for subject independent gait pattern classification. In *Proceedings of the IET Irish Signals and Systems Conference 2009 (ISSC09)*.
- Koch, J., Wettach, J., Bloch, E., and Berns, K. (2007). Indoor localisation of humans, objects, and mobile robots with rfid infrastructure. In *7th International Conference on Hybrid Intelligent Systems, 2007. HIS 2007.*, pages 271–276.
- Kotanen, A., Hannikainen, M., Leppakoski, H., and Hamalainen, T.D. (2003). Experiments on local positioning with bluetooth. In *Information Technology: Coding and Computing [Computers and Communications], 2003. Proceedings. ITCC 2003. International Conference on*, pages 297–303.
- Krumm, J., Cermak, G., and Horvitz, E. (2003). Rightspot: A novel sense of location for a smart personal object. In *Proceedings of Ubicomp 2003*, pages 36–43. Seattle, WA.
- Krumm, J. and Horvitz, E. (2004). Locadio: inferring motion and location from wi-fi signal strengths. In *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004.*, pages 4 – 13.
- Krumm, J., Williams, L., and Smith, G. (2002). Smartmovex on a graph - an inexpensive active badge tracker. In *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing*, pages 299–307. Springer-Verlag, London, UK. ISBN 3-540-44267-7.
- Kulyukin, V., Kutiyawala, A., LoPresti, E., Matthews, J., and Simpson, R. (2008). iwalker: Toward a rollator-mounted wayfinding system for the elderly. In *RFID, 2008 IEEE International Conference on*, pages 303–311.
- Kuncheva, L.I. (2004). *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience. ISBN 0471210781.
- Kupper, A. (2005). *Location Based Services - fundamentals and Operation*. John Wiley & Sons, Ltd.

-
- Ladd, A.M., Bekris, K.E., A.Rudys, G.Marceau, Kavvaki, L.E., and Wallach, D.S. (2002). Robotics-based location sensing using wireless ethernet. In *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 227–238. ACM, New York, NY, USA. ISBN 1-58113-486-X.
- Lamarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., and Schilit, B. (2005). Place lab: Device positioning using radio beacons in the wild. In *Pervasive 2005*. Munich, Germany.
- Lawrence, N., Seeger, M., and Herbrich, R. (2003). Fast sparse gaussian process methods: The informative vector machine. *Advances in Neural Information Processing Systems*, 15:609–616.
- Lee, L.-M. and J.-C.Lee (2006). A study on high-order hidden markov models and applications to speech recognition. In *Advances in Applied Artificial Intelligence*, pages 682–690.
- Liao, L., Fox, D., Hightower, J., Kautz, H., and Schulz, D. (2003). Voronoi tracking: location estimation using sparse and noisy sensor data. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 1, pages 723–728vol.1.
- Lightner, M.R. and Erdogmus, D. (2008). Signal processing challenges in cognitive assistive technology [exploratory dsp]. *Signal Processing Magazine, IEEE*, 25(5):103–108. ISSN 1053-5888.
- Lin, C.-C., Chiu, M.-J., Hsiao, C.-C., Lee, R.-G., and Tsai, Y.-S. (2006). Wireless health care service system for elderly with dementia. *IEEE Transactions on Information Technology in Biomedicine*, 10(4):696–704. ISSN 1089-7771.
- Loosli, G. (2008). Matlab SimpleSVM Toolbox. [Internet]. Available from: <http://sourceforge.net/projects/simplesvm> [Accessed 14 November 2008].
- Lorincz, K. and Welsh, M. (2006). Motetrack: a robust, decentralized approach to rf-based location tracking. *Personal and Ubiquitous Computing*, 11(6):489–503.
- Lundell, J., Hayes, T., Vurgun, S., Ozertem, U., Kimel, J., Kaye, J., Farzin, G, and Pavel, M. (2007). Continuous activity monitoring and intelligent contextual prompting to improve medication adherence. In *Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society - EMBC2007*, pages 6286 – 6289.
-

-
- Lundell, J., Kimel, J., Hayes, T., Dishongh, T., Kaye, J., and Pavel, M. (2006). Why elders forget to take their meds: A probe study to inform a smart reminding system. In *4th International Conference on Smart Homes and Health Telematics - ICOST2006*, pages 98 – 105.
- Madhavapeddy, A. and Tse, A. (2005). A study of bluetooth propagation using accurate indoor location mapping. In *Lecture Notes in Computer Science*, volume 3660, pages 105–122.
- Mantoro, T. and Johnson, C.W. (2005). nk-nearest neighbor algorithm for estimation of symbolic user location in pervasive computing environments. In *Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005*, pages 472–474.
- McKown, J.W. and Hamilton, Jr., R.L. (1991). Ray tracing as a design tool for radio networks. *IEEE Network*, 5(6):27–30. ISSN 0890-8044.
- Nawyn, J., Intille, S. S., and Larson, K. (2006). Embedding behavior modification strategies into consumer electronic devices. In P. Dourish and A. Friday, editors, *Proceedings of UbiComp 2006*, pages 297–314. Springer-Verlag, Berlin Heidelberg.
- Ni, L.M., Liu, Y., and Lau, A.P., Y.C. and Patil (2003). Landmarc: indoor location sensing using active rfid. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)*, pages 407–415.
- Niculescu, D. and Nath, Badri (2003). Ad hoc positioning system (aps) using aoa. In *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM 2003*, volume 3, pages 1734–1743 vol.3. ISSN 0743-166X.
- Nugent, C., Finlay, D., Davies, R., Paggetti, C., Tamburini, E., and Black, N. (2005). Can technology improve compliance to medicine? In *3rd International Conference on Smart Homes and Health Telematics - ICOST2005*, pages 65 – 72.
- Oberli, C. and Landau, D. (2008). Performance evaluation of uhf rfid technologies for real-time passenger tracking in intelligent public transportation systems. In *Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on*, pages 108–112.
- Ofstad, A., Nicholas, E., Szcodronski, R., and Choudhury, R.R. (2008). Aampl: accelerometer augmented mobile phone localization. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 13–18. ACM, New York, NY, USA. ISBN 978-1-60558-189-7.
-

-
- Orderud, F. (2005). Comparison of kalman filter estimation approaches for state space models with nonlinear measurements. In *Scandinavian Conference on Simulation and Modeling (SIMS05)*.
- Orooji, M. and Abolhassani, B. (2005). New method for estimation of mobile location based on signal attenuation and hata model signal prediction. *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 6025–6028.
- Otsason, V., Varshavsky, A., Lamarca, A., and de Lara, E. (2005). Accurate GSM indoor localization. In *UbiComp 2005: Ubiquitous Computing*, pages 141–158.
- Paalanen, P., Kamarainen, J.-K., Ilonen, J., and Kälviäinen, H. (2006). Feature representation and discrimination based on gaussian mixture model probability densities-practices and algorithms. *Pattern Recogn.*, 39(7):1346–1358. ISSN 0031-3203.
- Pandya, D., Jain, R., and Lupu, E. (2003). Indoor location estimation using multiple wireless technologies. In *14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003*, volume 3, pages 2208–2212.
- Patel, S., Lorincz, K., Hughes, R., Huggins, N., Growdon, J.H., Welsh, M., and Bonato, P. (2007). Analysis of feature space for monitoring persons with parkinson’s disease with application to a wireless wearable sensor system. In *Proceedings of the 29th Annual International Conference of the IEEE EMBS*, volume 1, pages 6291 – 6294.
- Pavel, M., Hayes, T., Tsay, I., Erdogmus, D., Paul, A., Larimer, N., Jimison, H., and Nutt, J. (2007). Continuous assessment of gait velocity in parkinson’s disease from unobtrusive measurements. In *Neural Engineering, 2007. CNE '07. 3rd International IEEE/EMBS Conference on*, pages 700–703.
- Pechac, P. and Klepal, M. (2001). Effective indoor propagation predictions. In *IEEE 54th Vehicular Technology Conference, 2001. VTC 2001 Fall. Proceedings of the*, volume 3, pages 1247–1250 vol.3.
- Polastre, J., Szewczyk, R., and Culler, D. (2005). Telos: enabling ultra-low power wireless research. In *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*, page 48. IEEE Press, Piscataway, NJ, USA. ISBN 0-7803-9202-7.
- Priyantha, N.B., Chakraborty, A., and Balakrishnan, H. (2000). The cricket location-support system. In *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43. ACM Press, New York, NY, USA. ISBN 1581131976.
-

-
- Quiñonero Candela, J. and Winther, O. (2003). Incremental gaussian processes. In *Advances in Neural Information Processing Systems 15*.
- QuietCare Systems (2009). QuietCare, from Living Independently. [Internet]. Available From: www.quietcaresystems.com [Accessed 30th September 2009].
- Rabiner, L.R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286. ISSN 0018-9219.
- Rappaport, T.S. (2002). *Wireless Communications - Principles and Practice*. Prentice Hall, 2nd edition.
- Rasmussen, C.E. and Williams, C.K.I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press. ISBN 026218253X.
- Ren, H., Meng, M.Q.-H., and Xu, L. (2007). Indoor patient position estimation using particle filtering and wireless body area networks. In *Proceedings of the 29th Annual International Conference of the IEEE EMBS*, pages 2277 – 2280.
- Reynolds, D.A. and Rose, R.C. (1995). Robust text-independent speaker identification using gaussian mixture speaker models. *Speech and Audio Processing, IEEE Transactions on*, 3(1):72–83. ISSN 1063-6676.
- Ristic, B., Arulampalam, S., and Gordon, N. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House.
- Rodas, J., Escudero, C.J., and Iglesia, D.I. (2008). Bayesian filtering for a bluetooth positioning system. In *Wireless Communication Systems. 2008. ISWCS '08. IEEE International Symposium on*, pages 633–637.
- Roos, T., Myllymaki, P., and Tirri, H. (2002a). A statistical modeling approach to location estimation. *Mobile Computing, IEEE Transactions on*, 1(1):59–69. ISSN 1536-1233.
- Roos, T., Myllymaki, P., Tirri, H., Misikangas, P., and Sievanen, J. (2002b). A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3):155 – 164.
- Ryynanen, M. and Klapuri, A. (2007). Automatic bass line transcription from streaming polyphonic audio. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–1437–IV–1440. ISSN 1520-6149.
- Savvides, A., C.-C.Han, and Strivastava, M.B. (2001). Dynamic fine-grained localization in ad-hoc networks of sensors. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 166–179. ACM, New York, NY, USA. ISBN 1-58113-422-3.
-

-
- Scanail, C., Carew, S., Barralon, P., Noury, N., Lyons, D., and Lyons, G. (2006). A review of approaches to mobility telemonitoring of the elderly in their living environment. *Annals of Biomedical Engineering*, 34(4):547–563. ISSN 0090-6964.
- Schwaighofer, A., Grigoras, M., Tresp, V., and Hoffmann, C. (2004). Gpps: A gaussian process positioning system for cellular networks. In *In NIPS*. MIT Press.
- Seidel, S.Y. and Rappaport, T.S. (1992). 914 mhz path loss prediction models for indoor wireless communications in multifloored buildings. *Antennas and Propagation, IEEE Transactions on*, 40(2):207–217. ISSN 0018-926X.
- Seshadri, V., Zaruba, G.V., and Huber, M. (2005). A bayesian sampling approach to indoor localization of wireless devices using received signal strength indication. In *Third IEEE International Conference on Pervasive Computing and Communications, 2005. PerCom 2005*, pages 75 – 84.
- Shi, G., Zoui, Y., Jin, Y., Zheng, Y., and Li, W.J. (2009). Multi-category human motion recognition based on mems inertial sensing data. In *Nano/Micro Engineered and Molecular Systems, 2009. NEMS 2009. 4th IEEE International Conference on*, pages 489–493.
- Srinivasan, K. and Levis, P. (2006). Rssi is under appreciated. In *Proc. of the Third Workshop on Embedded Networked Sensors, EmNets 2006*, pages 93 – 97.
- Sugano, M., Kawazoe, T., Ohta, Y., and Murata, M. (2006). Indoor localization system using rssi measurement of wireless sensor network based on zigbee standard. In *Proceedings of Wireless Sensor networks 2006*, pages 1 – 6.
- Tapia, E.M., Intille, S., and Larson, K. (2004). Activity recognition in the home setting using simple and ubiquitous sensors. In *PERVASIVE 2004*, pages 1–17.
- Tarrio, P., Bernardos, A.M., Besada, J.A., and Casar, J.R. (2008). A new positioning technique for rss-based localization based on a weighted least squares estimator. In *IEEE International Symposium on Wireless Communication Systems. 2008. ISWCS '08*, pages 633–637.
- Tipping, M. (2000). The relevance vector machine. In *Advances in Neural Information Processing Systems*.
- Tipping, M. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211 – 244.
- Tipping, M. (2002). Sparse bayesian models (& the RVM). [Internet]. Available From: <http://www.miketipping.com/index.php?page=rvm>.
-

-
- Tipping, M. and Faul, A. (2003). Fast marginal likelihood maximisation for sparse bayesian models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, pages 3–6.
- Varshavsky, A., Chen, M., de Lara, E., Froehlich, J., Haehnel, D., Hightower, J., LaMarca, A., Potter, F., Sohn, T., Tang, K., and Smith, I. (2006). Are GSM phones THE solution for localization? In *7th IEEE Workshop on Mobile Computing Systems and Applications*, pages 20–28.
- Wang, N., Ambikairajah, E., Redmond, S., Celler, B., and Lovell, N. (2009). An investigation of the impact of gait segmentation on accelerometry-based inclined terrain classification. In *Proceedings of the IET Irish Signals and Systems Conference 2009 (ISSC09)*.
- Want, R., Hopper, A., Faloutsos, V., and Gibbons, J. (1992). The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102. ISSN 1046-8188.
- Want, R., Schilit, B., Adams, N., Gold, R., Petersen, K., Ellis, J., Goldberg, D., and Weiser, M. (1995). The parctab ubiquitous computing experiment. Technical Report CSL-95-1, Xerox Palo Alto Research Center. Also appears In *Mobile Computing*, H.F. Korth and T Imielinski, eds., Kluwer Academic Press, 1996.
- Ward, A., Jones, A., and Hopper, A. (1997). A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47. ISSN 1070-9916.
- Welch, G. and Bishop, G. (1995). An introduction to the kalman filter. Technical report, Department of Computer Science, University of North Carolina, Chapel Hill, NC, USA.
- Widyawan, Klepal, M., and Beauregard, S. (2008). A backtracking particle filter for fusing building plans with pdr displacement estimates. In *5th Workshop on Positioning, Navigation and Communication, 2008. WPNC '08*. Submitted for publication in WPNC.
- Widyawan, Klepal, M., and Pesch, D. (2007a). A bayesian approach for rf-based indoor localisation. In *IEEE International Symposium on Wireless Communication Systems 2007*, pages 133 – 137. Trodheim, Norway.
- Widyawan, Klepal, M., and Pesch, D. (2007b). Influence of predicted and measured fingerprint on the accuracy of rssi-based indoor location systems. In *4th Workshop on Positioning, Navigation and Communication, 2007. WPNC '07*, pages 145–151.
- Williams, C.K.I. and Barber, D. (1998). Bayesian classification with gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351.
-

- Youssef, M.A., A.Agrawala, and Shankar, A.U. (2003). Wlan location determination via clustering and probability distributions. In *PERCOM '03: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, page 143. IEEE Computer Society, Washington, DC, USA. ISBN 0-7695-1893-1.
- Zàruba, G. V., Huber, M., Kamangar, F. A., and Chlamtac, I. (2007). Indoor location tracking using rssi readings from a single wi-fi access point. *Wireless Networks*, 13(2):221–235. ISSN 1022-0038.