

Road Features Extraction Using Terrestrial Mobile Laser Scanning System

by

Pankaj Kumar



NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

A thesis presented in fulfilment of the requirements for the Degree of
Doctor of Philosophy

Supervisors: Dr. Timothy McCarthy, Dr. Conor P. McElhinney

National Centre for Geocomputation

Faculty of Science

National University of Ireland Maynooth

Maynooth, Co. Kildare, Ireland

October, 2012

To My Wife, Parents & GrandParents.

DECLARATION

This thesis has not been submitted in whole or in part to this or any other university for any other degree and is, except where otherwise stated, the original work of the author.

Signed: -----

Pankaj Kumar

ACKNOWLEDGMENTS

I would like to thank my supervisor, Timothy McCarthy, for providing me this opportunity and keeping a faith from beginning to end of this Ph.D. research. A profound gratitude to my supervisor, Conor P. McElhinney, for his continuous guidance and intellectual support, without which it was difficult to complete this thesis. His suggestions and positive remarks provided a new direction to my thinking towards tackling the research problems. I would thank, Paul Lewis, for taking his time out to read my thesis draft and providing valuable corrections to it. Moreover, a weekly meeting of our Mobile Mapping Group helped in terms of providing a platform, where we could have a wider discussion on various research issues.

My sincere thanks to Jan Rigby, who supported and motivated me to continue my Ph.D. research study during my difficult period, especially in the first year. To Martin Charlton, for providing me valuable suggestions and support during the thesis.

I would like to thank my all other lab mates especially Binbin, Conor C., Ishwari, Cathal, Fergal, Carson, Burchin, Ambra, Zoe, Seamus who made my stay in Ireland pleasant by creating a helpful and friendly environment. This list of acknowledgement would be incomplete without mentioning the administrative and technical staff of NCG who provided me a valuable support. I would also thank to all those persons, who directly or indirectly, helped me to reach this stage of life.

I would acknowledge IRCSET Enterprise, PMS and StratAG as the research presented in this thesis was conducted with their financial support.

Contents

1	Introduction	1
1.1	Mobile Mapping	3
1.1.1	Mobile Mapping with Digital Cameras	4
1.1.2	Mobile Mapping with Laser Scanners	8
1.2	Road Safety Applications of Mobile Mapping	11
1.3	Road Features Extraction	15
1.3.1	Road Edges	17
1.3.2	Road Markings	19
1.3.3	Road Roughness	21
1.3.4	Snake Curves	27
1.4	Contributions of the Thesis	31
1.5	Outline of the Thesis	33
2	Terrestrial Mobile Mapping Technology	37
2.1	Components of Mobile Mapping Technology	38
2.1.1	Imaging System	38
2.1.2	Laser Scanning System	41
2.1.3	Navigation System	46
2.1.4	Data Acquisition System	50

2.2	XP-1 System	52
2.3	Conclusion	56
3	Approaches For Extracting Road Features	58
3.1	Road Edge Extraction	59
3.1.1	Hierarchical Thresholding	60
3.1.2	Canny Edge Detection	64
3.1.3	Active Contour Models	66
3.2	Road Marking Extraction	78
3.3	Road Roughness Estimation	81
3.3.1	RANSAC	81
3.4	Conclusion	84
4	Road Edge Extraction	87
4.1	Algorithm	88
4.1.1	Terrain Pyramids Generation	88
4.1.2	2D Raster Surfaces Generation	90
4.1.3	Snake Energy Estimation	92
4.1.4	Snake Curve Initialisation	94
4.1.5	Final Snake Curve	97
4.1.6	Batch Processing	97
4.1.7	3D Road Edges	99
4.2	Validation Algorithm	101
4.3	Automation Analysis	103
4.3.1	2D Raster Surfaces Generation	104
4.3.2	Optimal Cell Size	106
4.3.3	Optimal Road Length	119

4.3.4	Optimal Internal Energy Parameters	122
4.3.5	Optimal External Energy Parameters	124
4.4	Experimentation	129
4.4.1	Manual Processing	130
4.4.2	Automated Processing	141
4.4.3	Results Validation	148
4.5	Discussion	161
5	Road Marking Extraction	167
5.1	Algorithm	168
5.1.1	Road Surface Estimation	168
5.1.2	2D Raster Surface Generation	170
5.1.3	Range Dependent Thresholding	171
5.1.4	Morphological Operations	173
5.1.5	3D Road Markings	178
5.2	Automation Analysis	178
5.2.1	Optimal Cell Size	179
5.2.2	Range Dependent Threshold	184
5.3	Experimentation	187
5.3.1	Broken and Continuous Line Markings	187
5.3.2	Word Markings	190
5.3.3	Hatch Markings	195
5.3.4	Arrow Markings	196
5.3.5	Pedestrian Crossing Markings	198
5.3.6	Zig-Zag Markings	202
5.3.7	Results Validation	204
5.4	Discussion	204

6	Road Roughness Estimation	208
6.1	Algorithm	209
6.1.1	Road Surface Estimation	209
6.1.2	Data Rotation	211
6.1.3	Surface Grid	211
6.1.4	Road Roughness	213
6.2	Experimentation	214
6.2.1	Urban Roads	215
6.2.2	National Primary Roads	218
6.2.3	Results	219
6.3	Discussion	231
7	Conclusion	234
7.1	Road Edge Extraction	234
7.2	Road Marking Extraction	239
7.3	Road Roughness Estimation	241
8	Future Work	243
8.1	Road Edge Extraction	243
8.2	Road Marking Extraction	246
8.3	Road Roughness Estimation	247

List of Figures

1.1	Road accident statistics in Europe [CAR07].	12
1.2	Main causes of road accidents [TTM+79].	13
1.3	Road edges: (a) kerbs in an urban road section and (b) grass-soil in a rural road section.	18
1.4	Road markings: (a) continuous line, triangle and (b) arrow.	20
1.5	Roughness along the road surface.	22
1.6	Quarter car simulator.	23
1.7	Measurement of elevation points at a 1.5 m interval along the longitudinal profile of the road surface.	24
1.8	Snake curve and MRI scan [LS95].	28
2.1	XP-1's digital camera image of a road section.	39
2.2	XP-1's LiDAR data of a road section.	41
2.3	Principle of (a) multiple returns from targets and returned (b) echo pulses.	42
2.4	TOF method	44
2.5	Phase shift method.	44
2.6	Intensity and pulse width of reflected pulse.	45
2.7	XP-1's navigation data along a road section.	46
2.8	Georeferencing process of MMS.	47

2.9	Process for estimating navigation parameters of the mobile mapping vehicle using a GNSS base station and a navigation system.	50
2.10	Data acquisition system.	51
2.11	XP-1 MMS with an inset picture of the laser scanning and navigation system mounted on the roof rack.	52
2.12	XP-1's navigation data plotted over a Google earth image, highlighted in yellow, with an inset picture of its small section (Image courtesy: Google Earth).	53
2.13	Inclined laser scanner along with GPS antenna and LandINS GPS/INS mounted on the XP-1's roof rack.	54
2.14	XP-1's (a) imaging and (b) LiDAR data of a road section.	56
3.1	Segmentation approaches in the road edge extraction algorithm.	59
3.2	2D Gaussian distribution with mean $(0, 0)$ and $\sigma = 1$ [FPWW00].	61
3.3	Convolution kernel that produces a discrete approximation to the 2D Gaussian distribution function with mask size=5 and $\sigma = 1$ [FPWW00].	61
3.4	Hierarchical thresholding: (a) input image, (b) thresholded object cells in the lowest resolution image, at level n and (c) thresholded neighbourhood object cells in the next lowest resolution image, at level $n - 1$ [SHB08].	62
3.5	Hierarchical thresholding: (a) original LiDAR data, (b) input slope image and (b) estimated objects.	63
3.6	Sobel convolution kernels, which are used to find the gradients of an image along its (a) X and (b) Y axis.	64

3.7	Canny edge detection: (a) input hierarchical thresholded objects and (b) their estimated boundaries.	66
3.8	Initial snake curve in the form of parametric ellipse.	67
3.9	Parametric ellipse snake curve initialised on a gradient image of road object boundaries.	69
3.10	Traditional parametric active contour model applied to an object with a concave boundary [XP98].	70
3.11	Balloon external energy added to the snake curve.	71
3.12	Parametric ellipse snake curve initialised on GVF image of road object boundaries.	74
3.13	GVF active contour model applied to an object with a concave boundary [XP98].	75
3.14	Geometric active contour model.	76
3.15	Balloon energy pushes the snake curve and GVF energy attracts the snake curve toward the object boundaries.	78
3.16	Final position of the snake curve.	79
3.17	Range dependent thresholding in the road marking extraction algorithm.	79
3.18	Range dependent thresholding: (a) input 2D intensity raster surface and (b) extracted road markings.	80
3.19	RANSAC in the road roughness estimation algorithm.	81
3.20	RANSAC surface grid fitted to the LiDAR points, with an inset picture of a magnified portion.	84
4.1	Road edge extraction algorithm.	89
4.2	Terrain pyramids generated from the LiDAR attributes: (a) elevation, (b) reflectance and (c) pulse width.	91

4.3	Raster cell laid over the Voronoi polygons constituting the thinned LiDAR points in the natural neighbourhood interpolation.	92
4.4	2D raster surfaces generated from their respective terrain pyramids using natural neighbourhood interpolation: (a) slope, (b) reflectance and (c) pulse width.	93
4.5	Snake curve initialisation in (a) parametric ellipse form and (b) centre of the road estimation.	95
4.6	ϕ angle is calculated from θ , which is an average heading angle of the mobile van along the road section under investigation that can lie in between (a) 0° and 90° , (b) 90° and 180° , (c) 180° and 270° and (d) 270° and 360°	96
4.7	Snake curve positions: (a) initial, (b) iterative and (c) final.	97
4.8	Intersection point in between two overlapped snake curves.	98
4.9	Three processed road sections with (a) first intersection points highlighted, (b) second intersection points highlighted and (c) final left and right road edges.	100
4.10	3D left and right road edges.	101
4.11	Road edge validation algorithm.	102
4.12	Slope surface estimated from the elevation raster surface generated using linear interpolation applied to (a) full resolution, (b) first level terrain pyramid and using natural neighbourhood interpolation applied to (a) full resolution, (b) first level terrain pyramid.	105

4.13	Reflectance raster surface generated from (a) full resolution terrain pyramid using linear interpolation and (b) first level terrain pyramid using natural neighbourhood interpolation. Pulse width raster surface generated from (c) full resolution terrain pyramid using linear interpolation and (b) first level terrain pyramid using natural neighbourhood interpolation.	106
4.14	Final position of the snake curve over the slope raster surface with (a) 0.02 m ² , (b) 0.04 m ² , (c) 0.06 m ² , (d) 0.08 m ² , (e) 0.1 m ² and (f) 0.2 m ² cell size.	109
4.15	Plot of completeness obtained from the snake curve in the six test cases of an optimal cell size estimation analysis.	110
4.16	Plot of the time taken by the snake curve to move from its initial position to the final position in the six test cases of an optimal cell size estimation analysis.	111
4.17	3D LiDAR data: (a) 6 m ×10 m, (b) 4.5 m ×5 m and (c) 3 m ×4 m.	112
4.18	LiDAR point data along the (a) left and (b) right side of the road section.	113
4.19	Road edge digitised from the LiDAR data: (a) 3.2 m ×3.3 m and (b) 1.8 m ×2 m.	113
4.20	Box plots for the (a) 2D and (b) 3D accuracy of the extracted left edges in the six test cases of an optimal cell size estimation analysis.	114
4.21	Box plot for the (a) 2D and (b) 3D accuracy of the extracted right edges in the six test cases of an optimal cell size estimation analysis.	115

4.22	Missed points as circled in blue in the section of LiDAR data.	120
4.23	Final position of the snake curve over the slope raster surface with (a) 10 m, (b) 20 m and (c) 30 m road length.	121
4.24	Final position of the snake curve over the slope raster surface in the (a) first, (b) second, (c) third, (d) fourth and (e) fifth case of the internal energy parameter analysis.	123
4.25	Final position of the snake curve over the slope raster surface in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth, (f) sixth, (g) seventh, (h) eighth and (i) ninth case of the external energy parameter analysis.	126
4.26	Hierarchical thresholding applied to the reflectance raster surface provided the objects with road marking cells near the road edge points as circled in blue.	127
4.27	Hierarchical thresholding applied to the slope raster surface was not able to remove the noisy cells in between the road edge points.	129
4.28	Final position of the snake curve over the slope raster surface with (a) $\kappa_4 = 1$ and (b) $\kappa_4 = 3$ parameters.	130
4.29	Digital image of the rural road section consisting of grass-soil edges (Geographic location: 53°34'28.07"N 7°10'13.76"W). . .	131
4.30	Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the rural road.	133
4.31	Extracted 3D left and right edges in the rural road section obtained through the manual selection of parameters.	134

4.32	Digital image of the urban road section consisting of kerb edges (Geographic location: 53°36'37.42"N 7°5'41.00"W).	135
4.33	Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the urban road.	136
4.34	Extracted 3D left and right edges in the urban road section obtained through the manual selection of parameters.	137
4.35	Digital image of the national primary road section consist- ing of grass-soil edges with shoulders (Geographic location: 53°38'8.80"N 7°29'11.06"W).	138
4.36	Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the national primary road.	139
4.37	Extracted 3D left and right edges in the national primary road section obtained through the manual selection of parameters.	140
4.38	Automated final position of the snake curve over the slope raster surface for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the rural road.	142
4.39	Automatically extracted 3D left and right edges in the rural road section.	143
4.40	Automated final position of the snake curve over the slope raster surface for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the urban road.	144

4.41	Automatically extracted 3D left and right edges in the urban road section.	145
4.42	Automatic final position of the snake curve for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the national primary road.	146
4.43	Automatically extracted 3D left and right edges in the national primary road section.	147
4.44	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the rural road section.	149
4.45	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the rural road section.	150
4.46	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the urban road section.	154
4.47	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the urban road section.	155
4.48	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.	158
4.49	Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the national primary road section.	159
5.1	Road marking extraction algorithm.	169

5.2	Snake curve is (a) laid over the LiDAR points to (b) estimate the road surface.	170
5.3	2D raster surfaces generated from the LiDAR data: (a) intensity and (b) range.	171
5.4	Navigation data is used to select a range value to apply multiple threshold values to the intensity raster surface.	172
5.5	Side view of the non-planar road surface.	173
5.6	Road markings extracted from the intensity raster surface using range dependent thresholding.	174
5.7	Structuring elements: (a) diamond shaped with radius = 1, (b) linear shaped with length = 3 and angle, $\phi' = 45^\circ$ and (c) linear shaped with length = 5 and angle, $\phi' = 90^\circ$	174
5.8	Estimation of ϕ' angle of the linear shaped structuring element from the average heading angle of the mobile van, θ , along the road section that can lie in between (a) 0° and 90° , (b) 90° and 180° , (c) 180° and 270° and (d) 270° and 360°	176
5.9	Dilation operation: (a) input binary image with an inset picture of their road marking cells and (b) dilated image with an inset picture of their dilated road marking cells.	177
5.10	Noise removal process: (a) input dilated image and (b) noise cells removed from it.	178
5.11	Erosion operation: (a) input dilated image with an inset picture of their road marking cells and (b) eroded image with an inset picture of their eroded road marking cells.	179
5.12	Extracted road markings (a) before and (b) after applying the morphological operations.	180

5.13 LiDAR points (a) belonging to the road surface and the extracted (b) 3D road markings.	180
5.14 Road markings extracted from the intensity image with (a) 0.01 m ² , (b) 0.04 m ² , (c) 0.06 m ² , (d) 0.08 m ² and (e) 0.1 m ² cell size.	182
5.15 Range dependent thresholding applied to the intensity raster surface in the road section with (a) narrower and (b) greater width.	185
5.16 Road markings extracted using the (a) lower, (b) higher and (c) optimal threshold value applied to the intensity raster surface.	186
5.17 Digital image of (a) rural and (b) urban road section containing both broken and continuous line markings (Geographic locations: (a) 53°34'28.07" N 7°10'13.76" W and (b) 53°36'33.43" N 7°5'46.96" W).	187
5.18 2D broken and continuous line markings extracted from the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section of the rural road.	189
5.19 2D broken and continuous line markings extracted from the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section of the urban road.	190
5.20 Road markings extraction: (a) original LiDAR data and (b) extracted 3D broken and continuous line markings of the rural road section.	191

5.21	Road markings extraction: (a) original LiDAR data and (b) extracted 3D broken and continuous line markings of the urban road section.	192
5.22	Digital image of the rural road section containing word markings along with broken and continuous line markings (Geographic location: $53^{\circ}34'50.546''$ N $7^{\circ}8'57.62''$ W).	193
5.23	Output snake curve in the rural road section containing word markings along with broken and continuous line markings.	194
5.24	Road markings extraction: (a) original LiDAR data and (b) extracted 3D word, broken and continuous line markings of the rural road section.	194
5.25	Digital image of the national primary road section containing hatch and broken line markings (Geographic location: $53^{\circ}33'49.878''$ N $7^{\circ}21'24.148''$ W).	195
5.26	Output snake curve in the national primary road section containing hatch and broken line markings.	196
5.27	Road markings extraction: (a) original LiDAR data and (b) extracted 3D hatch and broken line markings of the national primary road section.	197
5.28	Digital image of the national primary road section containing arrow and broken line markings (Geographic location: $53^{\circ}35'30.558''$ N $7^{\circ}22'28.428''$ W).	198
5.29	Output snake curve in the national primary road section containing arrow and broken line markings.	199

5.30	Road markings extraction: (a) original LiDAR data and (b) extracted 3D arrow and broken line markings of the national primary road section.	199
5.31	Digital image of the urban road section containing pedestrian crossing, broken transverse and zig-zag markings (Geographic location: $53^{\circ}36'43.721''\text{N}$ $7^{\circ}5'32.871''\text{W}$).	200
5.32	Output snake curve in the urban road section containing pedestrian crossing, broken transverse and zig-zag markings.	201
5.33	Road markings extraction: (a) original LiDAR data and (b) extracted 3D pedestrian crossing, broken transverse line and zig-zag markings of the urban road section.	201
5.34	Digital image of the urban road section containing zig-zag markings (Geographic location: $53^{\circ}39'19.225''\text{N}$ $7^{\circ}31'9.792''\text{W}$).	202
5.35	Output snake curve in the urban road section containing zig-zag markings.	203
5.36	Road markings extraction: (a) original LiDAR data and (b) extracted 3D zig-zag markings of the urban road section.	203
5.37	2D hatch and broken line markings of the national primary road section with the noise (a) removed only once before applying the erosion operation and (b) removed twice, before and after applying the erosion operation.	206
6.1	Road roughness estimation algorithm.	210
6.2	Snake curve is (a) laid over the LiDAR data and (b) the points belonging to the road surface are identified.	211

6.3	Input LiDAR and navigation data in the (a) 3D, (b) 2D plane and rotated LiDAR and navigation data in the (c) 3D, (d) 2D plane.	212
6.4	Surface grid is fitted to the LiDAR points that belong to (a) whole road surface and (b) the left side of the road surface. . .	213
6.5	RANSAC surface grid fitted to the LiDAR points in the (c) 3D and (d) 2D plane.	214
6.6	Inversely rotated surface grid and LiDAR points along the navigation track of the mobile van in the (c) 3D and (d) 2D plane.	215
6.7	Digital image of (a) first, (b) second and (c) third section of urban road (Geographic locations: (a) $53^{\circ}36'33.68''N$ $7^{\circ}5'46.39''W$, (b) $53^{\circ}36'36.587''N$ $7^{\circ}5'41.671''W$ and (c) $53^{\circ}36'40.584''N$ $7^{\circ}5'37.789''W$).216	
6.8	Output snake curve in the (a) second and (b) third section of urban road.	217
6.9	Duplicate points were removed in the first LiDAR section to provide non-overlapped sections.	218
6.10	Surface grids and LiDAR points along the navigation track in the (a) first, (b) second and (c) third section of urban road. . .	220
6.11	Digital image of the national primary road section (Geographic locations: $53^{\circ}38'14.407''N$ $7^{\circ}29'24.622''W$).	221
6.12	Inversely rotated surface grids and LiDAR points along the navigation track in the national primary road section.	222
6.13	Road surface deviation maps for the first urban road section: (a) 3D and (b) 2D.	223

6.14	Road surface deviation maps for the second urban road section: (a) 3D and (b) 2D.	224
6.15	Road surface deviation maps for the third urban road section: (a) 3D and (b) 2D.	225
6.16	Road surface deviation maps for the national primary road section: (a) 3D and (b) 2D.	226
6.17	Plot of the LiDAR points and the surface grid points along the navigation track in (a) the first and (b) the second urban road section.	227
6.18	Plot of the LiDAR points and the surface grid points along the navigation track in (a) the third urban and (b) the national primary road section.	228
6.19	Box plot of the standard deviation of the elevation residual points along the navigation track in (a) the first and (b) the second urban road section.	229
6.20	Box plot of the standard deviation of the elevation residual points along the navigation track in (a) the third urban and (b) the national primary road section.	230
8.1	Fitted surface grid with an inset picture of its skewed cells. . .	248

List of Tables

1.1	Examples of terrestrial MMSs.	6
2.1	IXSEA LandINS navigation system specifications.	53
2.2	Riegl VQ-250 specifications.	55
4.1	Global minimum and maximum values of the LiDAR attributes.	107
4.2	Different parameters used in the six test cases of an optimal cell size estimation analysis.	108
4.3	Statistical analysis of the 2D accuracy of the left edges along with completeness and time in the six test cases.	116
4.4	Statistical analysis of the 3D accuracy of the left edges along with completeness and time in the six test cases.	116
4.5	Statistical analysis of the 2D accuracy of the right edges along with completeness and time in the six test cases.	117
4.6	Statistical analysis of the 3D accuracy of the right edges along with completeness and time in the six test cases.	117
4.7	Different parameters used in the three test cases of an optimal road length estimation analysis.	120
4.8	Internal energy weight parameters used in the five test cases. .	122
4.9	External energy weight parameters used in the nine test cases.	125

4.10	LiDAR point density over the left and right sides of different road sections, acquired with the XP-1 system.	128
4.11	ϕ angle calculated from θ , average heading angle in each navigation section in the rural road.	132
4.12	ϕ angle calculated from θ , average heading angle in each navigation section in the urban road.	135
4.13	ϕ angle calculated from θ , average heading angle in each navigation section in the national primary road.	138
4.14	Hierarchical threshold parameters selected empirically for each road section.	141
4.15	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the rural road section.	151
4.16	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted right edges in the rural road section.	151
4.17	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the urban road section.	153
4.18	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted right edges in the urban road section.	156
4.19	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.	157

4.20	Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.	160
5.1	Maximum and minimum values of the LiDAR intensity and range attributes.	181
5.2	Length and average width values of the extracted road markings in the five cases.	183
5.3	ϕ' angle calculated from θ angle in each navigation section of rural road section.	188
5.4	ϕ' angle calculated from θ angle in each navigation section of urban road section.	188
6.1	ϕ angle calculated from θ , average heading angle in the navigation sections of second and third urban road section.	217
6.2	t and ϵ parameters used for fitting the surface grid to each LiDAR section of the urban road.	219
6.3	t and ϵ parameters used for fitting the surface grid to each LiDAR section of the national primary road.	219
6.4	Statistical analysis of the standard deviation of the elevation residual points along the navigation track in the first, second, third urban and the national primary road section.	231

ABSTRACT

In this thesis, we present the experimental research and key contributions we have made in the field of road feature extraction from LiDAR data. We detail the development of three automated algorithms for the extraction of road features from terrestrial mobile LiDAR data. LiDAR data is a rich source of 3D geo-referenced information whose volume and scale have inhibited the development of automated algorithms. Automated feature extraction algorithms enable the wider geospatial industry to transition from traditional road feature surveying approaches to terrestrial mobile laser scanning technologies.

Our first contribution to this field is an automated road edge extraction algorithm which can be applied to LiDAR data and navigation information acquired by mobile survey vehicles. This novel algorithm relies on the combination of thresholding and a parametric active contour model to precisely extract road edges. We describe an automated validation algorithm we developed to determine the accuracy of our road edge extraction algorithm.

Using the extracted road edges, we are able to accurately extract the road surface from the LiDAR data. This enables us to develop an efficient automated road marking extraction algorithm which is our second contribution. Through the thresholding of the intensity values of road surface LiDAR points, we can extract the road marking LiDAR points. The third contribution of this thesis is the development of an automated road roughness estimation algorithm which is also dependent on the accurate detection of road surface LiDAR points. We fit a surface grid to the LiDAR points representing an ideal road surface and measure the elevation difference between this surface and the actual LiDAR points to compute the surface deviation

along a track representing a vehicle wheel.

We automated these algorithms through exhaustive examination of optimal parameters and methods for their implementation. To verify these novel algorithms, we tested them on varying types of road sections representing rural, urban and national primary road sections. The research work carried out in the course of this thesis provides valuable insights as well as a prototype road feature extraction tool-set, for both national road authorities and survey companies. These findings and knowledge contribute to a more rapid, cost-effective and comprehensive approach to surveying road networks which, in turn, enables a more efficient, comfortable and safer journey for all road users.

ACRONYMS

C/A	Coarse Acquisition
CCD	Charge Coupled Device
CDSS	Car Driven Survey System
CMOS	Complementary Metal Oxide Semiconductor
CW	Continuous Wave
DGPS	Differential Global Positioning System
DIA	Direct Inertial Aiding
DLM	Digital Landscape Model
DMI	Distance Measuring Instrument
DSM	Digital Surface Model
DTM	Digital Terrain Model
EU	European Union
FOGs	Fibre Optic Gyroscopes
FOV	Field Of View
GCPs	Ground Control Points
GDP	Gross Domestic Product
GIS	Geographic Information System
GNP	Gross National Product
GNSS	Global Navigation Satellite System
GVF	Gradient Vector Flow
IMC	Immersive Media Cooperation
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IO	Interior Orientation

IRI	International Roughness Index
KISS	KInematic Surveying System
LAN	Local Area Network
LED	Light Emitting Diode
LiDAR	Light Detection And Ranging
LSB	Least Square B-spline
MRI	Magnetic Resonance Imaging
MMS	Mobile Mapping System
NDSM	Normalised Digital Surface Model
NDVI	Normalised Differential Vegetation Index
NRA	National Roads Authority
NSM	Network Safety Management
NUIM	National University of Ireland Maynooth
PPK	Post Processing Kinematic
PPS	Pulse Per Second
QCS	Quarter Car Simulator
RANSAC	RANdom SAmples Consensus
RINEX	Receiver INdependent EXchange
RLG	Ring Laser Gyro
RMS	Root Mean Square
RSA	Road Safety Audit
RSI	Road Safety Inspection
RTRRMs	Response Type Road Roughness Meters
RTK	Real Time Kinematic
TIN	Triangulated Irregular Network
TOF	Time OF Flight

UHF	Ultra High Frequency
VISAT	Video Inertial SATellite
WHO	World Health Organization
XP-1	eXperimental Platform-1

Chapter 1

Introduction¹

The demand for accurate 3D mapping of natural environments and man-made features has increased due to the spatial detail required by scientists, engineers and planners [BMS06]. Light Detection And Ranging (LiDAR) is a relatively recent technology, enabling 3D modelling of real world environment by measuring the time of return of an emitted light pulses. The information obtained through laser scanning systems, which use LiDAR technology, have application in road safety, urban planning, flood plain, glacier and avalanche mapping, bathymetry, geomorphology, forest survey, bridge and transmission line detection [Bur02]. Laser scanning systems enable the acquisition of an accurately georeferenced set of dense LiDAR point cloud data [PT08]. Other benefit of this type of system is the high level of automation during data capture and the ability of this system to acquire data beneath tree's canopy.

Laser scanning systems are used to acquire LiDAR data from aerial and terrestrial platforms. The data acquired from these systems differs in terms of its intrinsic accuracy and resolution for a variety of reasons but primarily

¹Throughout the thesis, the terms 'we' and 'our' are used to describe the doctorate research carried out solely by the author of this thesis.

due to the distance of the scanner to the target objects [ROPV09]. In recent years, the use of laser scanners onboard terrestrial based moving vehicles has increased for the collection of high quality 3D data. The applicability of these terrestrial mobile laser scanning systems continue to prove their worth in route corridor mapping due to the rapid, continuous and cost effective 3D data acquisition capability compared with static terrestrial laser scanning systems [HPKH08, BMS06]. LiDAR data records a number of attributes including elevation, intensity, pulse width, range and multiple echo information, all of which can be used for extracting road features. The volume of data produced by a terrestrial mobile laser scanning system such as Riegl VQ-250 is large, generating 300,000 points per second resulting in approximately 20 GB of data per hour. However, manual processing of LiDAR data for road features extraction is very time consuming. These sensor characteristics, potential applications and challenges provide the underlying motivation for this thesis.

We present three key contributions we have made to the field of automated road feature extraction from LiDAR data. Correct identification of the road boundaries is essential in order to obtain a precise estimation of road geometry and associated features. Our first contribution is the development of an automated algorithm for extracting road edges from terrestrial mobile LiDAR data. A priori knowledge of the road boundaries and associated surface in the LiDAR data facilitates efficient extraction of the road markings and roughness. The second contribution deals with the development of an automated algorithm for extracting road markings from terrestrial mobile LiDAR data. The third contribution focuses on the development of an automated algorithm for estimating road roughness from terrestrial mobile

LiDAR data.

In this chapter, we introduce the reader to terrestrial Mobile Mapping Systems (MMSs). In Section 1.1, we describe terrestrial MMSs and their ability to provide georeferenced spatial data. We review various MMSs comprising digital cameras, navigation sensors and laser scanners over the past two decades. In Section 1.2, the importance of road safety and its linkage to road geometry and road features is discussed. Terrestrial MMSs can be used to acquire 3D information about the road environment that can, in turn, assist decision makers in identifying safety risk elements along road networks. We discuss an application of terrestrial MMSs in road safety. In Section 1.3, we review various methods developed for extracting road features from LiDAR data. We investigate methods based on snake curves that has been developed for segmentation. Following the review, we identify the research limitations which have been addressed by doctorate research. Section 1.4 deals with the key contributions of this thesis to the field of road feature extraction from LiDAR data. Finally, an outline of the thesis is presented in Section 1.5.

1.1 Mobile Mapping

Mobile mapping refers to a means of collecting geospatial data using mapping and navigation sensors that are mounted rigidly onboard a mobile platform [TL07]. The concept of mobile mapping dates back to early 1990's and since then has been primarily driven by the advances in kinematic positioning, machine vision, laser scanning systems, data fusion and spatial information technologies. The mobile platform term relates to the transportation mode of the MMS which can be land-based (car, train), air-borne (aircraft) or

marine (ship, submarine) [Nov93]. Mapping sensors can consist of imaging and laser scanning system while the navigation system is based on integrated Global Navigation Satellite System (GNSS) and Inertial Navigation System (INS). The navigation system may also be complemented with dead reckoning sensors such as a Distance Measuring Instrument (DMI), an odometer or a digital compass.

The effectiveness of mobile mapping lies in its ability to directly georeference the acquired spatial and spectral data with the recorded navigation data within the global coordinate frame. The direct georeferencing provided by MMSs removes the need for Ground Control Points (GCPs) or any other external geographic referencing system. This is different to some other spatial data acquisition systems where a suitable number of well spread GCPs are required from a field survey prior to or after data acquisition [EES02]. However, GCPs can be used to increase the data accuracy in MMSs where higher levels of accuracy are required for some projects. In the following sections, we review various terrestrial MMSs developed using digital cameras and laser scanners.

1.1.1 Mobile Mapping with Digital Cameras

Terrestrial MMSs have been actively researched and developed over the past two decades [BMS06]. In their initial development phase, the MMSs were developed based on GNSS/INS integration and digital cameras which were used for the acquisition of road inventories and surrounding route corridor environment. Some examples of terrestrial MMSs with digital cameras have been listed in Table 1.1. The first terrestrial MMS, GPSVan, was developed by the Centre of Mapping at The Ohio State University in the early 1990's

Name & Developers	Navigation Sensors	Mapping Sensors	Position Accuracy	Primary Application
GPSVan, Ohio State University	GPS, gyro based INS, odometer	Digital stereo-vision system, colour video cameras	1 to 3 m	Digital road maps and highway inventories
VISAT, University of Calgary	DGPS, strap down INS, DMI	eight digital cameras, one colour video camera	0.3 m	Highway surveying system
KISS, University of the Federal Armed Forces Munich	GPS, IMU, two inclinometers, odometer, barometer	Pair of digital cameras, one colour video camera	0.1 m	Road features and its nearby surrounding mapping
CDSS, Geodetic Institute Aachen	two C/A code GPS, two odometers, barometer	two video cameras	Less than 1 m in highways and up to 3 m in cities	Road features mapping using low cost hardware system
PHOTOBUS, Swiss Federal Institute of Technology	Dual frequency GPS, INS	Vertical and horizontal oriented video cameras	0.2 to 0.4 m	Road signs mapping
TruckMap, John E. Chance & Associates	Multiple GPS receivers, digital attitude sensor	Laser range finder, video camera	0.31 m in horizontal	Electric transmission line corridor surveying, urban mapping

GEOMOBIL, Cartographic Institute of Catalonia	two sets of GPS, IMU, DMI	Pair of stereoscopic digital cameras, laser scanner	0.18 m and 0.35 m in horizontal while 0.13 m in vertical positions	Urban mapping
StreetMapper, 3D Laser Mapping and IGI mBH	GPS, IMU, DIA	Digital camera, two laser scanners	0.01 to 0.03 m	Highways, road features and urban mapping
IP-S2, Topcon	GNSS, IMU, odometer	Panoramic camera unit, three laser scanners	0.015 to 0.040 m	Utilities, GIS asset management and transportation
LYNX M1, Optech	GPS, IMU, DMI	four digital cameras, two laser scanners	0.05 m	Road infrastructure, asset management, utilities
MoSES, 3D Mapping Solutions GmbH	DGPS, IMU, odometer	eight digital cameras, two laser scanners	0.03 m in horizontal and 0.003 m in vertical position	Road and rail construction site surveys, trajectory quality management, urban mapping
StreetView, Google	DGPS, IMU, odometer	nine digital camera, three laser scanners	N.A.	Street mapping for visualisation purpose

Table 1.1: Examples of terrestrial MMSs.

[Goa91]. The absolute object accuracy was 1-3 m in Easting and Northing which describes the accuracy achieved by GPSVan in object positioning with respect to the ground truth values. This limited accuracy was attributed to the use of a code-only Global Positioning System (GPS) receiver and gyro based inertial system [EES02]. In 1993, the Video Inertial SATellite (VISAT) system was developed at the University of Calgary. This system was based on dual frequency carrier-phase Differential Global Positioning System (DGPS) and a more accurate Inertial Measurement Unit (IMU) [SMES+93]. The VISAT system reported an improved horizontal accuracy of 0.3 m in Easting and Northing while operating at a speed of up to 60 km/hr but it had drawbacks including increased cost and a much higher level of system complexity [Es05].

By the mid 1990's additional systems based on a similar architecture were developed worldwide. In 1995, the KInematic Surveying System (KISS) was designed and developed by the University of the Federal Armed Forces for kinematic Geographical Information System (GIS) data acquisition [HCH+95]. During post-mission mode, the surveyed data was processed to produce 3D georeferenced information of the road network and its surrounding environment. In 1998, the Car Driven Survey System (CDSS) system was developed at the Geodetic Institute Aachen for acquiring road data using a low-cost navigation system which comprised a Coarse Acquisition (C/A) code GPS, odometer and barometer [BA98]. In 2000, the Geodetic Engineering Laboratory of the Swiss Federal Institute of Technology, Lausanne designed the PHOTOBUS system which was distinguishable from other systems by its ability to georeference the road centre line using a camera oriented vertically and acquire road sign data using a camera oriented horizontally [GSG03].

The object accuracy achieved was 20-40 cm with respect to the central axis of the road.

1.1.2 Mobile Mapping with Laser Scanners

Laser scanning has been accepted as an effective method for spatial data acquisition in MMSs due to the highly accurate and dense point cloud data that can be captured and recorded by these systems. The integration of laser scanners with terrestrial MMSs facilitated the rapid and cost effective capturing of 3D data for larger urban areas [HPKH08]. Some examples of terrestrial MMSs with laser scanners have also been listed in 1.1. In 1995, the applicability of terrestrial mobile laser scanning system for surveying and mapping transmission line corridors was demonstrated by Reed et al. [RLW96]. Their van-based system, TruckMAP, incorporated real time positioning, a reflectorless laser range finder and high resolution video cameras which were used to map the sub-stations and electric transmission line corridor while the sections of transmission line with no vehicular access were surveyed with an airborne laser scanning system, FLI-MAP. In 2000, the Cartographic Institute of Catalonia developed the GEOMOBIL system which included all the navigation and mapping sensors required for acquiring digital stereo pair images and subsequent direct georeferencing [TBA⁺04]. Later, the group integrated a laser scanner into their system which was able to collect 10,000 points per second [TAB⁺04]. The absolute accuracy of the laser scanning system was measured as 0.18 m in Easting, 0.35 m in Northing and 0.13 m in the vertical plane. The design and development of several other MMSs based on laser scanners were reported over subsequent years [ZS03, GNA⁺06, KAS⁺07].

The advent of LiDAR technology enabled the application areas of terres-

trial MMSs to be broadened to include 3D route corridor and urban mapping, traffic simulation studies, virtual reality modelling and utility mapping. This expansion of application areas has led to the development of a number of commercial enterprises based around MMS across the world. Over the last 20 years, MMSs have slowly developed from research projects in the academic sector to becoming commercially viable activities [Pet10]. According to a market research study conducted by ARC advisory group, the 3D laser scanning market is expected to double in size from 2010 to 2015 [Rio11]. There are a number of companies which provide MMS products for fast and automated data acquisition. 3D Laser Mapping is one such company which developed the StreetMapper system in collaboration with IGI mbH to meet the requirements of clients for rapid 3D mapping of highways, road features, buildings and infrastructure using vehicle-mounted lasers [HC10]. The StreetMapper system, which has been operating since early 2005, utilises a GPS receiver, fibre optic gyro based IMU, Direct Inertial Aiding (DIA) system, high resolution digital camera and two Riegl VQ-250 laser scanners. Each scanner offers a 360° Field Of View (FOV), a range of 300 m and a measurement rate of 300,000 points per second. In 2009, another company, Topcon positioning systems introduced IP-S2 (Integrated Positioning) MMS which integrates a Topcon dual frequency GNSS receiver operating at 20 Hz, Honeywell HG1700 tactical grade IMU based on Ring Laser Gyro (RLG), wheel-mounted odometer, the Ladybug multi-camera unit that is capable of 360° panoramic imaging and three LMS 291 laser scanners [Top10]. One other company, Optech, provides a LYNX mobile mapping product which incorporates an Applanix POSPac navigation system, imaging system and laser scanners that are built in-house by Optech [Opt10]. Their LYNX M1

model, launched in 2010, provides a laser measurement rate of 500,000 points per second, scan frequency of 200 Hz and range of 200 m. Some of the other MMS product suppliers include Mitsubishi Electric Corporation, Trimble and Riegl which are all well-established in the mapping and surveying industry [Pet10].

Apart from these MMS product suppliers, there are numerous companies which offer mapping services. 3D Mapping Solutions GmbH offers services in kinematic surveying of road networks with their mobile road mapping system MoSES. Their system integrates DGPS, IMU, linear odometer, eight multispectral cameras and two laser scanners [Gra08]. Google provides Street View images to the users online which are acquired using their MMS Street View cars. Their vehicle comprise a Topcon DGPS/IMU positioning system, a wheel mounted odometer, three SICK LMS 291 laser scanners and nine Elphel digital cameras that are configured to provide 360° horizontal and 290° degree vertical panoramic view [Wil07, Pet10]. Most recently, Google has introduced a new tricycle platform, named Trikes which are equipped with a similar set of positioning system, laser scanners and digital cameras. These Trikes are being used for data collection in areas which are not accessible by the cars. Similarly, other commercial companies like Tele Atlas and NAVTEQ provide digital map databases for navigation and cartographic applications which they generate using their own mobile mapping vehicles equipped with positioning and imaging systems [Pet10]. MMSs have many applications and in the next section, we describe one area where they can be used to improve safety along road networks.

1.2 Road Safety Applications of Mobile Mapping

Road transportation plays a vital role in the progress and socio-economic growth of society enabling the safe movement of goods, people and services. Roads are designed and built based on numerous design criteria, notably, travel time, user comfort and convenience, fuel consumption, construction, cost and environmental impact [ETS97]. A well designed and maintained route infrastructure assists in driver safety as well as in the efficient use of overall network in terms of route navigation.

Road accidents have become one of the main concerns for policy makers and road infrastructure developers due to thousands of deaths and the economic loss caused by them. Each year, around 1.2 million people die in road crashes around the world while around 50 million are severely injured [WHO09]. Furthermore, these accidents cost between 1 and 2 % of a country's annual Gross National Product (GNP) [WHO10]. According to the World Health Organization (WHO) report, road traffic accidents are likely to become the fifth leading cause of death in the world by 2030 [WHO11]. In the member states of the European Union (EU), road traffic accidents claim around 35,000 lives and leave more than 1.6 million people injured annually [BEY+10]. The economic cost has been estimated at around 2% of EU countries Gross Domestic Product (GDP), around 180 billion Euro, which is twice the EU's annual budget [Saf09]. The statistics of road accidents, fatalities and injuries occurred in the EU member states from 1990 – 2006 is graphically represented in Figure 1.1. The negative impact of road accidents can not be ignored in terms of the very sizeable social and economic loss.

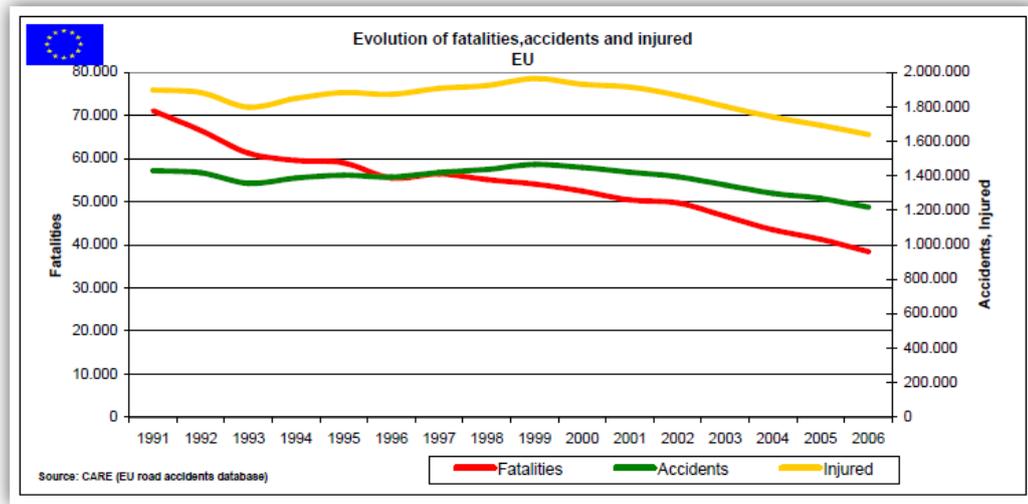


Figure 1.1: Road accident statistics in Europe [CAR07].

Thus, the main challenge for policy makers is to ensure that road networks are as safe as possible whilst maintaining quality and mobility.

The main causes of road accidents can be attributed to driver-behaviour, vehicle and road infrastructure or a combination of all these as described in Figure 1.2 [TTM+79]. Although driver behaviour is the main cause, the other two factors, vehicle and the road infrastructure, usually contribute to the final outcome. Road transport networks should be developed and maintained by taking into account the interaction between the above mentioned three factors. To date less consideration has been given to the road infrastructure element [IRF03]. Analysis shows that accidents occur due to human error mostly at specific accident hotspots. Road design has an immediate effect on accident risk as it influences driver behaviour in terms of speed, acceleration and lateral position. Safe road-way infrastructure has an important role in reducing the accident risk as road infrastructure contributes to one out of three fatal accidents [UNE08]. Road safety considerations must result in a

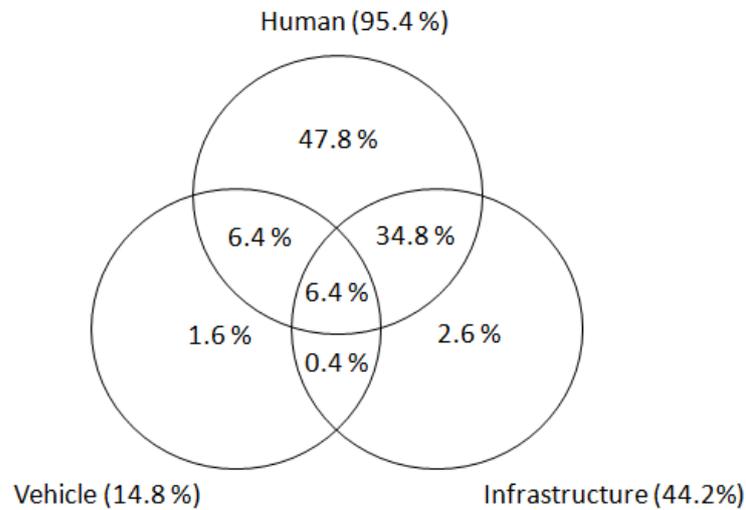


Figure 1.2: Main causes of road accidents [TTM⁺79].

road environment that should be self-explaining and forgiving, in the sense that users are not faced with unexpected situations and their mistakes can be, if not avoided, corrected [ERS06].

Recent research investigations have described a significant correlation between road infrastructure and accident analysis values [GPG⁺07]. Road user safety may be affected by road geometry and physical factors along the route corridor. Road geometry includes the parameters used for designing roads such as horizontal length section, curve radius, curvature change radius (CCR), vertical grade, cross-sectional lane width, shoulder width, median, number of lanes and stopping sight distance [GPG⁺07]. The physical factors refer to the objects along the route corridor such as traffic signs, light poles, trees, walls and signage. Route safety also depends on the existence and condition of road safety interventions along the roads. For example, road signs may be missing or suffer from reduced visibility due to temporary occlusion arising from vegetation growth, weather or from some other factors. Road geometry and physical road factors are required to be located, measured,

classified and recorded in a timely, cost effective manner in order to schedule maintenance and ensure maximum safety conditions for road users.

Various safety schemes and standards such as Road Safety Audit (RSA), Road Safety Inspection (RSI) and Network Safety Management (NSM) are implemented to qualitatively estimate potential road safety issues along the route corridor. The aim of these safety schemes is to identify the elements of the road that may present a safety concern and explore the various opportunities to eliminate identified safety concerns [ETS97]. Current road surveys collect this information manually which usually involves an engineer annotating a digital map or using spatially referenced video to manually classify various features along the route [ERE+08]. The information collected through these surveys is sometimes incomplete and insufficient for qualitative estimation of potential road safety issues. It can also be time consuming and expensive to conduct these inspections on a large scale. A recent research call highlighted the requirement for common evaluation tools and implementation strategies in carrying out these inspections and assessing risk along route corridors [PME+09]. One research project EuRSI [MM10] demonstrated that MMS could be used to collect physical route corridor information for rapid safety analysis.

With the potential of GIS technologies in road management, terrestrial MMSs present a reliable and cost effective alternative for carrying out road inspections along the route corridor. Terrestrial MMSs can be employed to capture 3D spatially referenced information about road geometry and physical road objects. This information can assist decision makers to identify the possible risk elements of the road which may present a safety concern. In the next section, we review various methods developed for extracting road

features from LiDAR data

1.3 Road Features Extraction

Accurate information about the road and its features is a prerequisite for effective management of road networks and to ensure maximum safe driving condition for road users. The extraction of road networks from aerial and satellite multi spectral optical images has been extensively researched. However, some limiting factors such as shadows, complex illumination and spatial accuracy prevailed in those approaches [SNG10]. The use of LiDAR technology for mapping road infrastructure provides accurate and dense 3D point cloud data which contain elevation, intensity, pulse width, range and multiple echo information. These data attributes can be used for reliable and precise extraction of the different road features. The methods developed for segmenting LiDAR data are mostly based on the identification of planar or smooth surfaces and the classification of point cloud data based on its attributes [Vos09]. In a related area, several methods have been developed over the past decade for extracting urban building features from LiDAR data [OTDS04, PV06, BH09, HDP09, RRP09, MEs10].

Some attempts have also been made to extract the road and its features from LiDAR data. Clode et al. [CKR04] segmented airborne LiDAR point cloud data into road and non-road objects using a hierarchical classification technique based on elevation and intensity information. The accuracy of their road segmentation approach was reduced due to the presence of car parks and private roads in their survey area. Hu et al. [HTH04] segmented LiDAR data into road and non-road areas based on elevation and intensity attributes. The Hough transformation was then applied to extract the candi-

date road stripes and parking areas. High resolution optical image data was also used to obtain road areas by extracting the concrete or asphalt pixels based on thresholding. Accuracy issues associated with the intermixing of road networks with parking areas were resolved using shape analysis and vehicle detection queues from the LiDAR and image data. Akel et al. [AKF⁺05] identified roads from airborne LiDAR data which were used for generalising the Digital Terrain Model(DTM). LiDAR data was segmented by applying a region growing approach on the basis of surface normal direction and height difference properties and then the extracted segments were classified into road and non-road objects based on a certain set of decision rules. Mumtaz et al. [MM09] identified buildings, trees and roads using a normalised Digital Surface Model (DSM) generated from LiDAR data and a Normalised Differential Vegetation Index (NDVI) estimated from high resolution aerial imagery. The resulting accuracy in road extraction was poor due to occlusions arising from buildings and tree shadows in the optical imagery. Oude Elberink et al. [OV09] developed an automated method for 3D modelling of highway infrastructure using airborne LiDAR data and 2D topographic map data. The road polygons were extracted from the topographic map data using a map based seed growing algorithm combined with a Hough transformation. The LiDAR points were added to the corresponding road polygons using a LiDAR based seed growing algorithm. Subsequently, 3D reconstruction was achieved by assigning the third dimension to the map polygons. Samadzadegan et al. [SBH09] used a multiple classifier system to classify the airborne LiDAR points into road and non-road objects using first pulse, last pulse, range and intensity attributes. Different combinations of LiDAR attribute layers were classified based on different features using maximum

likelihood and minimum distance methods. However, the optimum selection of features, type of classification technique and classifier fusion method were not conclusively addressed.

The majority of these road extraction methods attempt to delineate roads by distinguishing them from non-road objects but do not make any attempt to extract the road edges. The road boundary is a fundamental feature, knowledge of which can provide precise estimation of other road features such as road markings and roughness. In the following sections, we review various methods developed for extracting road edges, markings and roughness from LiDAR data. We also investigate various methods based on snake curves which were developed for extracting road and urban features.

1.3.1 Road Edges

Road edges usually distinguish the road surface from kerbs in urban roads and from grass-soil in rural roads. Road edges with kerbs and grass-soil are shown in Figure 1.3. We do not define the edge between a road and a hard shoulder as the shoulders are used for emergency stopping or access. The hard shoulders can be extracted based on a similar approach used for road markings extraction as they both possess retro-reflective surface characteristics.

Road edges need to be correctly identified and extracted in order to obtain precise information about road geometry and physical road objects. To date, little research has been focused on extracting precise road edges. Yuan et al. [YZC⁺08] proposed an algorithm for extracting road surface from terrestrial LiDAR data. The algorithm used a fuzzy clustering method to cluster LiDAR points. Straight lines were then fitted to the linearly clustered



Figure 1.3: Road edges: (a) kerbs in an urban road section and (b) grass-soil in a rural road section.

data using slope information for extracting the road surface area. Another approach for extracting the terrain surface from LiDAR point cloud data was formulated by Yoon et al. [YC09]. They calculated the slope and standard deviation characteristics from the LiDAR points and used these values to estimate the edges of the road. Vosselman et al. [VL09] developed a method for detecting kerbstones from airborne LiDAR data. The approach was based on the detection of small height jumps caused by the kerbstones in the LiDAR point cloud data. However, their extraction accuracy was affected by parked cars occluding the kerbstones. Zhang et al. [Zha10] proposed a method for detecting road edges in an urban environment using terrestrial LiDAR data. In their method, road edge points were identified based on elevation information. The identified 3D road edge points were then projected on a ground plane to estimate the road kerbs. Smadja et al. [SNG10] developed an algorithm for extracting roads from LiDAR data based on the detection of slope break points coupled with the RANdom SAMple Consensus (RANSAC)

algorithm [SNG10]. The extracted road boundaries were further processed to compute road curvature and road width information. McElhinney et al. [MKCM10] developed an algorithm for extracting road edges from terrestrial mobile LiDAR data. In the first stage of their algorithm, a set of lines were fitted to the road cross sections based on the navigation data and then LiDAR points within the vicinity of the lines were determined. In the second stage, these points were analysed along the Northing axis based on slope, intensity, pulse width and proximity to vehicle information in order to extract the road edges. The algorithm did not use the Easting values of LiDAR data to estimate the road edges.

Most of the methods reviewed have been developed for extracting road edges in an urban environment where algorithms rely on the existence of a sufficient height or slope difference between the road and kerb points for detecting road edges. Little or no research has been carried out to extract rural roads where the non-road surface comprises grass-soil and the edges are not as easily defined by slope changes alone. There is a need to develop a method that will provide an efficient and more accurate estimation of edges for different road types. Approaches developed to date make partial use of LiDAR data for extracting the road edges. The intensity and pulse width attributes from LiDAR data can be a useful source of information for extracting these road edges. Their use in urban and rural road sections has yet to be thoroughly explored.

1.3.2 Road Markings

Road markings play an important role in reducing accident frequency and severity as they provide guidance and instruction to the road users for safe

and comfortable driving. They are intended to direct traffic by indicating the direction of travel, warn road users about specific obstacles or hazards and define the territorial limit for traffic flows [GPG⁺07]. Road markings are retro-reflective surfaces having an ability to reflect most of the incident light back to its originating source. These markings retain their visibility criteria in day and night. Examples of road markings are shown in Figure 1.4. Laser

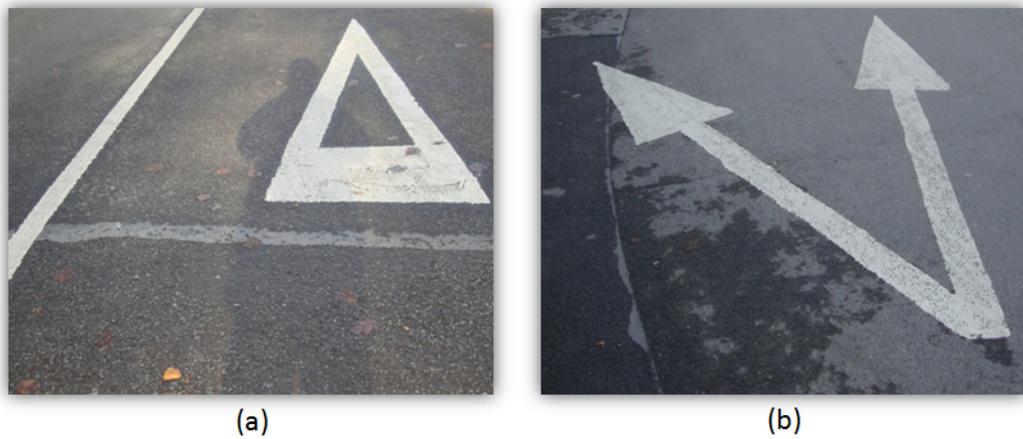


Figure 1.4: Road markings: (a) continuous line, triangle and (b) arrow.

scanners usually record the reflectance of the illuminated road surface in the form of intensity data which can be used to distinguish road markings.

Smadja et al. [SNG10] extracted road markings by applying a threshold to intensity data acquired using terrestrial mobile laser scanning system. Vosselman et al. [Vos09] recommended a normalisation of the intensity data prior to the threshold implementation or the use of a distance dependent threshold for extracting road markings from terrestrial LiDAR data. Jaakkola et al. [JHHK08] estimated road markings by first performing a radiometric correction of the LiDAR intensity data using a second order curve fitting function. Finally, road markings were estimated by applying a threshold and morpho-

logical filtering methods. Toth et al. [TPB08] used road pavement markings as ground control for assessing the positioning quality of aerial LiDAR data. The search window for finding the pavement markings in the LiDAR data was reduced by making use of the GPS survey data collected over the pavement. The pavement markings were extracted from LiDAR intensity values. Later, extracted pavement markings were compared with the GPS survey data to assess the quality of the LiDAR points. Chen et al. [CSW⁺09] developed a method to extract lane markings from the terrestrial LiDAR intensity data. After detecting the road surface using the elevation information from the LiDAR data, lane markings were extracted by applying a threshold. The extracted lane markings were clustered using the Hough transform.

The majority of methods developed for extracting road markings are based on threshold applied to the LiDAR intensity values. The development of a robust threshold approach will provide a more precise extraction of road markings. A threshold applied to the intensity values often introduces noise, which needs to be reduced. The reflected intensity values depend upon the distance from the laser scanner to the illuminated surface, incidence angle of the laser pulse and surface characteristics. The intensity values need to be normalised in relation to these factors. A priori knowledge of the road boundaries and its surface will facilitate a more efficient extraction of road markings.

1.3.3 Road Roughness

The roughness of the road surface can be considered an important factor that influences safety condition for road users. It can be defined as the deviation of a road surface from a designed surface grade that may develop as a result

of road use, construction process or a combination of them [dFdS09]. Road roughness affects rolling resistance, ride quality, vehicle operating cost and safety of the road users [SK98]. Examples of roughness along the road surface are shown in Figure 1.5. Several indices have been developed which are



Figure 1.5: Roughness along the road surface.

used to estimate roughness along a longitudinal profile of the road surface. These indices are computed as dynamic or geometrical values [dFdS09]. The dynamic indices such as International Roughness Index (IRI) provide continuous estimation of the roughness based on a model that simulates a dynamic response of a measuring vehicle along the road surface. The geometrical indices such as standard deviation of longitudinal roughness provide discrete estimation of the roughness in the form of standard deviation values of relative elevation points measured along the road surface.

The IRI was developed by the World Bank in the 1980's in response to a requirement for a reference scale for road roughness measurement [SGQ86]. This measure is used to provide a continuous estimation based on a model that applies a mathematical simulation of a standard vehicle moving along

the road surface profile at a certain speed. The model uses a Quarter Car Simulator (QCS), shown in Figure 1.6 [dFdS09]. The QCS consists of a

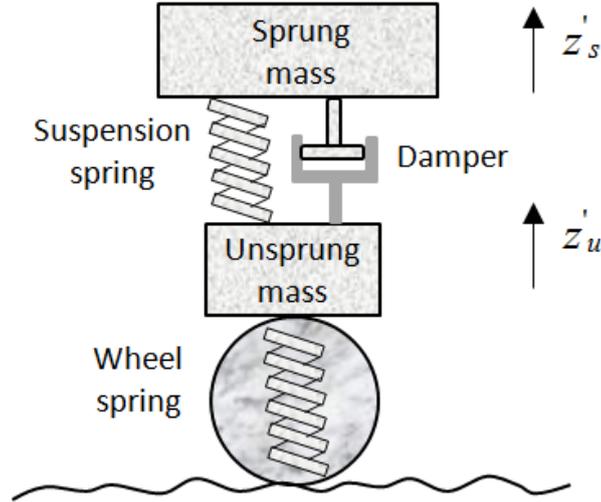


Figure 1.6: Quarter car simulator.

sprung mass that represents the vehicle body and an unsprung mass that represents the wheel and suspension. The sprung mass is connected to the unsprung mass with the suspension spring and damper. The unsprung mass is in a contact with the road surface using the wheel spring. During the simulation process, the QCS runs along the road surface profile at a constant speed V . The roughness along the road surface generates z'_s and z'_u vertical speeds in the sprung and unsprung mass respectively. The IRI value for a section of the road surface profile is estimated as

$$IRI = \frac{1}{L} \int_0^{x/V} |z'_s - z'_u| dt \quad (1.1)$$

where L is a length of the road section in meters, x/V is a time taken by the model to travel a certain distance x and dt is a time increment. Thus,

the IRI is an accumulation of a vertical displacement and divided by the distance travelled by the vehicle that, in turn, provides the roughness scale. Its value is estimated in m/km or inch/mile units and ranges between 0 to 20 m/km. The 0 m/km value of IRI represents a perfectly smooth road surface, approximately 6 m/km value represents a moderate road roughness and 20 m/km value represents a bumpy unpaved road surface [Pat87]. One key advantage of using the IRI scale for the roughness measurement is its reliability as it facilitates both repeatability and stability of results with respect to time [SK98].

The standard deviation of longitudinal roughness provides discrete estimation based on elevation points that are measured at a 1.5 m interval along the longitudinal profile of the road surface as shown in Figure 1.7 [dFdS09]. These elevation points along the longitudinal profile are measured

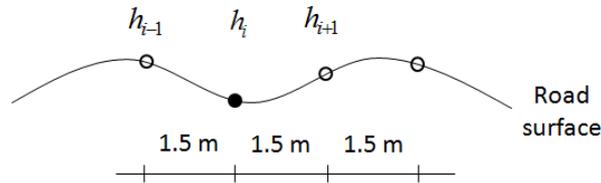


Figure 1.7: Measurement of elevation points at a 1.5 m interval along the longitudinal profile of the road surface.

using straight edge profilometers or laser profilers. The relative elevation d_i for each point is computed as

$$d_i = h_i - \frac{1}{2}(h_{i-1} + h_{i+1}) \quad (1.2)$$

where h_i , h_{i-1} and h_{i+1} are the current, previous and next measured elevation values respectively. A standard deviation of longitudinal roughness is

estimated as

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (d_i - \bar{d}_i)^2}{n}} \quad (1.3)$$

where \bar{d}_i is a mean of the d_i values and n is the number of points.

LiDAR data provides elevation values which have also been used for estimating road roughness. Pattnaik et al. [PHS03] estimated grade and cross-slope parameters of a road segment from LiDAR data. Road boundaries were delineated using multi-resolution orthophotos, GIS street database and a terrain model generated from LiDAR data. A centreline of the road was then determined using the estimated road boundaries. A plane was fitted to the LiDAR points using a linear regression model. Residuals for the grade and cross-slope were then estimated by finding a goodness of fit of the regression plane with the LiDAR points. Zhang et al. [ZF05] also presented a method for estimating road grade and banking from LiDAR data using a linear regression model, however road boundaries were extracted based on a priori knowledge of the road width instead of using a surface terrain model.

Some other approaches have been developed for estimating roughness over soil, river bed and other terrain surfaces. Zhang et al. [ZR04] demonstrated prototype system for estimating a ground surface roughness information using a combination of a camera and laser scanning system. A multiscale variance method was used over different ground surfaces to characterise an elevation profile at different spatial scales. Hollaus et al. [HH10] investigated two approaches for estimating terrain roughness from full waveform airborne LiDAR data. In the first approach, an orthogonal regression plane was fitted to the LiDAR data and then the standard deviation values of the residual elevation points were calculated. In the second approach, the roughness pa-

rameters were estimated from the echo width attribute of LiDAR data which provides information on the range distribution of scatterers. Both these approaches were found to be useful for extracting terrain roughness information. However, the echo width based approach produced a similar pattern results even for moderate point densities. Diaz et al. [DJS+10] characterised surface roughness in agricultural soil using terrestrial LiDAR data. LiDAR data was interpolated to create a surface grid model with a cell spacing of 1 cm which was used to estimate the surface roughness. Cavalli et al. [CTMF08] analysed the capability of high resolution airborne LiDAR data for recognising morphological features of a river channel bed. The analysis was carried out at 1D scale along the longitudinal channel profile and at 2D scale over the full extent of the channel bed. In the 1D approach, residuals of the elevation and slope were computed orthogonal to regression line drawn along the channel profile. In the 2D approach, elevation and slope residuals were estimated based on their local variation along the channel bed. Yen et al. [YAL+10] analysed terrestrial mobile laser scanning data to produce digital terrain models of pavement surfaces. In one of their analyses, a linear plane was fitted to the data points for 1 mile road section and then the vertical offsets were calculated.

LiDAR data provides elevation information which can be used to estimate road roughness without a requirement for any simulation process. Its ability to provide reliable information for estimating road roughness needs to be thoroughly explored. Unlike the traditional methods, laser scanning systems can also be used to provide the spatially referenced roughness information along the road surface. Most approaches used to compute roughness indices are cumulative in nature. LiDAR data can be used to provide a localised

roughness information along the longitudinal as well the transverse profile of the road surface. The methods developed for estimating road roughness from LiDAR data are based on fitting a regression plane and then computing elevation residuals. There is a need for a more robust surface fitting approach that will provide an ideal representation of the road surface. A priori information of the road boundaries and its surface will facilitate the process of estimating road roughness.

1.3.4 Snake Curves

The concept of snakes or parametric active contours was first introduced by Kass et al. [KWT88] and has been widely used in many applications including image segmentation, object boundary localisation, motion tracking, shape modelling and 3D reconstruction. In the parametric model, the snake is represented explicitly as a controlled spline curve, which is implemented based on computed energy. It is defined within a 2D image domain that moves towards a desired object boundary under the influence of an internal energy within the curve itself and an external energy derived from the image data. The internal energy is applied to the curve which controls the curve's elasticity and rigidity, while the external energy attracts the snake curve towards the object boundary. The energy of the snake is minimised by maintaining a state balance between the internal and external energy terms. When the snake's energy function reaches a minimum, it converges to the object boundary. Active contour models will be discussed in more detail in Section 3.1.3. The use of a snake curve for the estimation of brain parenchyma from Magnetic Resonance Imaging (MRI) scan is shown in Figure 1.8, with a yellow curve representing final position of the snake curve.

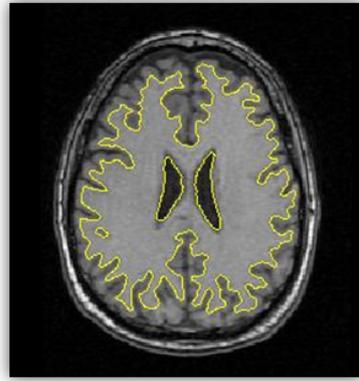


Figure 1.8: Snake curve and MRI scan [LS95].

Several methods based on the snake concept have been developed for extracting road and urban features from high resolution digital images. Kerschner et al. [Ker98] extended the concept of the traditional snake to twin snakes for extracting linear features with two boundaries from high resolution digital imagery. An additional energy term was introduced in the snake's energy function which formulated the attraction force in between the twin snake curves for detecting two parallel linear boundaries. Youn et al. [YB04] developed an approach for extracting urban roads from aerial digital imagery using an adaptive snake. In their method, an initial approximation was provided to the snake curve by detecting the preliminary road lines based on the dominant road directions. These dominant directions were estimated using the fact that road and buildings edges in an urban area are usually parallel. After initialisation, an adaptive snake algorithm was applied in which the weight coefficients provided to the snake energy terms were locally modified. Rocca et al. [RFFP04] used a wavelet based parametric active contour model for detecting linear features from satellite imagery. The wavelet transform was used to determine high gradient values in the images filtered at multiple

scales. An active contour model was applied over the image at the coarsest scale. The final snake curve obtained at the coarsest scale was used to initiate the snake curve at next coarsest scale and this process was repeated until the original scale of the image was reached.

Mayunga et al. [MZC05] developed a radial casting approach to initialise the snake curve for extracting building features from high resolution Quick-bird earth observation imagery. In their method, a single seed point was placed at the approximately the centre of each building object. The snake points were then generated in accordance with the radial lines projected outwards at various angular intervals from the centre seed point. Yagi et al. [YBKY05] used a parametric active contour model for tracking a road and reconstructing the 3D road shape from road scenes recorded using monocular cameras. They assumed that the left and right road edges would be parallel. This was applied as a constraint in the parametric active contour model to converge the control points to the left and right boundaries of the road. Zhang et al. [ZXZ08] used a self-adaptive template matching method to provide the initialisation to the Least Square B-spline (LSB) snake model for extracting road objects from high resolution satellite imagery. In the template matching method, various templates of road width and luminance attribute were matched with manually selected points in the image. The result with maximum matching was used to initialise the snake for extracting road features.

Apart from digital images, methods based on the snake concept have also been developed for extracting objects from LiDAR data. Tseng et al. [TTC07] proposed an approach for surface reconstruction from LiDAR data. In their algorithm, a surface was grown from an initial seed point in the

LiDAR data based on the extended snake model. The internal energy was provided by placing a constraint on the angle in between the normal vectors of two adjacent planar patches in order to maintain the smoothness of a reconstructed surface. The external energy was modelled as a function of distance from the LiDAR points to the corresponding planar patch. Goepfert et al. [GR09] applied a snake model to extract the road network for integrating and matching, 2D Digital Landscape Model (DLM) derived from 2D vector data and DTM generated from airborne LiDAR data. In their approach, the snake curve was initialised near the road feature using existing vector data, stored as a polyline, while the external energy terms were derived from intensity, elevation and surface roughness based images generated from the LiDAR data. The energy of the snake contour was minimised which led to its convergence to the road network. The extracted road network information was then used to integrate the two datasets. Later, they extended their work to include building and bridge information in order to support the road extraction process using the snake model [GR10]. Kabolizade et al. [KEA10] proposed an improved snake model for extracting buildings from aerial images and LiDAR data. The initial estimation of building edges was carried out by applying a threshold to the Normalised Digital Surface Model (NDSM). The snake model was implemented by deriving the external energy terms based on intensity and altitude variances of the snake curve points with their neighbourhood points.

Snake curves present a useful approach for extracting objects. One of the limiting factor associated with snake curves is a requirement for manual intervention or a priori information for their initialisation. The development of an automated initialisation method can provide for an efficient use of the

snake curves for the segmentation process. There is a need to develop a more robust approach for determining high gradient values in the 2D image. These values are required for generating external energy terms. The use of snake curves for the segmentation of LiDAR data is also required to be thoroughly explored. LiDAR data provides elevation, intensity and pulse width information which can be used for deriving external energy terms of the snake curve. In the following section, we detail our contributions to the field of road feature extraction from LiDAR data.

1.4 Contributions of the Thesis

In this thesis, we will primarily detail three key contributions we have made in the field of road feature extraction from LiDAR data. These contributions relate to the development of automated algorithms for extracting road features from terrestrial mobile LiDAR data. We also make novel use of navigation information from the mobile van in our algorithms.

Our first contribution is the development of an automated algorithm for extracting road edges from terrestrial mobile LiDAR data. We introduce a new approach based on the combination of two modified versions of the parametric active contour model that, in turn, provides a more precise extraction of the road edges. Through the use of LiDAR elevation, reflectance and pulse width attributes in the algorithm, we are able to differentiate the road surface from grass-soil edges in rural road sections and kerb edges in urban road sections. These attributes are input as 2D raster surfaces which significantly reduces the computing time. A point thinning process smooths these 2D raster surfaces as the raw LiDAR data is usually accompanied with high frequency noise. Object boundaries in the 2D raster surfaces are auto-

matically extracted through the application of hierarchical thresholding and Canny edge detection. These boundaries are then used in the external energy terms of the parametric active contour models which helps converge the snake curve to the road edges. Another novelty is in our automatic snake curve initialisation based on the navigation track of the mobile van. Extraction of road edges from large sections of road is facilitated through batch processing of consecutive individual sections. We have automated our road edge extraction algorithm through the recommendation of most applicable methods and optimal parameters. Finally, we have developed an automated algorithm for validating the extracted left and right road edges. In this validation algorithm, extracted road edges are validated with respect to manually digitised road edges.

Our second contribution is the development of an automated algorithm for extracting road markings from terrestrial mobile LiDAR data. We use our automated road edge extraction algorithm to estimate the road boundaries in the form of a snake curve. The output snake curve is used to identify LiDAR points that belong to the road surface. A priori knowledge of the road surface reduces false positives and the amount of processing in the road marking extraction process. We have developed a novel range dependent thresholding function to extract the road markings from intensity values. This is due to the variations which occur in the intensity values due to varying distance from the laser scanner to the illuminated surface and incidence angle of the laser pulse. The range dependent threshold is applied through converting the LiDAR intensity and range attributes to 2D raster surfaces. We use morphological operations to complete the shapes of extracted road markings and to remove noise that is introduced through the use of thresholding. These

operations are applied using the navigation heading information of the mobile van. To fully automate our algorithm, we have tested the most applicable values of cell size required for converting LiDAR data to 2D raster surfaces and range dependent threshold.

Our final contribution is development of an automated algorithm for estimating road roughness from terrestrial mobile LiDAR data. A priori knowledge of the road surface allows us to estimate the roughness over the road surface. We use a RANSAC algorithm to fit a surface grid to the LiDAR points belonging to left side of the road section. This is representative of what the ideal road surface should be. A cell size in the surface grid is provided based on the surface area of the foot-print of the mobile van's wheel that comes into contact with the road surface at any instant. We find a residual between the elevation values of each LiDAR point and its respective surface grid point and then the standard deviation of elevation residuals is calculated in each cell along the navigation track of the mobile van. The resulting values provide a discrete estimation of the longitudinal road roughness.

To validate our algorithms, we have tested them on a variety of road sections assessing the automated extraction of road edges, markings and roughness from terrestrial mobile LiDAR data. The development of these automated algorithms present a cost effective and time saving alternative to the manual processing of LiDAR data for road features extraction. In the next and final section, we provide an outline of the thesis.

1.5 Outline of the Thesis

Chapter 2, presents a detailed description of terrestrial mobile mapping technology and its components. It includes a description of imaging system and

new developments for acquiring imaging data. We discuss the different range measurement methods used by laser scanning systems to acquire 3D point cloud data. We discuss the navigation system which is used to record navigation parameters of the vehicle in the global coordinate frame. We describe the data acquisition system which consists of control, synchronisation, data storage and power supply units. We present the eXperimental Platform-1 (XP-1) MMS, which has been designed and developed at National University of Ireland Maynooth (NUIM). We describe its imaging, laser scanning, navigation and data acquisition components.

Chapter 3, describes the theoretical background of the various approaches, which have been used for developing our algorithms for extracting road edges, road markings and road roughness from terrestrial mobile LiDAR data. We discuss various segmentation approaches applied to extract road edges. This includes an overview of hierarchical thresholding and Canny edge detection, which are used to determine object boundaries from the LiDAR data. We present active contour models describing both parametric or geometric categories. We describe how parametric active contour models are applied in our road edge extraction algorithm. We discuss a range dependent thresholding applied to extract road markings from the LiDAR intensity values. Finally, we describe the RANSAC algorithm which is applied to fit a surface grid to the 3D LiDAR points. The fitted surface grid is then used to find the elevation residual values which in turn provides an estimation of the road roughness.

Chapter 4, presents our road edge extraction algorithm. We provide a stepwise description of our algorithm. It involves generating a 2D raster surface from LiDAR data and estimation of energy terms required to implement

the parametric active contour models. We describe an initialisation of the snake curve based on the navigation track of the mobile van. We describe the batch processing of consecutive individual sections to extract road edges from the larger road sections. We also present our algorithm to validate the extracted road edges with respect to the manually digitised road edges. We analyse the most applicable methods and optimal parameters involved in our algorithm to automate the process of extracting the road edges. We test our algorithm on various road sections in both manual and automated modes. We validate the results using our validation algorithm and carry out a comparative analysis of both manually and automatically extracted road edges. Finally, we conclude the chapter by discussing the novelty and limitations of our algorithm.

Chapter 5, describes our road marking extraction algorithm. We discuss various steps involved in our algorithm to extract road markings from terrestrial mobile LiDAR data. We detail how the output snake curve produced from our automated road edge extraction algorithm is used to identify the LiDAR points that belong to the road surface. We describe the 2D raster surface generation process from LiDAR intensity and range attributes. The range dependent thresholding function is presented, which is used to extract road markings from the intensity values. We describe the morphological operations used to complete the shapes of extracted road markings and to remove noise. To automate our algorithm, we recommend the most applicable values of input parameters through their experimental verification. We test our algorithm on various sections which contained distinct type of road markings. We validate the extracted road markings on the basis of their quantitative assessment. In the final section of the chapter we discuss the

advantages and limitations of our algorithm.

Chapter 6, presents our road roughness estimation algorithm. In a step-wise description of our algorithm, we again describe the use of the output snake curve to identify the LiDAR points belonging to the road surface. We detail the RANSAC surface grid fitting to the LiDAR points that belong to the left side of the road surface. We calculate the standard deviation of elevation residuals in each cell of the fitted grid along the navigation track of the mobile van, which in turn provides an estimation of road roughness. We test our algorithm on various road sections. We analyse the results to indicate higher, moderate and lower roughness along the tested road sections. The chapter is concluded by discussing the merits and demerits of our algorithm.

Chapter 7, summarises the three contributions, we have made to the field of road feature extraction from LiDAR data. We discuss the novelty of our algorithms and describe their limitations. Finally, Chapter 8, details future research directions which are enabled by our work. We describe improvements which can be made to extend the algorithms and methodologies presented in this thesis.

Chapter 2

Terrestrial Mobile Mapping Technology

Terrestrial mobile mapping technology has emerged as a reliable method for mapping infrastructure along roads, streets and utility networks. Multi sensor integrated mapping technology has enabled rapid and cost effective acquisition of georeferenced information about natural, urban and road network environments [LC08]. Their initial development was primarily driven by the advances in digital cameras and navigation technologies. Later, laser scanning systems were integrated with terrestrial MMSs which facilitated more accurate, dense collection of 3D point cloud data.

This chapter provides a detailed description of terrestrial mobile mapping technology and presents the XP-1 MMS, developed at NUIM. In Section 2.1, we describe the various components of mobile mapping technology. This includes a description of imaging systems and new developments for acquiring imaging data. Different range measurement methods are used by laser scanning systems to acquire 3D point cloud data. We describe these methods

and all the attributes returned by the laser scanning system such as intensity, pulse width, range and multiple echo. We describe a navigation system and provide a detailed overview of the direct georeferencing process involved with imaging, laser scanning and navigation systems. We detail a data acquisition system consisting of control, synchronisation, data storage and power supply units. Section 2.2 presents the XP-1 MMS and describes its imaging, laser scanning, navigation and data acquisition components in more detail.

2.1 Components of Mobile Mapping Technology

Terrestrial MMSs are based on the combination of four systems: imaging, laser scanning, navigation and data acquisition systems. The imaging and laser scanning systems are used to acquire spatial data and are generally referred to as the mapping sensors. The navigation system is the combination of multiple sensors which may include GNSS receiver, INS and DMI. The navigation system is used to record the navigation parameters of the mobile mapping vehicle in the global coordinate frame. The data acquisition system provides synchronisation of various sensors and also facilitates data storage. These components are discussed in detail in the following sections.

2.1.1 Imaging System

In a terrestrial MMS, the imaging system along with laser scanning and navigation systems are integrated and mounted on a mobile vehicle for spatial data acquisition [LC08]. An example of a digital image of a road section acquired with camera onboard the XP-1 MMS is shown in Figure 2.1.

With recent developments in imaging technology, film based optical cam-



Figure 2.1: XP-1's digital camera image of a road section.

eras have been replaced with multi-spectral scanners and digital cameras. Multi-spectral scanners are used to record large areas of the ground from airborne and satellite platforms in across track and along track scanning modes. Across track scanners record reflected radiations from terrain with an oscillating mirror that moves at right angles to the aircraft or satellite's direction of travel, while along track scanners consist of a linear array of detectors that record the reflected radiations along the aircraft or satellite's direction of movement [LKC08]. Digital frame cameras were generally developed for terrestrial based mapping. However, with the steady increase in spatial resolution of digital cameras, these cameras are now being used in airborne applications [SEs04]. Digital cameras are used to capture images using either a Charge Coupled Device (CCD) or a Complementary Metal Oxide Semiconductor (CMOS) system which converts acquired radiation into a charged signal. In CCD based cameras, each pixel's charge packet is transferred sequentially to a common output node which converts them into a

voltage while in CMOS based cameras, the conversion of charge to voltage takes place in each pixel [Lit01]. Both these systems provide high quality images but CMOS systems have a more complex circuit design [Lit05]. However, CMOS systems have the advantage of lower power consumption over CCD.

The selection of an imaging sensor in a MMS is dependent on the desired accuracy, reliability, operational flexibility and end-use application requirements [Es05]. These imaging sensors can capture imagery across various spectral bandwidths including visible, infrared and thermal regions. Stereoscopic imaging systems are popular for generating 3D georeferenced data. A stereoscopic camera system consists of a pair of digital cameras mounted on a mobile mapping vehicle which acquires a stereo image pair. After calibration, 3D positions can be computed for matching points. Spatial referenced 360° panoramic images can be recorded using multiple digital cameras. Several commercial companies such as Point Grey Research, Immersive Media Corporation (IMC), Cyclomedia and Google employ multiple camera systems to produce panoramic images. [Pet10].

Calibration of digital cameras is a requirement for extracting precise and reliable information from an imaging system. The purpose of camera calibration is to determine a true geometric model of the camera described by its Interior Orientation (IO) parameters such as focal length, principal point coordinates, radial and tangential distortion [LRKH06]. The computed geometric camera model is used to produce undistorted imaging data. The most popular approach for camera calibration has been bundle block adjustment which was introduced in the early 1970's [RF06]. This calibration process is carried out by selecting a scene with easily identifiable target points. A digi-

tal camera is then used to capture multiple images of the scene from different known locations of the camera. Afterwards, the target points are identified in each image and are input into a mathematical model of the bundle block which leads to the determination of all the IO parameters. Transformation parameters such as rotation offset and misalignment angle in between the imaging and GNSS/INS sensors in the MMS are also estimated using the calibration process which provides accurate positioning of image data at the time of capture [TBA⁺04].

2.1.2 Laser Scanning System

Laser scanning systems have been recognised as an efficient and reliable source of data acquisition in terrestrial MMSs for 3D mapping and modelling. An example of LiDAR data of a road section acquired using Riegl VQ-250 laser scanner in the XP-1 MMS is shown in Figure 2.2. LiDAR



Figure 2.2: XP-1's LiDAR data of a road section.

technology provides several benefits over conventional sources of data acquisition in terms of accuracy, resolution, information content and automation.

The laser scanning system emits a laser pulse which is monochromatic and coherent. The transmitted laser pulse may hit one or more targets which causes one or multiple echo pulses. These echo pulses return to a receiver instrument in the laser scanning system which converts the pulses into digital signals. In Figure 2.3(a), the principle of multiple returns from targets is described [Rie09]. A laser pulse is transmitted from the laser scanning system

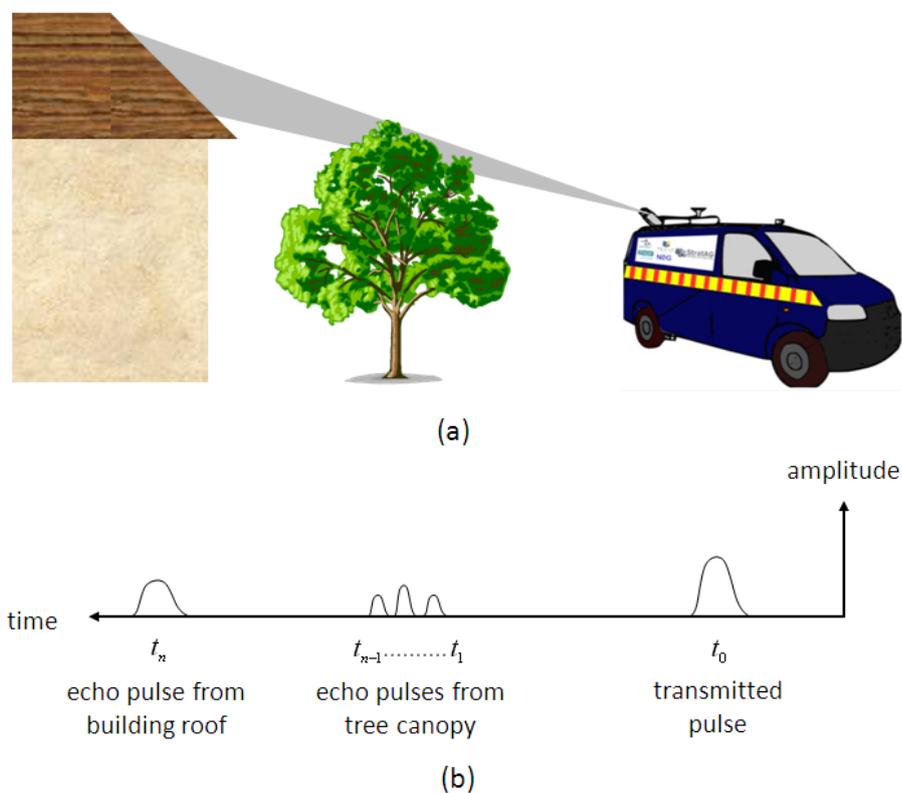


Figure 2.3: Principle of (a) multiple returns from targets and returned (b) echo pulses.

which strikes the tree canopy and produces echo pulses which are returned to the sensor. A fraction of the laser pulse which was not occluded by the tree canopy also strikes the building roof which leads to the return of an additional echo pulse. The returned echo pulses are depicted in more detail

in Figure 2.3(b). The first pulse refers to the transmitted laser pulse. The next $n - 1$ pulses correspond to reflection from the tree canopy, while the last pulse corresponds to the reflection from the building roof. The laser scanning system estimates the exact time position of each pulse as t_1, \dots, t_n for n targets from an initial time position t_0 of the transmitted laser pulse. The estimated time is used to measure the distance or range in between the laser scanner and target objects using a simple distance-time relation described as

$$R = s.t/2; \tag{2.1}$$

where R is the range, s is the speed of light and t is the estimated time interval. The laser scanning system is used in conjunction with navigation sensors which are used to measure the position and orientation parameters of the laser scanner in the global coordinate frame. The navigation measurements along with the sensor head orientation and range measurements are then used to obtain the 3D georeferenced information about the target objects.

The time interval of the reflected pulse can be determined based on either Time Of Flight (TOF) or phase shift ranging methods [PT08]. In a TOF method, a laser pulse is transmitted from the laser scanning system which strikes the target object and is reflected back to the sensor as shown in Figure 2.4. A reflected pulse returns to the receiver instrument and the time interval from transmission to return of the pulse is measured. The range is determined from the measured time interval using a simple distance-time relation described in Equation 2.1. In a phase shift method, the laser scanning system measures the phase difference between the transmitted and reflected pulse, shown in Figure 2.5 [PT08]. The time interval is then determined from

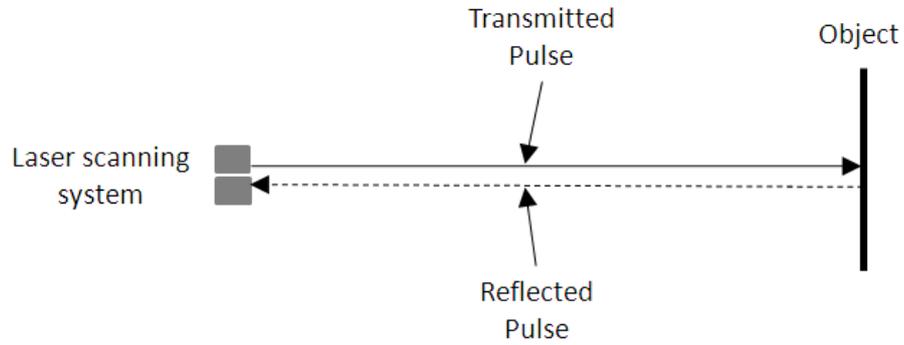


Figure 2.4: TOF method

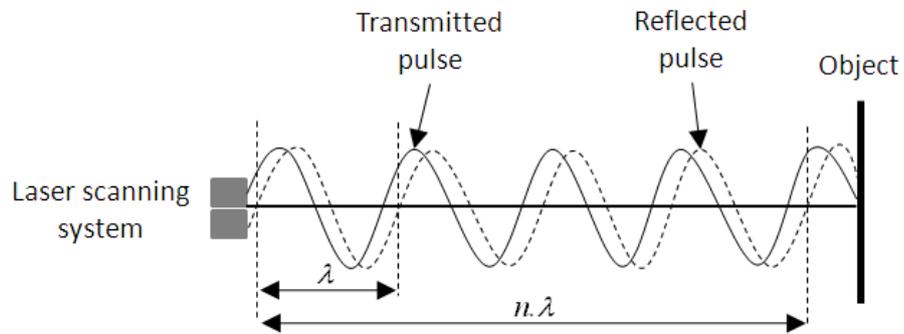


Figure 2.5: Phase shift method.

the measured phase difference as

$$T = \lambda(n + \xi/2\pi) \quad (2.2)$$

where ξ is the phase difference, λ is the wavelength of the pulse and n is the integer number which are measured using a digital pulse counting technique. The range is estimated from the measured time interval using the distance-time relation described in Equation 2.1.

The TOF method can be used to measure distances from a few hundred metres to several hundred kilometres [PT08]. The phase shift method is only suitable for short distances as an ambiguity can arise during long distance

measurement due to the periodical variation of the phase [Pas10]. The laser scanning system uses a Continuous Wave (CW) transmission in the phase shift method for scanning terrain objects, which requires the use of high power lasers. However, the use of the phase shift method provides better accuracy than the TOF method [PJAA11]. The phase shift method can also be used to measure the direction and velocity of a moving target in addition to the range measurements [BEF+96].

Laser scanning systems also provide intensity and pulse width information. Intensity is most commonly described as the maximum amplitude of a reflected pulse [HP07]. In Figure 2.6, the intensity of reflected pulse is shown as its maximum amplitude AB. The intensity value depends upon the sur-

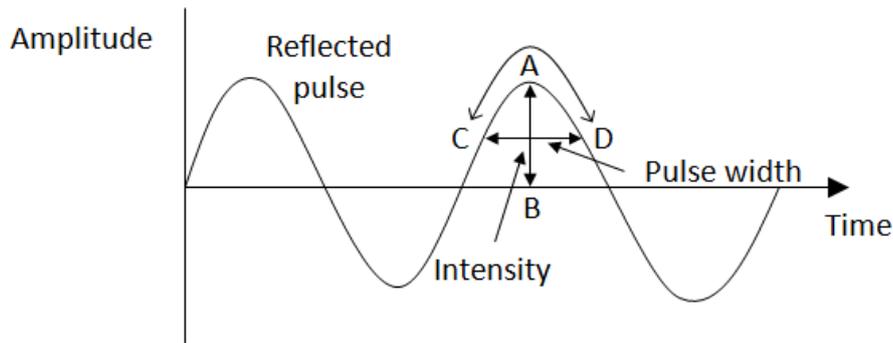


Figure 2.6: Intensity and pulse width of reflected pulse.

face characteristic of a target object, distance from the laser scanner to the target object and the incidence angle of laser pulse [JG09]. Intensity values can be used to differentiate terrain objects. However in most cases, they are found to be different for similar terrain objects. Intensity is required to be normalised in order to determine true reflectance values from the terrain objects. The pulse width from the laser scanning system is a recorded time difference between half maximum amplitudes of the pulse. In Figure 2.6, the

$$r_p^m = r_{GNSS}^m + R_{INS}^m (R_X^s(\alpha) \cdot R_Y^s(\beta) \cdot R_s^{INS} \cdot r_p^s + r_s^{INS} - r_{GNSS}^{INS}) \quad (2.3)$$

where r_p^m is the coordinates of the point p in the m-frame and r_{GNSS}^m is the coordinates of the GNSS in the m-frame determined using kinematic GNSS. R_{INS}^m is the rotation matrix between the INS body and the m-frame determined using INS measurements, R_X^s and R_Y^s are rotation matrices which describe a rotation of the mapping direction around the X and Y-axis of the s-frame respectively and R_s^{INS} is the rotation matrix between the INS and the s-frame determined through calibration. r_p^s are the coordinates of the point p in the s-frame, r_s^{INS} are the coordinates of the mapping sensor in the INS body frame determined through calibration and r_{GNSS}^{INS} are the coordinates of the GNSS in the INS body frame determined through calibration.

GNSS has revolutionised conventional surveying and mapping by providing reliable positioning services with an accuracy range up to few millimetres [KfBI03]. The GNSS sensor is used to determine translational and rotational parameters of a trajectory from range measurements between satellite and receiver. A roving GNSS sensor on a mobile mapping vehicle operates with respect to a local GNSS base station [Pet10]. The GNSS base station calculates its position based on a satellite signal and compares this position with its known position. This difference is then applied to correct the roving GNSS data in either Real time Kinematic (RTK) or Post Processing Kinematic (PPK) mode. In RTK mode, the roving GNSS sensor directly receives corrections from the base station GNSS using a Ultra High Frequency (UHF) modem communication. In PPK mode, the roving GNSS data and the base station GNSS corrections are stored in a data logger and are combined in

post processing [IXS09]. Thus, the processed GNSS data provides an accurate estimation of the vehicle's navigation parameters. The INS sensor consists of three gyroscopes and three accelerometers which determine the relative position and orientation parameters of the mobile mapping vehicle by sensing angular velocity and specific force [Es05].

In urban areas with high rise buildings or in areas with dense canopies, the GNSS sensor can output less accurate navigation due to satellite signal obstruction and distortion [Pet10]. However, the INS sensor can help compensate for this as it maintains reasonable level of position and orientation accuracy for short intervals of time. The combination of the GNSS and INS sensors provide an effective solution as both sensors can be used to update each other frequently and maintain high global positioning accuracy. Their integration is achieved using a Kalman filter algorithm which identifies and corrects navigation errors in both the GNSS and INS sensors [GWA07]. In the Kalman filter process, the navigation information from the GNSS and INS sensors is compared to estimate the errors [IXS09]. These errors are discriminated in the Kalman filter observation unit and are then fed back into the GNSS and INS error models for correction. Thus, drifts in the INS gyroscopes and accelerometers can be corrected with the GNSS derived trajectories while in areas of poor or absent GNSS signal coverage, the position and orientation parameters can be corrected with the INS observations [Es05]. Terrestrial MMSs are also supplemented with DMI's to provide additional information for relative positioning. The DMI is attached to the mobile mapping vehicle's wheel and provides a measure of distance travelled by the vehicle [GQS06]. The process for estimating navigation parameters of the mobile mapping vehicle using a GNSS base station and a navigation

system is shown in Figure 2.9. The measurements from a GNSS base station

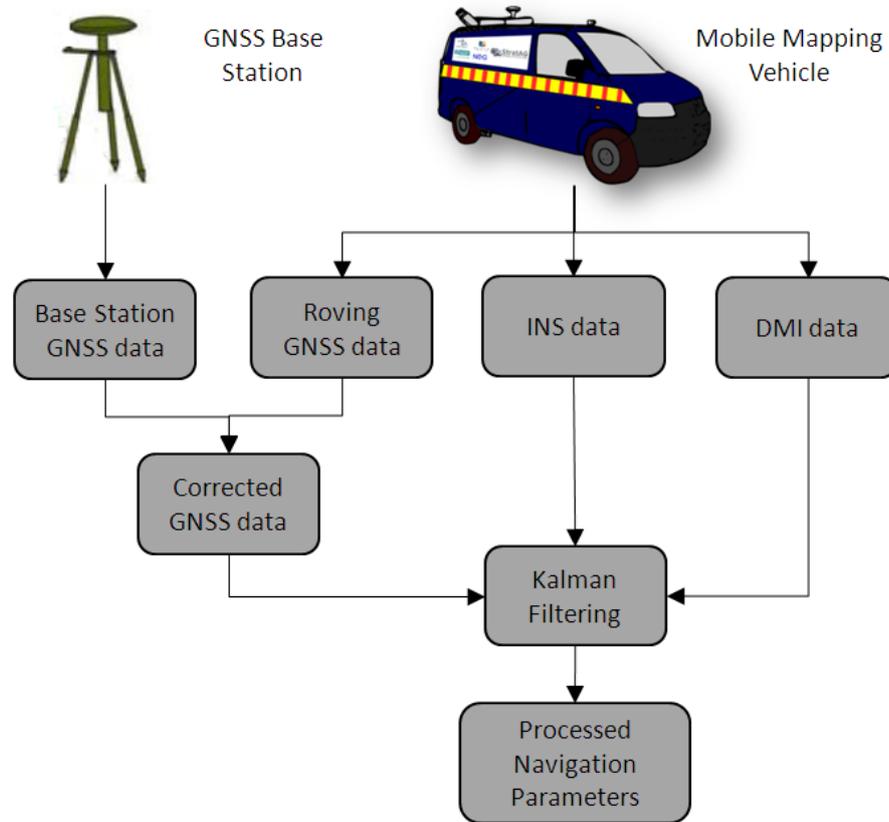


Figure 2.9: Process for estimating navigation parameters of the mobile mapping vehicle using a GNSS base station and a navigation system.

are used to make corrections to a roving GNSS data collected with the mobile mapping vehicle. The corrected GNSS data is then integrated with INS and DMI data in a Kalman filter process to provide more accurate estimation of the vehicle's navigation parameters.

2.1.4 Data Acquisition System

The MMS data acquisition system consists of control, synchronisation, data storage and power supply units which are shown in Figure 2.10. The pur-

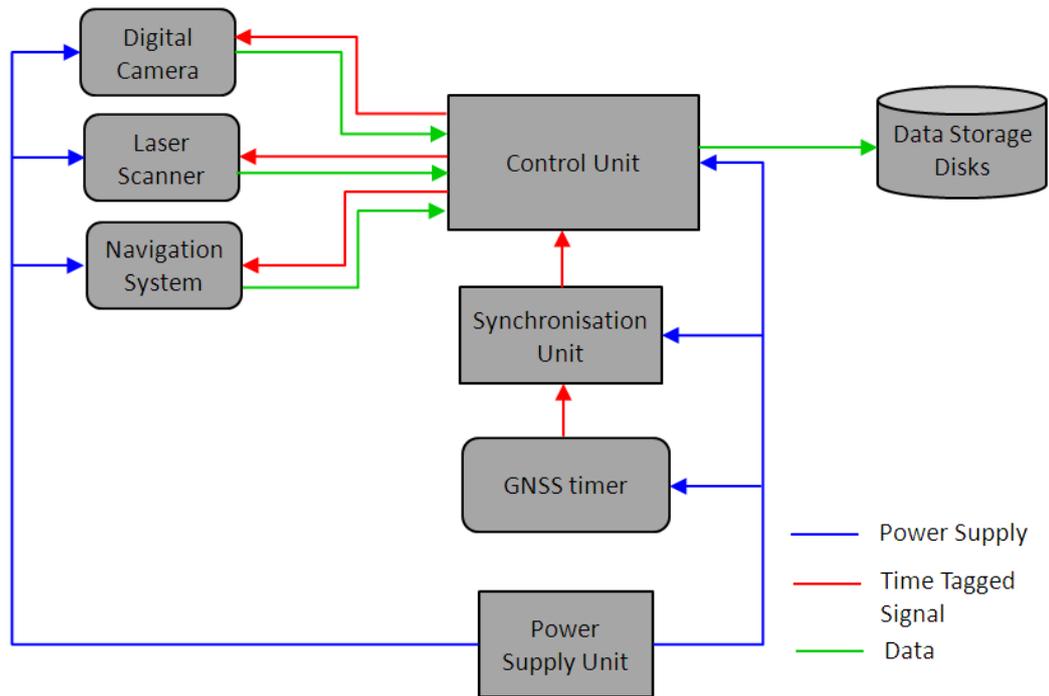


Figure 2.10: Data acquisition system.

pose of the synchronisation unit is to synchronise the mapping and navigation sensors in a common temporal reference frame. A GNSS timer provides an accurate and well defined Pulse Per Second (PPS) time signal to the synchronisation unit which is used as a reference for all sensors in the MMS. In the synchronisation unit, a time board device tags a Transistor-Transistor-Logic (TTL) digital signal with respect to the received PPS time signal [TBA⁺04]. The tagged TTL signals are then used to synchronise mapping and navigation sensors in RTK mode.

A suitable data logging and storage system is required to record the high volumes of data acquired by the mapping and navigation sensors in a MMS. The synchronisation and data storage units are controlled by a central control unit while the power supply unit provides electricity supply to enable contin-

uous operation of the mapping sensors, navigation sensors, synchronisation unit and control unit in the MMS.

2.2 XP-1 System

The XP-1 MMS at NUIM comprises of an IXSEA LandINS GPS/INS, a Riegl VQ-250 laser scanner and an imaging system which are mounted on a mobile van. A DMI is fitted to a wheel of the van which records the distance travelled and is used in computing the final navigation parameters during post processing. The XP-1 system is shown in Figure 2.11. The



Figure 2.11: XP-1 MMS with an inset picture of the laser scanning and navigation system mounted on the roof rack.

LandINS navigation system is based on an INS, a rover GPS receiver and antenna, a UHF modem and a data logger. In PPK mode, we correct the roving GPS data with Receiver INdependent EXchange (RINEX) files acquired from permanent GPS base stations. The core of the LandINS system is an INS which is an assembly of three Fibre Optic Gyroscopes (FOGs) and three accelerometers with a drift rate of better than 0.005 degree/hour [IXS09].

The navigation process in the LandINS system is based on Kalman filtering which integrates measurements from the INS, GPS and DMI sensors to provide an accurate estimation of navigation parameters. The performance of the LandINS system in the PPK mode are detailed in Table 2.1. The XP-1

True Heading	0.01 degree
Roll / Pitch	0.005 degree
Position X and Y	0.002 m
Position Z	0.05 m
Measurement Rate	100 Hz

Table 2.1: IXSEA LandINS navigation system specifications.

navigation data along a road section was plotted over Google earth image, shown in Figure 2.12.

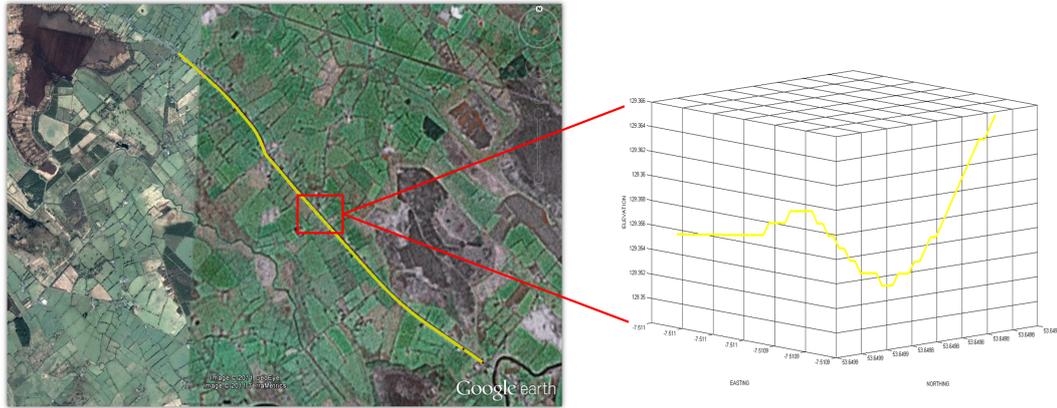


Figure 2.12: XP-1's navigation data plotted over a Google earth image, highlighted in yellow, with an inset picture of its small section (Image courtesy: Google Earth).

The Riegl VQ-250 laser scanner is mounted on the roof rack on the back of the mobile van at a 45° angle with respect to both the horizontal and vertical axis of the vehicle. This sensor orientation optimises sensor-target geometry and ensures that the laser scanner produces richer 3D information than if

mounted flat onboard the survey vehicle. The horizontal inclined position enables the laser scanner to scan those sides of terrain objects which are perpendicular to the direction of travel of the mobile van or else they may be missed. The vertical inclined position of the laser scanner is useful for scanning objects above the mobile van such as overhead road signs or bridge faces whose sides are perpendicular to the direction of travel of the mobile van. An inclined laser scanner configuration along with the navigation system in the XP-1 system is shown in Figure 2.13. The range measurement in the

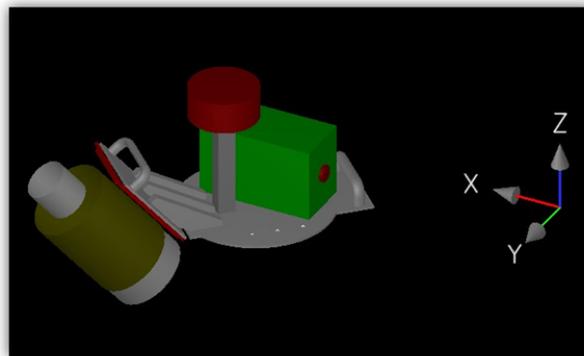


Figure 2.13: Inclined laser scanner along with GPS antenna and LandINS GPS/INS mounted on the XP-1's roof rack.

laser scanning system is based on the TOF method, while the use of online waveform analysis technology allows for the digitisation of the waveform of each reflected pulse. It captures up to 1 million points every 3.5 seconds using a 300 KHz sensor which leads to approximately 20 GB of data per hour. The specifications for the Riegl VQ-250 laser scanner are shown in Table 2.2.

The imaging system consists of two progressive scan digital cameras (1280 * 1024) that are mounted behind the windscreen of the mobile van. In the near future, 3 multispectral and a FLIR thermal SC-660 cameras

Minimum Range	1.5 m
Operating Range (@300 KHz)	75-200 m
Accuracy	0.01 m
Precision	0.01 m
Measurement Rate	300 KHz
Signal Intensity	16 bit
Laser Wavelength	1550 nm
Beam Divergence	0.3 mrad
FOV	360 degree
Scan Speed	100 Hz

Table 2.2: Riegl VQ-250 specifications.

are intended to be included in the XP-1 system. These camera sensors will provide visible, infrared and thermal imageries along route corridors. Active infrared Light Emitting Diode (LED) illumination enables the mobile mapping system to operate in dark or low-light conditions.

A power unit onboard the vehicle is capable of supplying 3 KW of power. Synchronisation and triggering of the mapping and navigation sensors is centrally controlled over a Local Area Network (LAN) using a Intelligent Reference/TM-4 GPS timer device. Three 4U 19" servers provide data logging services and are fitted with removable disks to facilitate fast data processing. System initialisation usually takes 20 minutes before the vehicle can begin surveying. This time is required to ensure that coarse and fine alignment of the navigation system is carried out correctly. An operator sits beside the driver and controls all onboard systems using a central data acquisition console. This includes monitoring image quality, navigation, LiDAR data as well as power and computing resources. Examples of imagery and LiDAR data of a road section collected using digital camera and Riegl VQ-250 laser scanner onboard the XP-1 system is shown in Figure 2.14.



Figure 2.14: XP-1's (a) imaging and (b) LiDAR data of a road section.

2.3 Conclusion

This chapter provided a detailed review of terrestrial mobile mapping technology. We discussed mapping, navigation and data acquisition systems which are considered the essential components of a MMS. The mapping sensors can include imaging and laser scanning systems which are used for spatial data acquisition. The navigation sensors are used to record navigation parameters of the mobile mapping vehicle in the global coordinate frame. The LiDAR and image data can be geocoded using the synchronised navigation sensor information.

The imaging sensors in terrestrial MMSs are usually based on digital progressive scan cameras which capture images of real world environments. Multiple digital cameras are also used in MMSs for acquiring 3D stereoscopic and 360⁰ panoramic images. The laser scanning systems are used to obtain 3D point cloud information about natural and man-made environments. These systems provide intensity, pulse width and multiple echo

information which can be used to classify different target objects. The navigation system in terrestrial MMSs comprises of multiple sensors including GNSS, INS and DMI. The integration of data outputs from these sensors is achieved using a Kalman filter process which identifies and corrects various errors to provide a more accurate estimation of the navigation parameters. The data acquisition system in terrestrial MMSs facilitates synchronisation, data storage and power supply to the system sensors. The synchronisation of the mapping and navigation sensors is carried out in a common temporal reference frame provided by the GNSS timing device. We also presented the XP-1 MMS and described its imaging, laser scanning, navigation and data acquisition components.

Multi-sensor integrated terrestrial mobile mapping technology presents an accurate, reliable and dynamic source of spatial data acquisition. This chapter assists us in understanding mobile mapping technology and its operational principle in terms of sensor synchronisation, spatial data acquisition, direct georeferencing and subsequent data storage processes. In the next chapter, we detail a theoretical background of the approaches we used to develop our road features extraction algorithm.

Chapter 3

Approaches For Extracting Road Features

The fundamental structure and intrinsic properties of LiDAR data enable more efficient and accurate road feature extraction approaches to be explored. Road features can be extracted from LiDAR data using a combination of the elevation, reflectance and pulse width attributes. In this thesis, we present our three algorithms which we have developed for extracting road edges, road markings and road roughness from the terrestrial mobile LiDAR data. In this Chapter, we introduce the theory and background of the approaches used to develop these algorithms. In Section 3.1, we discuss the segmentation approaches used to extract road edges from terrestrial mobile LiDAR data. We provide an overview of hierarchical thresholding and describe the different processing steps involved in Canny edge detection. We discuss two types of active contour models, the parametric and geometric active contour models. We also present the mathematical theory of these models as it pertains to our segmentation process. We then describe how active contour models are

applied in our road edge extraction algorithm. In Section 3.2, we describe our range dependent thresholding which is used to extract road markings from terrestrial mobile LiDAR data. In Section 3.3, we discuss the RANSAC surface grid fitting algorithm used to estimate road roughness from terrestrial mobile LiDAR data. In Section 3.4, we conclude with a discussion on the approaches explained in this chapter.

3.1 Road Edge Extraction

We have developed a road edge extraction algorithm to extract the left and right edges from terrestrial mobile LiDAR data. In the algorithm, we convert the LiDAR elevation, reflectance and pulse width attributes into 2D raster surfaces and then apply various segmentation approaches to extract the road edges, as shown in Figure 3.1. We will present our road edge extraction

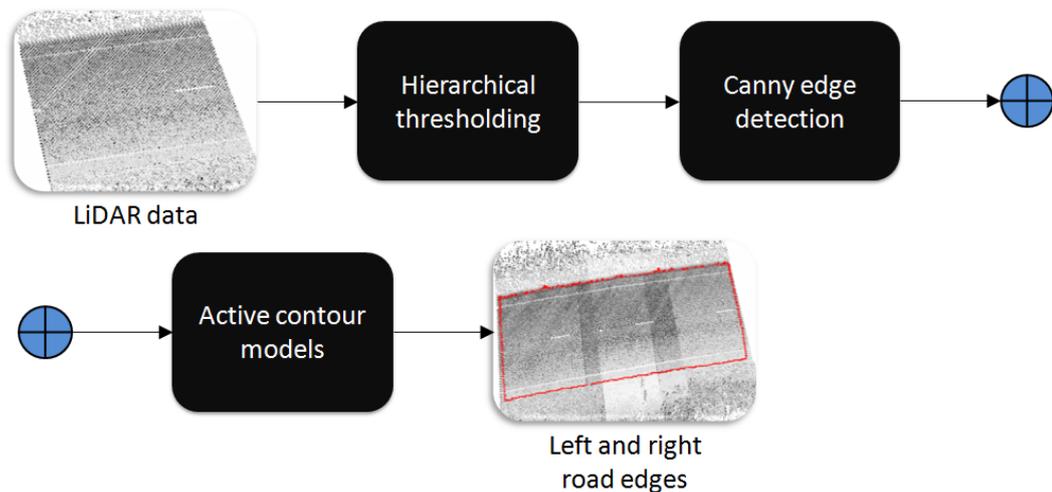


Figure 3.1: Segmentation approaches in the road edge extraction algorithm.

algorithm in detail in Chapter 4. In the following sections, we discuss the

segmentation approaches which are used for developing our algorithm.

3.1.1 Hierarchical Thresholding

In the road edge extraction algorithm, we apply hierarchical thresholding to the 2D raster surfaces generated from the LiDAR attributes to determine objects from them. Hierarchical thresholding is a segmentation approach which takes a 2D image as input. The input image is transformed into a set of images representing a hierarchy of resolution [SHB08]. High resolution images in the hierarchy provide finer details of the object. The image at the highest resolution is the original input image. Low resolution images in the hierarchy represent a blurred or smoothed version of the input image. These images are generated by blurring the input image with a Gaussian filter. This blurring is achieved by convolving each block in the image with a kernel that consists of a matrix of numbers. A block in the image is $N \times N$ group of cells that describes a mask size of the kernel. The value of each convolved block is obtained by multiplying each kernel element with its underlying image cell value and then adding them together.

The convolution kernel is a discrete approximation of a 2D Gaussian distribution function that can be represented as

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(x^2 + y^2)}{2\sigma^2} \quad (3.1)$$

where x and y are values along the X and Y axis respectively, while σ is the standard deviation of the distribution [FPWW00]. The 2D Gaussian distribution is shown in Figure 3.2. An example of the convolution kernel that produces a discrete approximation to this 2D Gaussian distribution

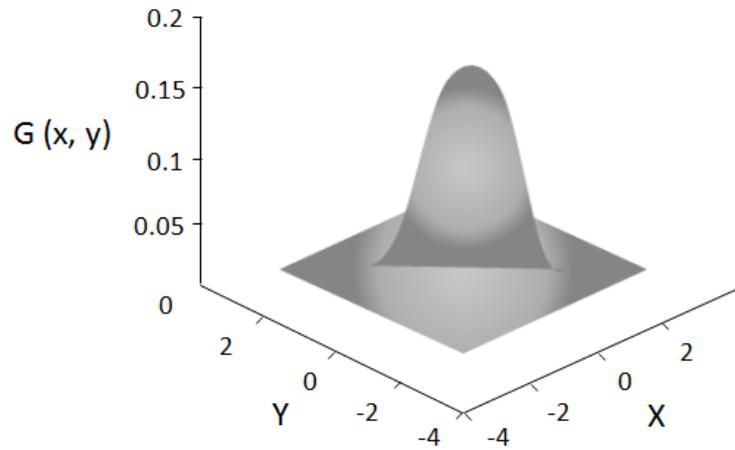


Figure 3.2: 2D Gaussian distribution with mean $(0, 0)$ and $\sigma = 1$ [FPWW00].

function is shown in Figure 3.3.

$$\frac{1}{273} \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Figure 3.3: Convolution kernel that produces a discrete approximation to the 2D Gaussian distribution function with mask size=5 and $\sigma = 1$ [FPWW00].

The Gaussian convolution kernel consists of weights where the highest weight is at the central cell and the weight is lowered based on the cells proximity to the central cell [FPWW00]. The degree of blurring is controlled by modifying the standard deviation of the Gaussian distribution, which in turn, is proportional to the mask size of the convolution kernel. One issue with applying a Gaussian filter is that the convolution kernel can introduce false boundaries at image borders. To overcome this, cells at the border of

the image are usually ignored.

The hierarchical thresholding approach operates in the following way. It takes an image I , a threshold value T and mask sizes s_n, s_{n-1}, \dots, s_1 , where $s_n > s_{n-1} > \dots > s_1$, as input. Starting at level $i = n$ and decrementing at each iteration until $i = 1$, the image is blurred with a Gaussian convolution kernel of mask size s_i . An initial estimation of an object O is made by thresholding the lowest resolution image, at level $i = n$, using the threshold value T , displayed in Figure 3.4(b). After this, corresponding neighbouring

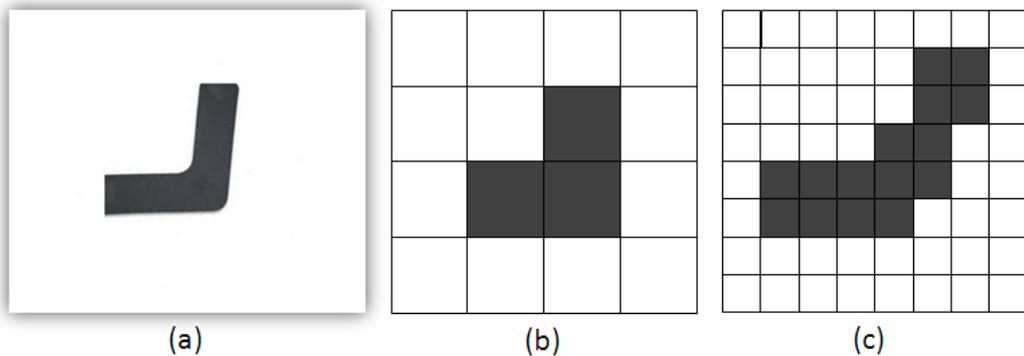


Figure 3.4: Hierarchical thresholding: (a) input image, (b) thresholded object cells in the lowest resolution image, at level n and (c) thresholded neighbourhood object cells in the next lowest resolution image, at level $n - 1$ [SHB08].

cells of the estimated object are thresholded in the next highest resolution image, at level $i = n - 1$, in order to update the estimation, shown in Figure 3.4(c). This process is repeated for each hierarchical level until $i = 1$. This is the process hierarchical thresholding applies to the input image which leads to a precise estimation of objects in the image.

Hierarchical thresholding is particularly useful in dealing with noisy images as it provides a better estimation of the object by minimising the influence of background noise. This makes a single threshold value more robust,

which in turn, allows for the hierarchical threshold to be applicable to automated algorithms. The threshold parameter can be estimated using the difference between the brightness value of an object cell and representative background cells in the image [MMHN08]. The mask size for the Gaussian convolution kernel is determined based on the level of noise present in the image. A larger mask size is used when there is a high amount of noise in the image, while a small mask size is used when there is little or no noise. In our road edge extraction algorithm, we have fixed the threshold value and mask size for all road sections. This allows for the fully automated application of hierarchical thresholding in our algorithm.

To give an example of the hierarchical thresholding used in our algorithm, we take the Easting, Northing and elevation values of the LiDAR data, shown in Figure 3.5(a). We convert these values to a 2D raster surface. The slope of these elevation values is then calculated and is shown in Figure 3.5(b). We apply hierarchical thresholding to this image with mask size=25 and

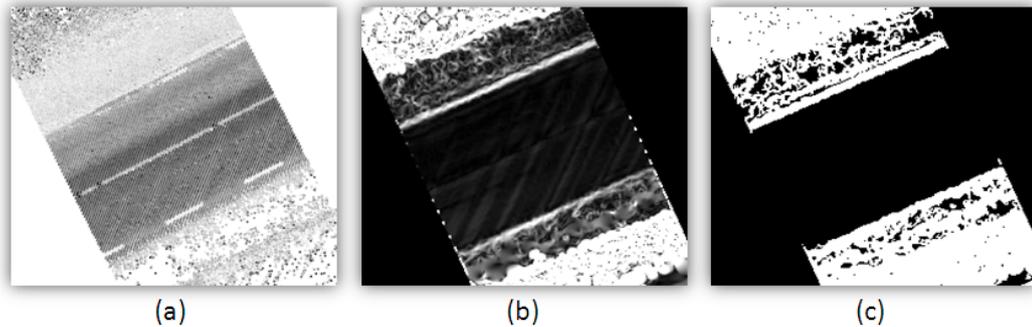


Figure 3.5: Hierarchical thresholding: (a) original LiDAR data, (b) input slope image and (c) estimated objects.

threshold=50, to estimate the objects, shown in Figure 3.5(c).

3.1.2 Canny Edge Detection

In our road edge extraction algorithm, we determine object boundaries by applying Canny edge detection to the objects estimated using hierarchical thresholding. A Canny edge detector is used to identify and locate sharp discontinuities that characterise boundaries of objects in an image. It was originally developed by Canny [Can86] and has become one of the standard object boundary detection methods in image processing.

The Canny edge detection approach involves various processing steps [Moe08]. These begin when an input image is blurred by applying a Gaussian filter, which we explained in Section 3.1.1. The Gaussian filter is applied to reduce any noise present in the image. Then, a Sobel operator is applied over the blurred image to determine gradients that describe sharp changes in the cell values of the image along the X and Y axis. The Sobel operator finds the gradient components along the X and Y axis by applying two 3×3 convolution kernels over the image, which are shown in Figure 3.6. The gra-

$$\begin{array}{cc} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} & \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \\ \text{(a)} & \text{(b)} \end{array}$$

Figure 3.6: Sobel convolution kernels, which are used to find the gradients of an image along its (a) X and (b) Y axis.

dent components are combined to provide an absolute value of the gradient magnitude and direction and can be described as

$$|G| = \sqrt{G_x^2 + G_y^2}, \quad (3.2)$$

$$\theta = \arctan \frac{|G_y|}{|G_x|} \quad (3.3)$$

where G_x , G_y are the gradient components along the X and Y axis, G is the gradient magnitude and θ represents the gradient direction.

In the next step, local maximum values of gradients are estimated by comparing the value of each cell with its neighbours along the gradient direction. The estimated local maximum gradient values are preserved and non-maximum values are removed from the image. After this, two threshold values are applied to the image in order to remove any spurious changes present in it. A cell with a gradient value larger than the upper threshold is marked as strong and is accepted as a boundary cell. A cell with the gradient value smaller than the lower threshold is rejected. The boundary cell in between the two thresholds is marked as weak and is accepted only if it is connected to the strong boundary cell. In this way, noise cells present in the image are removed as they are unlikely to be connected with the strong boundary cells.

Typically in Canny edge detection, the output is extremely sensitive to changes in the two threshold values. To automatically apply Canny edge detection to LiDAR data we would first have to overcome this threshold sensitivity problem. We apply hierarchical thresholding which outputs an image with only two values, 255 for object cells and 0 for non-object cells. Inputting this image into Canny edge detection makes the selection of two threshold values trivial. Setting 250 as an upper threshold limit and 5 as a lower threshold limit will produce the same results for all inputs. This makes our Canny edge detection fully automated and much more robust to noise. An example of the Canny edge detection applied to the hierarchical thresholded objects with upper threshold=250 and lower threshold=5 is shown in

Figure 3.7.

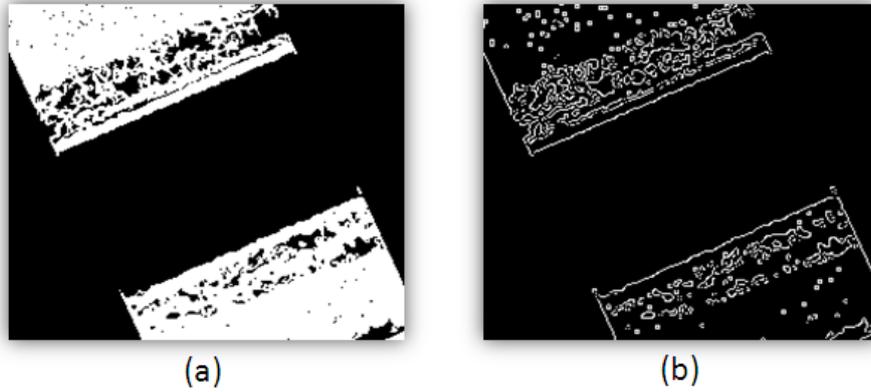


Figure 3.7: Canny edge detection: (a) input hierarchical thresholded objects and (b) their estimated boundaries.

3.1.3 Active Contour Models

After determining the object boundaries in the 2D raster surfaces, we use active contour models in our road edge extraction algorithm. Active contour models present a robust segmentation approach which make efficient use of specific information available about objects in the input data rather than processing all the data [BI98]. Active contour models are categorised as parametric or geometric [XYP00]. The difference between the two versions is in how the contour is defined and behaves. The parametric active contour is represented explicitly as a controlled spline curve that is implemented based on energy computations. The geometric active contour is represented implicitly as a level set and is evolved based on geometric computations. In the following sections, we discuss parametric and geometric active contour models in detail. We describe how the parametric active contour model is applied in our road edge extraction algorithm.

3.1.3.1 Parametric Active Contour Model

A parametric active contour, or snake, is defined as an energy minimising parametrised curve within a 2D image domain that moves towards a desired object boundary. It does this under the influence of an internal energy within the curve itself and an external energy derived from the image [KWT88]. The movement of the snake curve is controlled through balancing the internal and external energy terms until an energy minimisation condition is met. When the snake's energy function reaches a minimum, it converges to the object boundary.

The snake is defined parametrically in the x, y plane of an image as

$$v(s) = [x(s), y(s)], \quad (3.4)$$

where $x(s), y(s)$ are coordinates along the snake curve and s is the normalised arc length. The curve $v(s)$ is represented by a set of control points v_0, v_1, \dots, v_{n-1} and is linearly obtained by joining each control point as shown in Figure 3.8. The snake's energy function can be described as

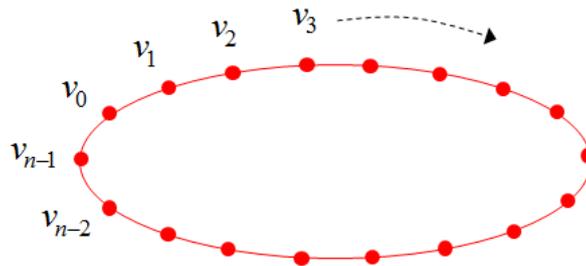


Figure 3.8: Initial snake curve in the form of parametric ellipse.

$$E_{snake} = \int_0^{n-1} E_{snake}(v(s)) ds. \quad (3.5)$$

The energy function constituting the internal and external energy terms is described as

$$E_{snake} = \int_0^{n-1} (E_{int}(v(s)) + E_{ext}(v(s))) ds, \quad (3.6)$$

where E_{int} represents the internal energy term and E_{ext} denotes the external energy term.

The internal energy controls the snake curve's elasticity and stiffness properties. The internal energy function E_{int} can be written as

$$E_{int} = \frac{1}{2}(\alpha(s)|\frac{dv}{ds}|^2 + \beta(s)|\frac{d^2v}{ds^2}|^2), \quad (3.7)$$

where $\alpha(s)$ and $\beta(s)$ are weight parameters. Equation 3.7 is composed of two terms, a first-order term designed to hold the curve together and a second-order term designed to keep the curve from bending. The α weight parameter controls elasticity while the β weight parameter controls stiffness in the snake curve.

The external energy creates a gradient image that attracts the snake curve toward the object boundaries as shown in Figure 3.9. The image gradient based external energy is described as

$$E_{ext} = -\kappa|\nabla f(v(s))|^2, \quad (3.8)$$

where κ is a weighting parameter and ∇f is the gradient image of object boundaries, f .

For the snake curve to converge on the object boundary, the snake's energy function, E_{snake} in Equation 3.6, should be minimised. To minimise the energy function, the snake curve must satisfy the Euler condition as

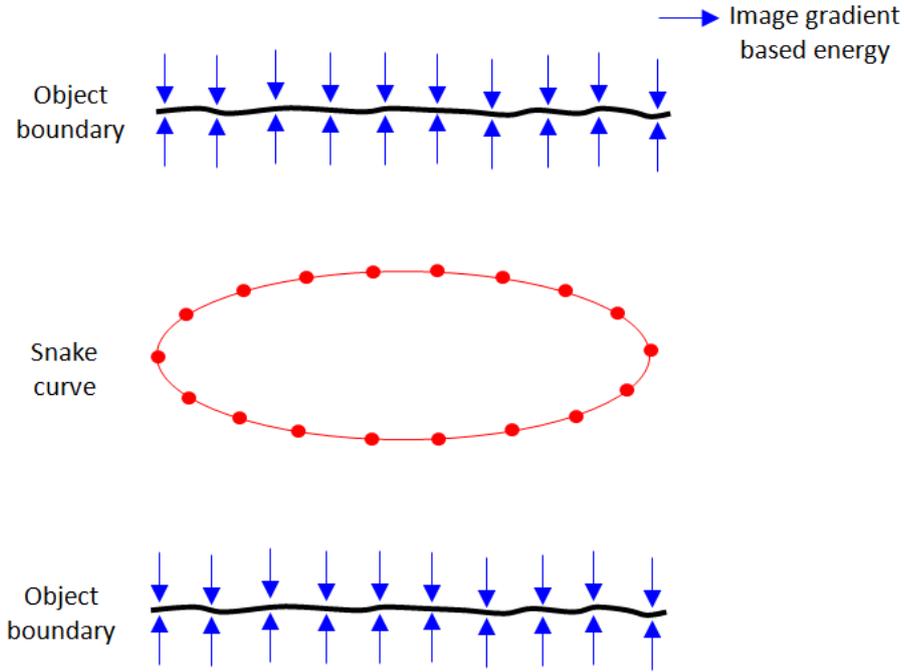


Figure 3.9: Parametric ellipse snake curve initialised on a gradient image of road object boundaries.

[SHB08]

$$\frac{d}{ds}(\alpha(s)\frac{dv}{ds}) - \frac{d^2}{ds^2}(\beta(s)\frac{d^2v}{ds^2}) - \nabla E_{ext}(v(s)) = 0, \quad (3.9)$$

where v_s is a derivative of v with respect to s . To find a solution to Equation 3.9, the snake is made dynamic by treating v as a function of time t which leads to

$$\frac{\partial v(s, t)}{\partial t} = \frac{\partial}{\partial s}(\alpha(s)\frac{\partial v(s, t)}{\partial s}) - \frac{\partial^2}{\partial s^2}(\beta(s)\frac{\partial^2 v(s, t)}{\partial s^2}) - \nabla E_{ext}(v(s, t)). \quad (3.10)$$

A solution to Equation 3.10 is found by discretising it and solving the discrete system iteratively [XP98]. When the term $\frac{\partial v(s, t)}{\partial t}$ approaches 0, the snake's energy function reaches its minimum and is expected to have converged on the object boundary.

The traditional parametric active contour model has two limitations in object boundary estimation. First, the initial snake curve must be initiated close to a desirable object boundary or else it won't converge to the object boundary. Second, the snake curve fails to detect concave boundaries [XP98], as shown in Figure 3.10. Several methods have been developed to overcome

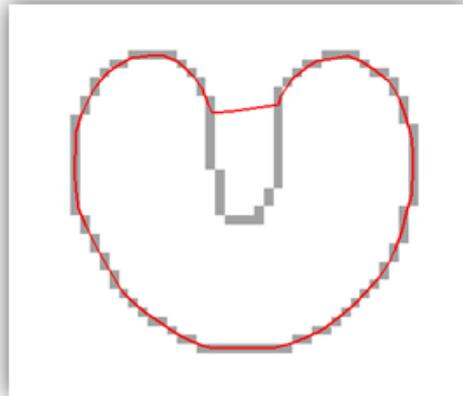


Figure 3.10: Traditional parametric active contour model applied to an object with a concave boundary [XP98].

these limitations in the parametric active contour model [KEA10]. In the following paragraphs, we discuss two modified versions of the parametric active contour model which have been developed to overcome these limitations, namely, the balloon and GVF models .

Balloon Model

To overcome the snake initialisation limitation, Cohen [Coh91] introduced a balloon concept in the parametric active contour model in which an additional energy is added to the external energy term which pushes a snake curve towards an object boundary. In the balloon model, the snake curve behaves like a balloon which is inflated by the additional energy. The balloon

energy acts in the normal direction to a point on the curve which makes the behaviour of the snake curve more dynamic. The modified external energy term in the parametric active contour model can be described as

$$E_{ext} = \kappa_1 n(s) - \kappa |\nabla f(v(s))|^2 \quad (3.11)$$

where κ_1 is the weight parameter for the balloon energy and $n(s)$ is the unit vector normal to the snake curve at point $v(s)$. If the sign of κ_1 is negative, it will have a deflating effect instead of an inflating effect over the snake curve.

The balloon energy pushes the snake curve towards object boundaries while the image gradient based energy attracts the snake curve toward the object boundaries, as shown in Figure 3.11. If the image gradient based en-

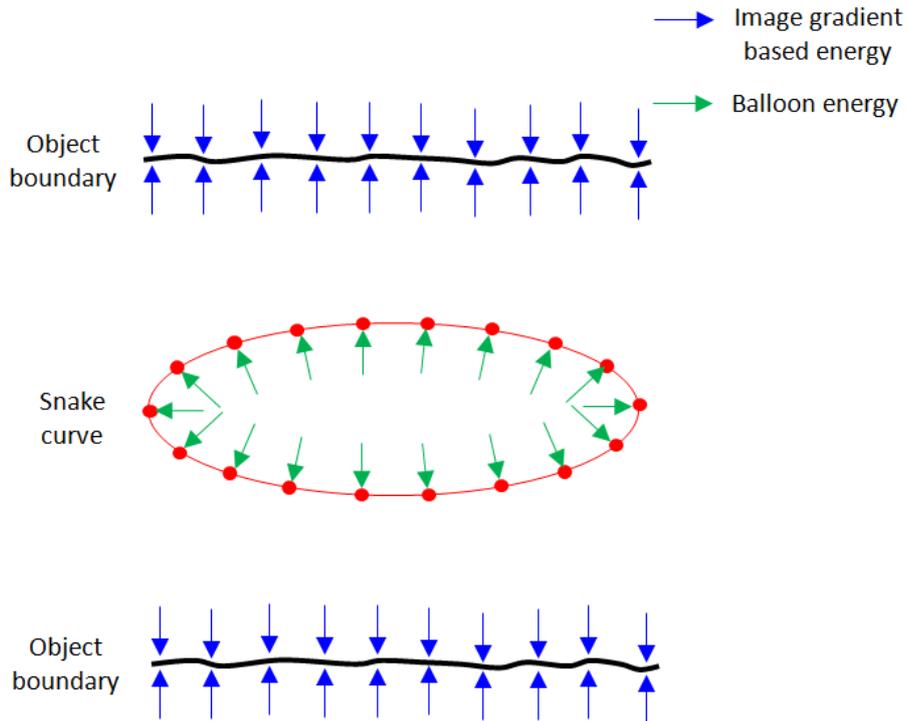


Figure 3.11: Balloon external energy added to the snake curve.

ergy at a point is weaker than the balloon energy, the snake curve passes beyond it. It stops at a point for which the image gradient based energy is higher than the balloon energy. This can have the added benefit of overcoming spurious noise in the data while detecting the object boundaries [SHB08].

The primary advantage of the balloon model is that it increases the movement range of the snake curve towards the object boundaries. It overcomes the snake curve initialisation drawback in the traditional parametric active contour model but it does not provide a solution for the concave boundary convergence problem. The value of balloon energy is also dependent on the image gradient energy and noise in the data. To date, there has been no method to automatically select this value.

GVF Model

Xu et al. [XP97] introduced a GVF external energy in the parametric active contour model in an attempt to detect the concave boundaries. It is based on diffused gradient vectors of the object boundaries. The GVF energy is described as the energy field $V(x, y) = (u(x, y), v(x, y))$, where u and v are its vector components in the x, y plane of an image. It minimises an energy function

$$E = \int \int \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |V - \nabla f|^2 dx dy. \quad (3.12)$$

Noise in the data can prohibit the GVF energy from being calculated effectively. To control the impact of this noise, a regularisation parameter, μ , is used to balance the first term, $\mu(u_x^2 + u_y^2 + v_x^2 + v_y^2)$ and the second term, $|\nabla f|^2 |V - \nabla f|^2$ of the Equation 3.12. An increased noise in the image will require a higher value for μ . When ∇f is small or negligible, the energy func-

tion is dominated by the first term which is a sum of the squares of partial derivatives of the vector components. When ∇f is large, the second term dominates and minimises the energy function, when $V = \nabla f$. It results in a smoothly varying energy being produced over a homogeneous image region while not affecting the gradient energy along the object boundary.

The GVF energy is obtained by solving the following Euler equations [XP98]

$$\mu \nabla^2 u - (f_x^2 + f_y^2)(u - f_x) = 0, \quad (3.13)$$

$$\mu \nabla^2 v - (f_x^2 + f_y^2)(v - f_y) = 0, \quad (3.14)$$

where ∇^2 is a laplacian operator and f_x, f_y are the gradient components along the X and Y axis of the image. Equations 3.13 and 3.14 are solved by treating u and v as functions of time t such that

$$u_t(x, y, t) = \mu \nabla^2 u(x, y, t) - (f_x(x, y)^2 + f_y(x, y)^2)(u(x, y, t) - f_x(x, y)), \quad (3.15)$$

$$v_t(x, y, t) = \mu \nabla^2 v(x, y, t) - (f_x(x, y)^2 + f_y(x, y)^2)(v(x, y, t) - f_y(x, y)). \quad (3.16)$$

Equations 3.15 and 3.16 are known as generalised diffusion equations [XP98]. After estimating the value of $V(x, y, t)$, the external energy term, $\nabla E_{ext}(v(s, t))$, in Equation 3.10 is replaced giving us

$$\frac{\partial v(s, t)}{\partial t} = \frac{\partial}{\partial s}(\alpha(s) \frac{\partial v(s, t)}{\partial s}) - \frac{\partial^2}{\partial s^2}(\beta(s) \frac{\partial^2 v(s, t)}{\partial s^2}) + V. \quad (3.17)$$

The GVF energy attracts the snake curve toward the object boundaries, as shown in Figure 3.12. The diffused energy allows the snake curve to detect the concave boundaries [XP98], shown in Figure 3.13. It also helps increase the movement range of the snake curve, which does not require it to be grown

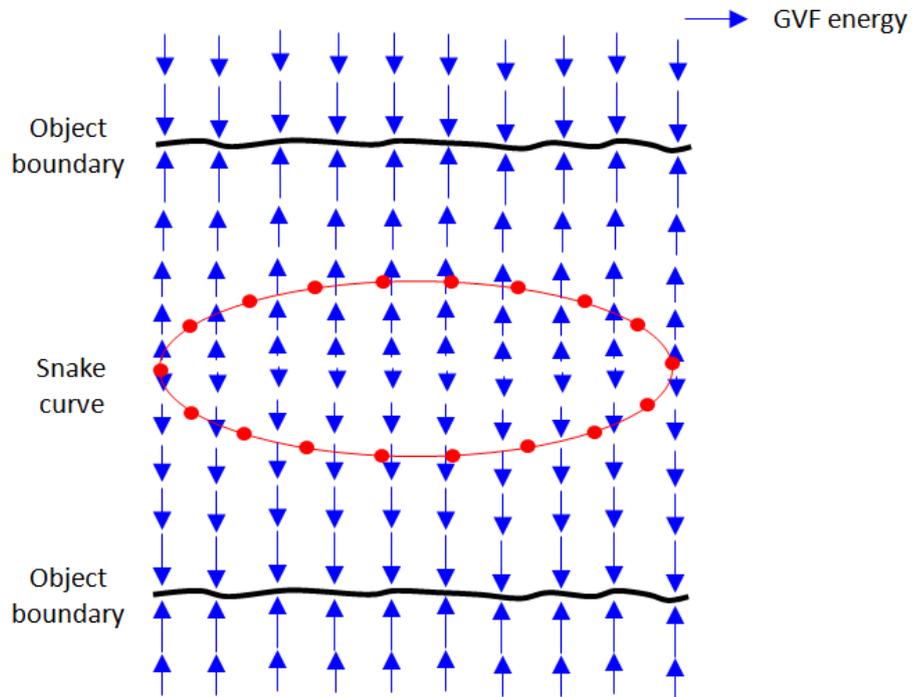


Figure 3.12: Parametric ellipse snake curve initialised on GVF image of road object boundaries.

close to a desirable object boundary. The GVF model can overcome both the limitations in the traditional parametric active contour model. However, its ability to overcome the snake initialisation problem is limited. For this reason, the balloon energy term is also required particularly in the case where there is a relatively large distance between the initial snake and the object boundaries.

The parametric active contour model presents a robust segmentation approach, which is implemented based on energy computations. The computation of external energy relies on the input gradient image having strong boundaries. It is for this reason that the selection and implementation of the thresholding function is significant, as detailed in Sections 3.1.1 and 3.1.2. The parametric active contour model often requires prior estimation of the

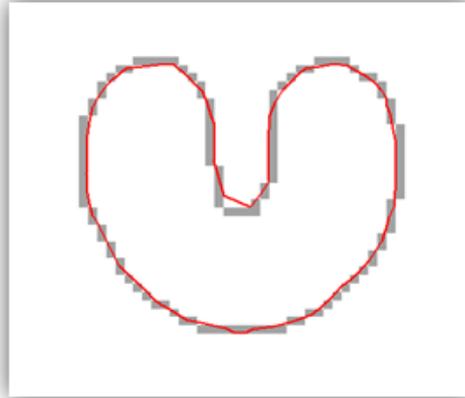


Figure 3.13: GVF active contour model applied to an object with a concave boundary [XP98].

energy weight parameters. To date, there has been no method to automatically select these parameters, which is required in order to make efficient use of the parametric active contour model for the purpose of segmentation.

3.1.3.2 Geometric Active Contour Model

The geometric active contour model is based on a level-set and curve-evolution theory [CCCD93, MSV95]. The primary advantage of the geometric active model is that it does not require any parameters for its implementation. A geometric curve is represented implicitly as a level-set and is evolved based on geometric computations, with its speed locally dependent on the image data [SHB08]. The level-set refers to a set of point variables for which their function is equal to some constant value. For example, the level-set of the variables (x, y, z) can be the sphere $x^2 + y^2 + z^2 = r^2$, with centre $(0,0,0)$ and radius r [Wei12].

In the geometric model, the curve is represented with a zero level-set $\psi(x, y, t)$, shown in Figure 3.14. The geometric curve evolves in time t on an

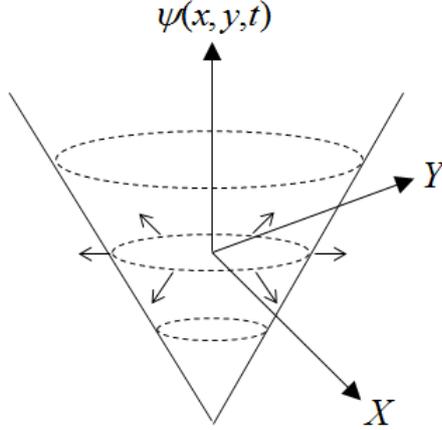


Figure 3.14: Geometric active contour model.

image with a speed function,

$$\frac{\partial \psi}{\partial t} = c(k + V_0)|\nabla \psi|, \quad (3.18)$$

where c is given by

$$c = \frac{1}{1 + |\nabla(G_\sigma * I)|} \quad (3.19)$$

and is the potential energy derived from the input image, I . This parameter serves as the stopping term for the curve at the object boundary. The term $\nabla(G_\sigma * I)$ is the gradient of the Gaussian blurred image with standard deviation of the Gaussian distribution, σ . k is curvature of the curve, which makes it smoother, V_0 is a constant which has the effect of controlling the shrinking or expansion of the curve and $\nabla \psi$ is a gradient of the level-set, ψ . The term $c(k + V_0)$ determines the speed with which the curve evolves along its normal direction.

The geometric curve does not stop at weak or indistinct boundary points due to its evolution speed but continues its movement with little or no energy drawing it back [XYP00]. To overcome this problem, an extra term can be

added to Equation 3.18 as follows

$$\frac{\partial\psi}{\partial t} = c(k + V_0)|\nabla\psi| + \nabla c\nabla\psi. \quad (3.20)$$

This additional stopping term $\nabla c\nabla\psi$ is used to pull back the curve in case it overpasses a weak object boundary.

The advantage of using the geometric active contour model is its ability to change a curve’s topology in accordance with the shape of the object during the curve evolution process. This can be useful in object tracking and motion detection applications. The geometric active contour model is generally not useful for noisy data where the evolved curve can be accompanied by topological inconsistencies [SHB08]. The implementation of the geometric active contour model is based on geometric computations which results in it being a computationally expensive process [KEA10].

3.1.3.3 Applied Parametric Active Contour Models

In our road edge extraction algorithm, we use the parametric active contour model as its implementation is less computationally expensive in comparison with the geometric active contour model and it gives us structured control on the snakes elasticity and stiffness. We make novel use of the balloon and GVF parametric active contour models. We combine both the models in our algorithm, such that the balloon energy pushes the snake curve and the GVF energy attracts the snake curve towards the object boundaries, as shown in Figure 3.15. The use of this combined approach in our algorithm allows the snake curve to detect the road edges automatically and more efficiently, as shown in Figure 3.16. The internal energy weight parameters used in the parametric active contour models can control the shape of the desired

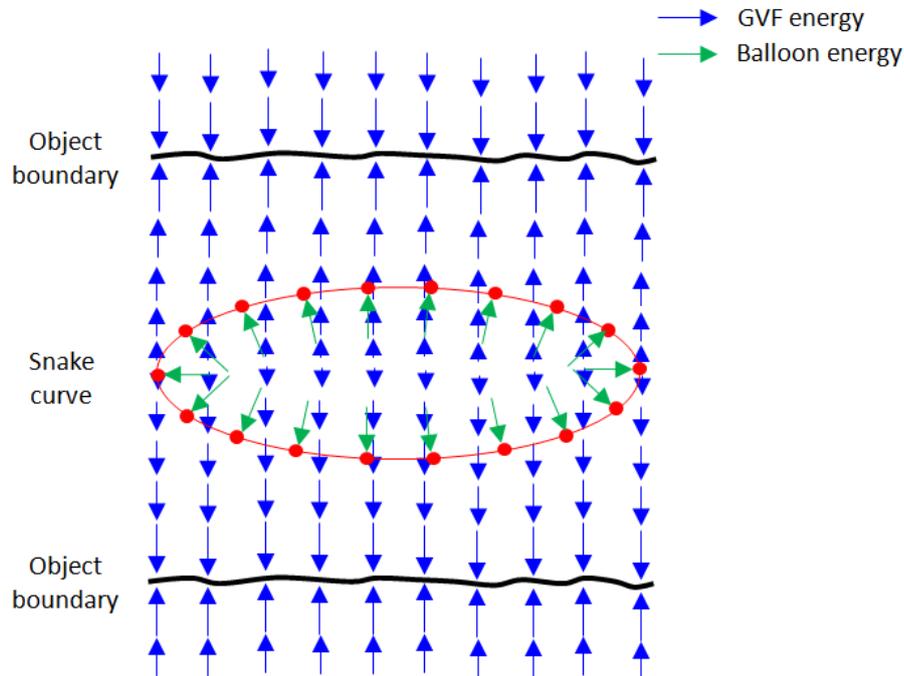


Figure 3.15: Balloon energy pushes the snake curve and GVF energy attracts the snake curve toward the object boundaries.

object. The topology of the left and right edges in the road sections do not vary much, thus, we can use the same internal energy weight parameters for each road section. We estimate the external energy weight parameters empirically and then apply the same values for each road section. In this manner, we have created a fully automated road edge extraction algorithm. In the next section, we discuss the range dependent thresholding used for developing the road marking extraction algorithm.

3.2 Road Marking Extraction

We developed a road marking extraction algorithm to extract the road markings from terrestrial mobile LiDAR data. To effectively do this we first need

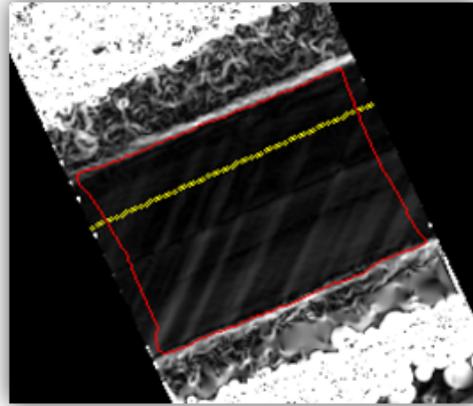


Figure 3.16: Final position of the snake curve.

to find the LiDAR points belonging to the road surface. We apply our road edge extraction algorithm to estimate the road boundaries in the form of a snake curve. This snake curve is then used to identify the LiDAR points that belong to the road surface. We convert the LiDAR intensity and range attributes from these road surface points into 2D raster surfaces and then apply a range dependent thresholding to extract the road markings, as shown in Figure 3.17.



Figure 3.17: Range dependent thresholding in the road marking extraction algorithm.

In our algorithm, we have developed range dependent thresholding func-

tion to deal with the factors which affect the intensity returns. These factors are the illuminated surface characteristics, distance from the laser scanner to the illuminated surface and the incidence angle of the laser pulse. The use of a threshold itself clusters data based on the surface characteristics, while the range dependent thresholding takes account of the two other factors. We apply multiple threshold values to the intensity raster surface as a function of the range across the road surface. The intensity values larger than the threshold limit are accepted as road markings, while the smaller intensity values are rejected. We then use binary morphological operations to complete the shapes of extracted road markings and to remove noise that is introduced through the application of a thresholding function.

We estimate a single threshold value empirically and then create a formula for applying the threshold values to the intensity raster surface as a function of the range. Through the use of range dependent thresholding, we have automated our road marking extraction algorithm. An example of the road markings extracted from the intensity raster surface using the range dependent thresholding is shown in Figure 3.18. We will present our road

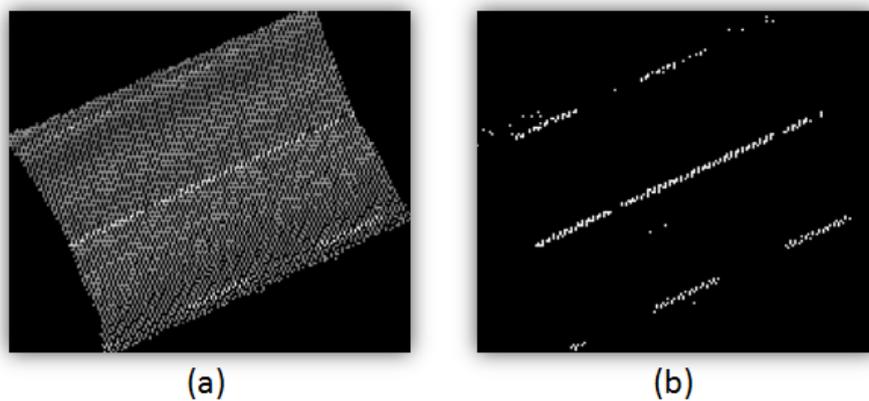


Figure 3.18: Range dependent thresholding: (a) input 2D intensity raster surface and (b) extracted road markings.

marking algorithm in detail in Chapter 5. In the next section, we describe a surface grid fitting approach used for developing the road roughness estimation algorithm.

3.3 Road Roughness Estimation

Our road roughness estimation algorithm requires the 3D LiDAR values of the road surface and a surface grid fitting approach. We identify the LiDAR points that belong to the road surface using the output snake curve. We use the RANSAC algorithm to fit a surface grid to the 3D LiDAR points as shown in Figure 3.19. We will describe our road roughness estimation

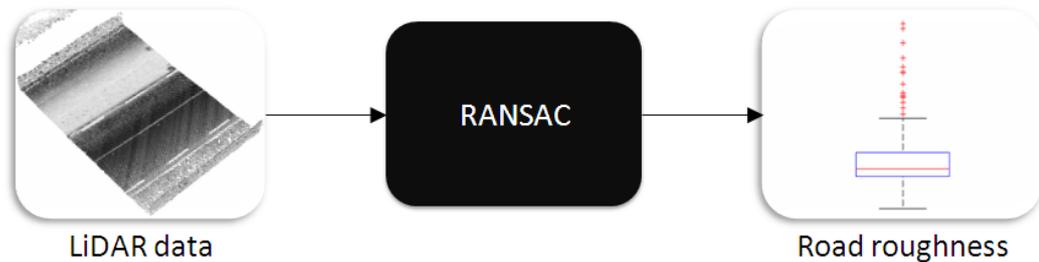


Figure 3.19: RANSAC in the road roughness estimation algorithm.

algorithm in detail in Chapter 6. In the following section, we discuss the RANSAC surface grid fitting, which is used to develop our algorithm.

3.3.1 RANSAC

The RANSAC algorithm was developed by Fischler et al. [FB81] and is used to provide a more robust fitting of a model to input data in the presence of data outliers. Its application can be found in computer vision, image processing, stereo camera calibration, panoramic image stitching and surface

plane extraction. The RANSAC algorithm has also been used to extract building facades and roof planes from the LiDAR data [FNSZ03, BLGTK07, EAH08].

Unlike conventional model fitting techniques that use as much data as possible to obtain an initial solution, the RANSAC algorithm uses the smallest set of initial data required to fit a model and enlarges this set with compatible data [Der10]. If there are enough compatible data, RANSAC can improve the estimation of the model, without having to deal with the data outliers. We will now describe the RANSAC algorithm in detail.

Suppose there are n points in a dataset, $X = x_1, x_2, \dots, x_n$. A minimum required number of m points are randomly selected, such that $m \leq n$, to fit a least-square model M . The least-square model is fitted to the points based on minimising the sum of square residuals which are the difference between the actual points and the fitted points. The model M is used to estimate data points in X which are within an error tolerance parameter, ϵ . These estimated data points are called consensus points. If the number of consensus points is equal to or larger than a threshold, t , then a new least-square model M^* is fitted to these points. Otherwise, the whole process is repeated beginning with a random selection of m points. After some pre-set number of iterations, K , if the number of consensus points equal to or larger than t is not found, then either the model fitted with the largest number of consensus points is accepted or the process is terminated unsuccessfully.

The RANSAC algorithm uses three specified parameters: ϵ , K and t . The ϵ parameter specifies an error tolerance which is used to determine whether a data point should be considered as a consensus point or not. Its value is calculated experimentally by fitting a model to the randomly selected data

points and then measuring the errors between the fitted model and the data points. The value of ϵ is then set as the sum of the mean and standard deviation of the measured errors [FB81]. This allows us to set ϵ automatically. The K parameter describes the number of iterations required to repeat the process to find the best consensus points. p is the probability that any selected data point is within the error tolerance value ϵ and q is the desired probability for getting a good set of data points, then the value of K is calculated as [FB81]

$$(1 - p^m)^K = (1 - q) \quad (3.21)$$

$$K = \frac{\log(1 - q)}{\log(1 - p^m)}. \quad (3.22)$$

The t parameter represents the number of consensus points required for estimating a correct model. Its value is calculated as [Col07]

$$t = p.n. \quad (3.23)$$

An example of a RANSAC surface grid fitted to LiDAR points using $\epsilon = 0.065$ m, $t = 7860$ and $K = 35$ is shown in Figure 3.20. We then find the elevation residual values by calculating the difference between the elevation of the LiDAR points and the surface grid points over a given area. We determine the standard deviation of the elevation residuals in each cell along the navigation track of the mobile van. These in turn provide an estimation of the road roughness. In the next section, we conclude with a discussion on the various approaches presented in this chapter.

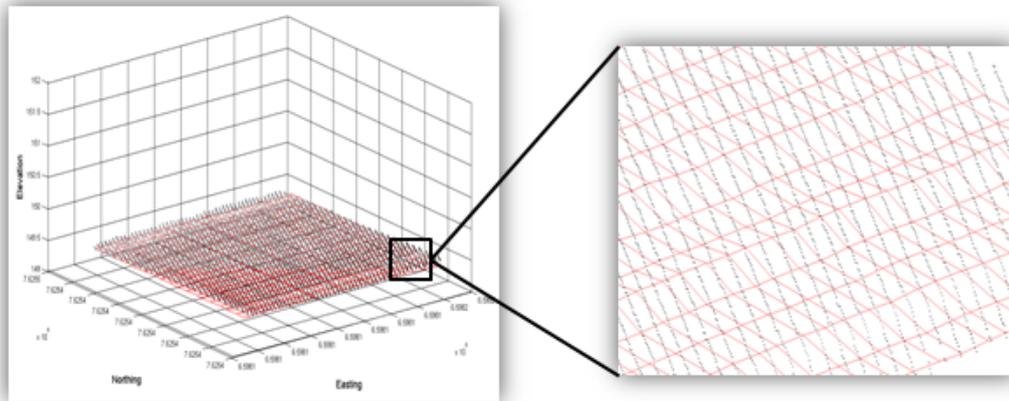


Figure 3.20: RANSAC surface grid fitted to the LiDAR points, with an inset picture of a magnified portion.

3.4 Conclusion

In this chapter, we presented the approaches used to develop algorithms for extracting road edges, road markings and road roughness from terrestrial mobile LiDAR data. These approaches were chosen due to their usefulness for automated and precise extraction of road features. We discussed the theoretical background of these approaches and described how they are used in our algorithms.

In our road edge extraction algorithm, we first convert the selected LiDAR attributes into 2D raster surfaces and apply various segmentation approaches to extract the road edges. We apply hierarchical thresholding to determine suitable objects in the raster surfaces. It segments an input image by thresholding a hierarchy of low to high resolution versions of the input image. An advantage of using the hierarchical thresholding approach is its ability to deal with noisy images. It minimises any background noise in the input image and so, provides a better estimation of the objects. The threshold and mask

size parameters are fixed which allows the hierarchical thresholding to be applied automatically. We apply Canny edge detection to the hierarchical thresholded objects to determine their boundaries. The binary nature of the input enables the automated computation of two threshold values. Through the use of both hierarchical thresholding and Canny edge detection, we have made the parameter estimation process more robust, which in turn facilitates integration with our automated road edge extraction algorithm. These object boundaries are input into the next stage of our algorithm, the active contour model. We use the parametric active contour which is represented implicitly as a controlled spline and is implemented based on energy computations. The parametric active contour model is less computationally expensive than the geometric active contour model and allows for more explicit control over the snake curve's growth. We make novel use of the modified versions of the parametric active contour models, the balloon and GVF models. We combine both models such that the balloon energy pushes the snake curve, while the GVF energy attracts the snake curve toward the object boundaries. We estimate the internal and external energy weight parameters empirically and then apply the same values for each road section. This enables the application of the balloon and GVF models to be fully automated in our algorithm.

In our road marking extraction algorithm, we convert the LiDAR road surface intensity and range attributes into 2D raster surfaces. We apply range dependent thresholding to the intensity raster surface which takes account of the factors that affect the intensity return values. We create a formula to automatically apply multiple threshold values as a function of the range across the road surface. We use morphological operations to deal with the incomplete shapes of the extracted road markings and noise introduced through

the use of a thresholding function.

In our road roughness estimation algorithm, we use the RANSAC algorithm to fit a surface grid to the 3D LiDAR points. The RANSAC algorithm is initiated by selecting a minimum number of points to fit a model. This sample is enlarged using compatible points to provide an improved estimation of the model. Data outliers are excluded using this bottom up surface grid fitting approach. The parameters used for this algorithm are calculated experimentally and modified using statistical computations. This has allowed us to produce an automated road roughness algorithm. We calculate the difference between the elevation of the LiDAR points and the surface grid points in order to compute the elevation residuals. We calculate the standard deviation of these elevation residuals in each cell, which provides an estimation of the road roughness. In the next chapter, we present our road edge extraction algorithm.

Chapter 4

Road Edge Extraction

Terrestrial mobile laser scanning systems provide 3D point cloud data which can be useful for extracting road features. Accurate knowledge of road edges increases the reliability and precision of road features extraction. LiDAR data provides elevation, intensity and pulse width information which can be useful in distinguishing the road surface from grass-soil edges in rural road sections and kerb edges in urban road sections.

In this chapter, we present an algorithm based on the combined use of the GVF and balloon parametric active contour models, to extract the road edges from terrestrial mobile LiDAR data. The algorithm outputs 3D left and right road edges extracted from the LiDAR data. In Section 4.1, we provide a stepwise description of the road edge extraction algorithm. This description provides details of 2D raster surfaces generation from the LiDAR attributes. We describe an estimation of the GVF and balloon external energy terms. The internal energy terms provided to a snake curve are also discussed. We present our approach to initialise the snake curve based on the navigation track of the mobile van along the road section. We explain

the use of batch processing to extract the left and right road edges from final position of the snake curves. In Section 4.2, we detail our automated approach for validating the extracted road edges. In Section 4.3, we analyse the most applicable method and optimal input parameters, which allow for the automation of our road edge extraction algorithm. In Section 4.4, we test our road edge extraction algorithm on different types of road sections in both manual and automated modes. We analyse the results of road edge extraction using our validation approach. In Section 4.5, we conclude by discussing merits and demerits of our road edge extraction algorithm.

4.1 Algorithm

A work flow of the road edge extraction algorithm is shown in Figure 4.1. In the following sections, we describe the various processing steps involved in our algorithm.

4.1.1 Terrain Pyramids Generation

We convert the LiDAR elevation, reflectance and pulse width attributes into 2D raster surfaces. This step ensures a more computationally efficient approach to road edge extraction. We use the reflectance attribute which attempts to represent normalised intensity values with respect to standard reflector targets [RSPU10]. LiDAR data can be affected by high frequency noise due to a lower vertical accuracy of points relative to their horizontal sampling distance [Cra09]. For example, if the vertical accuracy of the data points varies from 10 cm to 15 cm, then it can lead to a height difference of 20 cm to 30 cm in between two adjacent points. This height difference

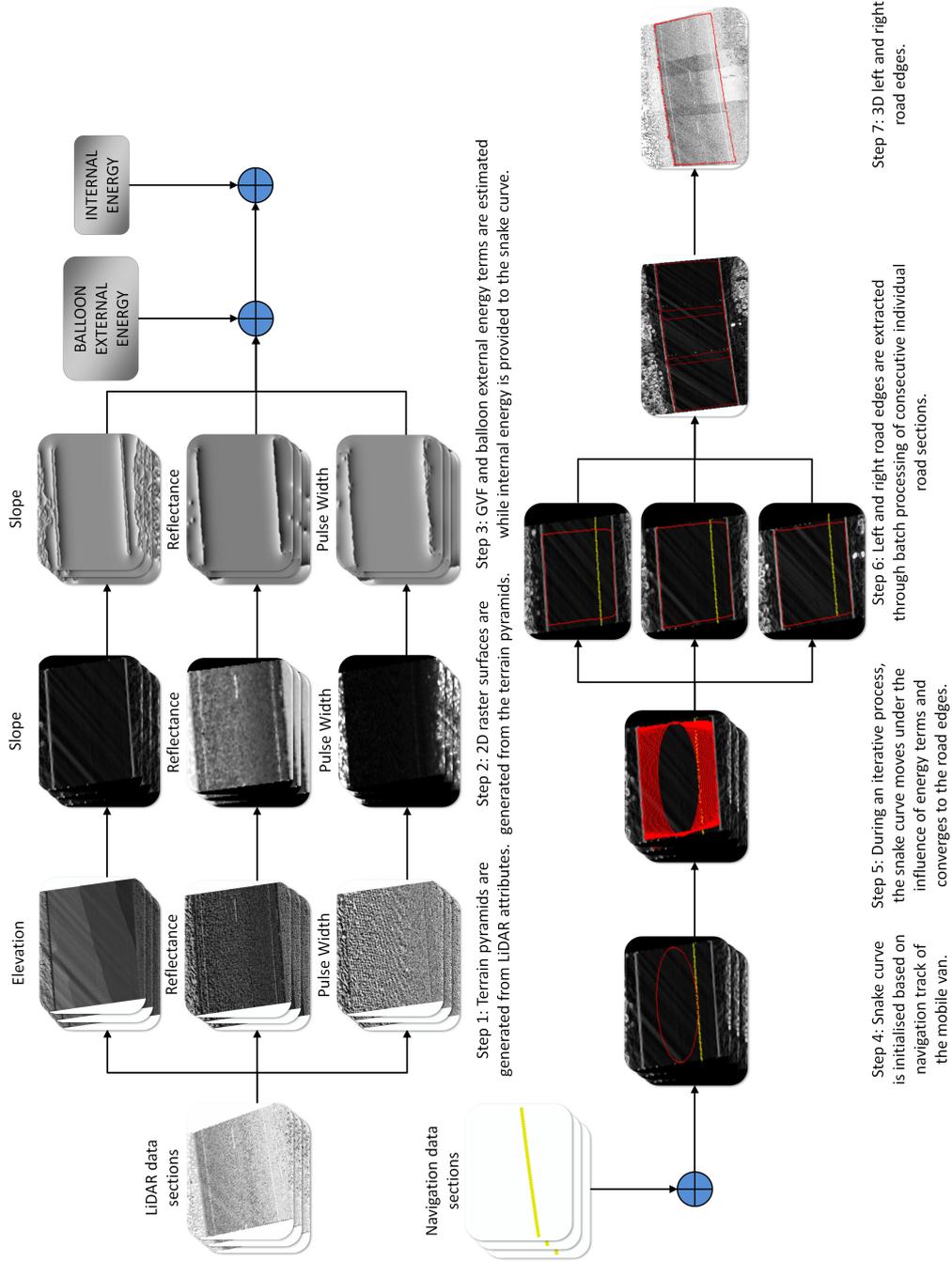


Figure 4.1: Road edge extraction algorithm.

becomes unusual for the data points that are only 1 m or less apart horizontally. Similarly, LiDAR data can suffer from unusual intensity and pulse width values. This high frequency noise present in LiDAR data can lead to the generation of noisy raster surfaces.

To generate smooth raster surfaces, we minimise the effect of this noise with a point thinning process [Cra09]. Point thinning is used to generate multi resolution terrain pyramids by reducing the number of data points required to represent a terrain model in each pyramid level [ESR10]. In Step 1 of our road edge extraction algorithm, we generate terrain pyramids from the LiDAR attributes. The point thinning operation is applied based on a window size filter method in which the data points are partitioned into equally sized window areas. In the first level terrain pyramid, the window size is twice the selected cell size of the raster surface, while the window size in the subsequent levels is increased by a power of two. A detailed analysis on the selection of an optimum cell size is provided in Section 4.3.2. In each window of the pyramid level, a data point nearest to the mean value is selected as representative of the terrain model in each pyramid level. Thus, the full resolution terrain pyramid corresponds to the highest resolution terrain model, the first level corresponds to the second-highest resolution terrain model while the last level corresponds to the lowest resolution terrain model. Examples of the elevation, reflectance and pulse width terrain pyramids generated from the LiDAR attributes are shown in Figure 4.2.

4.1.2 2D Raster Surfaces Generation

In Step 2 of our algorithm, we generate 2D raster surfaces from the first level terrain pyramids using natural neighbourhood interpolation. The first

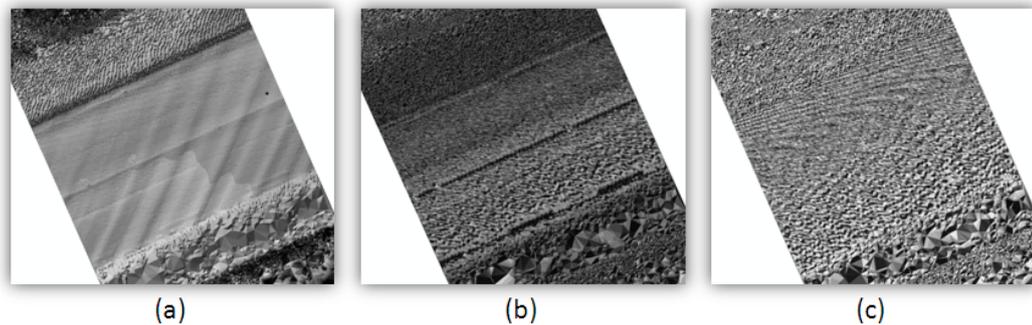


Figure 4.2: Terrain pyramids generated from the LiDAR attributes: (a) elevation, (b) reflectance and (c) pulse width.

level terrain pyramid is found to be useful for generating the smooth raster surface, as it minimises the noise effect without affecting much of the object details and accuracy [Cra09]. A detailed analysis on the use of the first level terrain pyramid along with natural neighbourhood interpolation is provided in Section 4.3.1. In the natural neighbourhood interpolation method, the thinned LiDAR points are partitioned into Voronoi polygons which are created with each polygon constituting a single point and every location within the polygon is closer to its constituted single point than to any other point [ESR11]. A raster surface with its selected cell size is laid over the Voronoi polygons. The value of each raster cell is then interpolated based on the proportion of overlapping areas between the raster cell and Voronoi polygons. An example of a raster cell laid over the Voronoi polygons is shown in Figure 4.3. Thus, the 2D elevation, reflectance and pulse width raster surfaces are generated from their respective terrain pyramids using this natural neighbourhood interpolation.

We estimate the slope values from the elevation raster surface as the rate of change in elevation of the raster cells to its neighbours [ESR10]. A plane

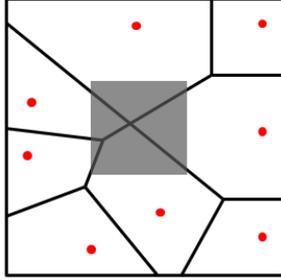


Figure 4.3: Raster cell laid over the Voronoi polygons constituting the thinned LiDAR points in the natural neighbourhood interpolation.

is fitted to the z elevation values of a 3×3 neighbourhood around each raster cell and then its slope value is calculated as the rate of change of z -values in the horizontal x and the vertical y direction, which can be described as

$$\arctan\left(\sqrt{\left(\frac{dz}{dx}\right)^2 + \left(\frac{dz}{dy}\right)^2}\right). \quad (4.1)$$

The slope, reflectance and pulse width raster surface values are normalised with respect to their global minimum and maximum values, and converted to an 8-bit data type. This allows for a two-way transformation between the 8-bit values and their original LiDAR values for all road sections. This in turn will allow for the use of a single threshold value for all road sections. Examples of slope, reflectance and pulse width raster surfaces generated from their respective terrain pyramids are shown in Figure 4.4.

4.1.3 Snake Energy Estimation

A snake curve is defined within a 2D raster surface domain that moves under the influence of internal and external energy. In Step 3 of our road edge extraction algorithm, we estimate these energy terms. The GVF external energy term is estimated as a dense vector energy field from the raster surfaces

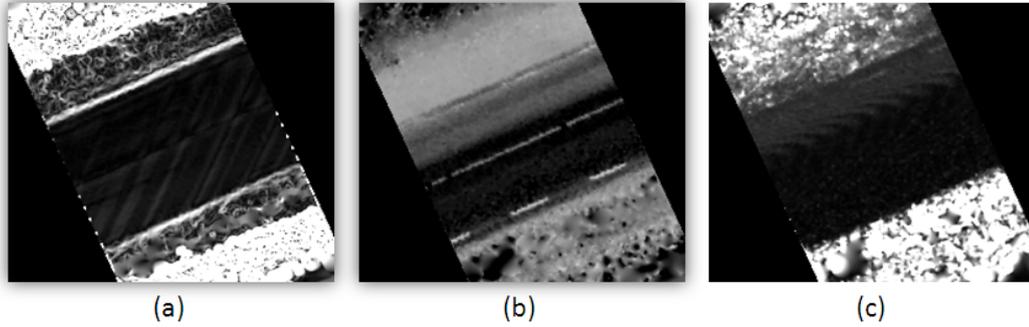


Figure 4.4: 2D raster surfaces generated from their respective terrain pyramids using natural neighbourhood interpolation: (a) slope, (b) reflectance and (c) pulse width.

which attract the snake curve towards the object boundaries. To estimate the GVF energy, the object boundaries are determined from the slope, reflectance and pulse width raster surfaces through the consecutive use of hierarchical thresholding and Canny edge detection, described in Chapter 3. The mask size M and threshold T_{slope}, T_{ref} and T_{pw} parameters applied to the slope, reflectance and pulse width raster surfaces in the hierarchical thresholding are found empirically. Their values are fixed for all the road sections, which allows for the fully automatic application of hierarchical thresholding. The upper threshold T_1 and lower threshold T_2 , parameters used in the Canny edge detection are selected based on the output binary cell values obtained from the hierarchical thresholding. This ensures that the Canny edge detection is fully automated in our algorithm. The GVF external energy terms are estimated by iteratively diffusing the gradient vector values of the object boundaries determined from the slope, intensity and pulse width raster surfaces. This iterative process is used to estimate the appropriate GVF energy terms required for attracting the snake curve towards the object boundaries. Finally, the balloon energy is included in the external energy by providing

a weight to the normal unit vector of the snake curve that helps push it outwards.

Internal energy is provided to the snake curve by adjusting its elasticity and stiffness properties. These properties are adjusted with the α and β weight parameters, while the γ weight parameter is used to control the snake curve step size in one iteration. The snake's energy function E_{snake} can be described as

$$E_{snake} = E_{int} + (\kappa_1 \times E_{slope}) + (\kappa_2 \times E_{reflectance}) + (\kappa_3 \times E_{pulsewidth}) + (\kappa_4 \times E_{balloon}) \quad (4.2)$$

where E_{int} is the internal energy term. E_{slope} , $E_{reflectance}$ and $E_{pulsewidth}$ are the GVF external energy terms estimated from the slope, reflectance and pulse width raster surfaces respectively, while κ_1 , κ_2 and κ_3 are their respective weight parameters. $E_{balloon}$ is the balloon external energy term and κ_4 is its weight parameter. These weight parameters are found empirically after examining several combinations. All parameters with the exception of κ_4 are the same for all road sections. κ_4 is modified in some cases due to the inconsistent point density in our data which is uncommon in professional surveyed terrestrial mobile LiDAR data.

4.1.4 Snake Curve Initialisation

In Step 4 of our algorithm, we initialise the snake curve over a 2D raster surface based on the navigation track of the mobile van along the road section. We choose a parametric ellipse for initialising the snake curve due to the dimensions of the LiDAR data, shown in Figure 4.5.

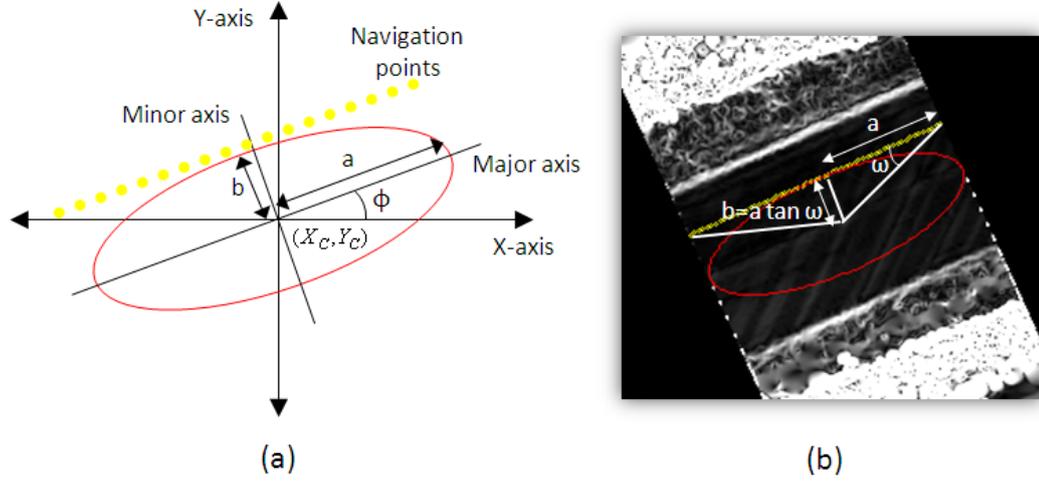


Figure 4.5: Snake curve initialisation in (a) parametric ellipse form and (b) centre of the road estimation.

The snake points are expressed in the form of parametric ellipse as

$$X(t) = X_C + a \cos(\delta t) \cos(\phi) - b \sin(\delta t) \sin(\phi) \quad (4.3)$$

$$Y(t) = Y_C + a \cos(\delta t) \sin(\phi) + b \sin(\delta t) \cos(\phi) \quad (4.4)$$

where δt is the parametric angle interval which is used to provide the number of points in the snake curve from 0 to 2π radian, ϕ is the angle between the X-axis and major axis of the ellipse and (X_C, Y_C) is the centre point of the ellipse. The semi-major axis of the snake ellipse, a , is computed as the difference between the middle and the first or last navigation points, described in Figure 4.5(a). The centre of the road is estimated based on the ω angle in between the major axis and slant height from the first or last navigation point, shown in Figure 4.5(b). This angle is found empirically and fixed for the road sections with similar width. The semi-minor axis of the snake ellipse, b , is computed as $a \tan \omega$.

The ϕ angle is calculated from the θ , average heading angle of the mobile van under investigation, described in Figure 4.6. The average heading angle

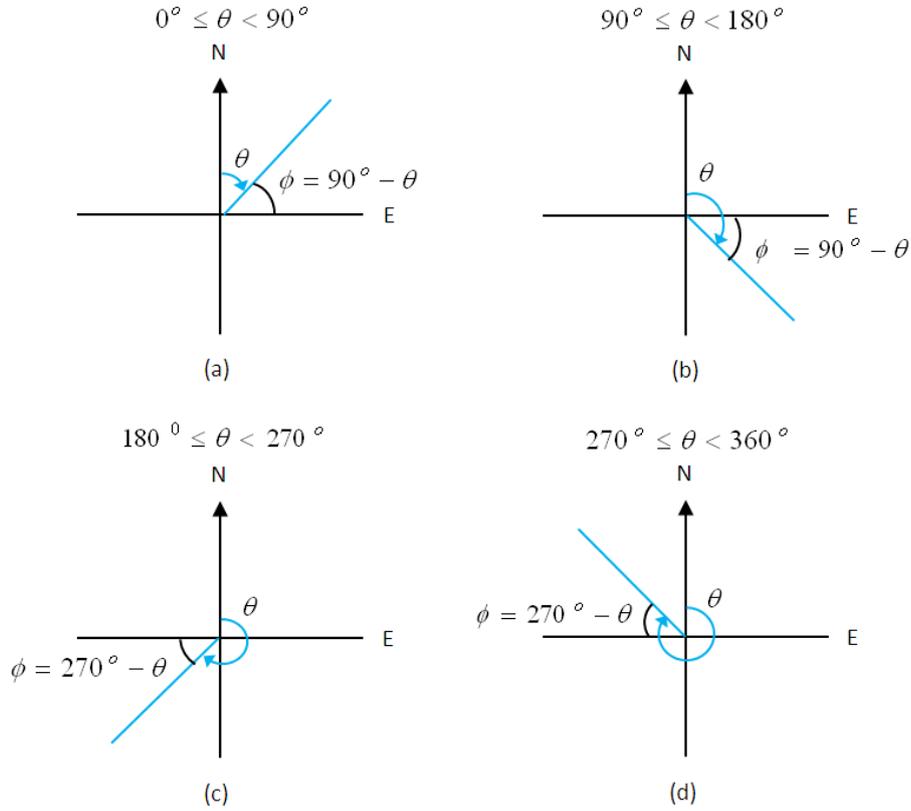


Figure 4.6: ϕ angle is calculated from θ , which is an average heading angle of the mobile van along the road section under investigation that can lie in between (a) 0° and 90° , (b) 90° and 180° , (c) 180° and 270° and (d) 270° and 360° .

is estimated as a mean of the mobile van's heading angle along the road section under investigation. If the value of θ angle lies in between 0° and 90° angles, then the ϕ angle is estimated as $90^\circ - \theta$, shown in Figure 4.6(a). If the θ angle lies in between 90° and 180° angles, then the ϕ angle is estimated as $90^\circ - \theta$, shown in Figure 4.6(b). Similarly, the ϕ angle can be estimated for other possible values of the θ angle, shown in Figure 4.6(c) and (d). Thus, the estimation of ϕ angle is completely automated in our algorithm.

4.1.5 Final Snake Curve

In Step 5 of our road edge extraction algorithm, the snake curve moves under the influence of the internal and external energy terms. During an iterative process it approaches the minimum energy state and converges to the road edges. Examples of the initial, iterative and final positions of the snake curve are shown in Figure 4.7.

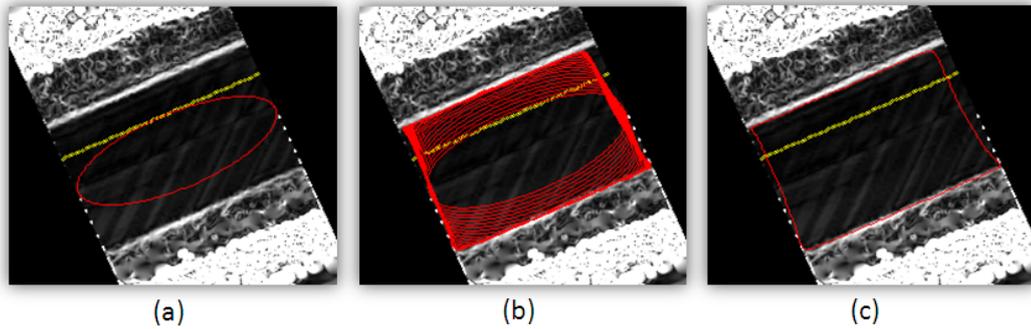


Figure 4.7: Snake curve positions: (a) initial, (b) iterative and (c) final.

4.1.6 Batch Processing

In Step 6 of our algorithm, we extract the left and right road edges using batch processing. We obtain overlapping snake curves by batch processing consecutive individual road sections, where there are small common areas between the adjacent sections. We then find the intersection points between the overlapping snake curves.

To estimate the intersection points, we use the following process. Let us consider that two overlapped snake curves, S_1 and S_2 , intersect at point $P(x, y)$, shown in Figure 4.8. Points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ belong to the S_1 snake curve while points $P_3(x_3, y_3)$ and $P_4(x_4, y_4)$ belong to the S_2 snake

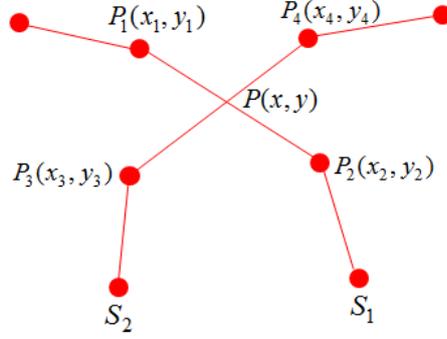


Figure 4.8: Intersection point in between two overlapped snake curves.

curve. The intersecting lines P_1P_2 and P_3P_4 can be described in the parametric form as

$$P = P_1 + m(P_2 - P_1) \quad (4.5)$$

$$P = P_3 + n(P_4 - P_3) \quad (4.6)$$

where m and n are unknown parameters. The above equations can be rewritten as

$$x_1 + m(x_2 - x_1) = x_3 + n(x_4 - x_3) \quad (4.7)$$

$$y_1 + m(y_2 - y_1) = y_3 + n(y_4 - y_3). \quad (4.8)$$

Solving the equations 4.7 and 4.8, for the m and n parameters provide

$$m = \frac{(x_4 - x_3)(y_1 - y_3) - (y_4 - y_3)(x_1 - x_3)}{(y_4 - y_3)(x_2 - x_1) - (x_4 - x_3)(y_2 - y_1)} \quad (4.9)$$

$$n = \frac{(x_2 - x_1)(y_1 - y_3) - (y_2 - y_1)(x_1 - x_3)}{(y_4 - y_3)(x_2 - x_1) - (x_4 - x_3)(y_2 - y_1)}. \quad (4.10)$$

Substituting the values of m and n in their corresponding line equations

provide the intersection point $P(x, y)$ as

$$x = x_1 + m(x_2 - x_1) \quad (4.11)$$

$$y = y_1 + n(y_2 - y_1). \quad (4.12)$$

After estimating the intersection points in between the overlapping snake curves, the non road edge points in between them are removed and only the left and right road edge points remain.

An example of the overlapping snake curves is shown in Figure 4.9, where the 30 m long road section has been processed with 2 m overlap in between the consecutive individual sections. The intersection points in between the overlapping snake curves are estimated and then intersection points with the highest and lowest Northing are selected, as circled in blue in Figure 4.9(a) and (b). In Figure 4.9(a) and (b), the non road edge points in between the circled intersection points are removed clockwise in the first snake curve and anti-clockwise in the next snake curve to provide the left and right road edge points, as shown in Figure 4.9(c). The selection of intersection points with the highest and lowest Northing is arbitrary. However, we will need to investigate the selection of those intersection points which will remove the points in the most irregular snake curve.

4.1.7 3D Road Edges

In Step 7 of our road edge extraction algorithm, we provide the third dimension to the extracted left and right road edge points. We provide the elevation value from the nearest LiDAR point to the road edge point. An example of the 3D left and right road edges is shown in Figure 4.10. In the next section,

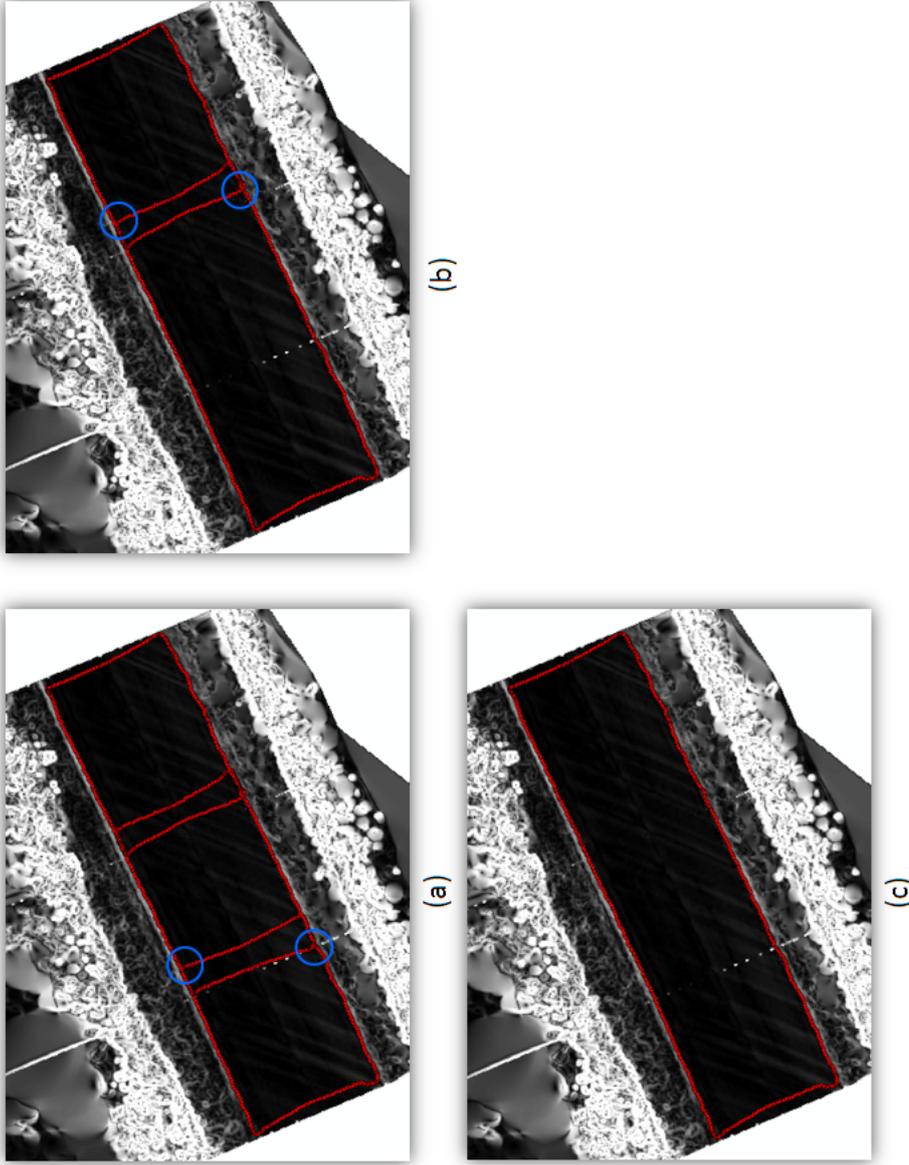


Figure 4.9: Three processed road sections with (a) first intersection points highlighted, (b) second intersection points highlighted and (c) final left and right road edges.

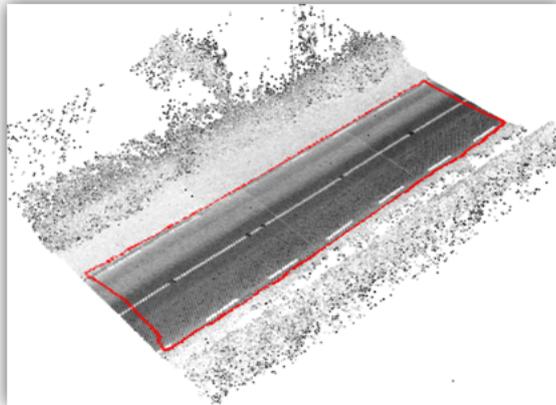


Figure 4.10: 3D left and right road edges.

we present our automated algorithm for validating the extracted road edges.

4.2 Validation Algorithm

We present an automated algorithm for validating the road edges extracted from terrestrial mobile LiDAR data. The developed algorithm is based on their comparison with manually digitised road edges. This comparison is carried out using navigation points that are considered as reference points. A stepwise description of our validation algorithm is provided in Figure 4.11 in which the navigation points are represented with yellow, the extracted road edge points are represented with red, while the digitised road edge points are represented with blue. In Step 1 of our validation algorithm, we select the 2D navigation points, at a 0.5 m interval. The selected navigation reference points are represented with green in Figure 4.11. In Step 2, the heading angle of each navigation point is used to rotate the points towards the Easting direction. The process of calculating the rotation angle from the heading angle is similar to that described in Figure 4.6. Smoothing splines

■ Navigation points
■ Navigation reference points
■ Extracted road edge points
■ Digitised road edge points

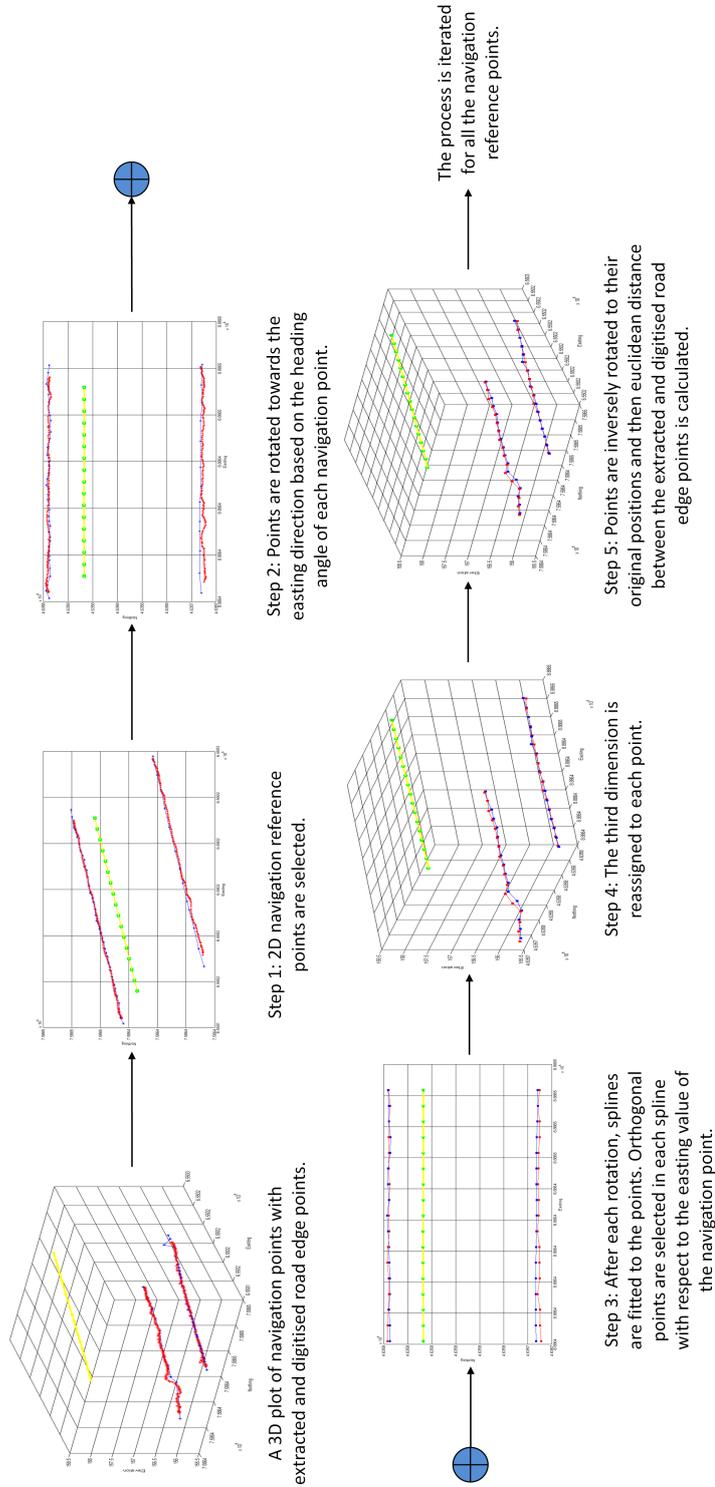


Figure 4.11: Road edge validation algorithm.

are fitted to both the extracted and digitised road edge points after each rotation in Step 3. The smoothing spline is fitted to the points using a minimised function that can be described as

$$p \sum_i (y_i - s(x_i))^2 + (1 - p) \int \left(\frac{d^2s}{dx^2}\right)^2 dx \quad (4.13)$$

where i is the number of points, s is the smoothing spline through the (x, y) points and p is the user specified smoothing parameter in between 0 and 1. A 0 value of p produces a straight line fitting to the points, while a 1 value produces a cubic spline interpolation in which the interpolation is based on a third degree polynomial function. The value of p can be specified based on the noise present in the data. We select orthogonal points in each spline with respect to the Easting value of the navigation point. In Step 4, we reassign the elevation value to the selected road edge points. This is achieved by providing the elevation value from the nearest LiDAR point to the road edge point. In Step 5, the points are inversely rotated to their original positions. We then calculate the euclidean distance between the extracted and digitised road edge points. The positive or negative sign is assigned to the calculated distance based on their Northing values of their rotated positions in Step 3. This process is iterated for all the navigation reference points to calculate the error in our road edge extraction process. In the next section, we present an analysis of our road edge extraction algorithm.

4.3 Automation Analysis

The automation of our road edge extraction algorithm required a detailed analysis of all the input parameters and their relative impacts. In the follow-

ing sections, we present these analyses.

4.3.1 2D Raster Surfaces Generation

This analysis involved the generation of 2D raster surfaces from the full resolution and the first level terrain pyramid using linear and natural neighbourhood interpolation. This comparative analysis was undertaken to investigate which pyramid level and interpolation method generated the most applicable raster surface.

In linear interpolation, the value of a raster cell is directly interpolated from the triangulated planes of the LiDAR points in which it falls. We applied linear interpolation to the full resolution and the first level terrain pyramids, resulting in a generation of elevation raster surfaces from which slope surfaces were estimated as shown in Figure 4.12(a) and (b) respectively. In both the cases, the estimated slope surfaces contained noisy cells due to the high frequency noise present in the LiDAR data.

In natural neighbourhood interpolation, the raster cell value is interpolated based on the proportion of its overlapped areas with Voronoi polygons, as described in Section 4.1.2. We applied natural neighbourhood interpolation to the full resolution and the first level terrain pyramids, resulting in the generation of elevation raster surfaces from which slope surfaces were estimated as shown in Figure 4.12(c) and (d). The natural neighbourhood interpolation along with the first level terrain pyramid, were found to be useful for estimating a smooth slope surface without losing much of the object details and accuracy.

Similarly, noisy reflectance and pulse width raster surfaces were generated using linear interpolation applied to the full resolution and the first level ter-

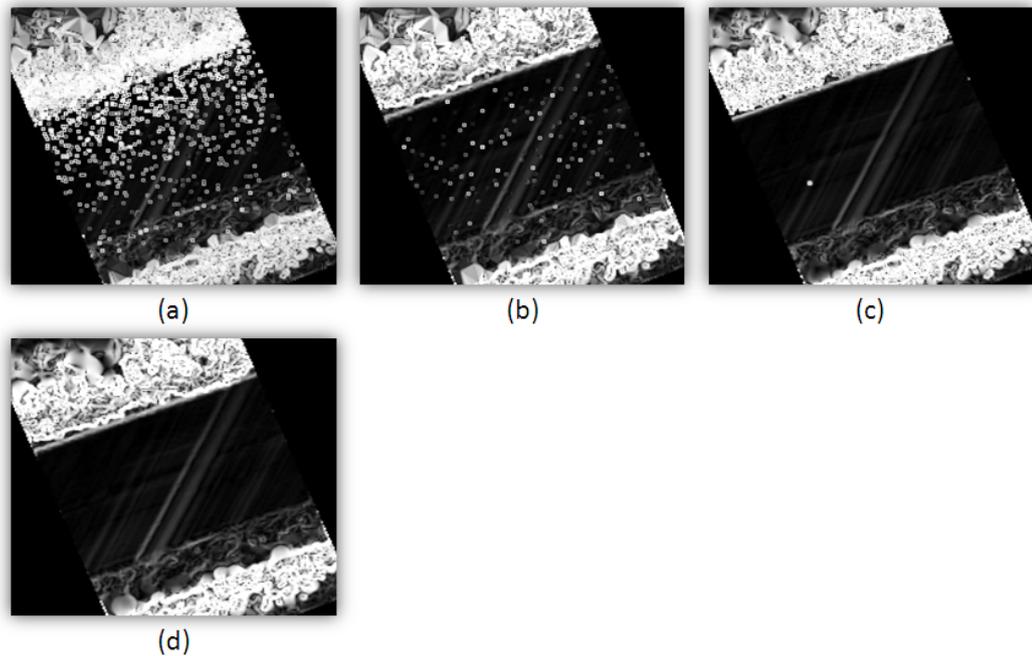


Figure 4.12: Slope surface estimated from the elevation raster surface generated using linear interpolation applied to (a) full resolution, (b) first level terrain pyramid and using natural neighbourhood interpolation applied to (a) full resolution, (b) first level terrain pyramid.

rain pyramids and using natural neighbourhood applied to the full resolution terrain pyramid. This was due to noise in the reflectance and pulse width LiDAR data. The use of natural neighbourhood interpolation along with the first level terrain pyramid was also found to be useful for generating smooth reflectance and pulse width raster surfaces. To give an example, the reflectance and pulse width raster surfaces generated using linear interpolation applied to the full resolution terrain pyramid and using natural neighbourhood interpolation applied to the first level terrain pyramid are shown in Figure 4.13.

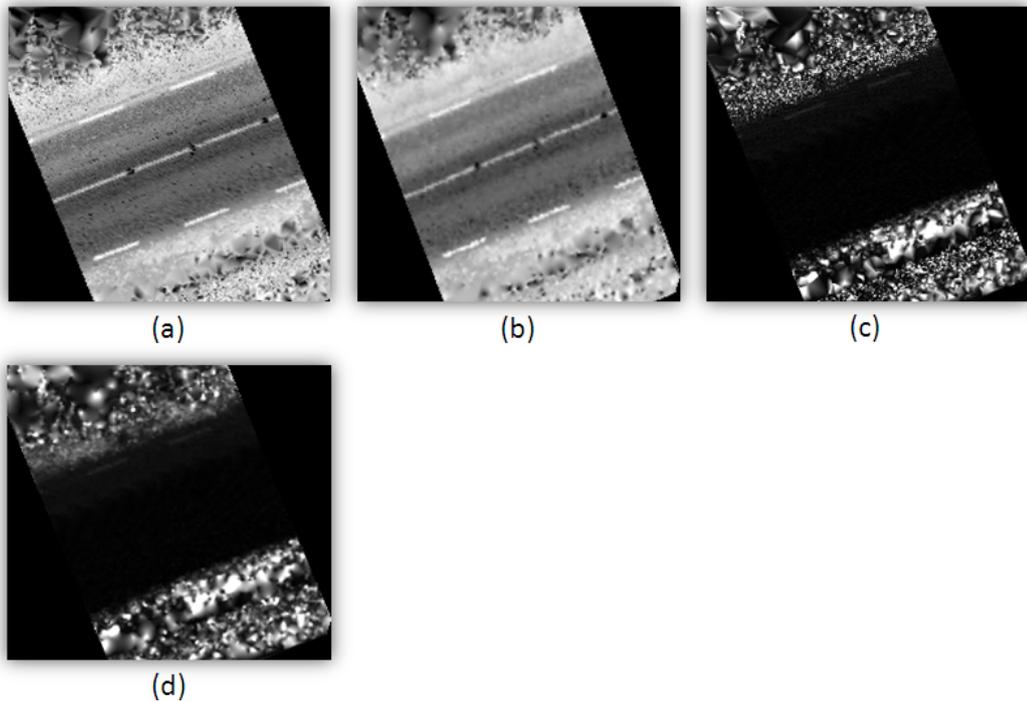


Figure 4.13: Reflectance raster surface generated from (a) full resolution terrain pyramid using linear interpolation and (b) first level terrain pyramid using natural neighbourhood interpolation. Pulse width raster surface generated from (c) full resolution terrain pyramid using linear interpolation and (b) first level terrain pyramid using natural neighbourhood interpolation.

4.3.2 Optimal Cell Size

A suitable cell size is required in order to calculate a 2D raster surface. The selection of the optimal cell size is essential as it may effect the accuracy and computational cost of our road edge extraction algorithm. To find its optimal value, we analysed the performance of our road edge extraction algorithm in raster surfaces generated with different cell sizes. We selected one 10 m section of rural road consisting of grass-soil edges. To process this road section, we used one 30 m \times 10 m \times 5 m section of LiDAR data and one 10 m section of navigation data which was collected with the XP-1 system.

We considered six test cases in which raster surfaces were generated with cell sizes 0.02 m², 0.04 m², 0.06 m², 0.08 m², 0.1 m² and 0.2 m² from the LiDAR elevation, reflectance and pulse width attributes. These cell sizes were selected with decreasing and increasing values based on an average point spacing of 0.08 m in the LiDAR points. We normalised the slope, reflectance and pulse width values with respect to their global minimum and maximum, shown in Table 4.1 and converted to 8-bit data type prior to processing them in our algorithm. These minimum and maximum values

LiDAR attribute	Global minimum	Global maximum
Slope	0	90
Reflectance	-2365.86	159.66
Pulse width	0	486.58

Table 4.1: Global minimum and maximum values of the LiDAR attributes.

were selected globally from all the data sections processed in the road edge extraction experiments.

In each test case, we applied our road edge extraction algorithm to the road section with the following parameters: $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 = 1$, $\omega = 20^\circ$ and $\phi = 21.49^\circ$. These parameters were found empirically after examining several values in the algorithm. The ϕ value was calculated from the $\theta = 68.51^\circ$ which is an average heading angle of the mobile van along the selected road section. The number of iterations required to converge the snake curve to the road edges was 40. Optimal parameters used in the algorithm were different in each case to correctly test the cell size effect, as shown in Table 4.2.

We used different sets of T, the hierarchical threshold, parameter in each case. The value of γ was increased to decrease the snake curve step size

Cell Size (m ²)	T_{slope}	T_{ref}	T_{pw}	γ	δt (radian)	GVF it- eration
0.02	75	100	85	1	0.01	1800
0.04	60	100	70	2	0.02	900
0.06	55	100	65	3	0.03	600
0.08	50	100	65	4	0.04	450
0.1	40	100	65	5	0.05	360
0.2	30	100	65	10	0.1	180

Table 4.2: Different parameters used in the six test cases of an optimal cell size estimation analysis.

during one iteration, δt was increased to decrease the number of points in the snake curve, while the number of GVF iterations were decreased. These parameters were changed to account for the effect of the change the raster surfaces resolution would have on the algorithm. The γ , δt and GVF iteration parameters were changed proportionally to the change in the cell size of the raster surfaces. The final positions of the snake curve in the six test cases are shown in Figure 4.14.

In order to complete our comparative analysis, we estimated the length of both the left and right road edges as extracted in each test case. In each case, a plot of completeness was obtained from the snake curve as determined from the left and right edges, as shown in Figure 4.15. For cell sizes 0.02 m², 0.04 m², 0.06 m² and 0.08 m², the snake curve converged to more than 79% of the left and right road edges. In the 0.1 m² cell size case, the snake curve converged to around 75% of the left road edge and 71.3% of the right road edge while in the large cell size of 0.2 m², the snake curve converged to around 55% of the left road edge and 46.7% of right edge. In our road edge extraction algorithm, the road edges are extracted using iterative partially overlapping road sections. The 0.1 m² and 0.2 m² cell cases produced snake curves which were not able to converge to the road edges to the extent with

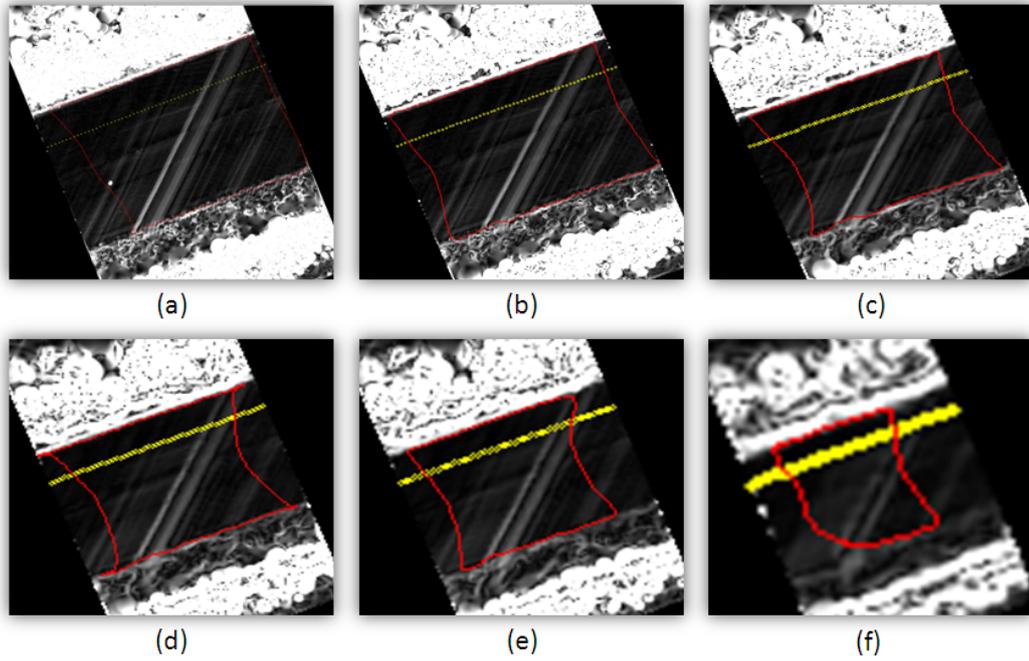


Figure 4.14: Final position of the snake curve over the slope raster surface with (a) 0.02 m^2 , (b) 0.04 m^2 , (c) 0.06 m^2 , (d) 0.08 m^2 , (e) 0.1 m^2 and (f) 0.2 m^2 cell size.

which the overlapping snake curves can be obtained.

For each test case, we also estimated the time taken by the snake curve to move from its initial position to the final position. A plot of these estimates is shown in Figure 4.16. For cell sizes 0.02 m^2 and 0.04 m^2 , the snake curve required more time, with 5280 and 1327 seconds respectively, to converge to the road edges. The increased time taken was due to the use of higher resolution raster surfaces. With cell sizes of 0.06 m^2 and 0.08 m^2 , the snake curve required 384 and 178 seconds respectively, to move to the final position. In the final two cases, the snake curve required less time with 78 and 9 seconds respectively. This reduced time was due to the use of lower resolution raster surfaces.

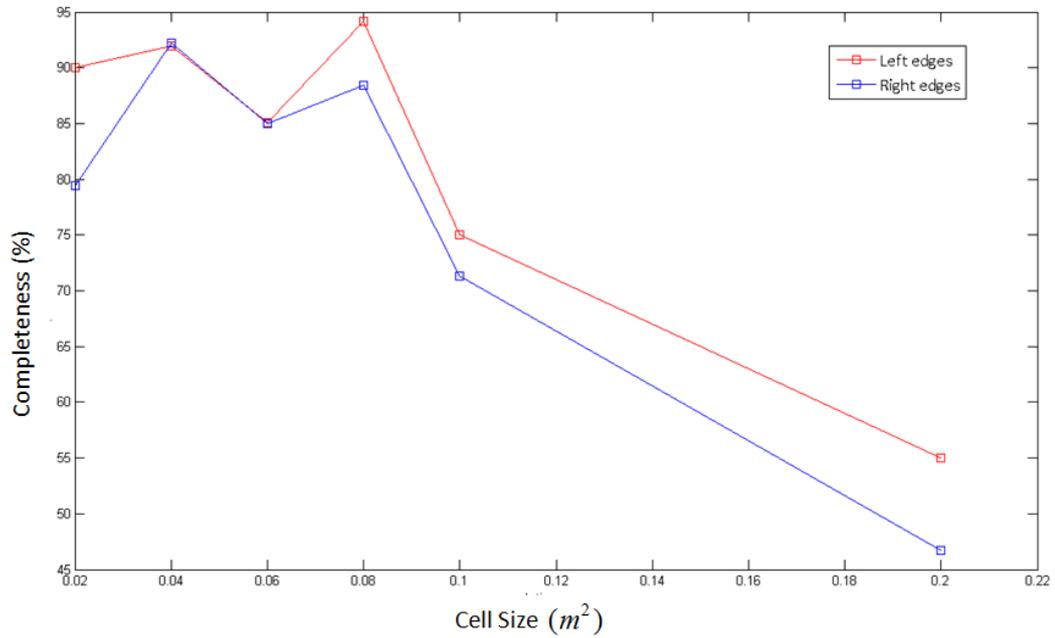


Figure 4.15: Plot of completeness obtained from the snake curve in the six test cases of an optimal cell size estimation analysis.

The road edges extracted in the six test cases were validated using our road edge validation algorithm. We manually digitised the left and right road edges in the road section from the 3D LiDAR data. Our validation algorithm is designed to measure the error between the extracted road edges and the manually digitised road edges. In the digitisation process, a user clicks the LiDAR point in a 3D environment where they perceive the road edge to be located. This can introduce an error source in the manual process of selecting this LiDAR road edge point. The error can arise primarily due to the trade off between selecting the road edge at lower zoom levels or higher zoom levels. At a lower zoom level, the road edge is much more identifiable because of the additional information of the road and its structure as shown in Figure 4.17(a). However, clicking on the road edge leads to uncertainty as

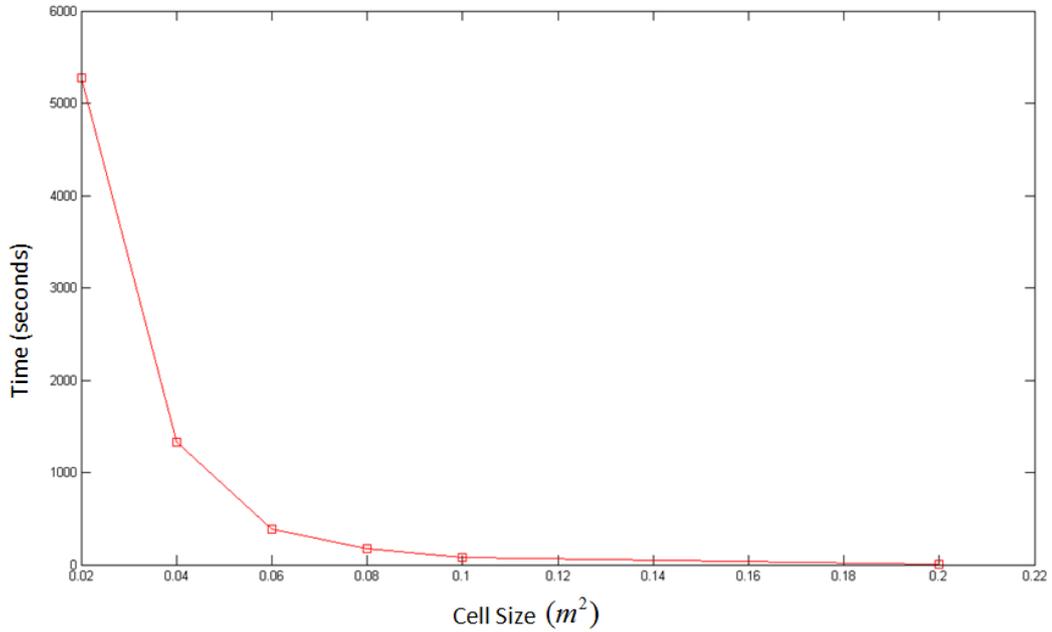


Figure 4.16: Plot of the time taken by the snake curve to move from its initial position to the final position in the six test cases of an optimal cell size estimation analysis.

to which LiDAR point was actually clicked. The converse is true at higher zoom levels where there is absolute certainty as to which LiDAR point was clicked but whether or not the point is actually a road edge candidate. This can be ambiguous without the wider perspective view showing the road, as seen in Figures 4.17(b) and (c). This is demonstrated for both the left and right road edges as shown in Figure 4.18. The left road edge has a much higher LiDAR point density relative to the right edge as the XP-1 system employed a single laser scanner during data acquisition process. A section of the road edge digitised from the LiDAR data in a 3D environment is shown in Figure 4.19. Our road edge extraction accuracy may be affected with the errors inherited through the digitisation process.

We applied our validation algorithm to estimate both the 2D and 3D

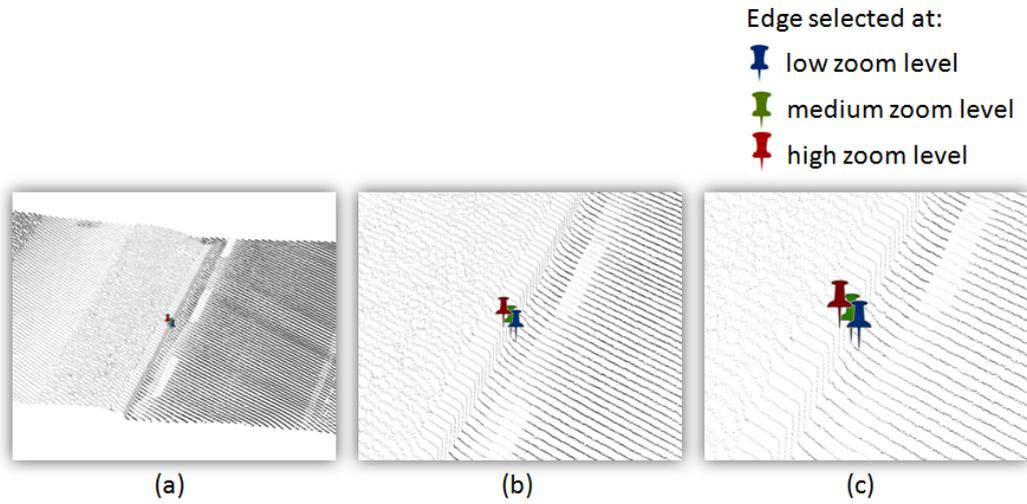


Figure 4.17: 3D LiDAR data: (a) 6 m \times 10 m, (b) 4.5 m \times 5 m and (c) 3 m \times 4 m.

accuracy of our extracted left and right road edges. In the 2D accuracy estimation, we calculated the euclidean distances between the 2D positions of the extracted and digitised road edge points, while in the 3D accuracy estimation, we considered their 3D positions.

Box plots for the 2D and 3D accuracy of the extracted left edges are shown in Figure 4.20. In each box, the central line mark in the red represents the median value while its lower and upper edges represent the 25th and 75th percentiles. Its lower and upper adjacent values represent the minimum and maximum values without outliers. Outlier points are drawn if their value is smaller than $p_1 - w(p_3 - p_1)$ or larger than $p_3 + w(p_3 - p_1)$, where p_1 and p_3 are the 25th and 75th percentiles, while w is the user specified whisker length. We provided the standard value of 1.5 to w after examining its other values in the box plot. In Figure 4.21, box plots for the 2D and 3D accuracy of the extracted right edges are shown.

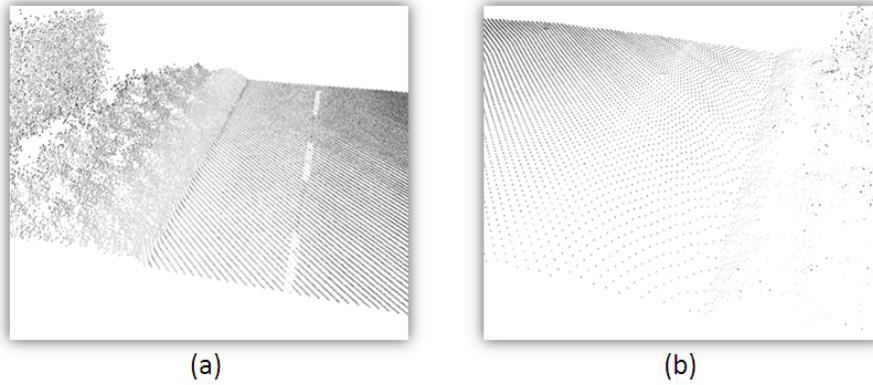


Figure 4.18: LiDAR point data along the (a) left and (b) right side of the road section.

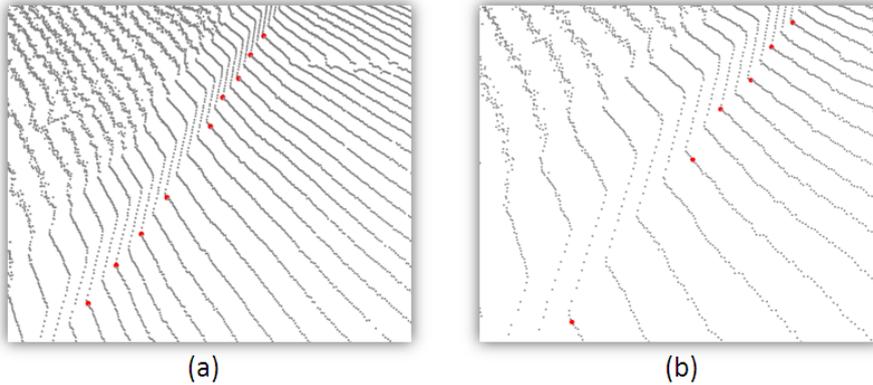
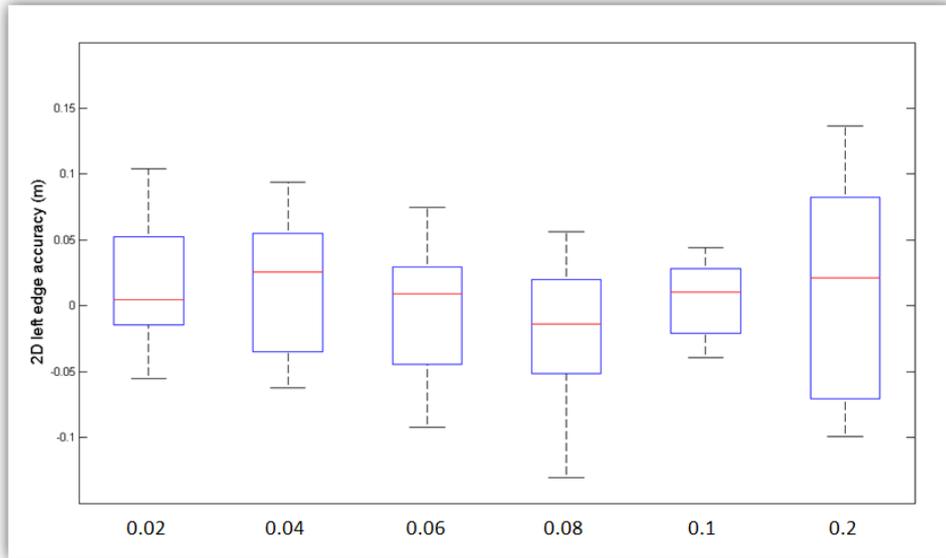


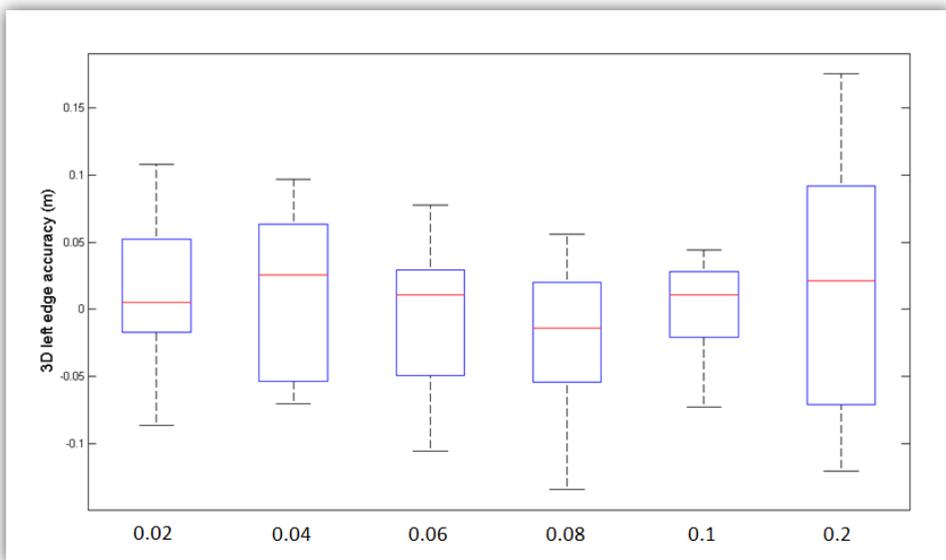
Figure 4.19: Road edge digitised from the LiDAR data: (a) 3.2 m \times 3.3 m and (b) 1.8 m \times 2 m.

We also carried out statistical analysis of the 2D and 3D accuracy of the extracted left edges, shown in Tables 4.3 and 4.4 respectively. Similarly, statistical analysis of the 2D and 3D accuracy of the extracted right edges are shown in Tables 4.5 and 4.6 respectively. In these tables, the completeness and time taken by the snake curve for the six test cases are also included to make a comparative analysis.

The negative and positive accuracy values of the extracted left and right edge points respectively indicate that they were outside the digitised edges of

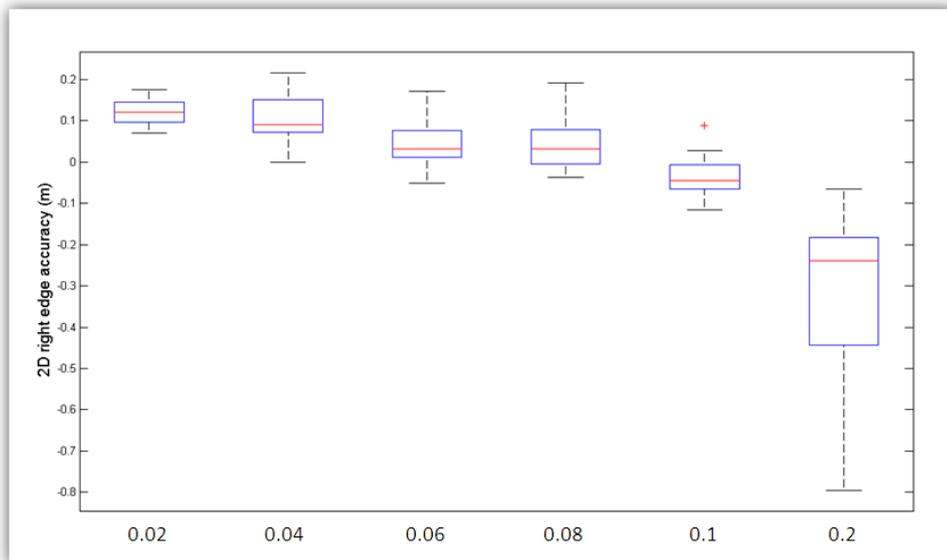


(a)

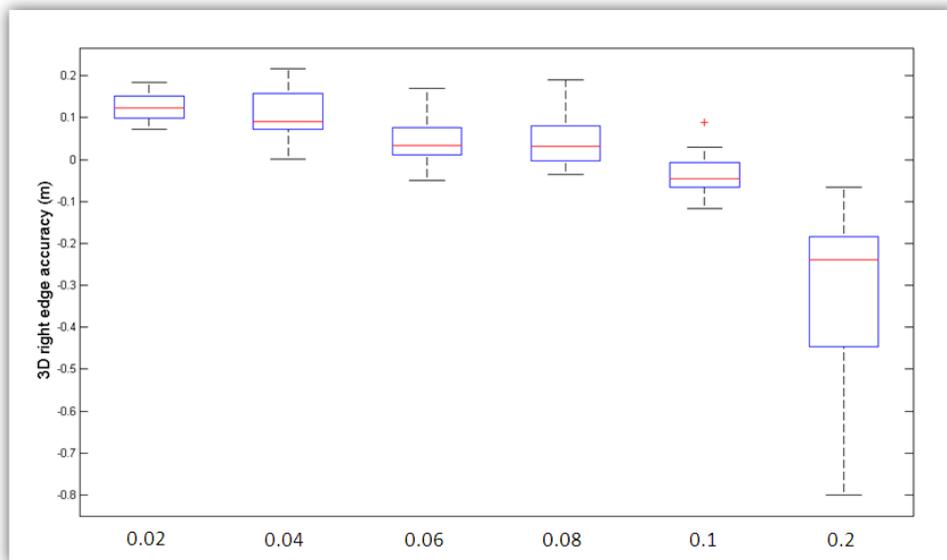


(b)

Figure 4.20: Box plots for the (a) 2D and (b) 3D accuracy of the extracted left edges in the six test cases of an optimal cell size estimation analysis.



(a)



(b)

Figure 4.21: Box plot for the (a) 2D and (b) 3D accuracy of the extracted right edges in the six test cases of an optimal cell size estimation analysis.

	0.02	0.04	0.06	0.08	0.1	0.2
minimum (m)	-0.055	-0.063	-0.092	-0.131	-0.039	-0.099
maximum (m)	0.104	0.094	0.075	0.056	0.044	0.137
lower adjacent (m)	-0.055	-0.063	-0.092	-0.131	-0.039	-0.099
upper adjacent (m)	0.104	0.094	0.075	0.056	0.044	0.137
25th percentile (m)	-0.015	-0.035	-0.045	-0.051	-0.021	-0.070
75th percentile (m)	0.052	0.055	0.029	0.020	0.028	0.083
mean (m)	0.014	0.014	-0.007	-0.016	0.005	0.018
median (m)	0.005	0.026	0.009	-0.014	0.01	0.021
outliers (%)	0	0	0	0	0	0
inside ± 0.01 (%)	40	5.88	6.25	5.88	21.43	0
inside ± 0.025 (%)	53.33	17.65	25	41.18	50	22.23
inside ± 0.05 (%)	66.67	58.82	68.75	70.59	100	22.23
outside ± 0.2 (%)	0	0	0	0	0	0
completeness (%)	90	92	85.1	94.2	75	55
time (sec)	5280	1327	384	178	78	9

Table 4.3: Statistical analysis of the 2D accuracy of the left edges along with completeness and time in the six test cases.

	0.02	0.04	0.06	0.08	0.1	0.2
minimum (m)	-0.087	-0.071	-0.106	-0.134	-0.073	-0.121
maximum (m)	0.108	0.097	0.078	0.056	0.044	0.175
lower adjacent (m)	-0.087	-0.071	-0.106	-0.134	-0.073	-0.121
upper adjacent (m)	0.108	0.097	0.078	0.056	0.044	0.175
25th percentile (m)	-0.017	-0.054	-0.049	-0.054	-0.021	-0.071
75th percentile (m)	0.053	0.063	0.029	0.020	0.028	0.092
mean (m)	0.008	0.011	-0.011	-0.019	0.001	0.024
median (m)	0.005	0.026	0.011	-0.014	0.01	0.021
outliers (%)	0	0	0	0	0	0
inside ± 0.01 (%)	40	5.88	6.25	5.88	21.43	0
inside ± 0.025 (%)	46.67	11.76	25	41.18	50	22.23
inside ± 0.05 (%)	60	41.18	68.75	70.59	92.86	22.23
outside ± 0.2 (%)	0	0	0	0	0	0
completeness (%)	90	92	85.1	94.2	75	55
time (sec)	5280	1327	384	178	78	9

Table 4.4: Statistical analysis of the 3D accuracy of the left edges along with completeness and time in the six test cases.

	0.02	0.04	0.06	0.08	0.1	0.2
minimum (m)	0.071	0.0007	-0.049	-0.036	-0.116	-0.794
maximum (m)	0.176	0.215	0.171	0.191	0.089	-0.065
lower adjacent (m)	0.071	0.0007	-0.049	-0.036	-0.116	-0.794
upper adjacent (m)	0.176	0.215	0.171	0.191	0.027	-0.065
25th percentile (m)	0.097	0.073	0.012	-0.003	-0.065	-0.443
75th percentile (m)	0.145	0.152	0.077	0.079	-0.007	-0.183
mean (m)	0.120	0.107	0.043	0.046	-0.036	-0.322
median (m)	0.122	0.090	0.032	0.031	-0.044	-0.238
outliers (%)	0	0	0	0	7.14	0
inside ± 0.01 (%)	0	5.88	0	11.76	14.29	0
inside ± 0.025 (%)	0	5.88	25	41.18	21.43	0
inside ± 0.05 (%)	0	11.76	62.5	58.82	57.14	0
outside ± 0.2 (%)	0	5.88	0	0	0	66.67
completeness (%)	79.4	92.2	85	88.4	71.3	46.7
time (sec)	5280	1327	384	178	78	9

Table 4.5: Statistical analysis of the 2D accuracy of the right edges along with completeness and time in the six test cases.

	0.02	0.04	0.06	0.08	0.1	0.2
minimum (m)	0.072	0.0007	-0.049	-0.036	-0.117	-0.799
maximum (m)	0.184	0.216	0.171	0.191	0.089	-0.065
lower adjacent (m)	0.072	0.0007	-0.049	-0.036	-0.117	-0.799
upper adjacent (m)	0.184	0.216	0.171	0.191	0.030	-0.065
25th percentile (m)	0.099	0.073	0.012	-0.003	-0.065	-0.447
75th percentile (m)	0.152	0.158	0.077	0.080	-0.007	-0.183
mean (m)	0.124	0.108	0.045	0.048	-0.036	-0.324
median (m)	0.124	0.091	0.034	0.031	-0.045	-0.238
outliers (%)	0	0	0	0	7.14	0
inside ± 0.01 (%)	0	5.88	0	11.76	14.29	0
inside ± 0.025 (%)	0	5.88	25	35.29	21.43	0
inside ± 0.05 (%)	0	11.76	62.5	52.94	57.14	0
outside ± 0.2 (%)	0	5.88	0	0	0	66.67
completeness (%)	79.4	92.2	85	88.4	71.3	46.7
time (sec)	5280	1327	384	178	78	9

Table 4.6: Statistical analysis of the 3D accuracy of the right edges along with completeness and time in the six test cases.

the road surface. Similarly, the positive and negative accuracy values indicate that they were inside. The 2D accuracy values were some what better than their respective 3D values and this was due to the inclusion of elevation points in the estimation of 3D accuracy values. The left edge displayed higher 2D and 3D accuracy than the right edge. This was due to the lower LiDAR point density along the right side of the road section when compared to the left side. Using just the minimum and maximum values as a metric, the highest accuracy for the left edge was obtained by applying the 0.1 m² cell size and for the right edge using the 0.02 m² cell size. However, this is not the best suited method to determine which cell size to apply. For the 2D and 3D left edge, the cell sizes whose mean and median values were closest to 0 were 0.1 m² and 0.02 m² cell size respectively. A mean and median value close to 0 indicates high accuracy. For the right edge, cell sizes of 0.1 m² and 0.08 m² produced mean and median values closest to 0. When examining the accuracy, we considered four cases for values inside ± 0.01 m, ± 0.025 m, ± 0.05 m and outside ± 0.2 m. These would demonstrate which cell sizes produced the highest accuracy edges for various acceptable distances. For the left edge, the highest accuracy values inside ± 0.01 m, ± 0.025 m and ± 0.05 m were in the 0.02 m², 0.02 m² and 0.1 m² cell size respectively, while for the right edge, they were in the 0.1 m², 0.08 m² and 0.06 m² cell size respectively.

We examined the results obtained in the six test cases. For the 0.02 m² and 0.04 m² cell sizes, the snake curves were computationally expensive without providing any considerable improvement in the accuracy of the extracted road edges. In the 0.1 m² and 0.2 m² cell size cases, the snake curves did not converge to the road edges to the extent that will allow for overlapping snake

curves to be obtained. Overlapping snake curves are required to process multiple road sections and to facilitate general application of the algorithm. For the 0.06 m² cell size, the minimum-maximum range, mean and median accuracy values were better than in the 0.08 m² cell size. The snake curve in the 0.06 m² case required a reasonable amount of time to move from its initial position to the final position and was able to converge to the road edges to the extent with which the overlapped snake curves can be obtained. Thus, we chose an optimal cell size value of 0.06 m² to generate the raster surfaces from the LiDAR attributes.

4.3.3 Optimal Road Length

We performed this analysis to find the optimal length of the road section to which our road edge extraction algorithm can be applied. To find the optimal length with respect to computational efficiency, we considered three test cases in which a temporal performance of our road edge extraction algorithm was analysed in a road section with 10 m, 20 m and 30 m length. We selected one national primary road section consisting of grass-soil edges with shoulders. To process this road section, we used three 30 m × 10 m × 5 m, 30 m × 20 m × 5 m, 30 m × 30 m × 5 m sections of LiDAR data and three 10 m, 20 m, 30 m section of navigation data which was collected with the XP-1 system.

In each test case, we applied our road edge extraction algorithm to the road section using a cell size $c=0.06$ m², $M_{slope} = 185$, $M_{ref} = 25$, $M_{pw} = 25$, $T_{slope} = 45$, $T_{ref} = 132$, $T_{pw} = 55$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\gamma = 3$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 = 1$ and $\phi = -37.27^\circ$. The value of ϕ was calculated from the average heading, $\theta = 127.27^\circ$. The number of GVF iterations used were 600 while the number of iterations required to move the

snake curve were 40. The variation in road length required slight modification of two parameters in the algorithm, as shown in Table 4.7.

Road Length (m)	ω (degree)	δt (radian)
10	34°	0.03
20	17°	0.015
30	11°	0.01

Table 4.7: Different parameters used in the three test cases of an optimal road length estimation analysis.

We applied the higher M_{slope} parameter to the slope raster surface due to inherent noise. This noise was due to missing points in the section of LiDAR data, shown in Figure 4.22, which are due to the mobile van overtaking a stationary vehicle during the data acquisition process along the road section. In the reflectance and pulse width raster surfaces, the use of a lower, M_{ref}

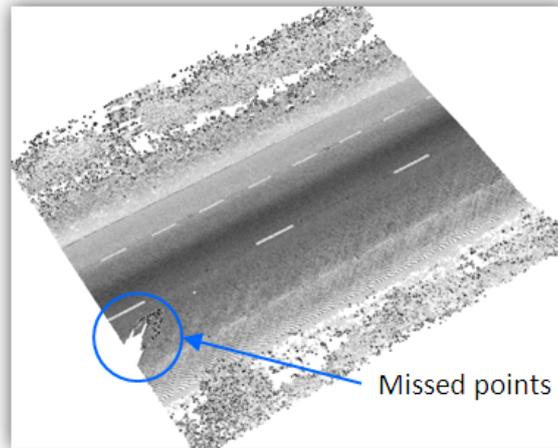


Figure 4.22: Missed points as circled in blue in the section of LiDAR data.

and M_{pw} , parameters respectively was sufficient to remove the inherent noise. The values of ω and δt were decreased proportionally to reflect the change in the length of the road section. The final positions of the snake curve in the three test cases are shown in Figure 4.23.

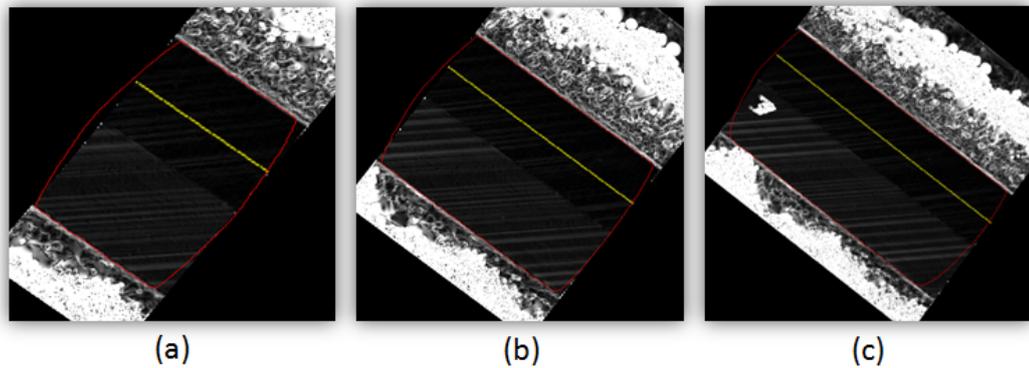


Figure 4.23: Final position of the snake curve over the slope raster surface with (a) 10 m, (b) 20 m and (c) 30 m road length.

We examined the results obtained in the three test cases of this analysis. In all cases, the snake curve was able to converge to the road edges. In the first case, the snake curve required around 623 seconds to move from its initial position to the final position. In the second and third case, the snake curve was computationally expensive as it required 1570 and 3191 seconds respectively to converge to the road edges. This analysis was performed on a computer with 2 Intel Xeon E5607 processors @2.27 GHz, 12 GB RAM and a 64-bit operating system. For example, to process a 60 m section of road with $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ would require approximately 3738 seconds, with $30\text{ m} \times 20\text{ m} \times 5\text{ m}$ would require 4710 seconds and with $30\text{ m} \times 30\text{ m} \times 5\text{ m}$ would require 6382 seconds, without overlap. Thus, on the basis of computational cost, we chose the 10 m length of the road section to which our road edge extraction algorithm can be applied.

4.3.4 Optimal Internal Energy Parameters

This analysis was performed to demonstrate the importance of the internal energy weight parameters in our road edge extraction algorithm. We considered five test cases in which the performance of our algorithm was analysed with the use of optimally higher and lower values of the internal energy weight parameters. We selected one 10 m section of national primary road consisting of grass-soil edges with shoulders. To process this road section, we used one 30 m \times 10 m \times 5 m section of LiDAR data and one 10 m section of navigation data which was collected with the XP-1 system.

In each test case, we applied our road edge extraction algorithm to the road section with the following parameters: $c=0.06 \text{ m}^2$, $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_{slope} = 45$, $T_{ref} = 132$, $T_{pw} = 55$, $T_1 = 250$, $T_2 = 5$, $\gamma = 3$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 = 1$, $\phi = -37.27^\circ$, $\omega = 34^\circ$ and $\delta t = 0.03$. The value of ϕ was calculated from the average heading, $\theta = 127.27^\circ$. The number of GVF iterations were 600, while the number of iterations required to move the snake curve were 40. The internal energy parameters used in the algorithm in each case are shown in Table 4.8. The final positions of the snake curve

Test Case	α	β
1	9	0.001
2	40	0.001
3	0	0.001
4	9	40
5	9	0

Table 4.8: Internal energy weight parameters used in the five test cases.

in the five test cases are shown in Figure 4.24.

We examined the results obtained in the five test cases of this analysis. In the first case, we provided the optimal values of α and β parameters in

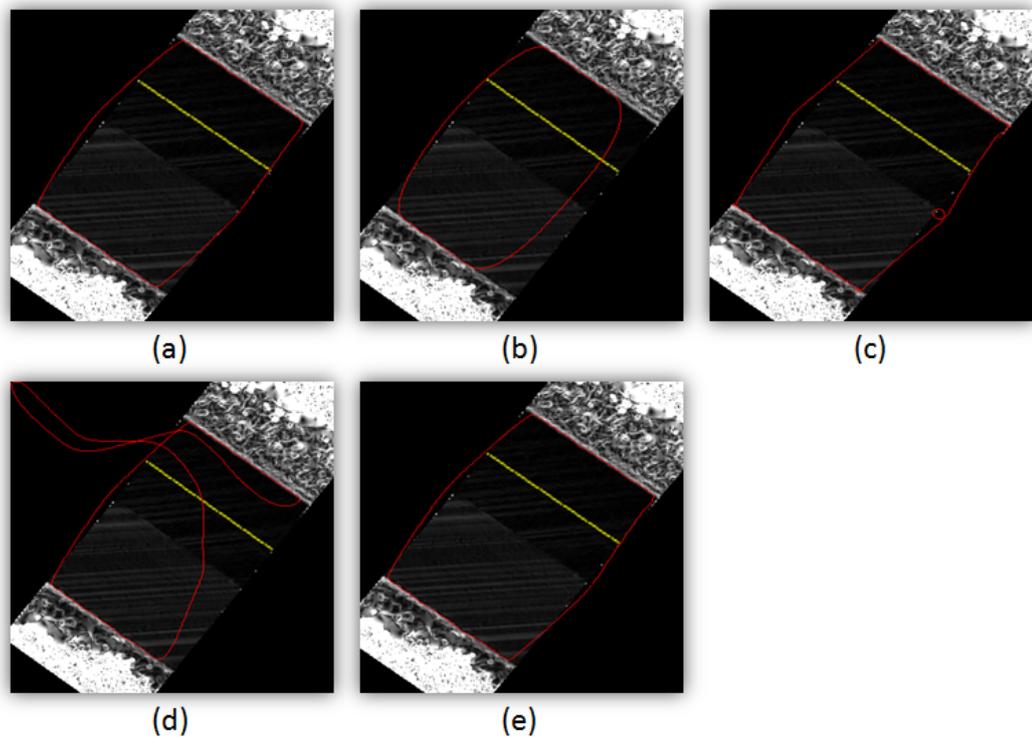


Figure 4.24: Final position of the snake curve over the slope raster surface in the (a) first, (b) second, (c) third, (d) fourth and (e) fifth case of the internal energy parameter analysis.

the algorithm that enabled the snake curve to converge to the road edges. Their respective optimal values were found empirically after examining several combinations in the algorithm. In the second case, with a higher value of α parameter, the snake curve was unable to fully converge to the road edges, while in the third case, with the lower value of α parameter, the snake curve was found to be jagged at some of the points. The α parameter is used to hold the snake curve together and controls its elasticity. Its higher value increased the binding energy in the snake curve while its lower value caused the snake curve to move without any binding energy. The optimal value of the α parameter is essential to obtain better results in the road edge

extraction process.

In the fourth case, with a higher value of β parameter, the snake curve was affected with extreme bending at some of its points, while in the fifth case, with the lower value of β parameter, there was no change in the final position of the snake curve from its position in the first case. The β parameter is used to keep the snake curve from bending and controls its stiffness. The lower value allowed the snake curve to move without any bending but its higher value caused extreme bending in the snake curve. The extraction of the road edges does not require much bending in the snake curve so a lower value of the β parameter was found to be optimal.

4.3.5 Optimal External Energy Parameters

We performed this analysis to demonstrate the importance of the external energy weight parameters in our road edge extraction algorithm. We considered nine test cases in which the performance of our algorithm was analysed with the use of various combinations of GVF and balloon energy weight parameters. We selected one 10 m section of national primary road consisting of grass-soil edges with shoulders. To process this road section, we used one 30 m \times 10 m \times 5 m section of LiDAR data and one 10 m section of navigation data which was collected with the XP-1 system.

In each test case, we applied our road edge extraction algorithm to the road section with the following parameters: $c=0.06 \text{ m}^2$, $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_{slope} = 45$, $T_{ref} = 132$, $T_{pw} = 55$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\gamma = 3$, $\phi = -37.27^\circ$, $\omega = 34^\circ$ and $\delta t = 0.03$. The external energy parameters used in the algorithm in each case are shown in Table 4.9. The value of ϕ was calculated from the average heading, $\theta = 127.27^\circ$. The number

Test Case	κ_1	κ_2	κ_3	κ_4
1	4	2	2	1
2	2	4	2	1
3	2	2	4	1
4	4	0	0	0
5	4	2	0	0
6	0	2	2	0
7	4	2	2	0
8	4	2	2	3
9	4	2	2	5

Table 4.9: External energy weight parameters used in the nine test cases.

of GVF iterations were 600, while the number of iterations required to move the snake curve were 40. The final positions of the snake curve in the nine test cases are shown in Figure 4.25.

We examined the results obtained in the nine test cases of this analysis. In the first case, we set our optimal weights with the highest weight to the slope GVF energy. Their respective optimal values were found empirically after examining several combinations. In the second and the third case, we set the highest weights to the reflectance GVF energy and the pulse width GVF energy respectively. In all three cases, the snake curve converged to the road edges in a similar way. The difference in the slope values over the road surface and its nearby grass-soil surface was found to be useful in providing the required external energy to the snake curve to converge to the road edges. The reflectance values provided by the Riegl VQ-250 laser scanner are not accurately normalised, which leads to different values over the road surface. In most cases, the hierarchical thresholding applied to the reflectance raster surface failed to remove the road marking cells near the road edge points, as shown in Figure 4.26. These road marking cells create an obstruction for the snake curve to move towards the road edges. The orientation of

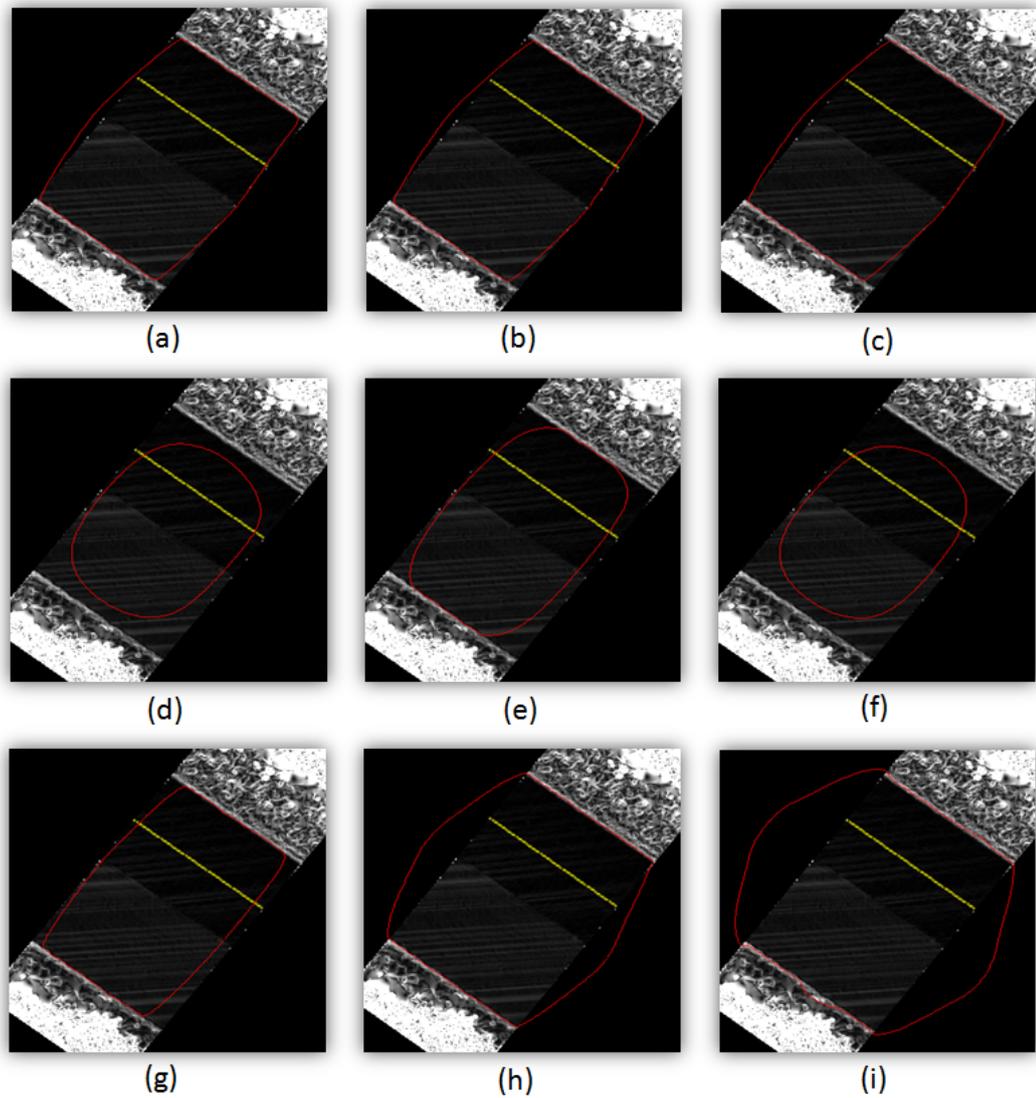


Figure 4.25: Final position of the snake curve over the slope raster surface in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth, (f) sixth, (g) seventh, (h) eighth and (i) ninth case of the external energy parameter analysis.

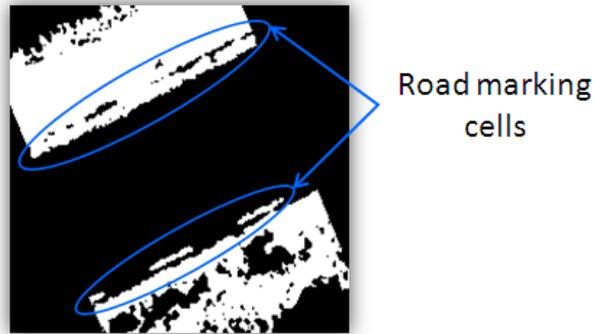


Figure 4.26: Hierarchical thresholding applied to the reflectance raster surface provided the objects with road marking cells near the road edge points as circled in blue.

top of the kerb edges relative to the vehicle is very similar to that of the road surface. This together with their similar surface composition results in the pulse width having similar values in these cases. This leads to pulse width being less important in urban regions. Thus, we did not choose to set the highest weight to the reflectance GVF energy and the pulse width GVF energy in our selected optimal values.

In the next four cases, fourth to seventh, we tested the following permutations without the balloon energy in our algorithm: the slope GVF energy, the slope and reflectance GVF energy, the reflectance and pulse width GVF energy and the slope, reflectance and pulse width GVF energy respectively. In all these cases, the snake curve was not able to fully converge to the road edges. This demonstrates the importance of using all the external energy weight parameters including the balloon energy in our algorithm.

In the eighth case, we increased the weight of the balloon energy when used with the optimal weights of the other GVF energy terms. The snake curve fully converged to the road edges but also expanded along the non road edge sides. This increased weight to the balloon energy can be useful

in cases of noisy data as it can provide an additional inflation energy to the snake curve to overcome the noise. In the final case, we increased the balloon energy weight parameter from its value in the eighth case. However, the snake curve grew beyond the road edges.

The XP-1 system employed a single laser scanner during the data acquisition process, which leads to the acquisition of LiDAR data with a lower point density along the right side of the road section than along its left side, as shown for different road sections in Table 4.10. This lower point density

Road section	Left side (number of points/ m^2)	Right side (number of points/ m^2)	Van speed (km/hr)	Geographic location
Rural	702.83	127.32	61.12	53°34'28.375" N 7°10'12.833" W
Rural	697.74	137.51	60.97	53°34'28.681" N 7°10'11.658" W
Rural	702.83	138.78	61.61	53°34'28.88" N 7°10'10.877" W
Urban	861.98	143.88	51.48	53°36'33.11" N 7°5'47.186" W
Urban	884.90	140.06	51.57	53°36'33.583" N 7°5'46.183" W
Urban	895.09	142.60	51.76	53°36'33.902" N 7°5'45.514" W
National primary	1056.79	44.56	56.66	53°38'15.168" N 7°29'26.328" W
National primary	1033.87	45.84	56.79	53°38'14.707" N 7°29'25.314" W
National primary	1004.59	47.11	56.89	53°38'14.407" N 7°29'24.622" W

Table 4.10: LiDAR point density over the left and right sides of different road sections, acquired with the XP-1 system.

along the right side of the road section resulted in noisy cells in between the road edge points in the slope raster surfaces generated from the LiDAR ele-

vation attribute. In most cases, the hierarchical thresholding applied to the slope raster surface was not able to remove the noisy cells, as shown in one of the examples in Figure 4.27. These noisy cells created an obstruction for the

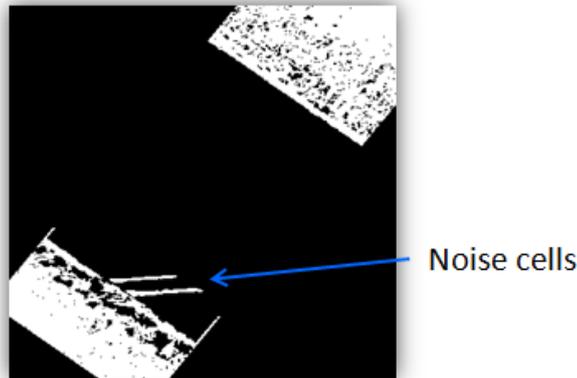


Figure 4.27: Hierarchical thresholding applied to the slope raster surface was not able to remove the noisy cells in between the road edge points.

snake curve to move towards the right road edges. Figure 4.28(a) shows the final position of the snake curve over the slope raster surface accompanied with noisy cells in between the road edge points where a value of $\kappa_4 = 1$ was provided. These noisy cells can be overcome by increasing the balloon energy weight parameter. Figure 4.28(b) shows the final position of the snake curve where we provided an increased balloon energy weight parameter of $\kappa_4 = 3$.

4.4 Experimentation

We selected three sections of road to test our road edge extraction algorithm. These sections covered 150 m of rural, urban and national primary roads which consisted three types of road edges. We selected these road sections to demonstrate the ability of our algorithm to extract these varying distinct road edges without manual intervention. The processed data was acquired

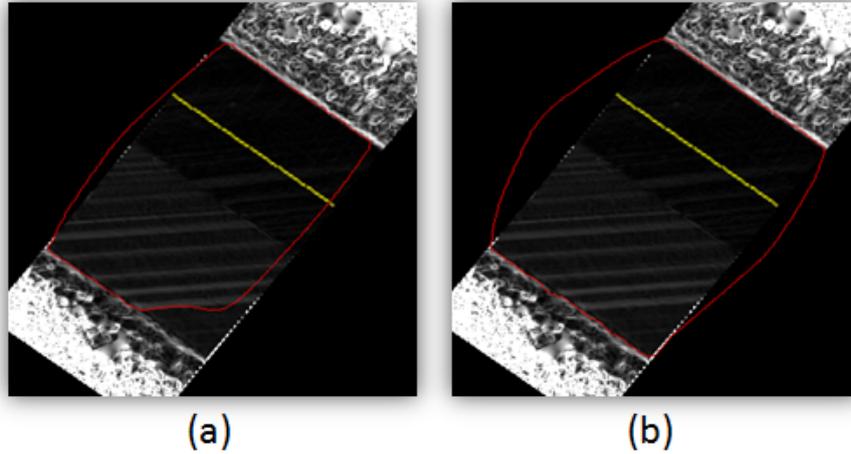


Figure 4.28: Final position of the snake curve over the slope raster surface with (a) $\kappa_4 = 1$ and (b) $\kappa_4 = 3$ parameters.

using the XP-1 system along these road sections. Examples of the point density for these three road sections along the left and right edge is shown in Table 4.10. We applied our algorithm to these road sections in both manual and automated modes. In the following sections, we present the road edge results and discuss their qualitative validation.

4.4.1 Manual Processing

We applied our road edge extraction algorithm to the road sections where we selected the hierarchical threshold parameters empirically for each road section. This step is required to produce the qualitative best results from our algorithm for later comparison with our fully automated algorithm. Other parameters, barring balloon energy, used in the algorithm were constant for all the road sections. In the following sections, we present our algorithm applied to three types of the road section through the manual selection of parameters.

4.4.1.1 Grass-Soil Edges

We selected one 50 m section of rural road consisting of grass-soil edges on both sides of the road, as shown in Figure 4.29. To process the 50 m road



Figure 4.29: Digital image of the rural road section consisting of grass-soil edges (Geographic location: $53^{\circ}34'28.07''\text{N}$ $7^{\circ}10'13.76''\text{W}$).

section, we used six $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ sections of LiDAR data and six 10 m section of navigation data. These data sections were selected with an overlap of 2 m between them.

We applied our road edge extraction algorithm to this road section with the following parameters: $c=0.06\text{ m}^2$, $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\gamma = 3$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 = 1$, $\omega = 20^{\circ}$ and $\delta t = 0.03$. The selected hierarchical threshold parameters were $T_{slope} = 50$, $T_{ref} = 100$ and $T_{pw} = 65$. The ϕ angle was calculated from θ , as shown in Table 4.11. The number of GVF iterations were 600 while the number of iterations used to converge the snake curve to the road edges were 40. The final positions of the snake curve over the slope surface in the six data sections are shown in Figure 4.30. The extracted 3D left and right edges in

Navigation Section	θ (degree)	ϕ (degree)
1	65.59°	24.41°
2	65.77°	24.23°
3	65.97°	24.03°
4	66.24°	23.76°
5	66.46°	23.54°
6	66.85°	23.15°

Table 4.11: ϕ angle calculated from θ , average heading angle in each navigation section in the rural road.

the rural road section are shown in Figure 4.31.

Our algorithm was able to extract the left and right road edges in the rural road section through the manual selection of parameters. In Figure 4.30(a), the snake curve extended incorrectly into the grass and soil area along the left side. This was due to the stronger edge between the grass and soil area compared to the left edge points between the road surface and the soil. Otherwise, the extracted road edges were of high quality.

4.4.1.2 Kerb Edges

We selected one 50 m section of urban road consisting of kerb edges, as shown in Figure 4.32. To process the road section, we used six 30 m \times 10 m \times 5 m sections of LiDAR data and six 10 m sections of navigation data. These data sections were again selected with an overlap of 2 m.

The following parameters were input: $c=0.06 \text{ m}^2$, $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\gamma = 3$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 = 1$, $\omega = 20^\circ$ and $\delta t = 0.03$. The hierarchical threshold parameters we selected were $T_{slope} = 55$, $T_{ref} = 85$ and $T_{pw} = 35$. The ϕ angle was calculated from θ , as shown in Table 4.12. The number of GVF iterations were 600 while the snake curve converged to the road edges in 40 iterations.

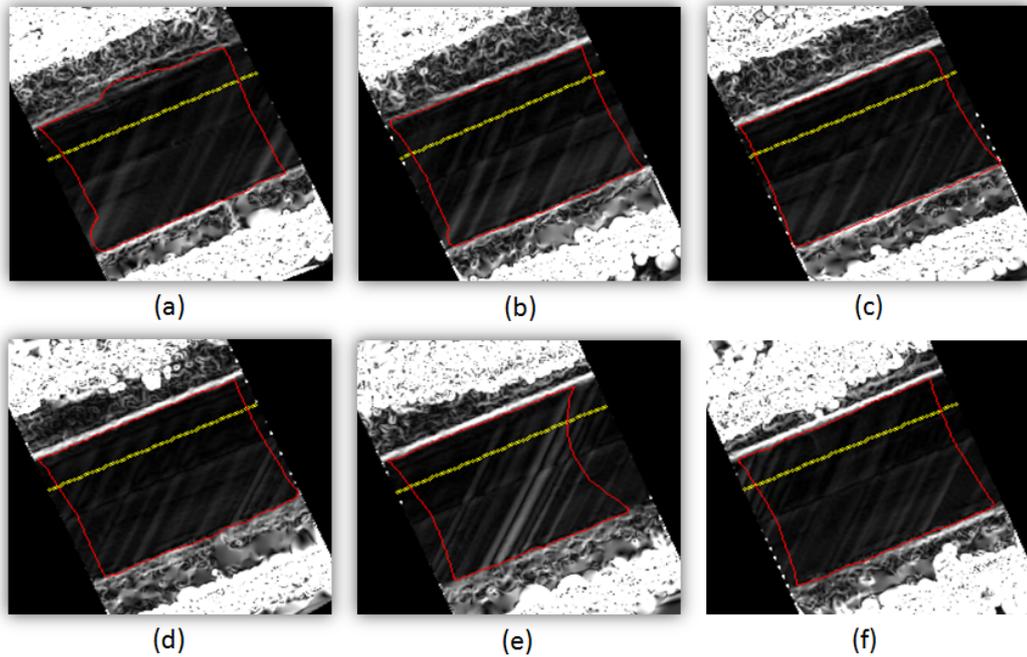


Figure 4.30: Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the rural road.

The final positions of the snake curve over the slope raster surface in the six data sections are shown in Figure 4.33. The extracted 3D left and right edges in the urban road section are shown in Figure 4.34.

The algorithm successfully extracted the left and right road edges in the urban road section through the manual selection of parameters. In Figure 4.33(e), the snake curve was also able to identify the drainage system along the left edge. The snake curve did not properly converge to some of the points along the right edge. This was due to the lower LiDAR point density along the right side of the road section, as highlighted in Table 4.10, which resulted in the noisy cells in the generated 2D raster surfaces.

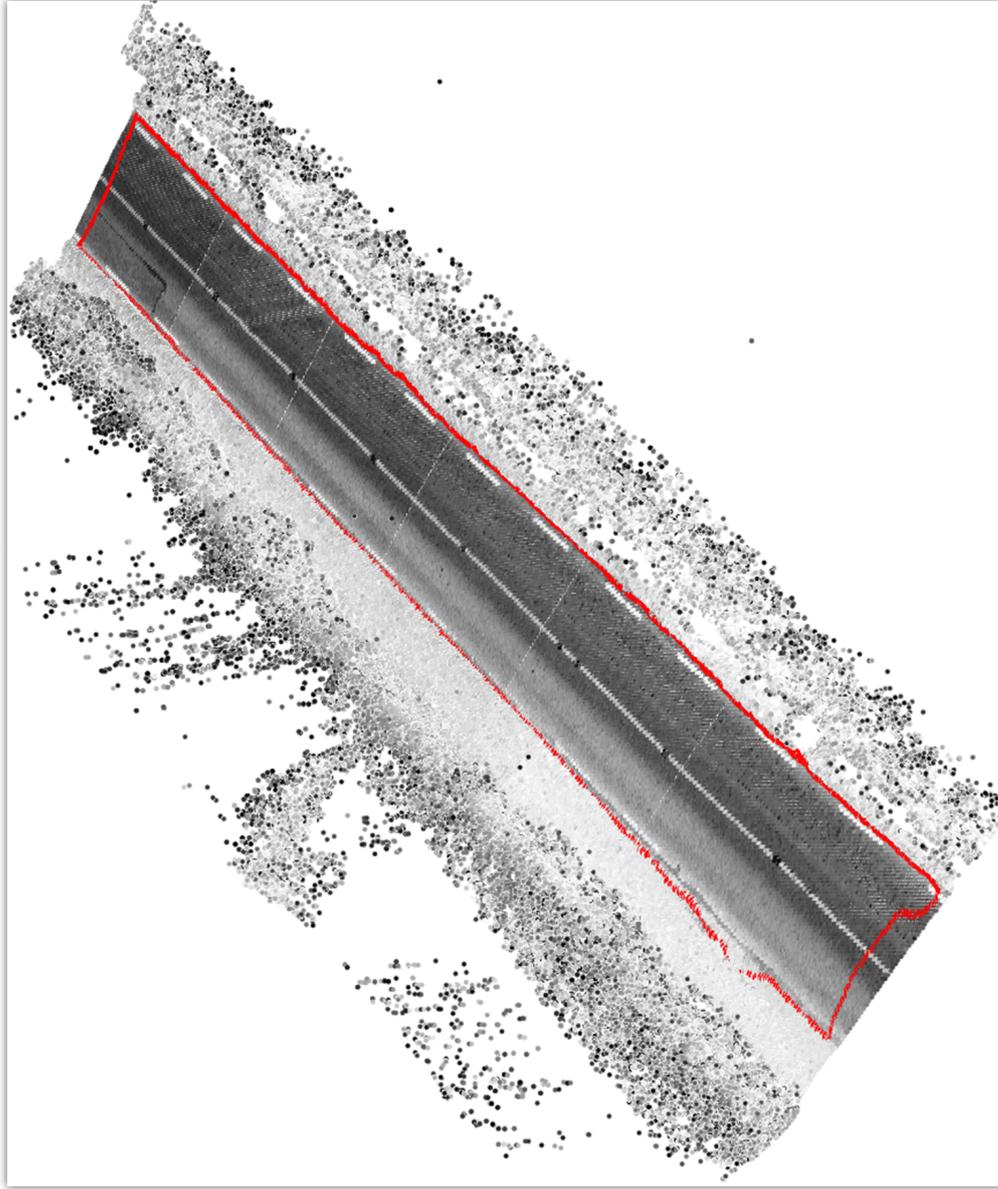


Figure 4.31: Extracted 3D left and right edges in the rural road section obtained through the manual selection of parameters.



Figure 4.32: Digital image of the urban road section consisting of kerb edges (Geographic location: $53^{\circ}36'37.42''\text{N}$ $7^{\circ}5'41.00''\text{W}$).

Navigation Section	θ (degree)	ϕ (degree)
1	51.90°	38.10°
2	51.59°	38.41°
3	51.62°	38.38°
4	51.64°	38.36°
5	51.34°	38.66°
6	50.08°	39.92°

Table 4.12: ϕ angle calculated from θ , average heading angle in each navigation section in the urban road.

4.4.1.3 Grass-Soil Edges with Shoulders

The final experiment consisted of one 50 m section of national primary road consisting of grass-soil edges with shoulders, shown in Figure 4.35. Six $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ sections of LiDAR data and six 10 m sections of navigation data were processed. These data sections were selected with an overlap of 2 m overlap in between them.

The standard parameters of $c=0.06\text{ m}^2$, $M_{slope} = 25$, $M_{ref} = 25$, $M_{pw} = 25$, $T_1 = 250$, $T_2 = 5$, $\alpha = 9$, $\beta = 0.001$, $\gamma = 3$, $\kappa_1 = 4$, $\kappa_2 = 2$, $\kappa_3 = 2$, $\kappa_4 =$

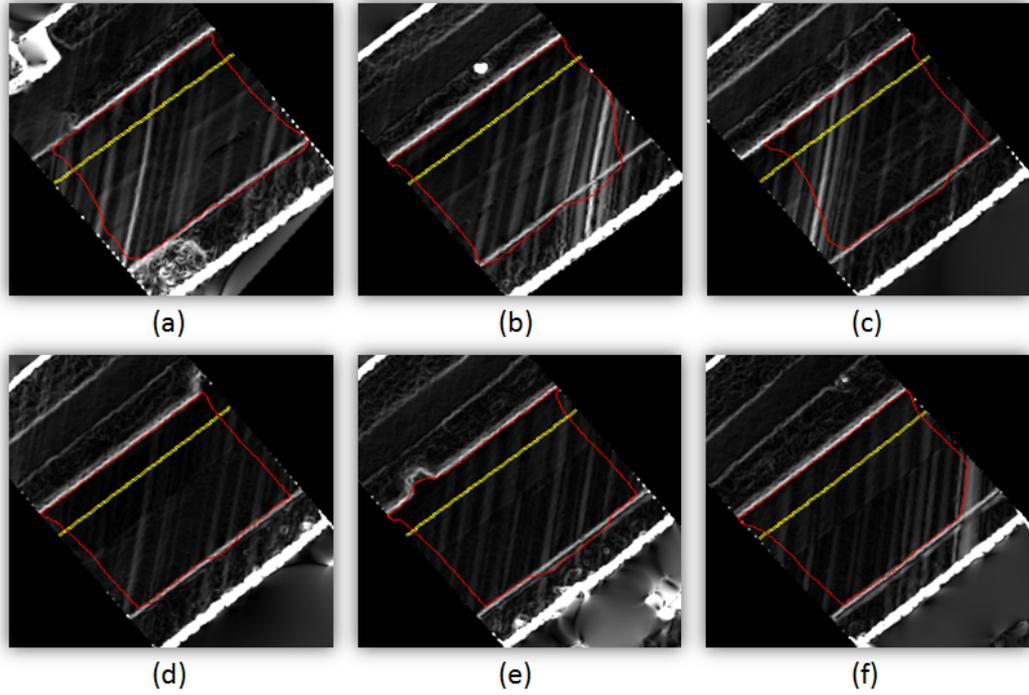


Figure 4.33: Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the urban road.

$3, \omega = 34^\circ$ and $\delta t = 0.03$ were input. We provided an increased weight to the balloon energy parameter to overcome noise which created an obstruction for the snake curve to move towards the road edges. As shown in Table 4.10, the point density for the right edges was $1/3$ of that for the previous two examples. This was the primary cause for the noise. The hierarchical threshold parameters we selected were $T_{slope} = 45, T_{ref} = 132$ and $T_{pw} = 55$. The ϕ angle was calculated from θ , as shown in Table 4.13. The number of GVF iterations were 600 while the snake curve converged to the road edges in 40 iterations. The final positions of the snake curve over the slope surface in the six data sections are shown in Figure 4.36. The extracted 3D left and

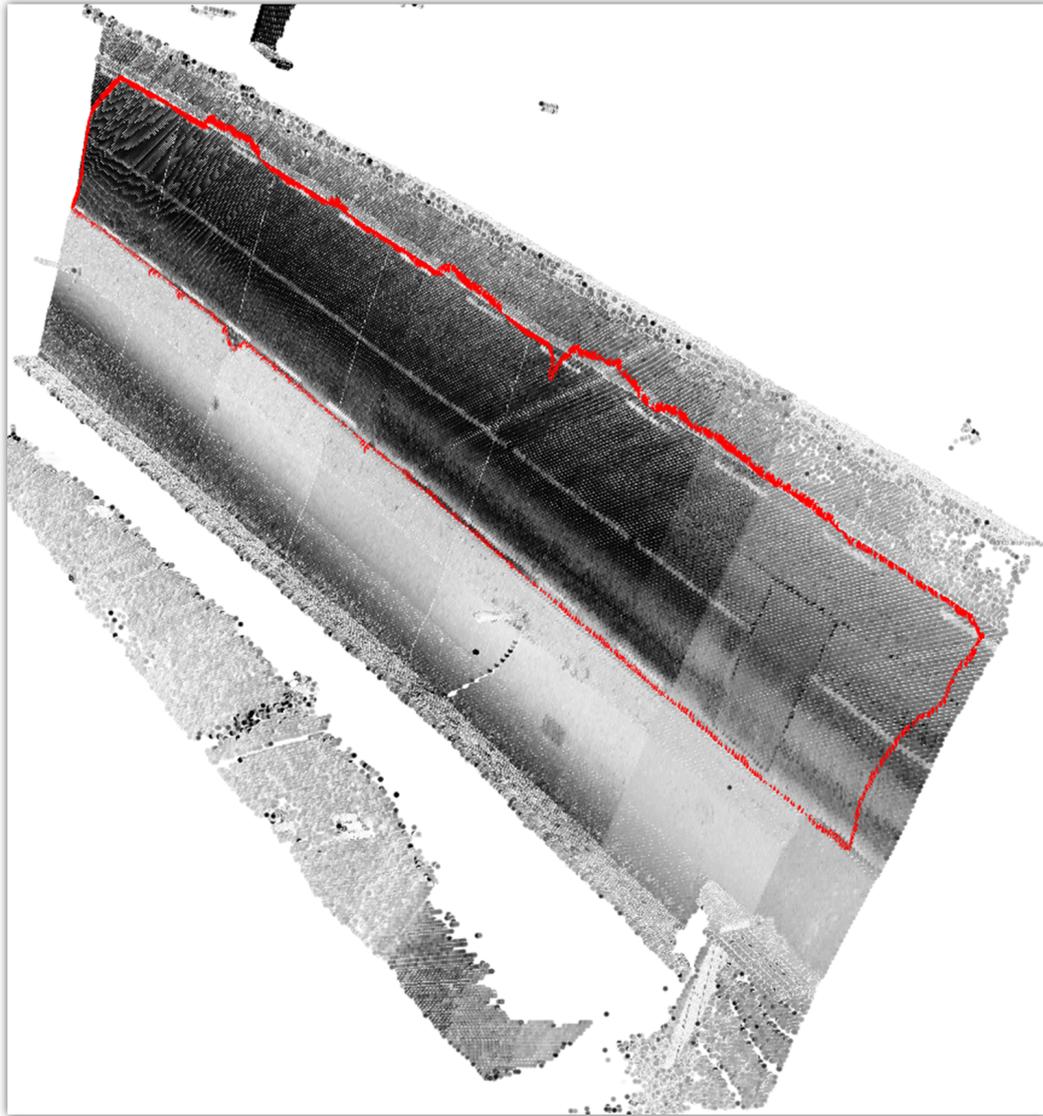


Figure 4.34: Extracted 3D left and right edges in the urban road section obtained through the manual selection of parameters.



Figure 4.35: Digital image of the national primary road section consisting of grass-soil edges with shoulders (Geographic location: $53^{\circ}38'8.80''\text{N}$ $7^{\circ}29'11.06''\text{W}$).

Navigation Section	θ (degree)	ϕ (degree)
1	127.27°	-37.27°
2	127.10°	-37.10°
3	126.90°	-36.90°
4	126.67°	-36.67°
5	126.41°	-36.41°
6	126.16°	-36.16°

Table 4.13: ϕ angle calculated from θ , average heading angle in each navigation section in the national primary road.

right edges in the national primary road section are shown in Figure 4.37.

Our algorithm was able to successfully extract the road edges in the national primary road section through the manual selection of parameters. In Figure 4.36(e), the snake curve extended incorrectly into the grass and soil area along the left edge. This was due to the increased balloon energy pushing the snake curve beyond the weak left edge points. The increased balloon energy was applied due to the lower LiDAR point density along the

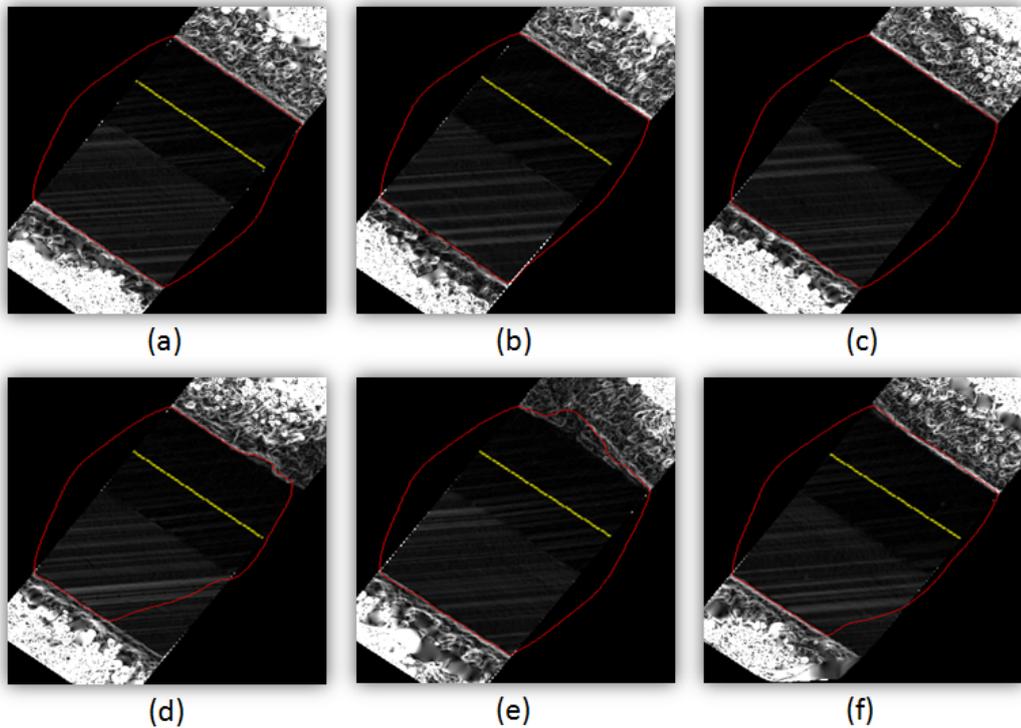


Figure 4.36: Final position of the snake curve over the slope raster surface obtained through the manual selection of parameters in the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the national primary road.

right side of the road section. A dataset with uniform point density should not exhibit these errors. In Figure 4.36(d) and (f), the snake curve was not able to fully converge to the right edges which was due to the lower LiDAR point density along that side. The increased balloon energy weight parameter provided was not sufficient for the snake curve to overcome the stronger noisy cells present at those points.

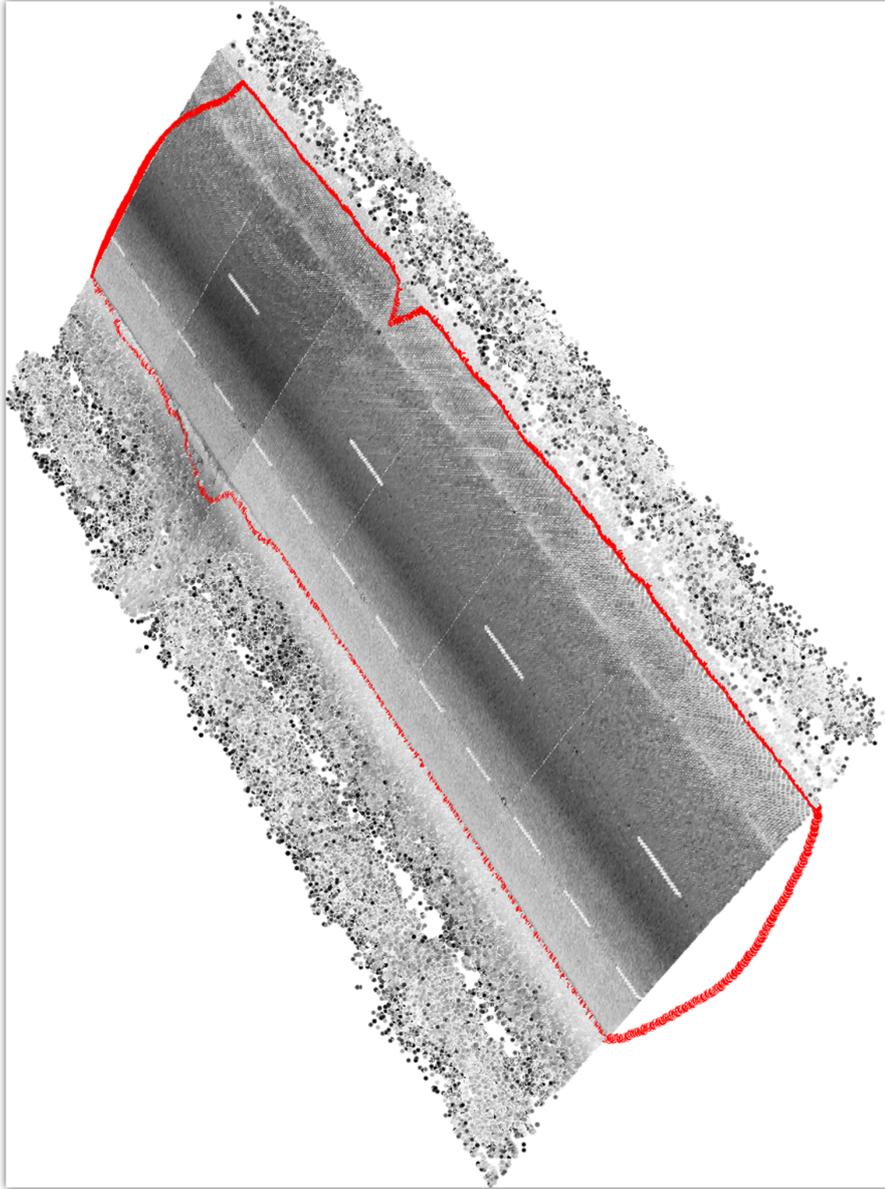


Figure 4.37: Extracted 3D left and right edges in the national primary road section obtained through the manual selection of parameters.

4.4.2 Automated Processing

We applied our road edge extraction algorithm to the road sections with a fixed set of hierarchical threshold parameters for all the road sections. In the previous section, we applied our algorithm with hierarchical threshold parameters selected empirically for each road section, as shown in Table 4.14. We calculated the median of these empirically selected threshold parameters

Road Section	T_{slope}	T_{ref}	T_{pw}
Rural	50	100	65
Urban	55	85	35
National Primary	45	132	55

Table 4.14: Hierarchical threshold parameters selected empirically for each road section.

as 50, 100 and 55 respectively. We use these median values in the algorithm to provide the hierarchical threshold parameters for all the road sections. In the following sections, we present our automated algorithm applied to these road sections. In each case, we used the same parameters as used in Section 4.4.1.

4.4.2.1 Grass-Soil Edges

We applied our automated road edge extraction algorithm to the same rural road section as described in Section 4.4.1.1. The hierarchical threshold parameters used in the algorithm were $T_{slope} = 50$, $T_{ref} = 100$ and $T_{pw} = 55$, while all other parameters were the optimal parameters used in the manual processing. The final positions of the snake curve over the slope raster surface in the six data sections are shown in Figure 4.38. The extracted 3D left and right edges in the rural road section are shown in Figure 4.39.

The automated road edge extraction results were found to be as high

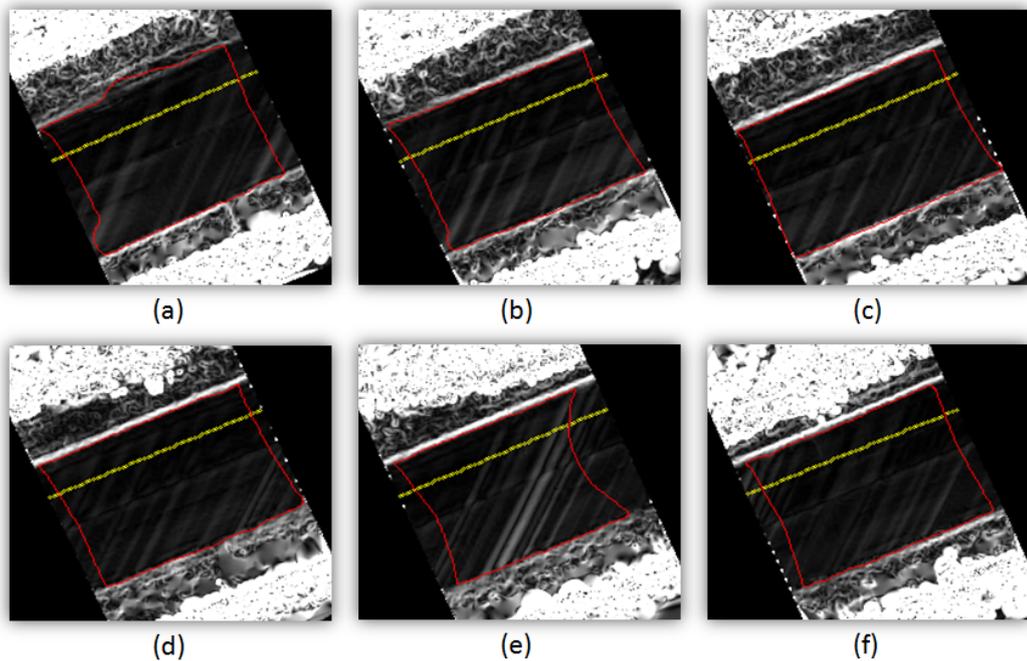


Figure 4.38: Automated final position of the snake curve over the slope raster surface for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the rural road.

quality as the results obtained through the manual approach. This is linked to the fact that the median and the empirically selected, T_{slope} and T_{ref} , parameters for this rural road section were the same in both processes.

4.4.2.2 Kerb Edges

We then applied our automated road edge extraction algorithm to the urban road section from Section 4.4.1.2. The median hierarchical threshold parameters were input along with the fixed optimal parameters. The final positions of the snake curve over the slope raster surface for the six data sections are shown in Figure 4.40. The extracted 3D left and right edges in the urban road section are shown in Figure 4.41.

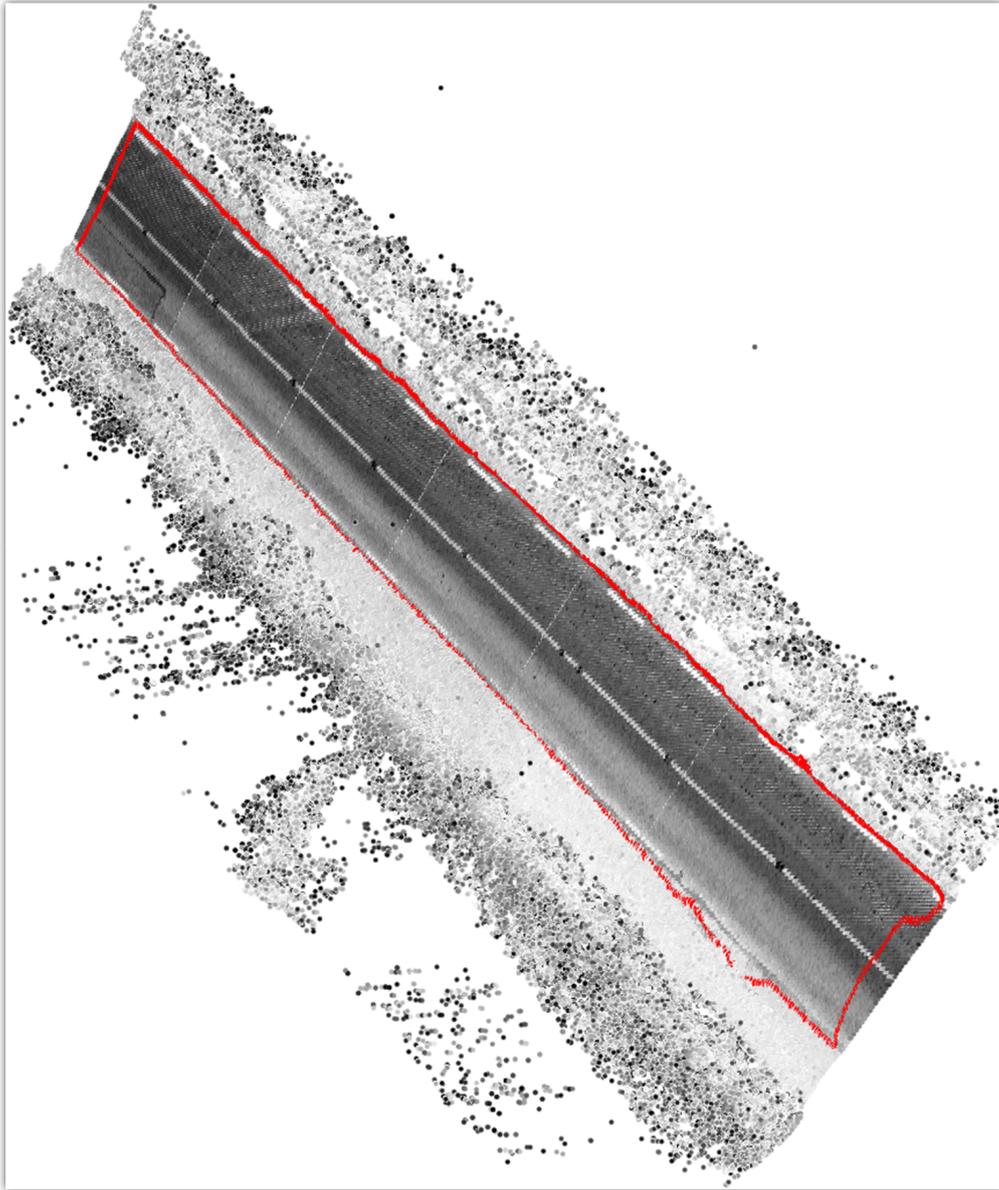


Figure 4.39: Automatically extracted 3D left and right edges in the rural road section.

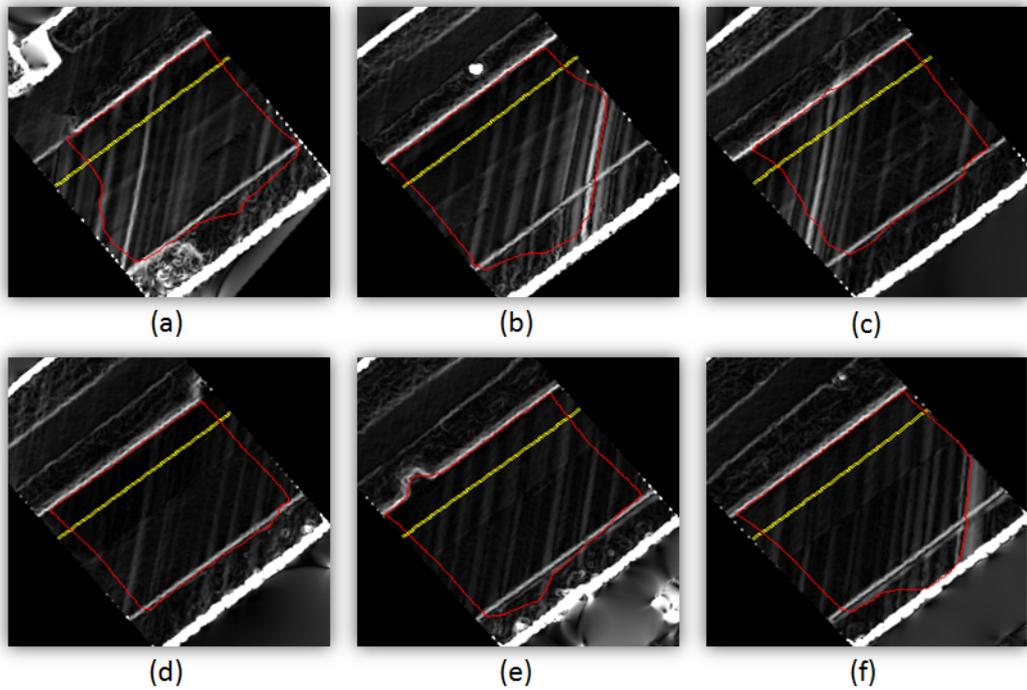


Figure 4.40: Automated final position of the snake curve over the slope raster surface for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the urban road.

Our automated algorithm was able to extract the road edges in the urban road section. In Figure 4.40(a), (b), (e) and (f), the snake curves extended beyond the right edge at some of the points. The lower LiDAR point density along the right side of the road section led to the generation of raster surfaces with noisy and weak cells along the right road edge. The median values of T_{ref} and T_{pw} for this road section were higher when compared with their empirically selected values. The use of these median values removed the weak cells along the right road edge, which caused the snake curve to extend beyond them.

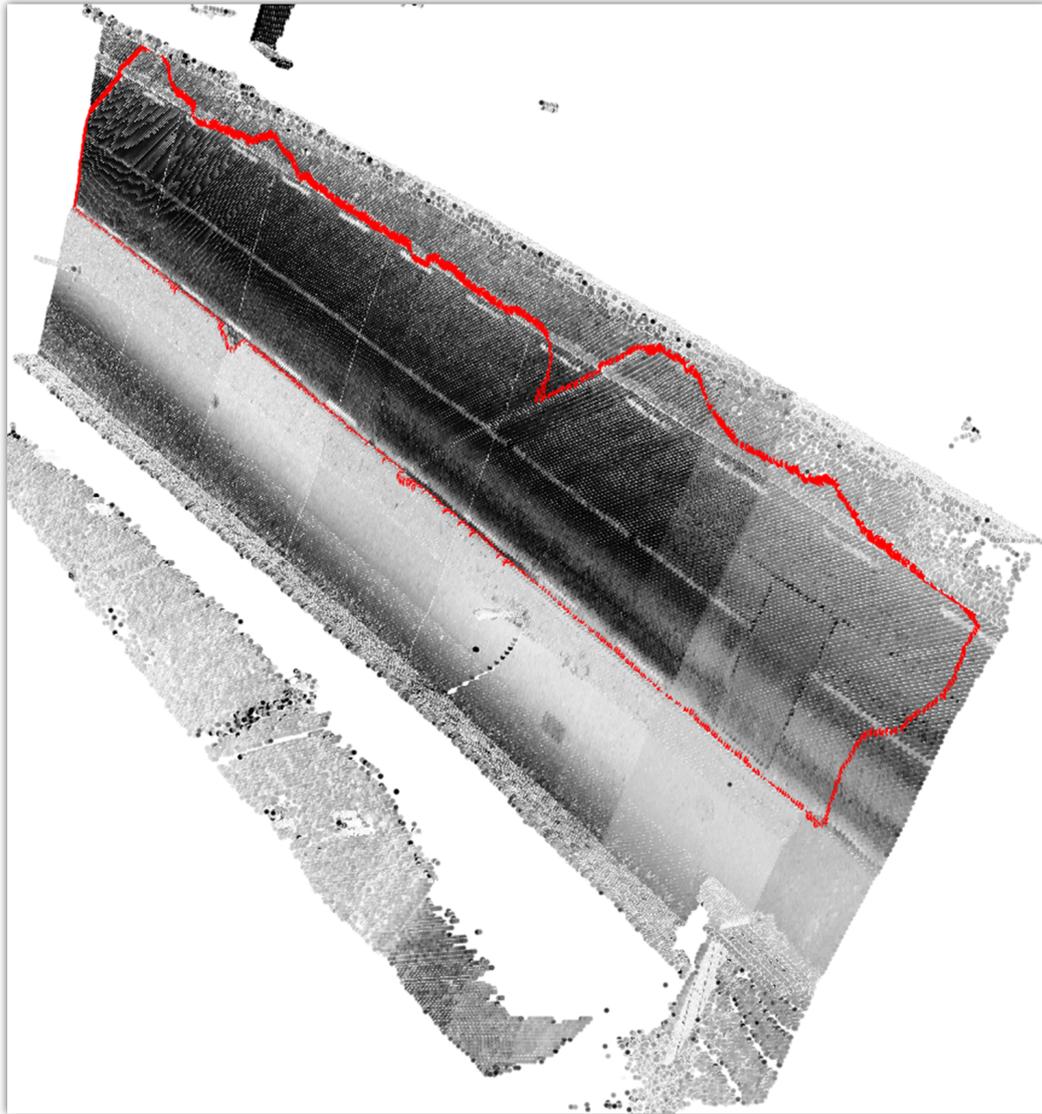


Figure 4.41: Automatically extracted 3D left and right edges in the urban road section.

4.4.2.3 Grass-Soil Edges with Shoulders

Finally, we applied our automated road edge extraction algorithm to the national primary road section from Section 4.4.1.3. The final positions of the snake curve over the slope raster surface in the six data sections are shown in Figure 4.42. The extracted 3D left and right edges in the national primary

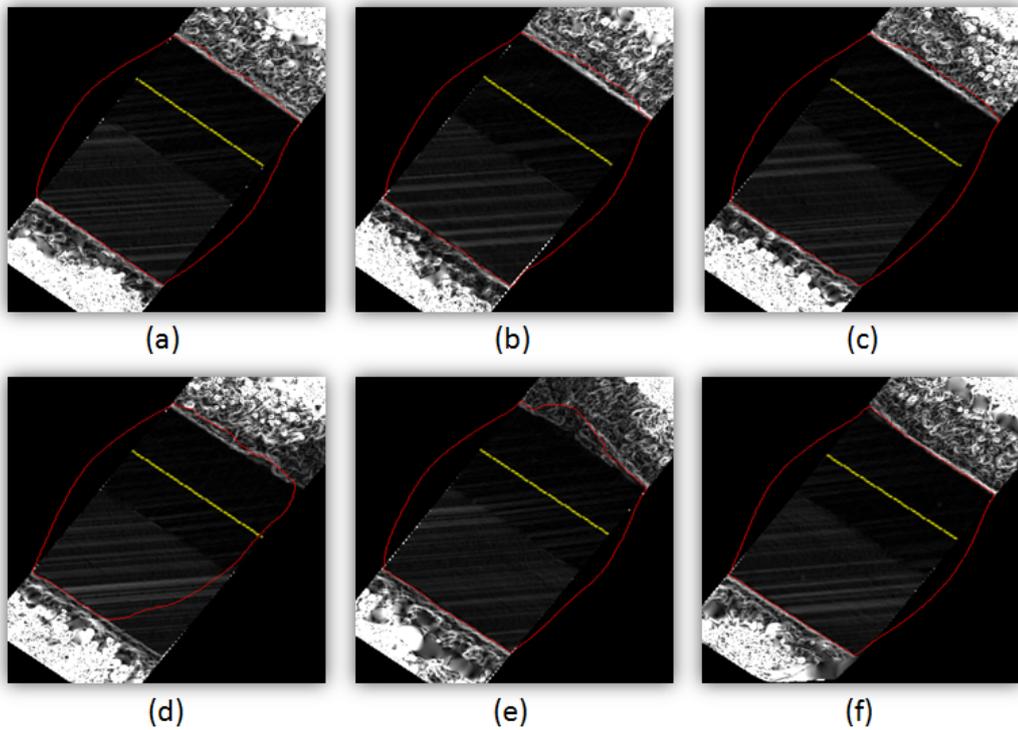


Figure 4.42: Automatic final position of the snake curve for the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section in the national primary road.

road section are shown in Figure 4.43.

The automated algorithm successfully extracted the left and right edges in the national primary road section. In Figures 4.42(b), (c), (d) and (e), the snake curves extended incorrectly into the grass and soil area along the

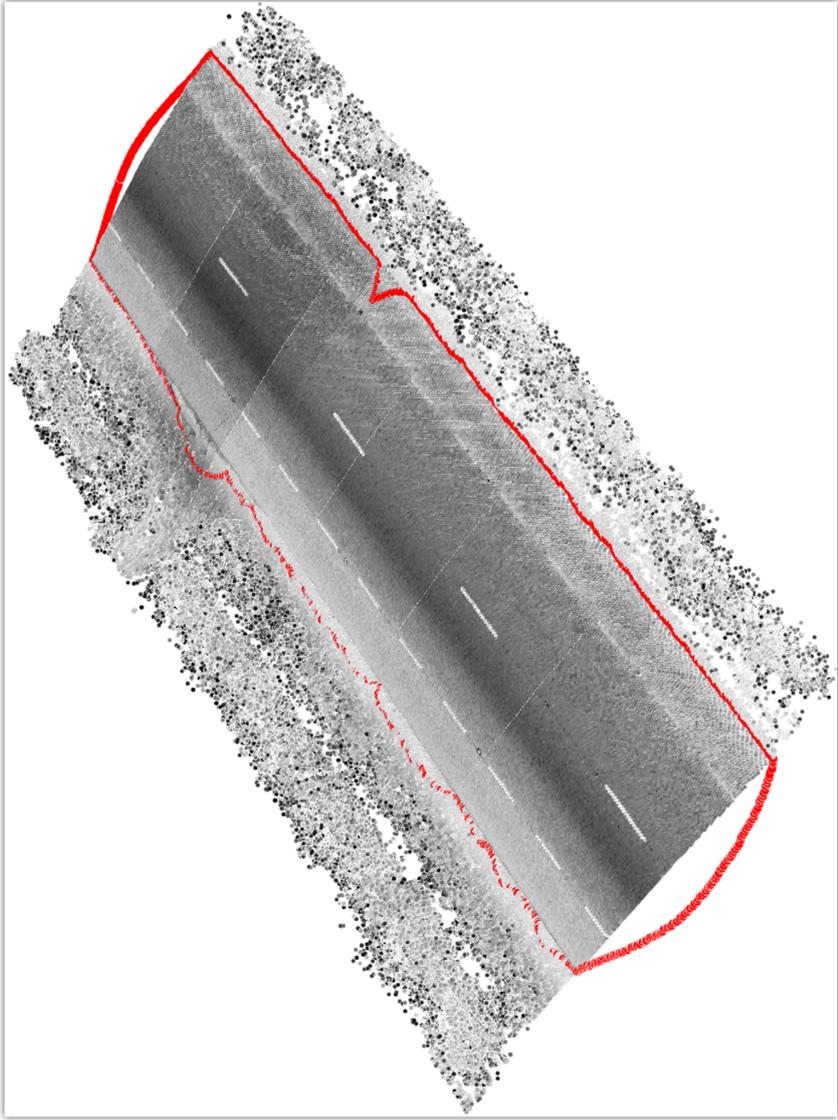


Figure 4.43: Automatically extracted 3D left and right edges in the national primary road section.

left edge. In this section of the national primary road, the stronger edge was between the grass and the soil rather than at the left road edge between the road surface and the soil. The median value of T_{slope} for this road section was higher than its empirically selected value. The use of this median value for the automated application removed the weak soil edges, which caused the snake curve to extend incorrectly into the grass and soil area in some areas.

4.4.3 Results Validation

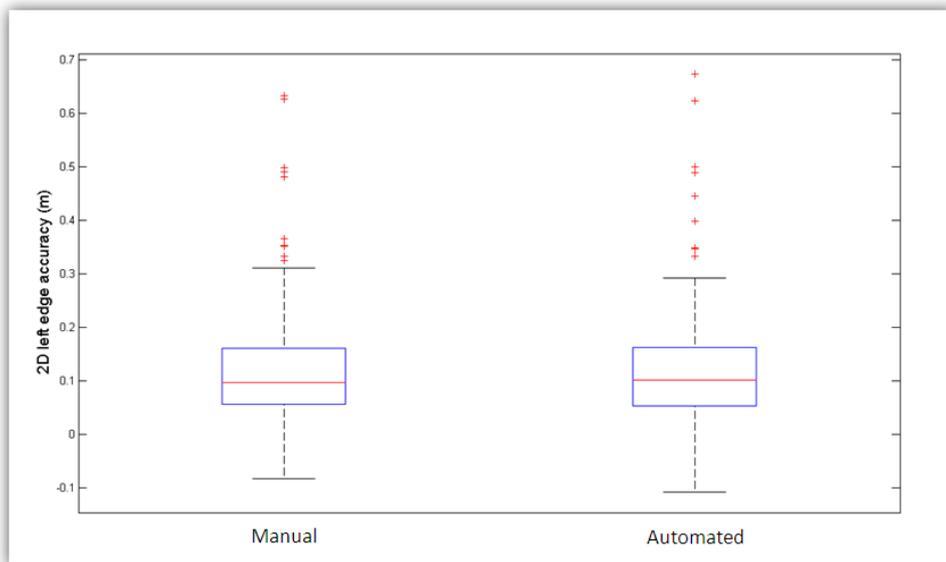
We tested the road edge extraction results using our validation algorithm. These results compared how closely our road edge extraction algorithm was able to extract the road edges when compared to their manual digitisation. The road edge validation results in the three road sections are presented in the following sections.

4.4.3.1 Grass-Soil Edges

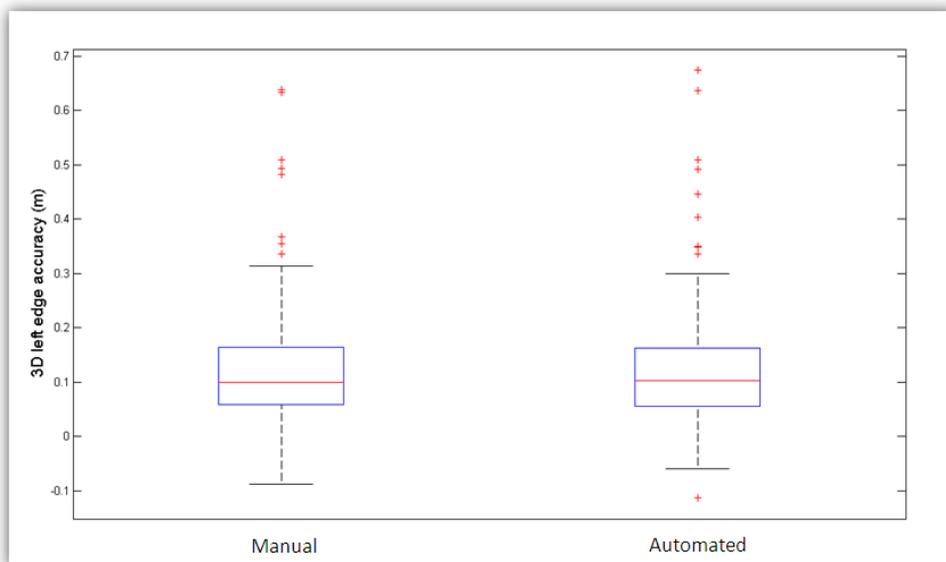
We digitised the left and right edges in the rural road section from the 3D LiDAR data. We applied our validation algorithm to estimate both the 2D and 3D accuracy of our manually and automatically extracted edges in the rural road section.

Box plots for the 2D and 3D accuracy of the manually and the automatically extracted left edges are shown in Figure 4.44, while box plots for the manually and the automatically extracted right edges are shown in Figure 4.45. We also carried out statistical analyses of the 2D and 3D accuracy of the manually and the automatically extracted left and right edges, as shown in Tables 4.15 and 4.16 respectively.

The 2D accuracy values were better than their respective 3D values. This

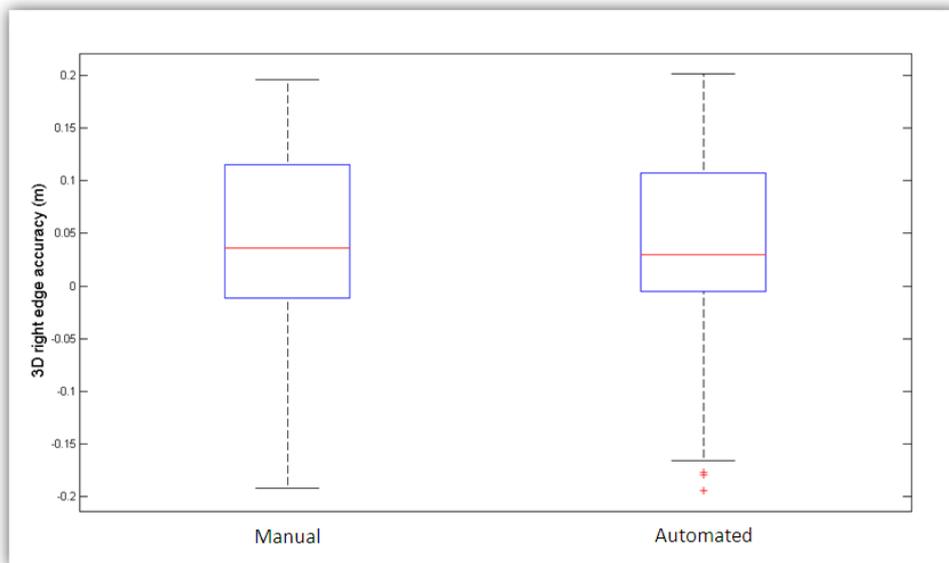


(a)

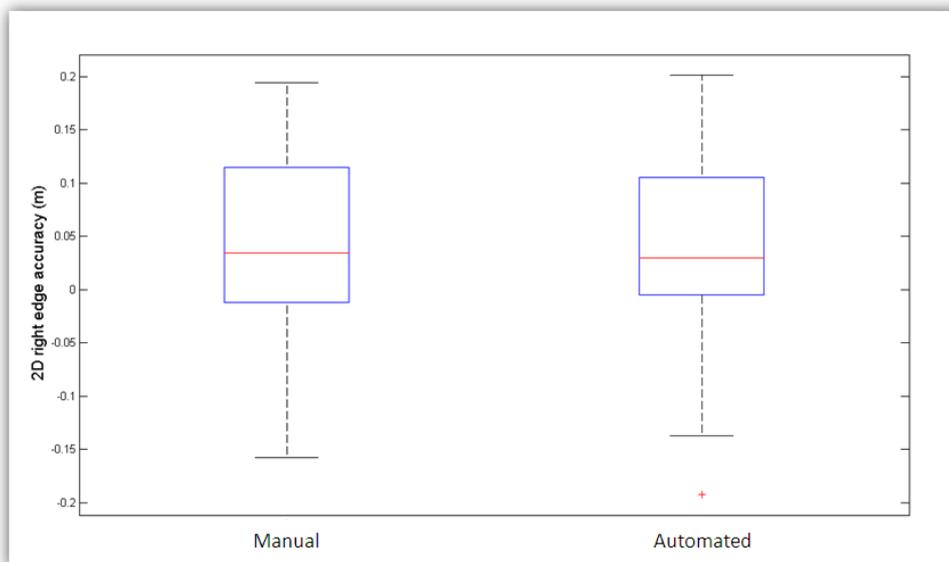


(b)

Figure 4.44: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the rural road section.



(a)



(b)

Figure 4.45: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the rural road section.

	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-0.082	-0.108	-0.088	-0.112
maximum (m)	0.632	0.672	0.638	0.674
lower adjacent (m)	-0.082	-0.108	-0.088	-0.059
upper adjacent (m)	0.310	0.291	0.314	0.30
25th percentile (m)	0.056	0.053	0.059	0.055
75th percentile (m)	0.160	0.162	0.164	0.163
mean (m)	0.129	0.126	0.132	0.128
median (m)	0.096	0.100	0.099	0.102
outliers (%)	8.55	7.69	8.55	8.55
inside ± 0.01 (%)	3.51	0.85	2.56	0.85
inside ± 0.025 (%)	5.98	8.55	5.13	7.69
inside ± 0.05 (%)	17.95	21.37	15.38	18.80
outside ± 0.2 (%)	21.38	21.38	21.38	21.38

Table 4.15: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the rural road section.

	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-0.157	-0.192	-0.192	-0.194
maximum (m)	0.193	0.201	0.196	0.201
lower adjacent (m)	-0.157	-0.137	-0.192	-0.166
upper adjacent (m)	0.193	0.201	0.196	0.201
25th percentile (m)	-0.012	-0.005	-0.012	-0.005
75th percentile (m)	0.115	0.105	0.115	0.107
mean (m)	0.043	0.039	0.042	0.038
median (m)	0.034	0.029	0.036	0.030
outliers (%)	0	0.85	0	2.56
inside ± 0.01 (%)	11.97	16.24	11.97	16.24
inside ± 0.025 (%)	26.50	30.78	25.64	30.78
inside ± 0.05 (%)	41.88	44.44	38.46	43.59
outside ± 0.2 (%)	0	0.85	0	1.71

Table 4.16: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted right edges in the rural road section.

was again expected and due to the inclusion of elevation points in the estimation of 3D accuracy values. In most cases, the minimum-maximum range was lowest for both the manually extracted left and right edges. The mean value closest to 0 was for both the automatically extracted left and right edges. The median value closest to 0 was for the manually extracted left edge and the automatically extracted right edge. The highest percentage of accuracy inside ± 0.01 m was for the manually extracted left edge and the automatically extracted right edge, while the percentages inside ± 0.025 m and ± 0.05 m were highest for both the automatically extracted left and right edges. The percentage outside ± 0.2 m was the same for the manually and the automatically extracted left edges but it was lowest for the manually extracted right edge.

Both the manual and the automated processes displayed high accuracy with their mean values ranging from 0.126 m to 0.132 m for the left edge and from 0.038 m to 0.043 m for the right edge. The left edge displayed poorer results than the right edge in most cases. This was due to the snake curve extending into the grass and soil area at some points along the left edge. The difference between the accuracy results of the manual and the automated road edges were minimal. This is because the median and the empirically selected, T_{slope} and T_{ref} , parameters used for this rural road section were the same in both processes.

4.4.3.2 Kerb Edges

We asked a different LiDAR user to digitise the left and right edges in the urban road section from the 3D LiDAR data. The reason for choosing a different user was to remove any bias in the digitisation process of the road

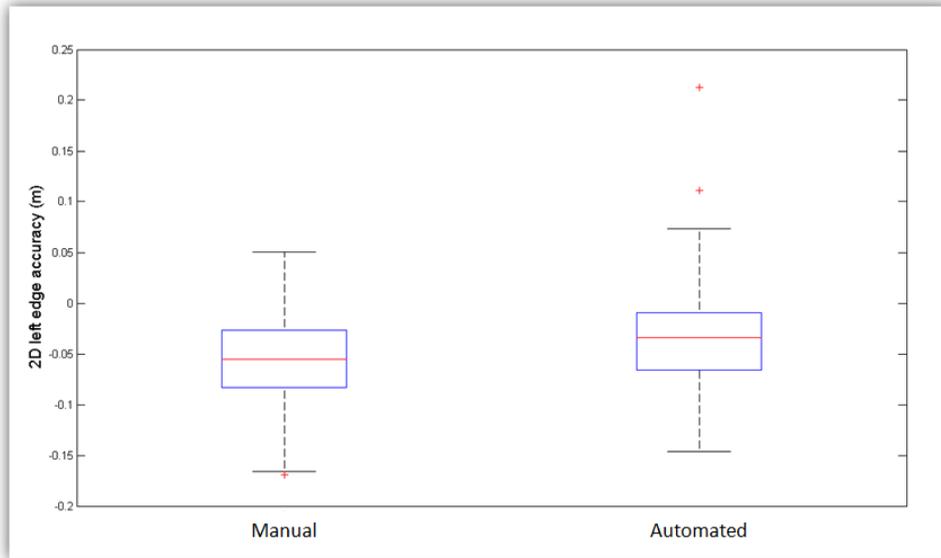
	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-0.169	-0.146	-0.169	-0.151
maximum (m)	0.051	0.213	0.075	0.239
lower adjacent (m)	-0.166	-0.146	-0.169	-0.151
upper adjacent (m)	0.051	0.073	0.025	0.078
25th percentile (m)	-0.083	-0.066	-0.092	-0.073
75th percentile (m)	-0.026	-0.009	-0.033	-0.011
mean (m)	-0.057	-0.033	-0.063	-0.037
median (m)	-0.056	-0.034	-0.069	-0.051
outliers (%)	1.01	2.06	4.04	4.12
inside ± 0.01 (%)	4.04	14.43	3.03	12.37
inside ± 0.025 (%)	21.21	31.96	17.17	23.71
inside ± 0.05 (%)	41.41	59.79	27.27	41.24
outside ± 0.2 (%)	0	1.03	0	1.03

Table 4.17: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the urban road section.

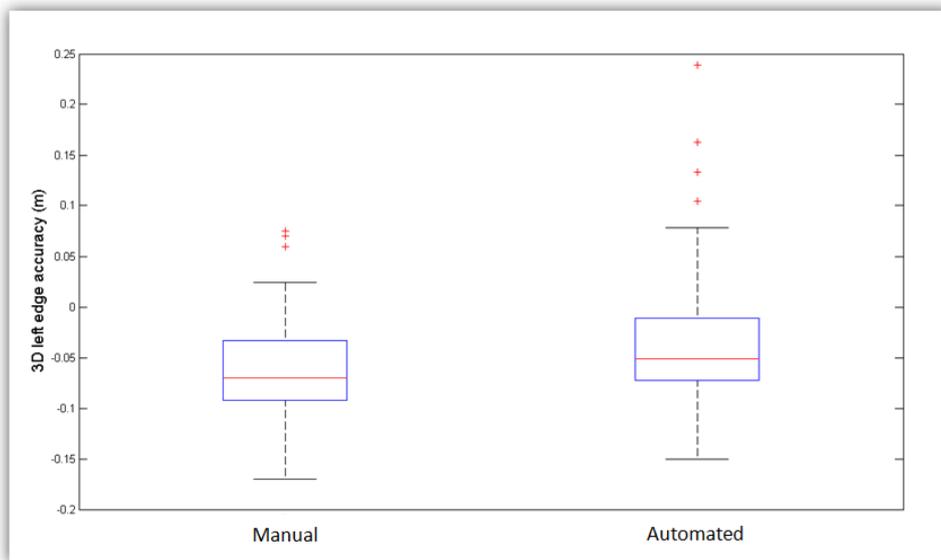
sections, as different users might have different perceptions when digitising 3D LiDAR data. We applied our validation algorithm to estimate both the 2D and 3D accuracy of the manually and the automatically extracted edges in the urban road section.

Box plots for the 2D and 3D accuracy of the manually and the automatically extracted left edges are shown in Figure 4.46, while box plots for the manually and the automatically extracted right edges are shown in Figure 4.47. Statistical analyses of the 2D and 3D accuracy of the manually and the automatically extracted left and right edges are shown in Tables 4.17 and 4.18.

The 2D accuracy values were again better than their respective 3D values. The minimum-maximum range was lowest for both the manually extracted left and right road edges. The mean and median values closest to 0 were for the automatically extracted left edge and the manually extracted right edge. The highest percentages of accuracy inside ± 0.01 m, ± 0.025 m and

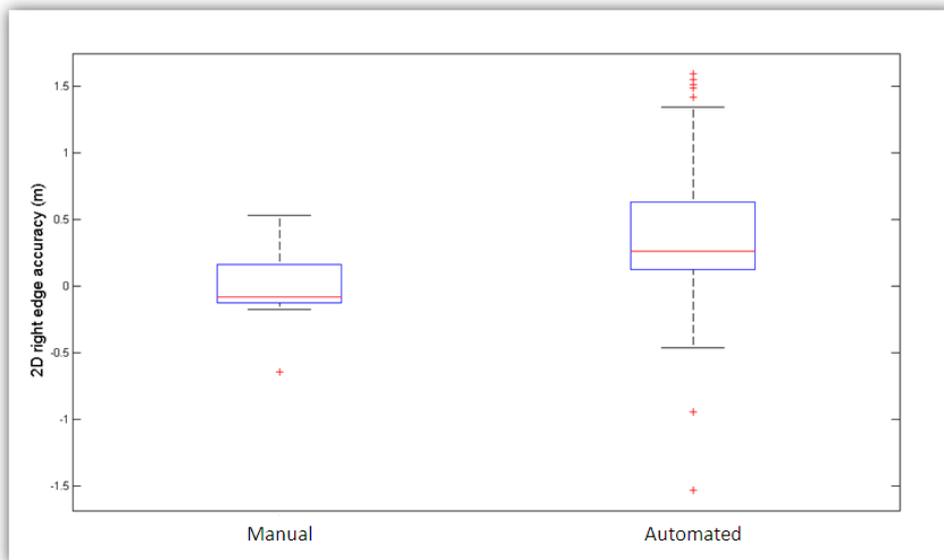


(a)

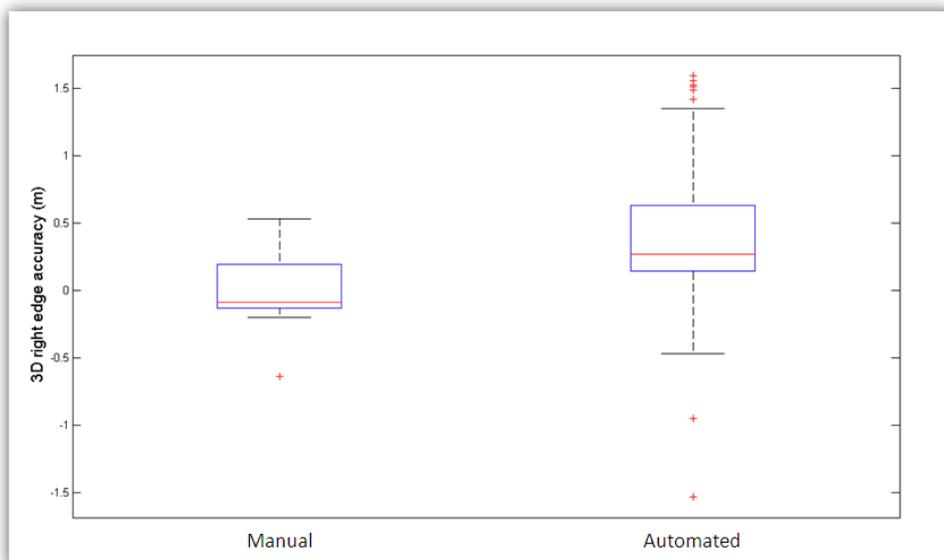


(b)

Figure 4.46: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the urban road section.



(a)



(b)

Figure 4.47: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the urban road section.

	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-0.64	-1.528	-0.64	-1.529
maximum (m)	0.532	1.594	0.535	1.594
lower adjacent (m)	-0.174	-0.46	-0.196	-0.47
upper adjacent (m)	0.532	1.35	0.535	1.35
25th percentile (m)	-0.121	0.124	-0.129	0.144
75th percentile (m)	0.163	0.635	0.192	0.635
mean (m)	0.013	0.396	0.014	0.404
median (m)	-0.078	0.267	-0.084	0.268
outliers (%)	1.01	8.25	1.01	8.25
inside ± 0.01 (%)	0	1.03	0	1.03
inside ± 0.025 (%)	1.01	2.06	0	2.06
inside ± 0.05 (%)	5.05	3.09	5.05	2.06
outside ± 0.2 (%)	23.23	60.82	24.24	42.27

Table 4.18: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted right edges in the urban road section.

± 0.05 m were associated, for the most part, with automatically extracted left and right edges. The lowest percentage outside ± 0.2 m was for both the manually extracted left and right edges.

Our algorithm was able to manually and automatically extract the left and right road edges with their mean accuracy values ranging from -0.063 m to -0.033 m for the left edge and from 0.013 m to 0.404 m for the right edge. In most cases, the accuracy results for the left edge were better than the right edge. This is due to the lower LiDAR point density along the right side of the road section. The accuracy results for the automated right edge were poor. This is related to the fact that the median values of T_{ref} and T_{pw} for this road section were higher than their empirically selected values. The use of these median values for automatically extracting the edges removed the weak cells along the right road edge which caused the snake curve to move beyond them.

4.4.3.3 Grass-Soil Edges with Shoulders

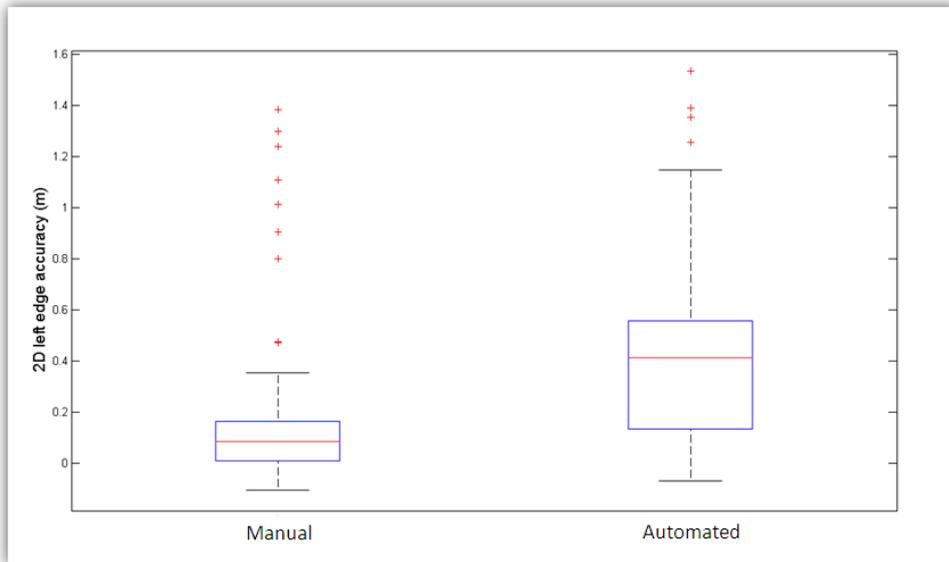
We asked a user to digitise the left and right edges in the national primary road section from the 3D LiDAR data. We applied our validation algorithm to estimate both the 2D and 3D accuracy of the manually and the automatically extracted edges in the national primary road section.

Box plots for the 2D and 3D accuracy of the manually and the automatically extracted left edges are shown in Figure 4.48, while box plots for the manually and the automatically extracted right edges are shown in Figure 4.49. We made a statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left and right edges as shown in Tables 4.19 and 4.20.

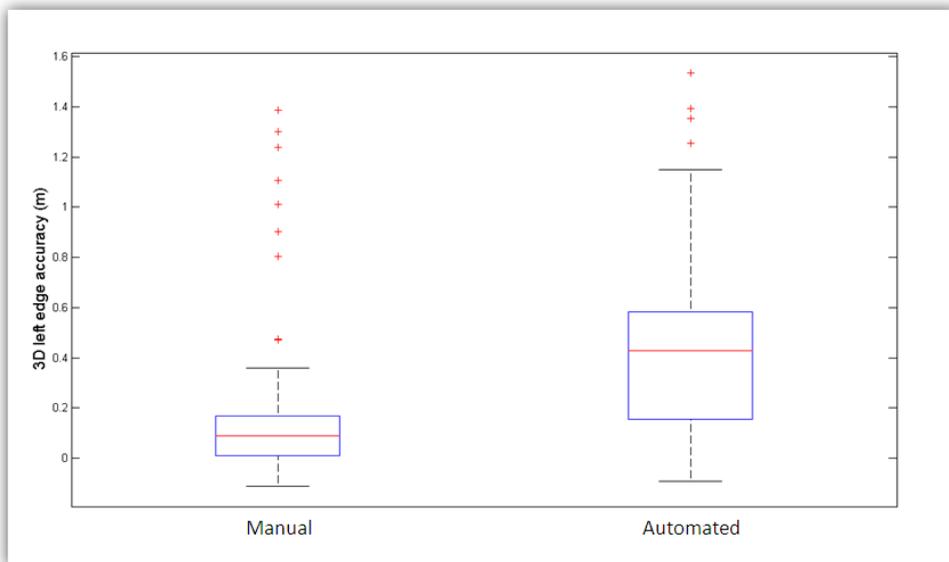
	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-0.105	-0.069	-0.109	-0.091
maximum (m)	1.385	1.533	1.385	1.533
lower adjacent (m)	-0.105	-0.069	-0.109	-0.091
upper adjacent (m)	0.35	1.15	0.358	1.149
25th percentile (m)	0.008	0.133	0.01	0.156
75th percentile (m)	0.162	0.557	0.169	0.582
mean (m)	0.165	0.41	0.169	0.426
median (m)	0.082	0.412	0.089	0.427
outliers (%)	9.89	4.25	9.89	4.25
inside ± 0.01 (%)	8.79	2.13	7.69	1.06
inside ± 0.025 (%)	18.68	9.57	18.68	4.26
inside ± 0.05 (%)	31.87	12.77	29.67	8.51
outside ± 0.2 (%)	19.78	69.15	21.98	70.21

Table 4.19: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.

The minimum-maximum range was lowest for the manually extracted left edges and for the automatically extracted right edges. The mean and median values closest to 0 were for both the manually extracted left and right edges.

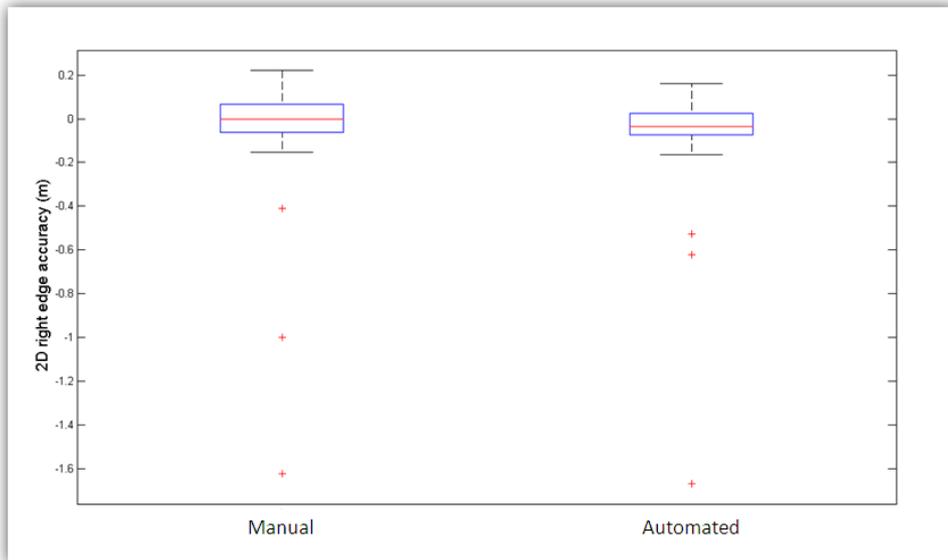


(a)

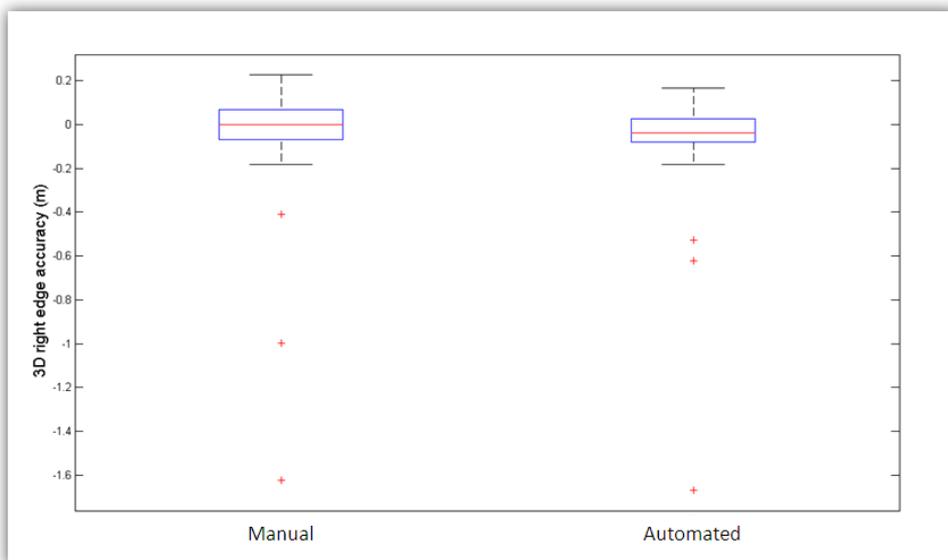


(b)

Figure 4.48: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.



(a)



(b)

Figure 4.49: Box plot for the (a) 2D and (b) 3D accuracy of the manually and the automatically extracted right edges in the national primary road section.

	2D		3D	
	Manual	Automated	Manual	Automated
minimum (m)	-1.622	-1.67	-1.622	-1.67
maximum (m)	0.219	0.162	0.225	0.165
lower adjacent (m)	-0.154	-0.166	-0.181	-0.181
upper adjacent (m)	0.219	0.162	0.225	0.165
25th percentile (m)	-0.063	-0.072	-0.068	-0.079
75th percentile (m)	0.067	0.025	0.067	0.025
mean (m)	-0.03	-0.049	-0.031	-0.052
median (m)	-0.001	-0.036	-0.001	-0.038
outliers (%)	3.29	3.19	3.29	3.19
inside ± 0.01 (%)	8.79	7.45	8.79	7.45
inside ± 0.025 (%)	28.57	21.28	25.27	19.15
inside ± 0.05 (%)	40.66	44.68	38.46	43.62
outside ± 0.2 (%)	4.39	3.19	4.39	3.19

Table 4.20: Statistical analysis of the 2D and 3D accuracy of the manually and the automatically extracted left edges in the national primary road section.

The highest percentages of accuracy inside ± 0.01 m, ± 0.025 m and ± 0.05 m were, in most cases, for the manually extracted left and right edges, while the percentage outside ± 0.2 m was lowest for the manually extracted left edge and the automatically extracted right edge.

The algorithm was able to manually and automatically extract the left and right edges with their mean accuracy values ranging from 0.165 m to 0.426 m for the left edge and from -0.052 m to -0.03 m for the right edge. The left edge results were not good as the snake curve incorrectly extended into the grass and soil area at some of the points. This was caused by the increased balloon energy pushing the snake curve beyond the weak left edge points. The manually extracted road edges displayed better results than the automatically extracted process. The median value of T_{slope} for this road section was higher than its empirically selected value. The use of this median value for automatically extracting the edges removed the weak cells which

caused the snake curve to move beyond them.

4.5 Discussion

We presented our algorithm, based on the combined use of GVF and balloon parametric active contour models, to extract road edges from the terrestrial mobile LiDAR data. Our algorithm was based around the assumption that the LiDAR elevation, reflectance and pulse width attributes can be used to distinguish the road surface from grass-soil edges and kerb edges. We converted the LiDAR attributes into 2D raster surfaces to simplify the problem domain and reduce computational expense. For example, one of a $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data consisted of 92,737 points, thus an iterative estimation of the GVF external energy for each point attribute would be computationally expensive.

We identified six challenges in developing an automated algorithm. The first challenge was to generate smooth 2D raster surfaces from the LiDAR attributes which are usually accompanied with high frequency noise. This issue was dealt through the use of a point thinning process. In this process, multi resolution terrain pyramids were generated from the LiDAR attributes and then smooth 2D raster surfaces were generated from the first level terrain pyramids using natural neighbourhood interpolation.

The second challenge was to estimate the GVF external energy terms from the raster surface values. First, we determined the object boundaries from the raster surfaces through the consecutive use of hierarchical thresholding and Canny edge detection. The mask size and threshold parameters used in the hierarchical thresholding were found empirically and were fixed for all the road sections. The upper and lower threshold parameters used in the Canny

edge detection were selected based on the output binary cell values obtained from the hierarchical thresholding. Once we identified the parameters for the internal and external energy terms, we were then able to apply our algorithm to a road section given a valid initialisation area.

The third challenge was to initialise the snake curve automatically. We developed a novel approach in which the snake curve was initialised over 2D raster surfaces based on the navigation track of the mobile van along the road section. We chose a parametric ellipse for initialising the snake curve as it suited the dimensions of the LiDAR data. The navigation points were used to estimate the semi-major and semi-minor axis of the snake ellipse, while the average heading angle was used to estimate an angle in between the major axis and the X-axis. The snake curve was initialised at the centre of the road based on the slant angle. This angle was selected empirically and was fixed for the road sections with similar width. Our developed initialisation approach negated the limitation associated with the manual intervention generally required in the parametric active contour models.

The fourth challenge was to extract the left and right edges from the snake curve. We batch processed consecutive individual road sections to obtain multiple snake curves. Including an overlap between the road sections ensured that the snake curves overlapped. We removed non road edge points in between the intersections points of the overlapping regions of the snake curves. It left the snake curves with just the left and right road edge points.

The fifth challenge was to validate the road edge extraction results. We developed an automated algorithm to test the accuracy of extracted left and right road edges. The validation algorithm was used to find out how closely our road edge extraction algorithm was extracting road edges, when com-

pared to their manual digitisation. In our validation algorithm, we considered navigation points as a reference to compute euclidean distances between the extracted and digitised road edge points.

The final challenge was to automate our road edge extraction algorithm. This was carried out through the recommendation of a set of methods and parameters. We analysed the generation of 2D raster surfaces, finding the first level terrain pyramid and natural neighbourhood interpolation to be efficient for generating smooth raster surfaces from the LiDAR attributes. We analysed the performance of our algorithm in relation to the raster surfaces generated with different cell sizes. The optimal value was selected for the algorithm that produced the best results in terms of completeness obtained by the snake curve, time taken by the snake curve to converge to the road edges and the quality of the extracted road edges. We demonstrated that choosing a small cell size increased the computational cost without providing any considerable improvement in the road edge extraction results, while choosing a large cell size affected the completeness and quality of the extracted road edges. By analysing the temporal performance of the algorithm for road sections with different dimensions, we identified the optimal length where the snake curve required least time to converge accurately to the road edges. We found the optimal internal energy parameters empirically after examining several combinations in the algorithm and their output. We demonstrated the importance of their optimal values in the algorithm. A higher value of the α parameter caused an inability in the snake curve to fully converge to the road edges, while a lower value caused the snake curve to be jagged at some of its points. Similarly, a higher value of the β parameter caused extreme bending in the snake curve, but a lower value produced the best

results. We found the best GVF and balloon external energy parameters where the highest weight was given to the slope GVF energy. Their values were found empirically after examining several combinations in the algorithm and their output. We also highlighted the importance of using all the external energy weight parameters in our algorithm. We demonstrated that the use of an increased balloon energy parameter can overcome noisy data as it can provide an additional inflation energy to the snake curve to overcome the noise. This noise can be due to a lower LiDAR point density along the right side of the road section.

We tested our road edge extraction algorithm in a variety of examples. We selected three different sections that covered 150 m of rural, urban and national primary roads. These road sections consisted of three different types of road edges. We selected these road sections to demonstrate the effectiveness of our algorithm to extract these distinct road edges using the same parameter set. We applied our algorithm to these road section using the manually selected optimal and the automatically selected parameters. In the manual processing, the optimal hierarchical threshold parameters were selected manually for each road section, while in the automated processing, the same set of these parameters were selected for all the road sections. All other parameters were common for both sets of experiments.

We tested the accuracy of road edge extraction using our validation algorithm. We calculated the accuracy of our manually extracted and automatically extracted road edges when compared to their manual digitisation. We asked different users to digitise the left and right edges in the road sections from the 3D LiDAR data. The reason for this was to remove any bias in the digitisation process of the road sections. Our road edge extraction

algorithm successfully extracted high quality road edges. In the rural road section, the best mean accuracy values were 0.126 m and 0.038 m for the automated left and right edge respectively. The best median accuracy values were 0.096 m and 0.029 m for the manual left edge and the automated right edge respectively. The right edge accuracy results were better than the left edge. This was due to the weak cells along the left edge which caused the snake curve to extend incorrectly into the grass and soil area. The difference between the manual and the automated accuracy results was minimal due to the similar values of the empirically and the median selected, T_{slope} and T_{ref} parameters for this road section. In the urban road section, the mean and median accuracy values, -0.033 m and -0.034 m, were found to be best for the automated left edge, while 0.013 m and -0.078 m for the manual right edge. For this road section, the left edge accuracy was better than the right edge in most results. This is due to the lower LiDAR point density along the right side of the road section. The automated right edge results were poor due to the use of median values of T_{ref} and T_{pw} which were higher than their empirically selected values. These values removed the weak cells along the right edge which resulted the snake curve to move beyond the actual edge location. In the national primary road section, the mean and median accuracy values, 0.165 m and 0.082 m, were best for the manual left edge, while -0.03 m and -0.001 m for the manual right edge. The left edge accuracy results were poor due to the increased balloon energy pushing the snake curve beyond the weak left edge points. The automated road edge results were poor due to the use of median value of T_{slope} which was higher than its empirically selected value. In the next chapter, we present our automated algorithm for extracting road markings from terrestrial mobile LiDAR data

using the identified road edges.

Chapter 5

Road Marking Extraction

Terrestrial mobile laser scanning systems usually record the reflectance of an illuminated surface in the form of intensity data. The recorded intensity data can be used to distinguish road markings that produce high reflectivity due to their retro-reflective property. Road markings are used to provide guidance and instruction to road users for safe and comfortable driving. Knowledge of the location, dimension and condition of road markings can be useful for driver assistance systems, road safety and route network maintenance.

In this chapter, we present an automated algorithm to extract road markings from terrestrial mobile LiDAR data. We apply our automated road edge extraction algorithm to estimate road boundaries in the form of a snake curve. The output snake curve is used to identify the LiDAR points that belong to the road surface. Knowledge of the road surface area facilitates a more efficient and accurate road marking extraction algorithm. In Section 5.1, we present a stepwise description of the road marking extraction algorithm. It involves a description of 2D intensity and range raster surfaces generation from the LiDAR points belonging to the road surface. We do not use the

reflectance attribute in our algorithm as its values provided by the Riegl VQ-250 laser scanner are not correctly normalised. We provide a detailed description of a range dependent thresholding function which we have developed and applied to the intensity raster surface to extract road markings. We discuss the application of morphological operations to complete the shapes of extracted road markings and to remove noise. In Section 5.2, we present our analysis to find the most applicable values of input parameters required to automate our road marking extraction algorithm. In Section 5.3, we test our algorithm on various road sections, demonstrating the successful extraction of different types of road markings. We present the results following validation of the road marking extraction process. In Section 5.4, we conclude the chapter by discussing the novelty and limitations of our algorithm.

5.1 Algorithm

A work flow of our road marking extraction algorithm is shown in Figure 5.1. In the following sections, we describe a number of processing steps involved in our algorithm.

5.1.1 Road Surface Estimation

We input the LiDAR data and the snake curve estimated using our automated road edge extraction algorithm, described in Chapter 4. In Step 1 of our road marking extraction algorithm, we use the snake curve to identify the LiDAR points that belong to the road surface. A snake curve is laid over the LiDAR data, as shown in Figure 5.2(a). The LiDAR points outside the snake curve are removed, while the inner points are retained to estimate the

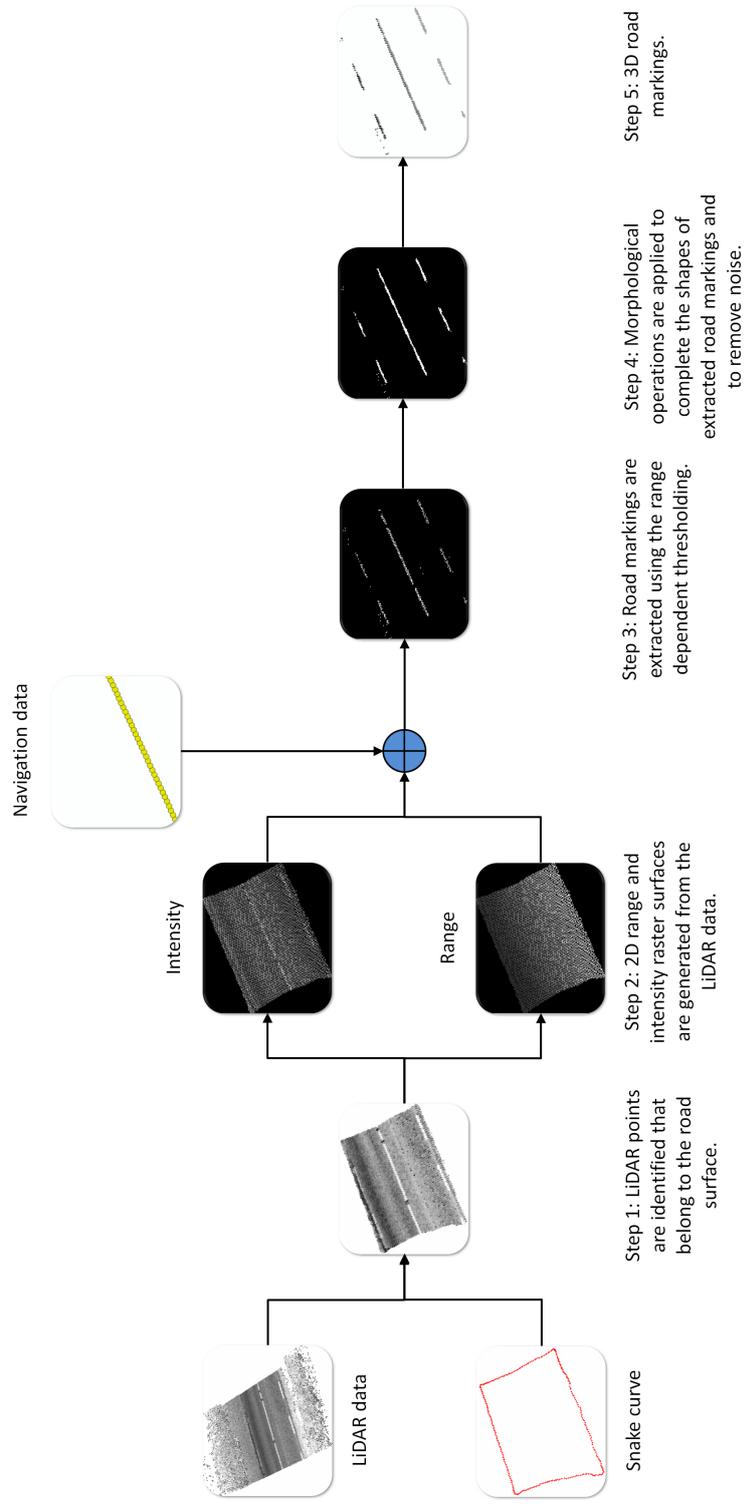


Figure 5.1: Road marking extraction algorithm.

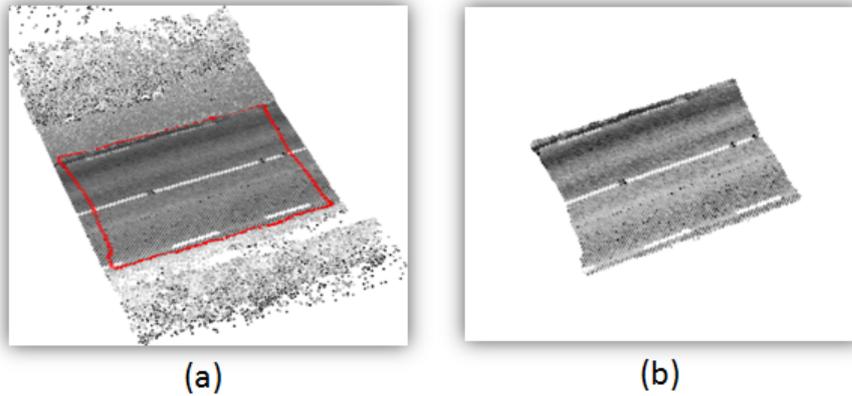


Figure 5.2: Snake curve is (a) laid over the LiDAR points to (b) estimate the road surface.

road surface, as shown in Figure 5.2(b).

5.1.2 2D Raster Surface Generation

We use LiDAR intensity and range attributes in our algorithm to extract the road markings. In Step 2 of our algorithm, we generate 2D intensity and range raster surfaces from the LiDAR data using a preselected cell size. A detailed analysis on the selection of an optimum cell size is provided in Section 5.2.1. The value of each cell in the raster surfaces is estimated as the average of the intensity and range values of the LiDAR points that fall within the 2D boundary of the cell. The intensity and range raster surface values are normalised with respect to their global minimum and maximum, and converted into an 8-bit data type. This will allow for the use of one set of values for all road sections. An example of 2D intensity and range raster surfaces generated from the LiDAR data is shown in Figure 5.3.

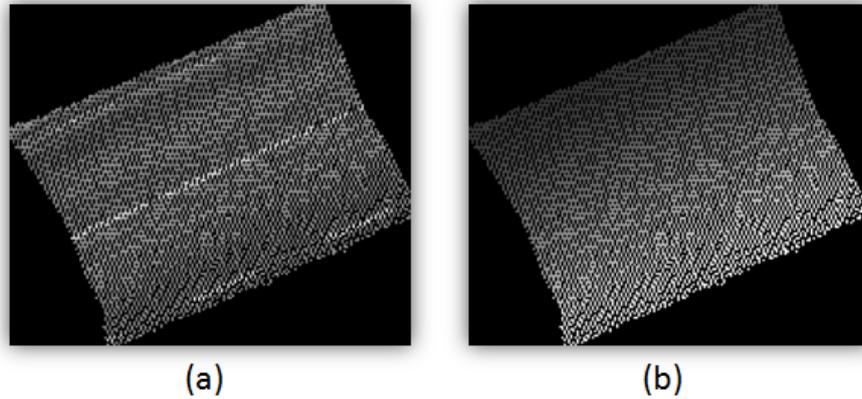


Figure 5.3: 2D raster surfaces generated from the LiDAR data: (a) intensity and (b) range.

5.1.3 Range Dependent Thresholding

Road markings are generally more reflective than the road surface. This results in the intensity values of the laser returns from the road markings being higher than the background surface. However, apart from the reflectivity of the illuminated surface, there are two other factors affecting the intensity values, the distance from the laser scanner and the incidence angle of the laser pulse. These factors need to be addressed in any algorithm extracting road markings. In Step 4 of our algorithm, we apply range dependent thresholding to the intensity raster surface. We use the navigation data to select a range value that is used to apply multiple threshold values. In most terrestrial mobile laser scanning systems, the laser scanner is mounted on a mobile van at some horizontal and vertical inclined position in order to produce rich 3D information. Let us suppose that a laser scanner is mounted on the back of the mobile van at a η angle from both the horizontal and vertical axes of the vehicle. This inclined position modifies the initial scanning point from directly below the scanner to the position, O , shown in Figure 5.4. R is

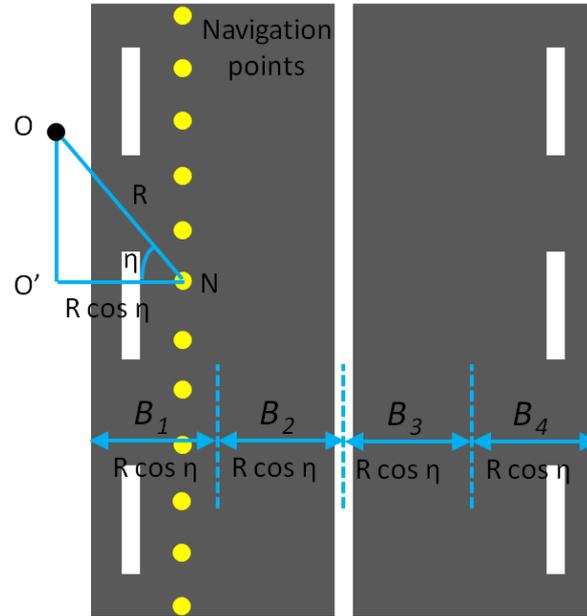


Figure 5.4: Navigation data is used to select a range value to apply multiple threshold values to the intensity raster surface.

the range of a laser pulse from the navigation point N to the initial scanning point O . The transverse range from the point N to O' is estimated as $R \cos \eta$. For each laser return, we replace their range value with the new transverse range value using the inclination angle η . We use the estimated transverse range value $R \cos \eta$ to divide the intensity raster surface into different blocks which allows us to threshold the intensity values based on their range from the scanner.

We apply a different threshold value to each block of the intensity raster surface to deal with the factors that effect intensity values. The road surface is usually constructed with a non-planar shape, shown in Figure 5.5. This surface is engineered to allow rain water to run off the road surface to reduce water pooling which can damage the road surface over time. This type of road surface changes the laser pulses incidence angle, resulting in an increased

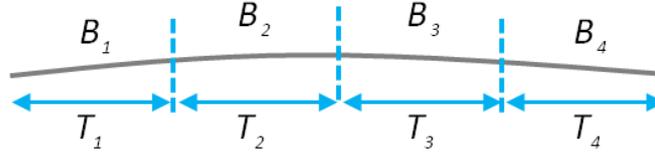


Figure 5.5: Side view of the non-planar road surface.

reduction in intensity returns as the range increases over a flat road surface. We apply the threshold value T_1 to data in block B_1 to extract road markings. In B_2 block, the road surface is oriented towards the laser scanner relative to block B_1 which results in an acquisition of stronger reflections from the road surface. For this reason, we apply a different threshold value T_2 in block B_2 such that $T_2 > T_1$. The road surface is oriented away from the laser scanner in blocks B_3 and B_4 , for which we apply different threshold values T_3 and T_4 . We select a single optimal threshold value empirically and use it to estimate the values of T_1, T_2, T_3 and T_4 . A detailed analysis on the selection of the range dependent threshold values is provided in Section 5.2.2. An example of road markings extracted from the intensity raster surface using range dependent thresholding is shown in Figure 5.6.

5.1.4 Morphological Operations

The extracted road markings from Step 3 may be incomplete and contain noise that is introduced through the use of thresholding. To overcome this, we propose the use of binary morphological operations [HG92]. In Step 4 of our algorithm, the thresholded raster surface is converted into a binary image and is processed using morphological operations. The two morphological operators we will use are dilation and erosion. We apply the dilation operation to the binary image in which a structuring element is placed over the



Figure 5.6: Road markings extracted from the intensity raster surface using range dependent thresholding.

cells of the image. The purpose of dilation is to use the structuring element to grow cells with a value of 1 in order to fill in any holes. A structuring element consists of a binary matrix that represents the selected shape and size. Examples of structuring elements with different shapes and sizes are shown in Figure 5.7. A central element of the matrix represents an origin and

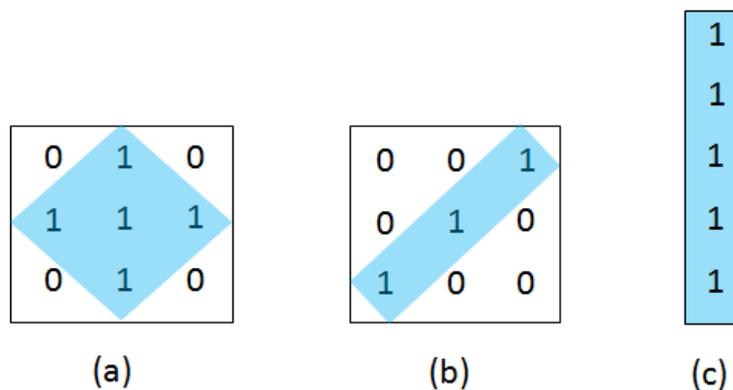


Figure 5.7: Structuring elements: (a) diamond shaped with radius = 1, (b) linear shaped with length = 3 and angle, $\phi' = 45^\circ$ and (c) linear shaped with length = 5 and angle, $\phi' = 90^\circ$.

the elements with a value of 1 describe a neighbourhood of the structuring

element. The origin of the structuring element is positioned over each cell in the binary raster surface to dilate that cell along the neighbourhood of the structuring element.

A linear shaped structuring element is used to dilate the cells of each road marking. We choose a linear shape for the structuring elements due to the general linear patterns of road markings. The linear shaped structuring element is used with a ϕ' angle that is calculated from the average heading of the mobile van along the road section under investigation. This angle is used in order to dilate the road markings along the longitudinal direction. A process of calculating the ϕ' angle from the average heading angle of the mobile van, θ , is described in Figure 5.8. The θ angle provides the direction of the mobile van's trajectory with respect to the north direction and the ϕ' angle is measured in the anticlockwise direction from the horizontal axis. If the value of θ angle lies in between 0° and 90° , then the ϕ' angle is estimated as $90^\circ - \theta$ in the anticlockwise direction as shown in Figure 5.8(a). Similarly, the ϕ' angle can be estimated for other possible values of the θ angle as shown in Figure 5.8(b), (c) and (d).

The length of the linear shaped structuring element is selected empirically. We use the same value in each road section, which allows us to automate the morphological operations in our algorithm. An example of the extracted road markings dilated using a linear shaped structuring element with length=9 and $\phi' = 38.37^\circ$ is shown in Figure 5.9.

As can be seen from Figure 5.9(b), the process of dilation can negatively affect the results by increasing the significance of the noise. We propose a method to deal with this, which involves two processes. The first process is based on a priori knowledge of the dimensions of road markings used to

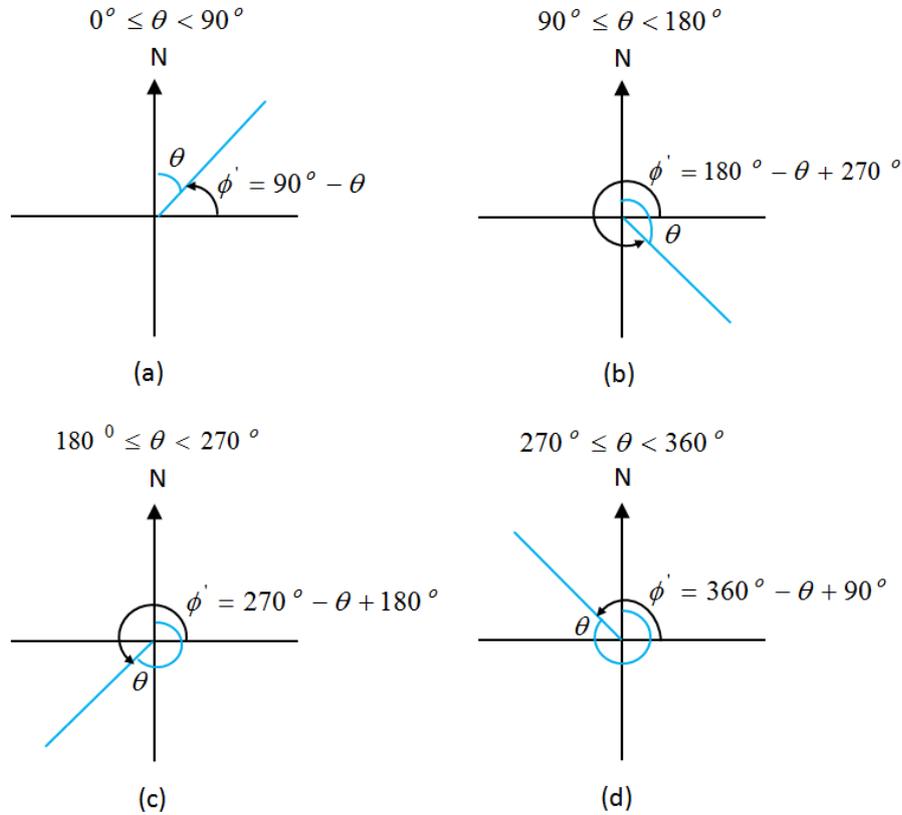


Figure 5.8: Estimation of ϕ' angle of the linear shaped structuring element from the average heading angle of the mobile van, θ , along the road section that can lie in between (a) 0° and 90° , (b) 90° and 180° , (c) 180° and 270° and (d) 270° and 360° .

remove the noise. The second is using another morphological operator called erosion to counteract the artificial growth in the dimensions of the road markings we caused using dilation to fill in the holes.

In the first process, we group cells into objects in the dilated image using connectivity. If a cell has a value of 1 then it is connected to the cells whose values are 1 and are directly above, below, left or right of that cell. We calculate the length and average width values of each object in the dilated image. Objects whose length and average width values are less than a min-

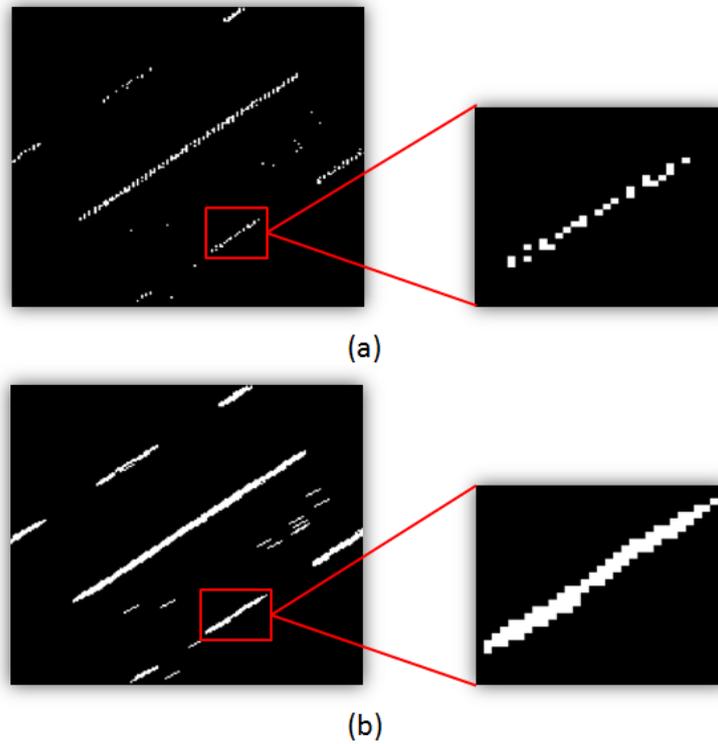


Figure 5.9: Dilation operation: (a) input binary image with an inset picture of their road marking cells and (b) dilated image with an inset picture of their dilated road marking cells.

imum length and width threshold are considered as noise and are removed from the image. An example of a noise cell removal process from the dilated image is shown in Figure 5.10.

In the second process, we apply an erosion operation to the dilated image. In an erosion operation, cells are removed from the road marking cells using a structuring element. The linear shaped structuring element used for dilation is also applied to erode the road markings. This is carried out in order to retain the original boundary shape of the road markings. An example of the dilated road markings eroded using the linear shaped structuring element with length=9 and $\phi' = 38.37^\circ$ angle is shown in Figure 5.11.

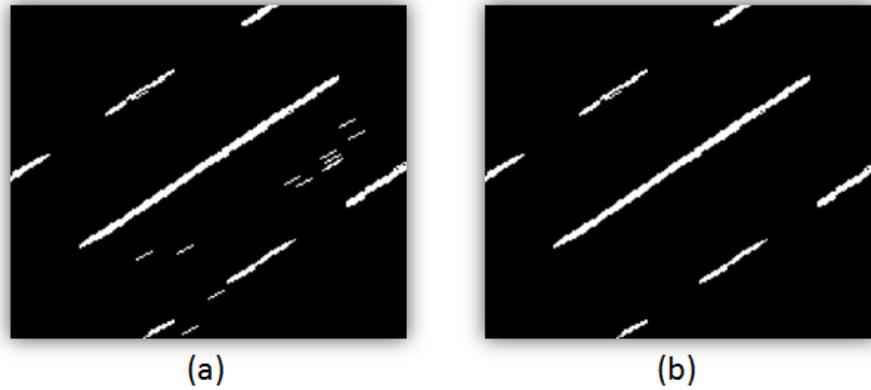


Figure 5.10: Noise removal process: (a) input dilated image and (b) noise cells removed from it.

Thus, the morphological operations applied to the extracted road markings are able to reconstruct their inherent shape and to remove noise, as can be seen in Figure 5.12.

5.1.5 3D Road Markings

In Step 5 of our algorithm, we extract the 3D road markings using the 2D output. The original 3D LiDAR points which are contained within the 2D road marking cell boundaries are extracted. An example of the extracted 3D road markings is shown in Figure 5.13. In the next section, we present our analysis of the input parameters used to automate our road marking extraction algorithm.

5.2 Automation Analysis

Our road marking extraction algorithm requires two input parameters, the cell size value for converting the LiDAR data into 2D raster surfaces and the range dependent threshold value. To implement an automated algorithm, we

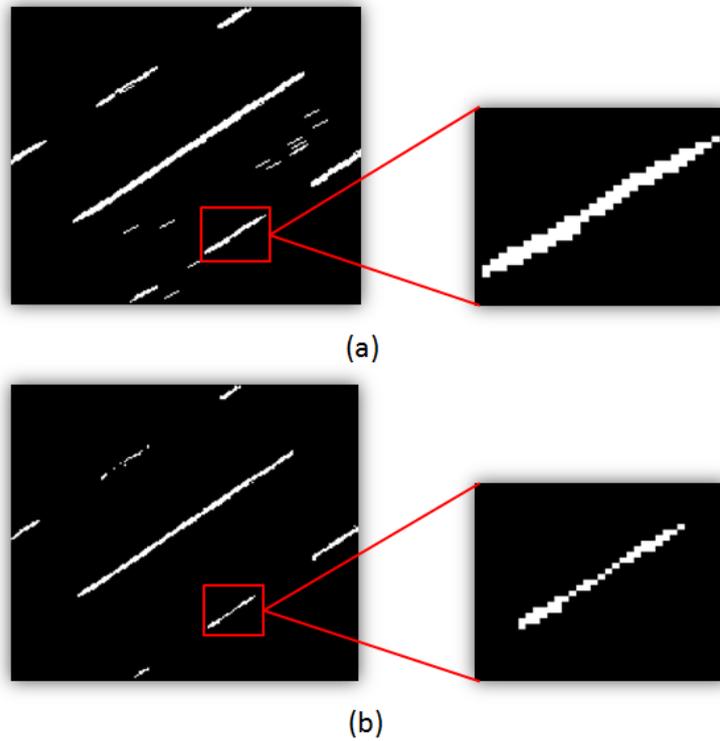


Figure 5.11: Erosion operation: (a) input dilated image with an inset picture of their road marking cells and (b) eroded image with an inset picture of their eroded road marking cells.

need the most applicable value for each of these. In the following sections, we will detail our recommended values and show the effect of changing these.

5.2.1 Optimal Cell Size

As described in Section 5.1.2, we convert the LiDAR data into 2D raster surfaces. Each cell in the raster surface has a physical dimension. A selection of the optimal cell size is essential as it may affect both the accuracy and computational cost of any process. Our road marking extraction algorithm is primarily based on thresholding which is not computationally expensive. Therefore, we considered accuracy as the main criteria for cell size selection.

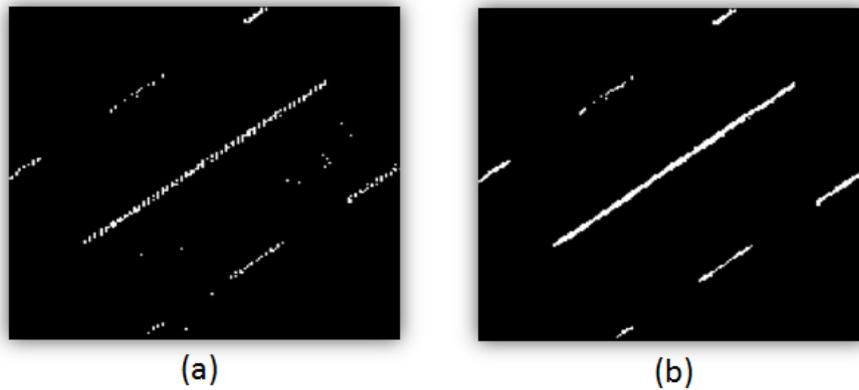


Figure 5.12: Extracted road markings (a) before and (b) after applying the morphological operations.

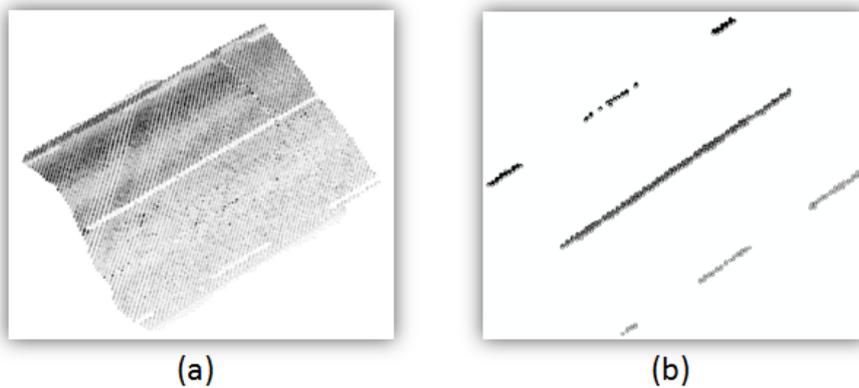


Figure 5.13: LiDAR points (a) belonging to the road surface and the extracted (b) 3D road markings.

To find its optimal value, we analysed the performance of our road marking extraction algorithm in raster surfaces generated with different cell sizes. We selected one 10 m section of rural road which contained broken and continuous line markings over its surface. To process this road section, we used one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data, which was collected with the XP-1 system.

We used the output snake curve as obtained through the automated road edge extraction algorithm applied to the data section, shown in Figure

4.38(b). We considered five test cases in which raster surfaces were generated from the LiDAR intensity and range attributes with cell sizes 0.01 m², 0.04 m², 0.06 m², 0.08 m² and 0.1 m². These cell sizes were selected with decreasing and increasing values based on an average point spacing of 0.08 m in the LiDAR points. We normalised the intensity and range values with respect to their global minimum and maximum values, shown in Table 5.1 and converted into 8-bit data type. These minimum and maximum values were

LiDAR attribute	Minimum value	Maximum value
Intensity	1085	5528
Range	1.73	21.92

Table 5.1: Maximum and minimum values of the LiDAR intensity and range attributes.

selected globally from all the data sections processed for the road marking extraction experiments.

In each test case, we applied our road marking extraction algorithm using a threshold value of 70. This value was modified as a function of range and applied to four blocks of the intensity raster surface. A detailed analysis on the selection of range dependent threshold values is provided in Section 5.2.2. For the five cases, the length of the structuring element was 51, 13, 9, 7 and 5, while $\phi' = 24.23^\circ$ was calculated from $\theta = 65.77^\circ$, which is an average heading of the mobile van along the selected road section. We found 0.5 m and 0.1 m as a minimum length and width for road markings in the Ireland. These values are detailed in a traffic signs manual published by the National Roads Authority (NRA), Ireland [NRA10]. The road marking cells were dilated along the longitudinal direction, with a minimum length threshold for the five cases being 1 m, 1 m, 1.02 m, 1.04 m and 0.9 m. The minimum width threshold was 0.1 m as the cells were not dilated along the transverse

direction. The road markings extracted in the five cases are shown in Figure 5.14.

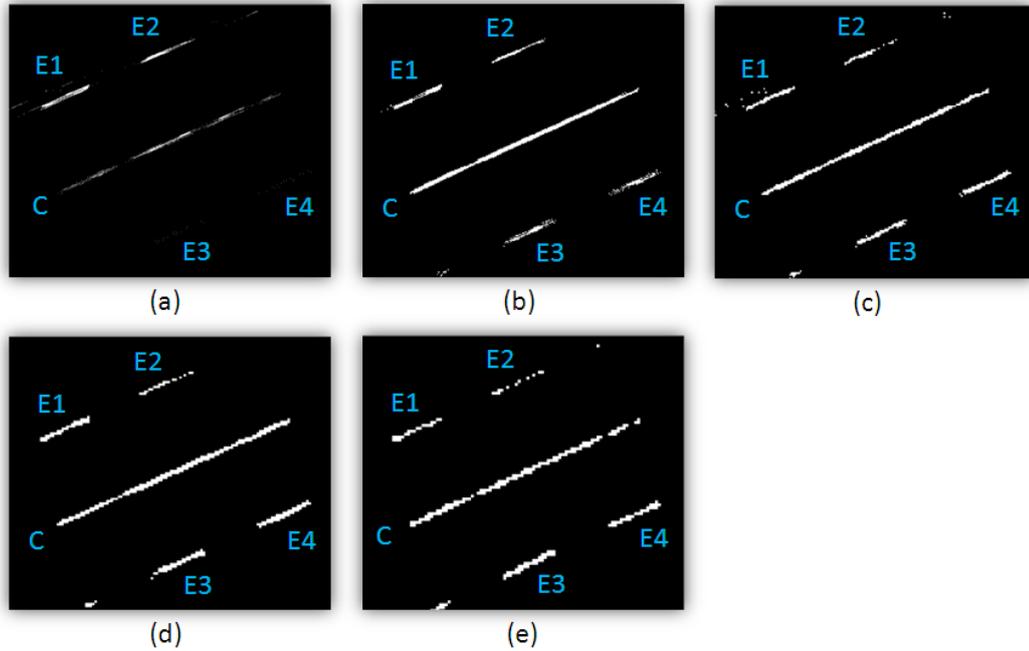


Figure 5.14: Road markings extracted from the intensity image with (a) 0.01 m^2 , (b) 0.04 m^2 , (c) 0.06 m^2 , (d) 0.08 m^2 and (e) 0.1 m^2 cell size.

In order to carry out a comparative analysis, we calculated the length and average width of the final extracted road markings in the five cases. We considered four broken line markings named as $E1$, $E2$, $E3$ and $E4$ and one continuous line marking named as C , shown in Figure 5.14. The calculated length and average width values are listed in Table 5.2. We also found a standard length and width of the five road markings from the NRA's traffic signs manual [NRA10]. The standard length and width values of the $E1$, $E2$, $E3$ and $E4$ road markings were 2 m and 0.15 m. The standard width of the C road marking was 0.15 m and its length was found from the longitudinal length of the snake curve as 8.6 m. We compared the extracted length and

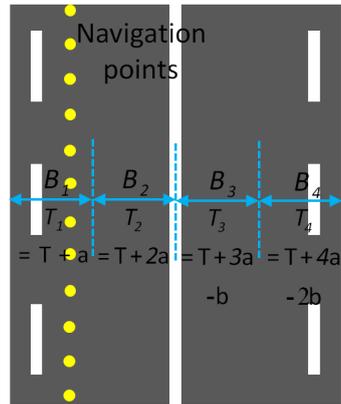
Road Markings	0.01 m ²		0.04 m ²		0.06 m ²		0.08 m ²		0.1 m ²	
	Length	Width	Length	Width	Length	Width	Length	Width	Length	Width
E1 (2 m × 0.15 m)	1.75 m	0.12 m	1.68 m	0.12 m	1.68 m	0.12 m	1.68 m	0.16 m	1.70 m	0.13 m
E2 (2 m × 0.15 m)	1.84 m	0.07 m	1.84 m	0.07 m	1.80 m	0.08 m	1.84 m	0.08 m	1.80 m	0.07 m
E3 (2 m × 0.15 m)	1.71 m	0.002 m	1.80 m	0.10 m	1.80 m	0.13 m	1.84 m	0.17 m	1.80 m	0.20 m
E4 (2 m × 0.15 m)	1.78 m	0.002 m	1.80 m	0.08 m	1.80 m	0.14 m	1.84 m	0.18 m	1.80 m	0.16 m
C (8.6 m × 0.15 m)	7.80 m	0.034 m	7.84 m	0.145 m	7.86 m	0.141 m	7.92 m	0.17 m	7.90 m	0.16 m

Table 5.2: Length and average width values of the extracted road markings in the five cases.

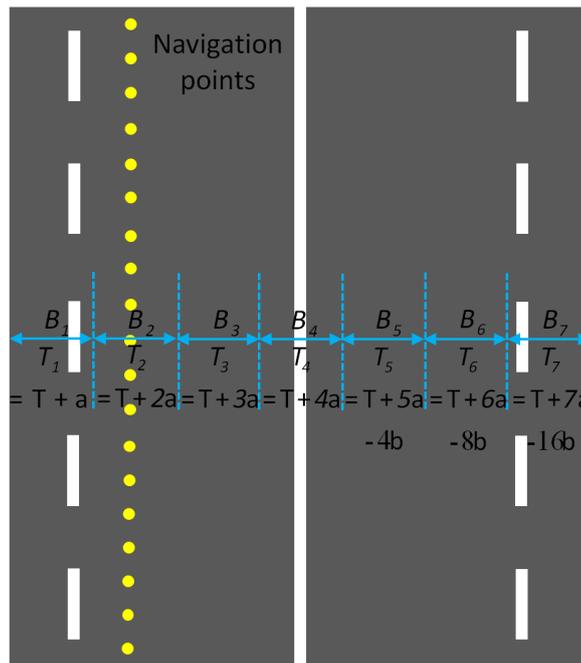
average width values of the road markings with the expected standard dimensions. For cell sizes 0.08 m^2 and 0.1 m^2 , the average width values of the road markings were generally found to be more than their standard width values. For the 0.06 m^2 cell size, the length and average width values of the road markings were closest to their standard values. Thus, we selected 0.06 m^2 cell size as the most applicable value to generate 2D intensity and range raster surfaces from the LiDAR data.

5.2.2 Range Dependent Threshold

After selecting the optimal cell size, the final parameter requiring selection is the range dependent threshold. The aim of this analysis was to determine a method for automatically selecting this threshold value irrespective of the road dimension. As described in Section 5.1.3, we use the navigation data to select the range value that is used to divide the intensity raster surface into different blocks. In a road section with a narrower width, the intensity raster surface was divided into four blocks, shown in Figure 5.15(a). In a road section with a greater width, the intensity raster surface was divided into seven blocks, shown in Figure 5.15(b). We selected a single threshold value $T = 70$ empirically and modified it as a function of the range. We applied a $T + ma$ threshold to each block of the intensity raster surface, where $a = 10$ and m represents a block number. We applied the $T + ma - nb$ threshold to the blocks after the centre of the road, where $b = 5$ and $n = 1, 2, 4, 8, \dots$ representing an integer for the third, fourth, fifth, sixth, etc. blocks respectively. The threshold values applied before the centre of the road were consecutively increased with the ma term, as the road surface in those blocks begins to orient towards the laser scanner. The threshold values



(a)



(b)

Figure 5.15: Range dependent thresholding applied to the intensity raster surface in the road section with (a) narrower and (b) greater width.

applied after the centre of the road were increased with the term ma and then decreased with the nb term to account for the increased range and the change in surface orientation. The purpose of finding these variables was to automate the process of applying the range dependent thresholding to the intensity raster surface.

To demonstrate the importance of the selection of this threshold, we applied a lower, higher and optimal value of T to the intensity raster surface as 55, 90 and 70. Each value was modified as a function of the range in accordance with the aforementioned formulae and used to extract the road markings, shown in Figure 5.16(a), (b) and (c). The use of a lower threshold

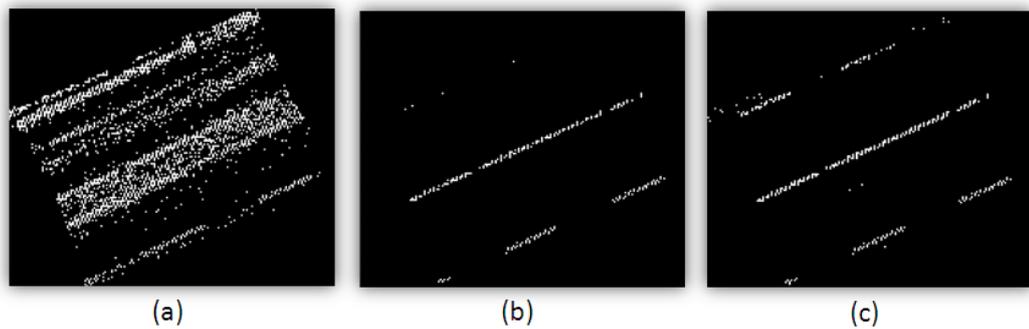


Figure 5.16: Road markings extracted using the (a) lower, (b) higher and (c) optimal threshold value applied to the intensity raster surface.

value led to the extraction of road markings with large areas of noise, while its higher value removed the noise at the expense of extracting all the road markings. Thus, the selection of optimal threshold value is essential for the robust extraction of road markings. In the next section, we test our road marking extraction algorithm on various road sections.

5.3 Experimentation

We selected seven sections of road to test our road marking extraction algorithm. These seven sections covered 150 m of rural, urban and national primary roads, which contained six distinct types of road markings to demonstrate the effectiveness of our algorithm. The processed data was collected with the XP-1 along these road sections. In the following sections, we will present the results for these road marking types and discuss our validation.

5.3.1 Broken and Continuous Line Markings

The most common road marking types are broken and continuous line markings in the road centre and lane boundaries, which are used to guide and confine traffic to its lane. We selected two 50 m sections of rural and urban road containing both broken and continuous line markings, shown in Figure 5.17. To process each road section, we used six $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ sections



Figure 5.17: Digital image of (a) rural and (b) urban road section containing both broken and continuous line markings (Geographic locations: (a) $53^{\circ}34'28.07''\text{N } 7^{\circ}10'13.76''\text{W}$ and (b) $53^{\circ}36'33.43''\text{N } 7^{\circ}5'46.96''\text{W}$).

of LiDAR data and one 10 m section of navigation data in the same manner

as in Chapter 4, with a 2 m overlap.

We used the automatically extracted snake curves on these data sections, shown in Figure 4.38. We applied our road marking extraction algorithm to the rural and urban road sections using a cell size $c = 0.06 \text{ m}^2$ and a threshold value $T = 70$, with each road section being split horizontally into four blocks. The length of the linear structuring element was 9, while ϕ' was calculated from θ , as shown in Tables 5.3 and 5.4. The 2D broken and continuous line markings extracted from the six data sections of the rural and urban road sections are shown in Figures 5.18 and 5.19 respectively. The original LiDAR data and extracted 3D broken and continuous line markings of the rural and urban road sections are shown in Figures 5.20 and 5.21 respectively.

Navigation Section	θ	ϕ'
1	65.59°	24.41°
2	65.77°	24.23°
3	65.97°	24.03°
4	66.24°	23.76°
5	66.46°	23.54°
6	66.85°	23.15°

Table 5.3: ϕ' angle calculated from θ angle in each navigation section of rural road section.

Navigation Section	θ	ϕ'
1	51.90°	38.10°
2	51.59°	38.41°
3	51.62°	38.38°
4	51.64°	38.36°
5	51.34°	38.66°
6	50.08°	39.92°

Table 5.4: ϕ' angle calculated from θ angle in each navigation section of urban road section.

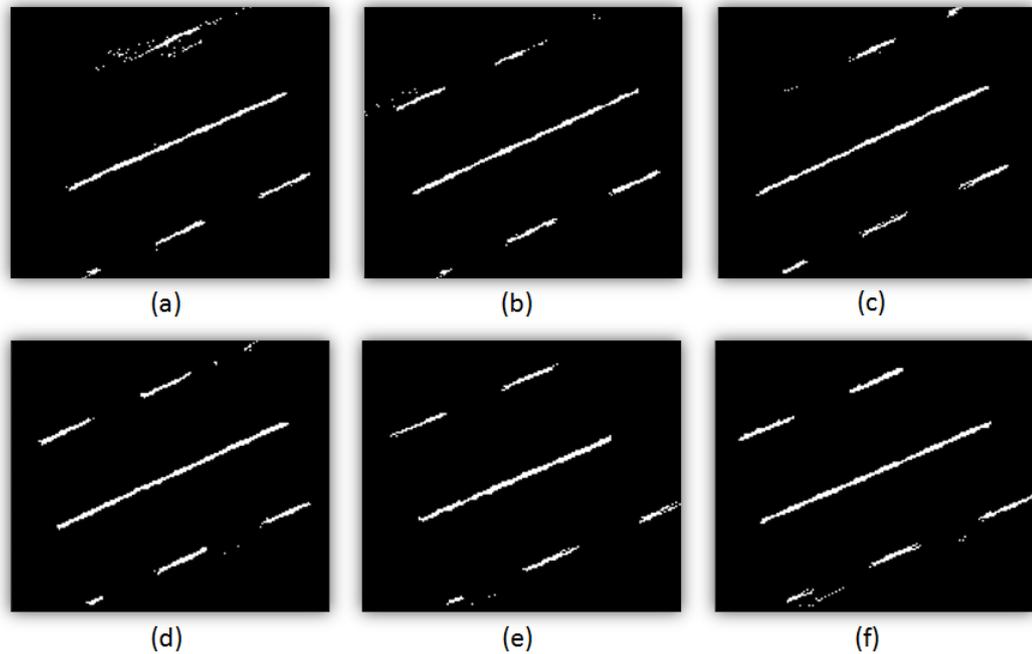


Figure 5.18: 2D broken and continuous line markings extracted from the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section of the rural road.

Our algorithm extracted the broken and continuous line markings of both the rural and urban road sections. Some broken line markings along the left side of both road sections were not detected due to lower intensity values of the laser return from them. The extracted markings contained some noise along the left edge, as shown in Figure 5.18(a). This was primarily due to extracted road edges extending incorrectly into a grass and soil area, as seen in Figure 4.38(a). This surface produced high intensity values which were not considered as noise and were not removed by our algorithm due to their large physical dimension. The broken markings that were not detected along the right side of the urban road section were attributed to a lower point density of the LiDAR data along the right side. This was due to the use of a single

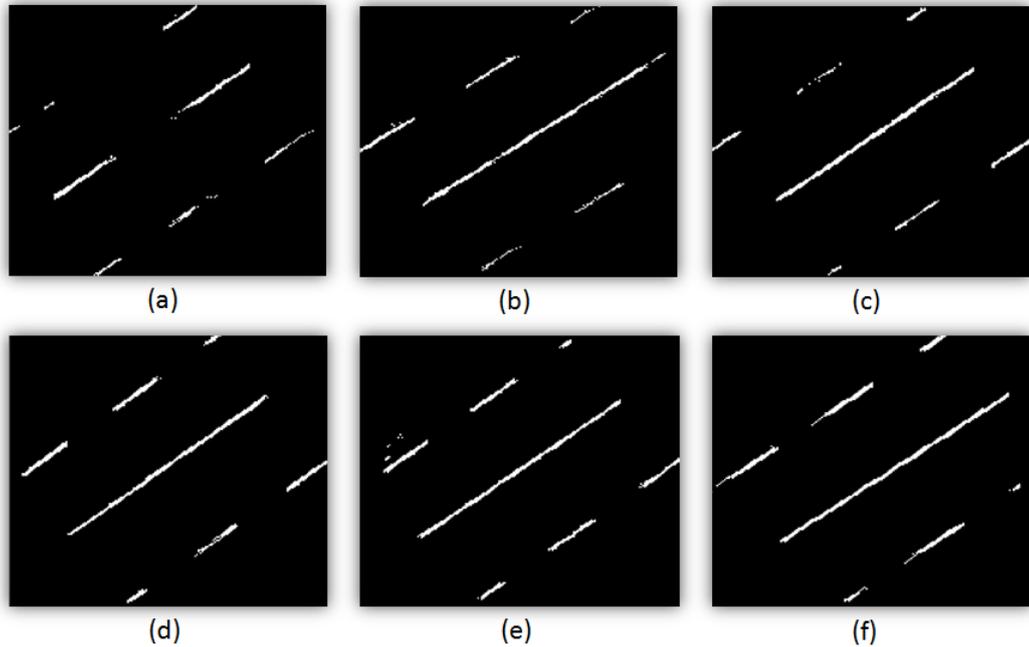


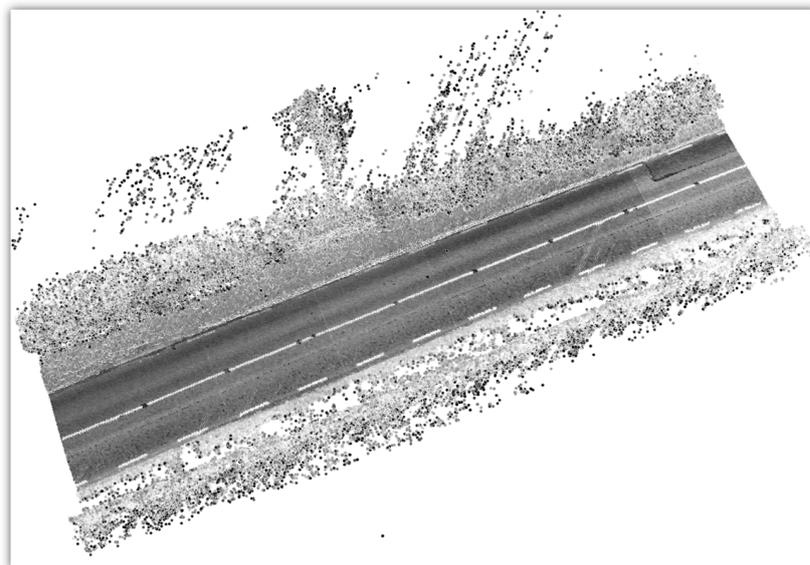
Figure 5.19: 2D broken and continuous line markings extracted from the (a) first, (b) second, (c) third, (d) fourth, (e) fifth and (f) sixth data section of the urban road.

laser scanner in the XP-1 system during the data acquisition process.

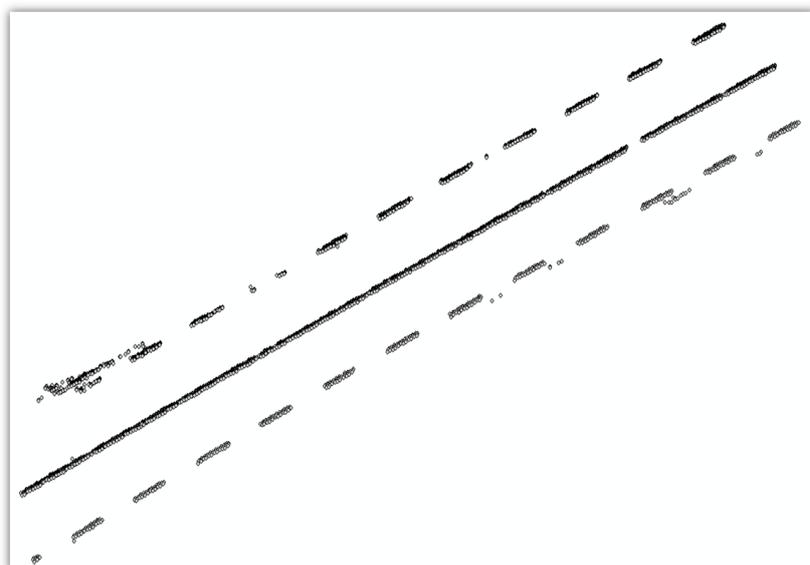
5.3.2 Word Markings

The second road marking type selected was word markings, as defined by the NRA, Ireland [NRA10]. These markings are inscribed as a word message on the road surface to guide, regulate or warn the road user [MK06]. We selected one 10 m section of rural road containing word markings along with broken and continuous line markings, shown in Figure 5.22. To process the selected road section, we used one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data.

We applied our automated road edge extraction algorithm to the road

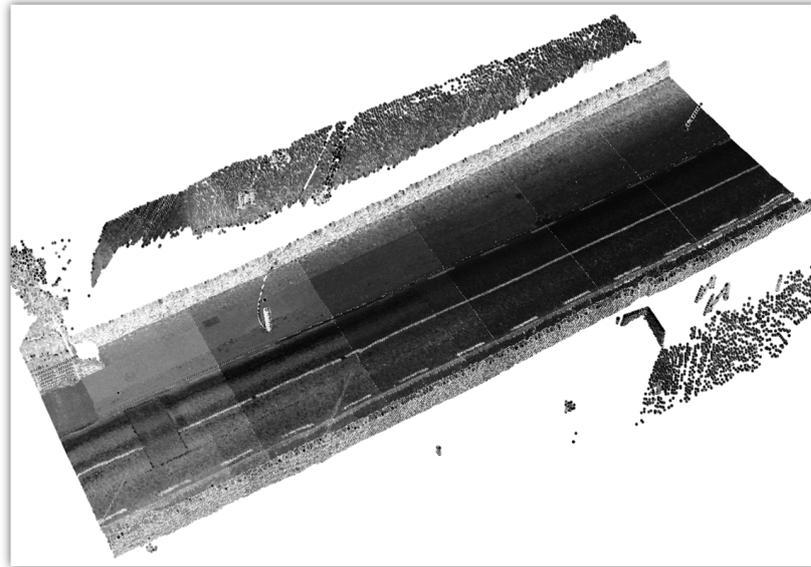


(a)

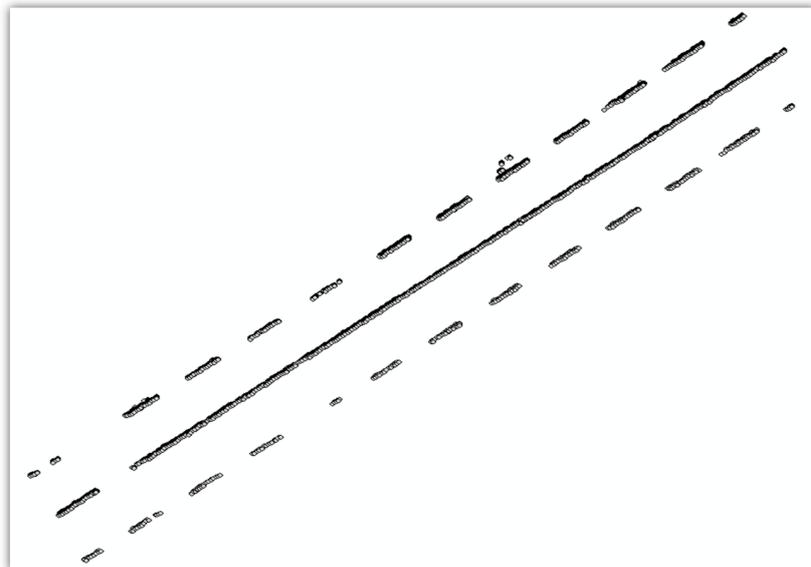


(b)

Figure 5.20: Road markings extraction: (a) original LiDAR data and (b) extracted 3D broken and continuous line markings of the rural road section.



(a)



(b)

Figure 5.21: Road markings extraction: (a) original LiDAR data and (b) extracted 3D broken and continuous line markings of the urban road section.



Figure 5.22: Digital image of the rural road section containing word markings along with broken and continuous line markings (Geographic location: $53^{\circ}34'50.546''\text{N}$ $7^{\circ}8'57.62''\text{W}$).

section with one variation to the parameters outlined in Section 4.4.2. We increased κ_4 to 3 to overcome noise due to a lower point density in the LiDAR data along the right side of the road section. The output snake curve obtained from this process is shown in Figure 5.23. We applied our road marking extraction algorithm using $c = 0.06 \text{ m}^2$ and $T = 70$, with the road section being divided into four blocks. The length of the linear structuring element was 9, while $\phi' = 343.52^{\circ}$ was calculated from $\theta = 106.48^{\circ}$. The original LiDAR data and the extracted 3D word, broken and continuous line markings of the rural road section are shown in figure 5.24.

Our algorithm successfully extracted the majority of the word, broken and continuous line markings from the rural road section. The extracted road markings contained some noise along the right edge. This was due to the extracted road edges, extending incorrectly into grass and soil area at some points, as shown in Figure 5.23. The primary reason for this was the increased balloon energy pushing the snake curve beyond the weak edge points. Some of the extracted words along the right side were incomplete.

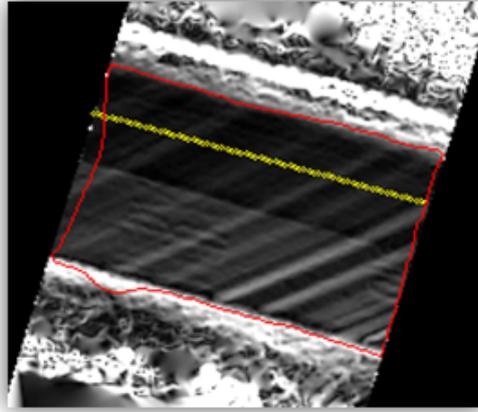
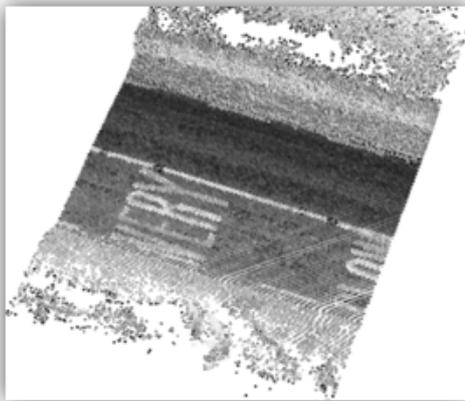
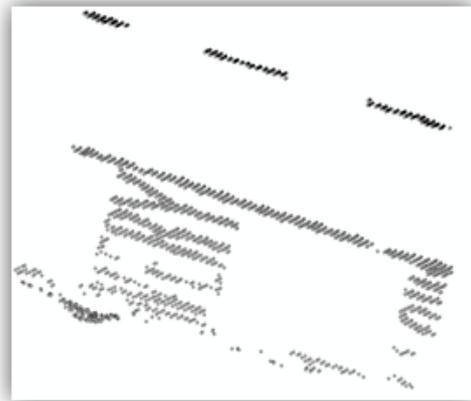


Figure 5.23: Output snake curve in the rural road section containing word markings along with broken and continuous line markings.



(a)



(b)

Figure 5.24: Road markings extraction: (a) original LiDAR data and (b) extracted 3D word, broken and continuous line markings of the rural road section.



Figure 5.25: Digital image of the national primary road section containing hatch and broken line markings (Geographic location: $53^{\circ}33'49.878''\text{N}$ $7^{\circ}21'24.148''\text{W}$).

This was due to a lower LiDAR point density along that side of the road section.

5.3.3 Hatch Markings

We selected hatch markings as the third road marking type to test our road marking extraction algorithm. The hatch markings are generally used to channel or control the traffic flow, reserve central traffic islands and indicate a reduction in road width ahead [NRA10]. We selected one 10 m section of national primary road containing hatch and broken line markings, shown in Figure 5.25. We used one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data, to process the selected road section.

We applied our automated road edge extraction algorithm to the road section with one modified parameter $\kappa_4 = 3$. This was again to overcome noise due to the lower point density of the LiDAR data along the right side of the road section. The output snake curve obtained is shown in Figure 5.26. We then applied our road marking extraction algorithm using $c = 0.06\text{ m}^2$

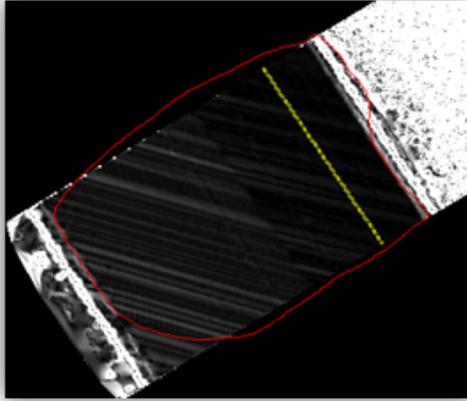


Figure 5.26: Output snake curve in the national primary road section containing hatch and broken line markings.

and $T = 70$. The road section was divided into seven blocks due to the larger width of road section. The length of the linear structuring element was 9, while $\phi' = 302.44^\circ$ was calculated from $\theta = 147.56^\circ$. The original LiDAR data and the extracted 3D hatch and broken line markings of the national primary road section are shown in Figure 5.27.

Our algorithm was able to extract the hatch and broken line markings from the national primary road section. Noise present along the left side of the road section was due to the extracted road edges extending incorrectly into a grass and soil area, as can be seen in Figure 5.26. A reason for this extension was the increased balloon energy which caused the snake curve to move beyond the weak left edge points.

5.3.4 Arrow Markings

The fourth road marking type selected was arrow markings. Arrow markings are used to give drivers advance indication of the correct lane to take when approaching busy intersections [NRA10]. We selected one 10 m section of

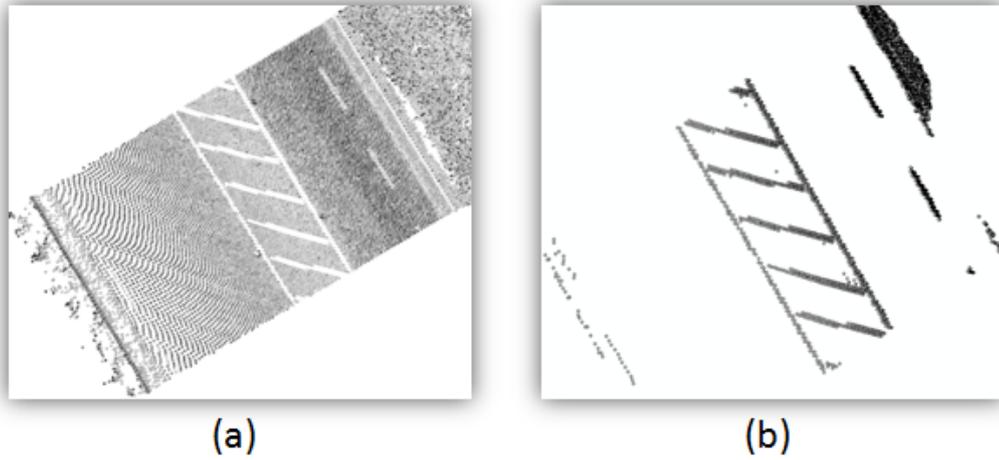


Figure 5.27: Road markings extraction: (a) original LiDAR data and (b) extracted 3D hatch and broken line markings of the national primary road section.

national primary road containing arrow and broken line markings, shown in Figure 5.28. To process this road section, we used one 30 m \times 10 m \times 5 m section of LiDAR data and one 10 m section of navigation data.

We applied our automated road edge extraction algorithm to the road section with a modified parameter, $\kappa_4 = 3$. The output snake curve obtained from this process is shown in Figure 5.29. We then applied our road marking extraction algorithm using $c = 0.06 \text{ m}^2$ and $T = 60$, with the road section being split into six blocks. The length of the linear structuring element was 9, while $\phi' = 296.35^\circ$ was calculated from $\theta = 153.65^\circ$. The original LiDAR and the extracted 3D arrow and broken line markings are shown in Figure 5.30.

The algorithm was able to extract the arrow and broken line markings from the national primary road section. The broken line markings along the right side of the road section were not extracted due to a lower point density



Figure 5.28: Digital image of the national primary road section containing arrow and broken line markings (Geographic location: $53^{\circ}35'30.558''\text{N}$ $7^{\circ}22'28.428''\text{W}$).

of the LiDAR data along that side. Noise in the form of a continuous line along the left side of the road section was due to the extracted road edges extending into the grass and soil area, as can be seen in Figure 5.29. This was again attributed to the increased balloon energy.

5.3.5 Pedestrian Crossing Markings

We selected pedestrian crossing markings as the fifth road marking type to test our algorithm. The pedestrian crossing markings are used to provide a safe road crossing facility for pedestrians [NRA10]. We selected one 10 m section of urban road containing a pedestrian crossing, broken transverse line and zig-zag markings, shown in Figure 5.31. To process this road section, we used one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data.

We applied our automated road edge extraction algorithm to the road section with a modified parameter, $\kappa_4 = 5$. This parameter was increased compared to previous values in order to overcome noise due to a lower point

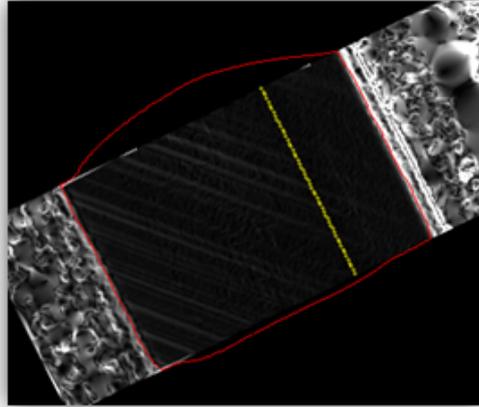


Figure 5.29: Output snake curve in the national primary road section containing arrow and broken line markings.



Figure 5.30: Road markings extraction: (a) original LiDAR data and (b) extracted 3D arrow and broken line markings of the national primary road section.



Figure 5.31: Digital image of the urban road section containing pedestrian crossing, broken transverse and zig-zag markings (Geographic location: $53^{\circ}36'43.721''\text{N}$ $7^{\circ}5'32.871''\text{W}$).

density of the LiDAR data along the right side of the road section. The output snake curve obtained from this process is shown in Figure 5.32. We applied our road marking extraction algorithm using $c = 0.06 \text{ m}^2$ and $T = 70$, with the road section being divided into five blocks. The length of the linear structuring element was 9, while $\phi' = 49.10^{\circ}$ was calculated from $\theta = 40.90^{\circ}$. The original LiDAR data and the extracted 3D pedestrian crossing, broken transverse line and zig-zag markings are shown in Figure 5.33.

The algorithm was able to extract the pedestrian crossing, broken transverse line and zig-zag markings from the urban road section. It failed to detect a portion of the markings along the right side of the road section as the snake curve was not able to fully converge to the right edge, as shown in Figure 5.32. This was due to the presence of noise along that side of the road section.

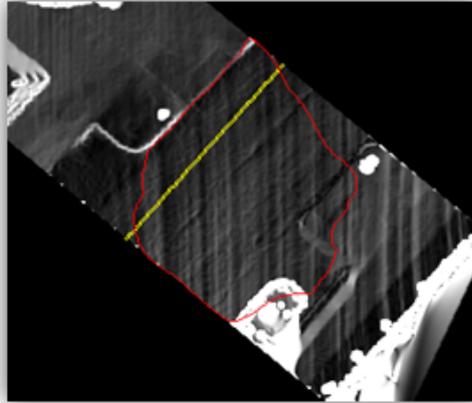


Figure 5.32: Output snake curve in the urban road section containing pedestrian crossing, broken transverse and zig-zag markings.

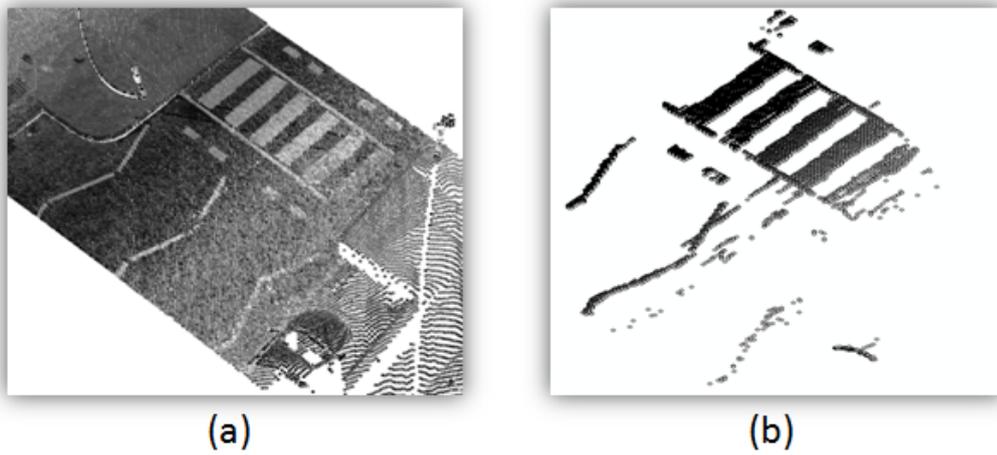


Figure 5.33: Road markings extraction: (a) original LiDAR data and (b) extracted 3D pedestrian crossing, broken transverse line and zig-zag markings of the urban road section.



Figure 5.34: Digital image of the urban road section containing zig-zag markings (Geographic location: $53^{\circ}39'19.225''\text{N}$ $7^{\circ}31'9.792''\text{W}$).

5.3.6 Zig-Zag Markings

The final road marking type selected was zig-zag markings. Zig-zag markings are generally used with the pedestrian crossing to indicate that vehicles are prohibited to park at that location as well as providing a visual warning to drivers approaching the pedestrian crossing [NRA10]. We selected one 10 m section of urban road containing zig-zag markings as shown in Figure 5.34. We used one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data, to process the selected road section.

We applied our automated road edge extraction algorithm to the road section. The output snake curve obtained from this process is shown in Figure 5.35. We then applied our road marking extraction algorithm using $c = 0.06\text{ m}^2$ and $T = 70$, with the road section being divided into four blocks. The linear structuring element was used with length 9, while $\phi' = 312.93^{\circ}$ was calculated from $\theta = 137.07^{\circ}$. The original LiDAR data and the extracted 3D zig-zag markings are shown in Figure 5.36.

Our algorithm was able to extract two zig-zag markings but failed to

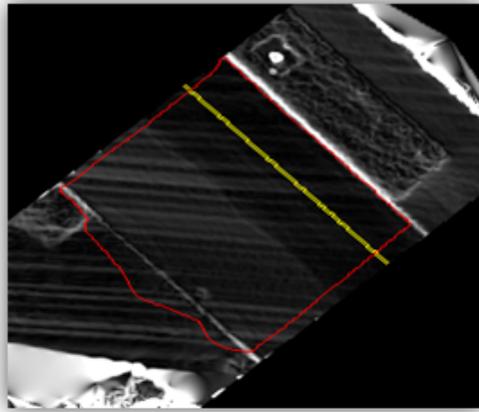


Figure 5.35: Output snake curve in the urban road section containing zig-zag markings.

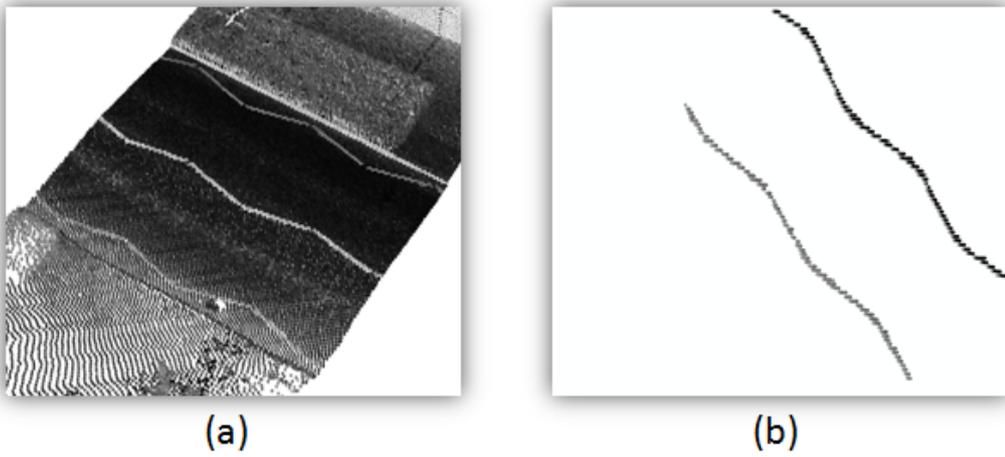


Figure 5.36: Road markings extraction: (a) original LiDAR data and (b) extracted 3D zig-zag markings of the urban road section.

detect the third one along the right side of the road section. This was due to the lower point density of the LiDAR data along the right side of the road section.

5.3.7 Results Validation

A quantitative assessment was used to validate the extracted road markings. We counted a total of 93 road markings from 3D LiDAR data of the tested road sections. Our road marking extraction algorithm correctly extracted 80 road markings but failed to detect 13 out of the total number. Thus, we were able to detect 86% of the road markings. We identified 13 groups of noisy raster cells or LiDAR points which were incorrectly labelled as extracted road markings on the basis of their length and width.

5.4 Discussion

We presented our automated algorithm for extracting the road markings from terrestrial mobile LiDAR data. Our algorithm was based on the assumption that the intensity values of the laser returns from the road markings are higher than those from other road surface elements. We expected to extract these road markings by applying a threshold to the intensity values. We considered four challenges while developing this algorithm. The first challenge was knowledge of the road surface. This was required to reduce false positives and the processing effort in the road marking extraction process. We applied our automated road edge extraction algorithm to determine the road boundaries in the form of a snake curve. The output snake curve was then used to identify the LiDAR points that belong to the road surface.

The second challenge was to apply a threshold to the intensity values. The LiDAR intensity attribute is dependent on the distance from the laser scanner to the illuminated surface, the incidence angle of the laser pulse and the illuminated surface characteristics. To successfully extract the road markings from the intensity attribute, we developed a range dependent thresholding function which attempts to deal with all three of the factors that affect the intensity value. The use of a threshold itself clusters data based on the reflectivity from the illuminated surface, while the range dependent threshold takes account of the distance and incidence angle factors. We used the navigation data to select a range value that is used to apply the threshold to the intensity values. We applied the range dependent threshold by converting the LiDAR intensity and range attributes into 2D raster surfaces.

Our third challenge was to deal with the incomplete shapes of the road markings and noise that is primarily introduced through the use of thresholding. We proposed the use of morphological operations to overcome these issues. To complete the road markings shapes, we first converted the raster surface into a binary image and then applied the dilation operation. The binary cells were dilated using a linear shaped structuring element. The length of the linear structuring element was selected empirically, while its angle was calculated from the average heading of the mobile van. The noise was removed based on a priori knowledge of the dimensions of the dilated road markings. We then applied an erosion operation to counteract the artificial growth in the dimensions of the road markings. The binary dilated cells were eroded using the same linear shaped structuring element. After applying the erosion operation, we can again remove any additional noise based on the dimensions of the eroded road markings. But it could have a negative impact

on the extracted road markings. An example of noise removed only once before applying the erosion operation, and removed twice, before and after applying the erosion operation, is shown in Figure 5.37.

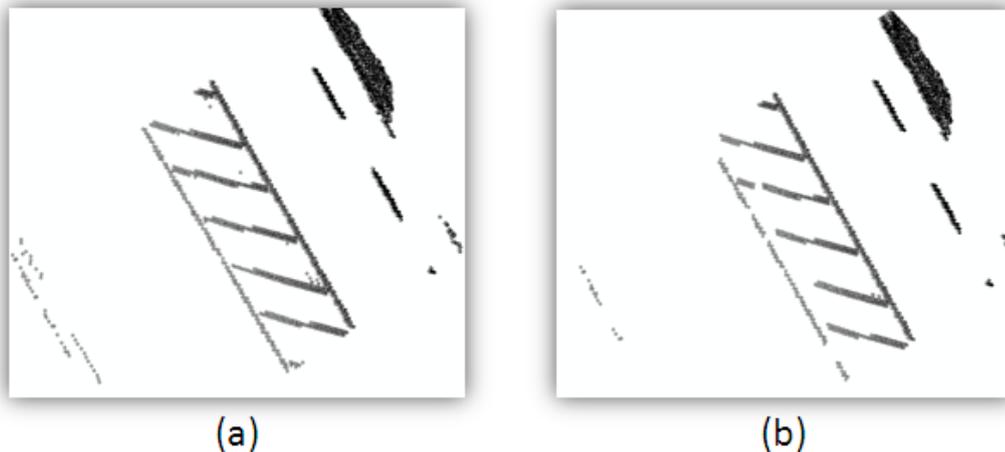


Figure 5.37: 2D hatch and broken line markings of the national primary road section with the noise (a) removed only once before applying the erosion operation and (b) removed twice, before and after applying the erosion operation.

The final challenge was to automate our road marking extraction algorithm through the recommendation of the most applicable values of input parameter. This involves selecting an optimal cell size values for converting the LiDAR data into 2D raster surfaces as well as threshold values for applying to the intensity as a function of the range. We demonstrated that choosing a cell size that is too small will reduce the completeness and that choosing a large cell size will output the road markings whose dimensions are greater than the actual road markings. We recommended 0.06 m^2 cell size as the most applicable value which consistently produced the best results. We showed the importance of choosing an optimal threshold value for extracting the road markings. If the threshold value is small, it will output

road markings accompanied with large areas of noise. If the threshold value is large, it will remove the noise at the expense of extracting all the road markings. We selected 70 as an optimal threshold value and then created a formula for applying the threshold values as a function of the range along the road surface.

We tested our road marking extracting algorithm on seven different road sections. These seven sections covered 150 m of rural, urban and national primary roads which, in turn, contained six distinct types of the road marking. Out of the total of 93 road markings, our algorithm was able to extract 80 road markings, resulting in an 86% detection rate. The majority of undetected road markings were along the right side of the road section, this was due to a lower LiDAR point density along that side, while the road markings missed along the left side were generally due to lower intensity values in the associated laser returns. Our algorithm identified 13 groups of noise cells which were incorrectly labelled as road markings. The majority of these false road markings were detected in rural and national primary road sections. These false detections were due to extracted road edges extending incorrectly to the nearby grass and soil area which was primarily attributed to the increased balloon energy pushing the snake curve beyond the weak edge points. In the next chapter, we present our algorithm for estimating the road roughness from terrestrial mobile LiDAR data.

Chapter 6

Road Roughness Estimation

Terrestrial mobile LiDAR data provides 3D information that can be used to estimate roughness along a road surface. Road roughness is generally considered to be the deviation of the road surface from a designed surface grade that can, in turn, influence safety conditions for road users. LiDAR data can be used to provide a good estimation of roughness values along the road surface without the requirement for any simulation processes. Road roughness can be used to provide valuable information to road authorities when planning routine route network maintenance schedules.

In this chapter, we present an automated algorithm to estimate road roughness from terrestrial mobile LiDAR data. In the algorithm, we apply our automated road edge extraction algorithm to determine the boundaries of the road in the form of a snake curve. The output snake curve is used to identify the LiDAR points that belong to the road surface. A priori knowledge of the road surface allows us to more accurately estimate the roughness along the road surface. Our algorithm provides discrete estimation of road roughness in the form of standard deviation values of elevation residual points.

These residual points are obtained after fitting a surface grid to the LiDAR points belonging to the road surface. In Section 6.1, we present a stepwise description of our road roughness estimation algorithm. This involves the identification of LiDAR points that belong to the road surface and fitting a surface grid to them. We describe the standard deviation calculation of the elevation residual points along the navigation track of the mobile van. In Section 6.2, we test our road roughness estimation algorithm in various road sections. In Section 6.3, we conclude by discussing the advantages and limitations of our algorithm.

6.1 Algorithm

A work flow of the road roughness estimation algorithm is shown in Figure 6.1. In the following sections, we describe a number of processing steps involved in our algorithm.

6.1.1 Road Surface Estimation

We apply our automated road edge extraction algorithm to determine the road boundaries in the form of a snake curve, as described in Chapter 4. In Step 1 of our road roughness estimation algorithm, we use the snake curve to identify the LiDAR points that belong to the road surface. A snake curve is laid over the LiDAR data, as shown in Figure 6.2(a). The LiDAR points outside the snake curve are removed and those inside are retained as they belong to the road surface, as shown in Figure 6.2(b).

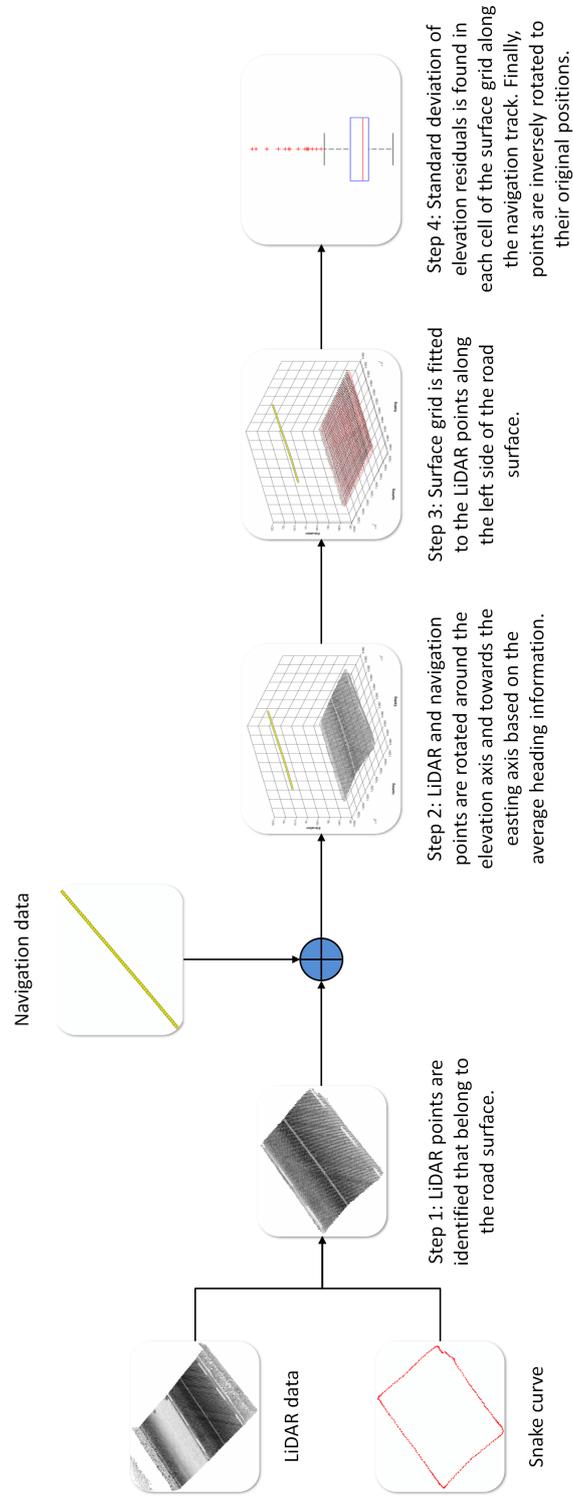


Figure 6.1: Road roughness estimation algorithm.

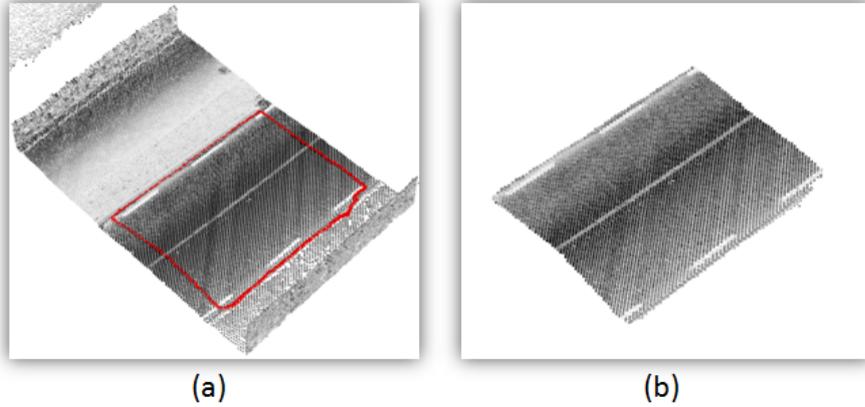


Figure 6.2: Snake curve is (a) laid over the LiDAR data and (b) the points belonging to the road surface are identified.

6.1.2 Data Rotation

In Step 2 of our road roughness estimation algorithm, we rotate the LiDAR and navigation points around the elevation axis and towards the Easting axis. This rotation is carried out to assist the process of fitting a surface grid to the LiDAR points described in the next section. Both the LiDAR and navigation points are rotated through a ϕ angle, which is calculated from the average heading angle, θ , of the mobile van. The process of calculating ϕ from the θ is similar to that described in Figure 4.6.

The rotation of LiDAR and navigation points around the elevation axis and towards the Easting axis is shown in Figure 6.3.

6.1.3 Surface Grid

In Step 3 of our algorithm, we use the RANSAC algorithm to fit a surface grid to the LiDAR points that belong to the left side of the road surface. The fitted surface grid is a representative of what the ideal road surface should be. This surface is used to measure the elevation residuals along the

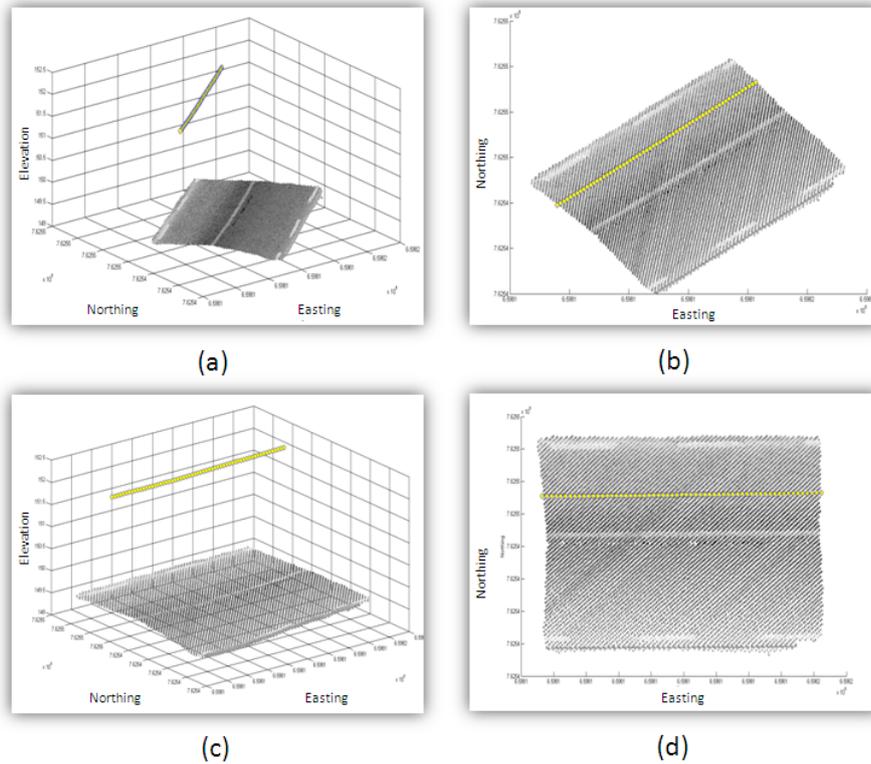


Figure 6.3: Input LiDAR and navigation data in the (a) 3D, (b) 2D plane and rotated LiDAR and navigation data in the (c) 3D, (d) 2D plane.

navigation track described in the next section. We do not fit a surface grid to the LiDAR points that belong to the complete road surface as it does not provide an accurate representation of the road surface due to its non-planar shape, as can be seen in Figure 6.4.

The LiDAR points are divided into the left and right sides of the road surface based on their rotated Northing values. We assume the centre of the road is in the middle of the extracted road surface. Using the road boundaries from our road edge extraction algorithm, we can easily split the road into two parts. A surface grid is fitted to the LiDAR points belonging to the left side of the road surface. The RANSAC approach uses the smallest set of initial points and enlarges this set with compatible points based on the number

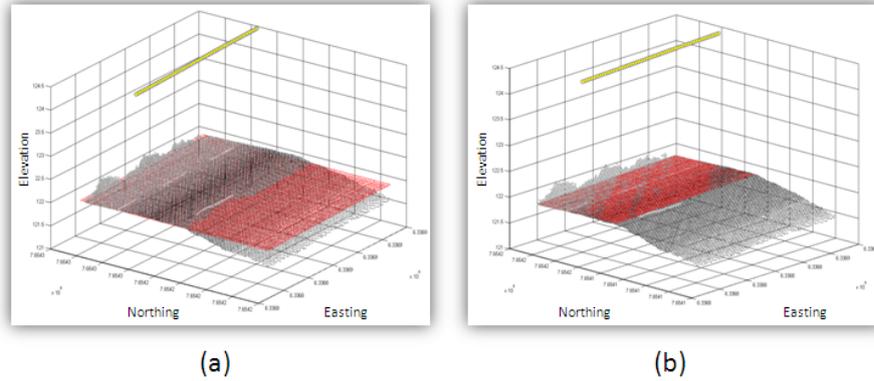


Figure 6.4: Surface grid is fitted to the LiDAR points that belong to (a) whole road surface and (b) the left side of the road surface.

of iterations, K , the size of the consensus set, t and the error tolerance, ϵ . The ϵ value is estimated experimentally in the algorithm, while the K and t values are provided based on the formulae described in Section 3.3.1. In each iteration, we randomly select three minimum points that are required to estimate an initial plane using a least square model and then enlarge them with compatible points to provide a surface grid fitting to the LiDAR points. The size of the cell in the surface grid is selected based on the surface area of the foot-print of the mobile van’s wheel that comes into a contact with the road surface. This cell size is chosen to measure the roughness experienced at each discrete instant of the vehicle’s movement over the road surface. The RANSAC surface grid fitted to the LiDAR points is shown in the 3D and 2D planes in Figure 6.5.

6.1.4 Road Roughness

In Step 4 of our algorithm, we find the elevation residual values by calculating the difference between the elevation of each LiDAR point and its respective surface grid point. We determine a standard deviation of the elevation resid-

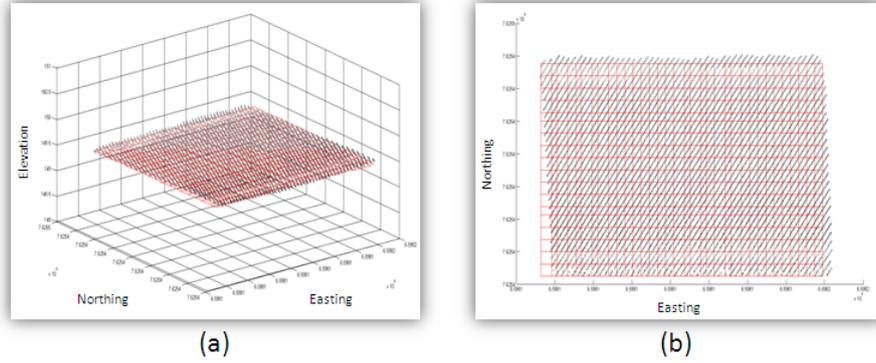


Figure 6.5: RANSAC surface grid fitted to the LiDAR points in the (c) 3D and (d) 2D plane.

uals in each cell along the navigation track of the mobile van as

$$s = \sqrt{\frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2}, \quad (6.1)$$

where,

$$\bar{z} = \frac{1}{n} \sum_{i=1}^n z_i, \quad (6.2)$$

z is the residual and n is the number of residuals in each cell. The estimated standard deviation values provide the roughness information along the longitudinal road surface. Finally, the LiDAR points are inversely rotated to their original position along the road surface. The inversely rotated surface grid and LiDAR points along the navigation track for which the standard deviation is estimated are shown in the 3D and 2D planes in Figure 6.6.

6.2 Experimentation

We selected four sections of road to test our road roughness estimation algorithm. These four sections covered 120 m of urban and national primary

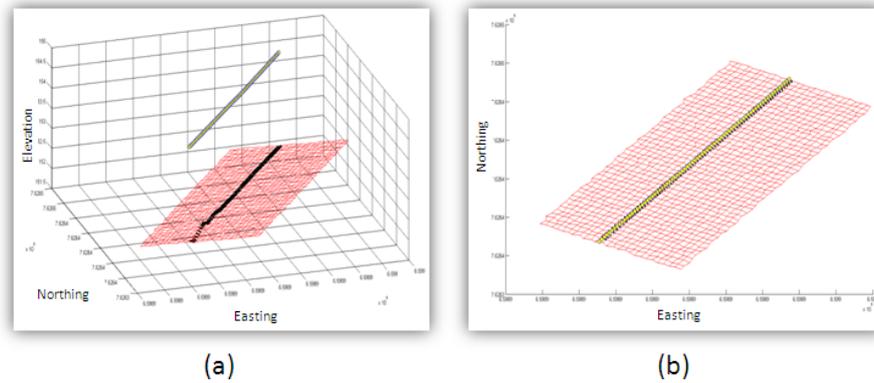


Figure 6.6: Inversely rotated surface grid and LiDAR points along the navigation track of the mobile van in the (c) 3D and (d) 2D plane.

roads. We selected these road sections to demonstrate the effectiveness of our algorithm to estimate the roughness present along their surfaces. The processed data was collected with the XP-1 system along these road sections. In the following sections, we present the roughness estimation results.

6.2.1 Urban Roads

We selected one 50 m and two 10 m sections of urban road as shown in Figure 6.7. To process the 50 m road section, we used six $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ sections of LiDAR data and six 10 m sections of navigation data in the same manner as that described in Chapter 4, with 2 m overlaps. We used the output snake curves as obtained through the automated road edge extraction algorithm applied to the data sections, as shown in Figure 4.40. Each 10 m road section was processed using one $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ section of LiDAR data and one 10 m section of navigation data. We applied our automated road edge extraction algorithm to the second and third road sections in the same manner as that described in Section 4.4.2 and used the output snake curves obtained from this process, as shown in Figure 6.8.

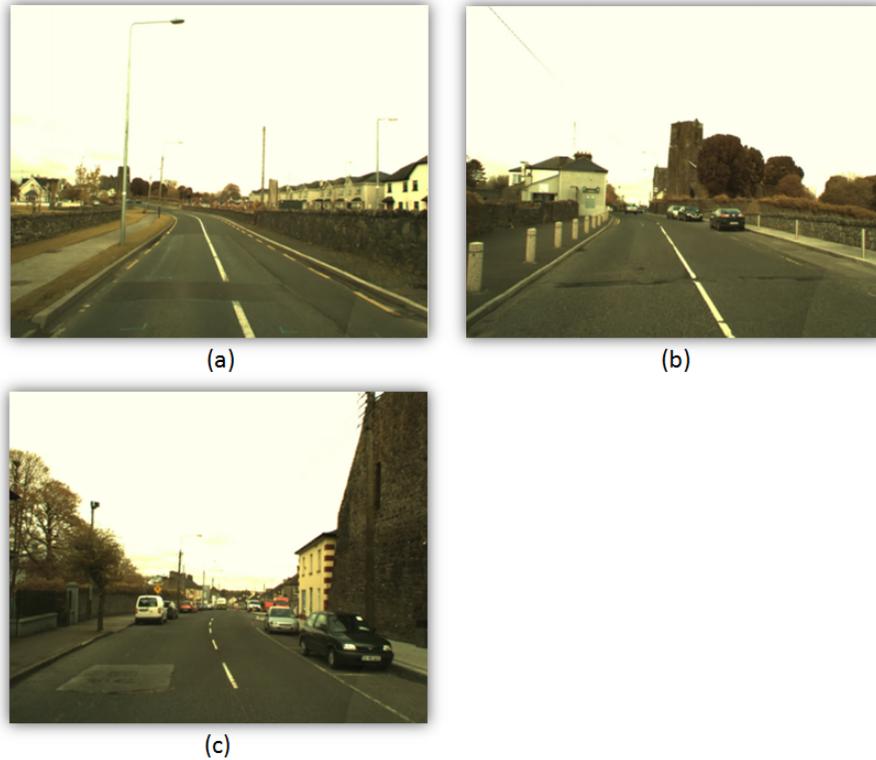


Figure 6.7: Digital image of (a) first, (b) second and (c) third section of urban road (Geographic locations: (a) $53^{\circ}36'33.68''\text{N } 7^{\circ}5'46.39''\text{W}$, (b) $53^{\circ}36'36.587''\text{N } 7^{\circ}5'41.671''\text{W}$ and (c) $53^{\circ}36'40.584''\text{N } 7^{\circ}5'37.789''\text{W}$).

In the first road section, we used LiDAR sections with an overlap of 2 m between them. In order to avoid the dual estimation of roughness in the overlapped portions, we selected the first and second consecutive LiDAR sections and removed the duplicate points in the first section. This step ensured that sections did not overlapped as shown in Figure 6.9. The second and third LiDAR sections were then selected and the duplicate points in the second section were removed. This process was repeated to remove duplicate points in all the LiDAR sections.

We applied our road roughness estimation algorithm to the first, second and third road sections. In the first road section, we used the ϕ angle shown

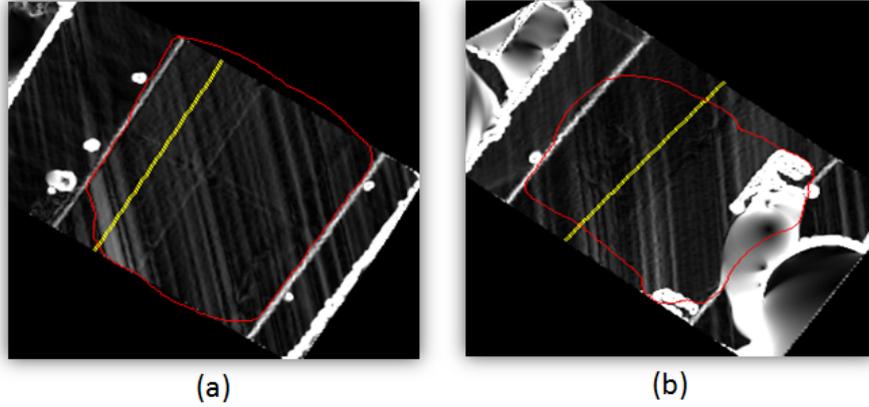


Figure 6.8: Output snake curve in the (a) second and (b) third section of urban road.

in Table 4.12. In the second and third road sections, the ϕ angle was calculated from the θ as shown in Table 6.1. The ϵ parameter was estimated

Road Section	Navigation Section	θ (degree)	ϕ (degree)
Second	1	31.85°	58.15°
Third	1	39.06°	50.94°

Table 6.1: ϕ angle calculated from θ , average heading angle in the navigation sections of second and third urban road section.

experimentally in the algorithm. We selected $p = 0.50$ and $q = 0.99$, considering 50% probability for selecting any data point within the error tolerance ϵ and 99% probability for producing a good set of points. Using p , q , $m = 3$ and Equation 3.22, the K parameter was calculated as 35. The t parameter was computed using the number of points in the respective LiDAR section, p and Equation 3.23. The values of the t and ϵ parameters used for fitting surface grids to the LiDAR sections are shown in Table 6.2. The length and width of each cell was 0.215 m and 0.18 m respectively. These values were estimated based on the measured surface area of the foot-print of the mobile van's wheel that comes into contact with the road surface. The surface grids

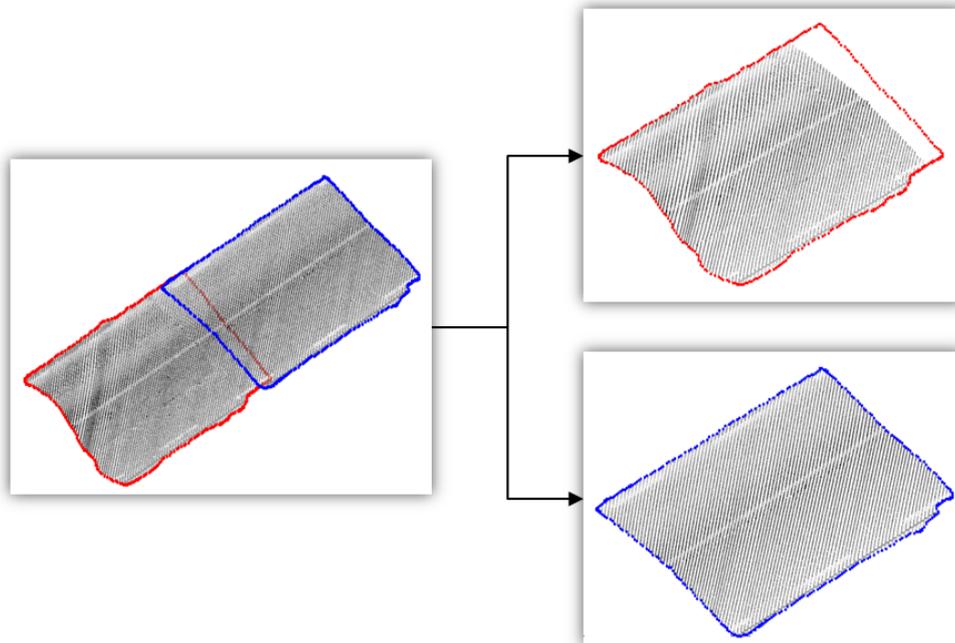


Figure 6.9: Duplicate points were removed in the first LiDAR section to provide non-overlapped sections.

and LiDAR points along the navigation track in the first, second and third road sections are shown in Figure 6.10.

6.2.2 National Primary Roads

We selected one 50 m section of the national primary road, shown in Figure 6.11. We used six $30\text{ m} \times 10\text{ m} \times 5\text{ m}$ sections of the LiDAR data and six 10 m sections of the navigation data to process the 50 m road section in the same manner as that described in Chapter 4, with a 2 m overlap. The output snake curves obtained through the automated road edge extraction algorithm applied to the road section are shown in Figure 4.42.

We removed the duplicate points in the LiDAR section in the same manner as described in the previous section. We then applied our road roughness

Road Section	LiDAR Section	t	ϵ (m)
First	1	6986	0.078
	2	10174	0.059
	3	7767	0.029
	4	7860	0.065
	5	9392	0.032
	6	11389	0.044
Second	1	16495	0.025
Third	1	23868	0.019

Table 6.2: t and ϵ parameters used for fitting the surface grid to each LiDAR section of the urban road.

LiDAR Section	t	ϵ (m)
1	19642	0.020
2	20665	0.036
3	20662	0.033
4	20399	0.013
5	20969	0.047
6	24327	0.029

Table 6.3: t and ϵ parameters used for fitting the surface grid to each LiDAR section of the national primary road.

estimation algorithm to the road section using the ϕ rotation angle, shown in Table 4.13. We used $K = 35$ while the values of t and ϵ parameters are shown in Table 6.3. The length and width of each cell was again 0.215 m and 0.18 m respectively. The surface grids and LiDAR points along the navigation track in the road section are shown in Figure 6.12.

6.2.3 Results

We generated road surface deviation maps for the first, second, third urban and the national primary road sections, as shown in Figures 6.13, 6.14, 6.15 and 6.16. These deviations were calculated as the elevation residuals between

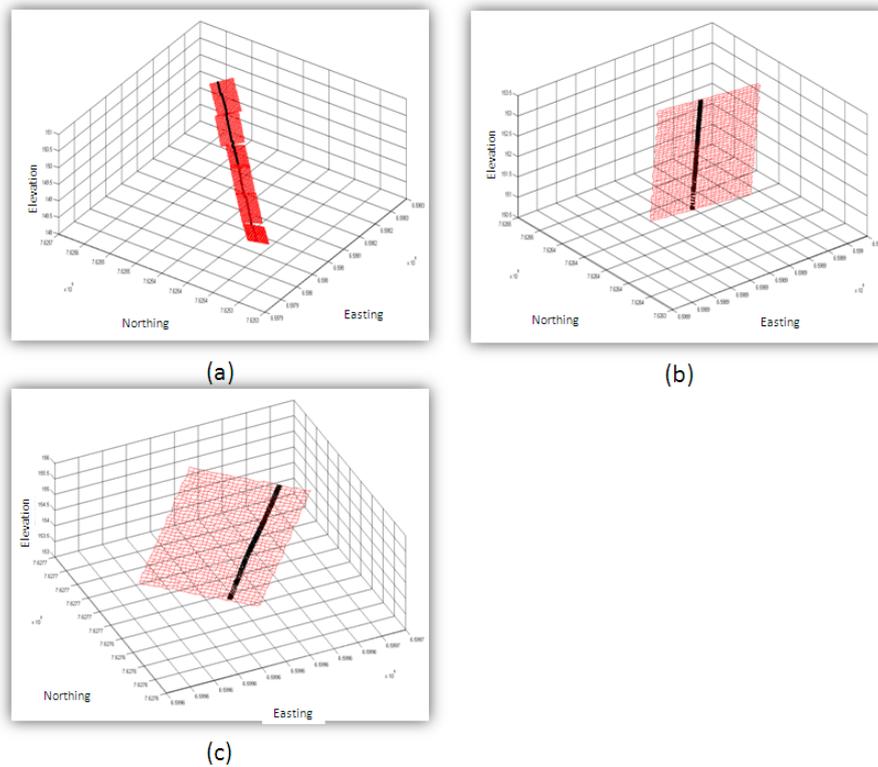


Figure 6.10: Surface grids and LiDAR points along the navigation track in the (a) first, (b) second and (c) third section of urban road.

the LiDAR points and the fitted surface grid points. The deviation maps were generated with red and blue representing the highest and lowest values respectively at their extreme ends. In the urban road sections, there was more variation on the surface, while as expected in the national primary road section, the variation was less.

We plotted a graph between the LiDAR points and their respective surface grid points along the navigation track in the road sections. By summing the mean and standard deviation of the elevation residuals we were able to plot and highlight the difference between the LiDAR points and the fitted surface grid points. The plots of the points along the navigation track in the first and the second urban road section are shown in Figure 6.17. Similarly, the



(a)

Figure 6.11: Digital image of the national primary road section (Geographic locations: $53^{\circ}38'14.407''$ N $7^{\circ}29'24.622''$ W).

plots of the points along the navigation track in the third urban and the national primary road section are shown in Figure 6.18. In these plots, the red symbols represent the LiDAR points, the blue symbols represent the surface grid points, while the green symbols highlight their elevation difference. In the urban road sections, some of the LiDAR points were found to vary from the surface grid points, while in the national primary road section, there was less variation between the points.

Box plots of the standard deviation of the elevation residual points along the navigation track in the first and the second urban road section are shown in Figure 6.19, while box plots for the points along the navigation track in the third urban and the national primary road section are shown in Figure 6.20. We also made statistical analysis of the standard deviation values calculated for the four road sections as shown in Table 6.4.

Considering just the minimum and maximum values, the best results were produced for the national primary road section, while results were found to

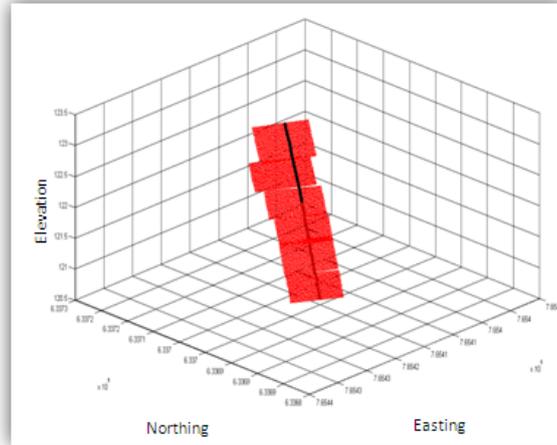
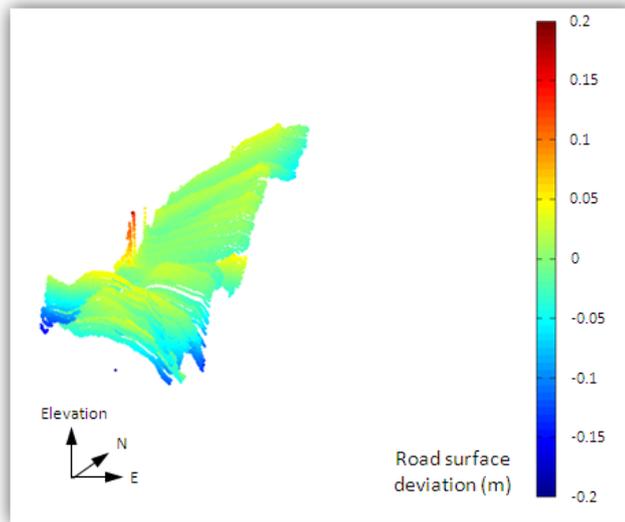


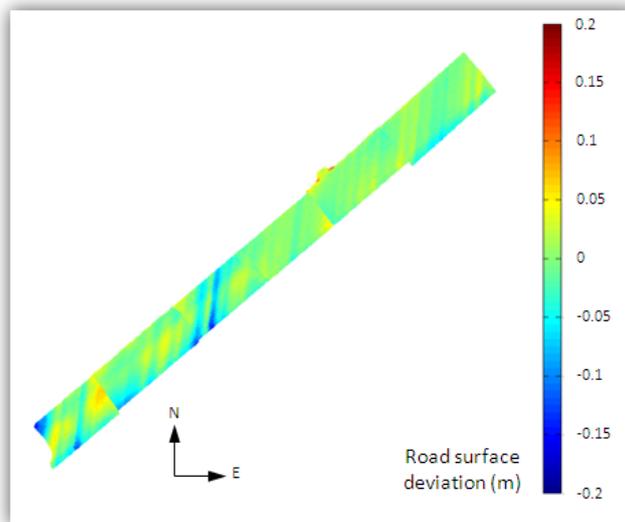
Figure 6.12: Inversely rotated surface grids and LiDAR points along the navigation track in the national primary road section.

be poorest for the first urban road section. The mean and median values were lowest for the national primary road section, while they were highest for the third urban road section. Similarly, the number of outliers were least for the national primary road section but highest for the second urban road section. These values indicate a smoother surface in the national primary road section and uneven surface at some of the points in the urban road sections.

We can also find a correlation between the calculated standard deviation and the IRI values, which will allow us to quantify the estimated road roughness into their standard reference scale values. However, we were unable to carry out the comparison as the IRI values were not available. Applying our algorithm to LiDAR data provides spatially referenced and localised roughness information along the road surface. We can also estimate the roughness across any track on the road section including the cross-section road surface profile.

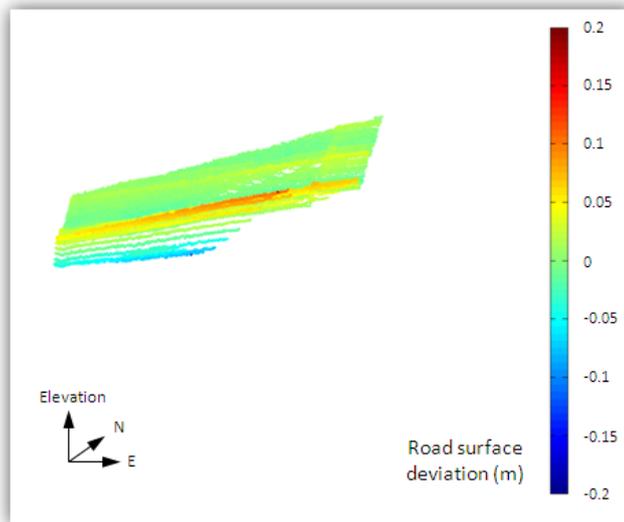


(a)

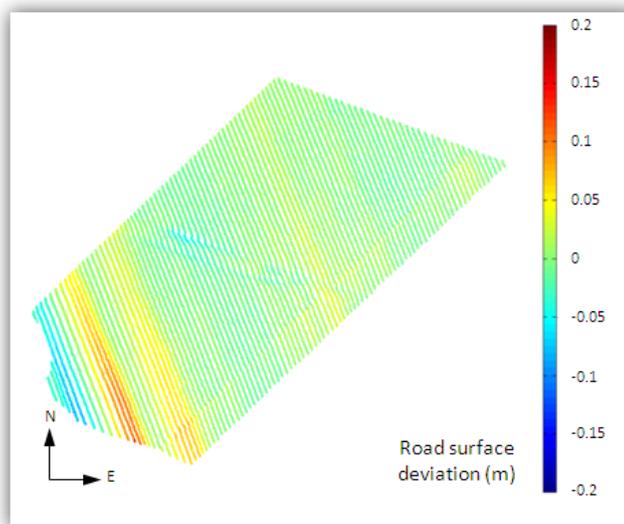


(b)

Figure 6.13: Road surface deviation maps for the first urban road section: (a) 3D and (b) 2D.

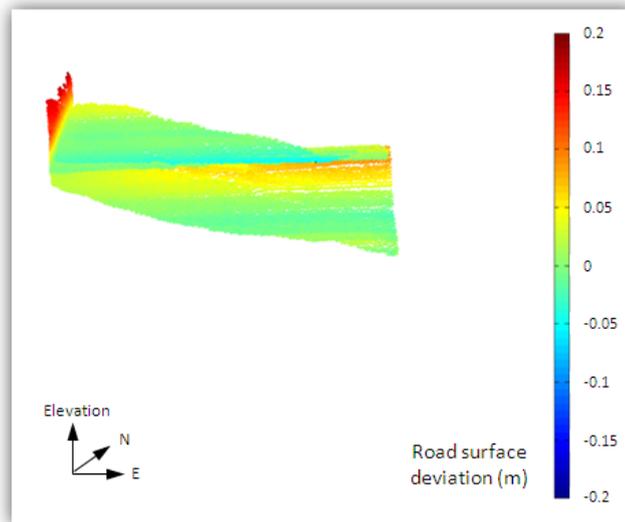


(a)

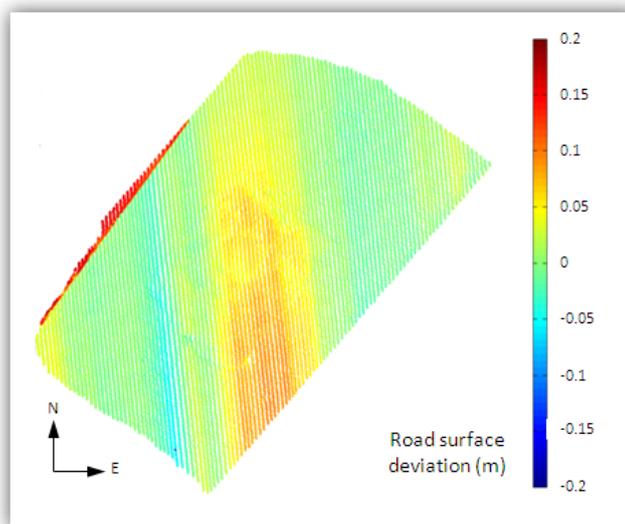


(b)

Figure 6.14: Road surface deviation maps for the second urban road section: (a) 3D and (b) 2D.

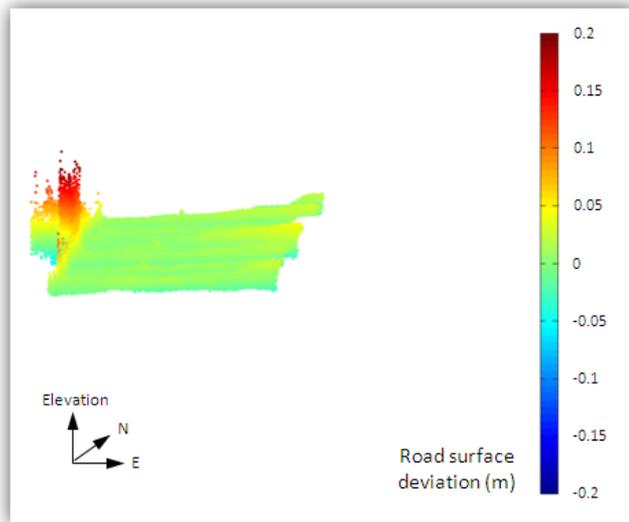


(a)

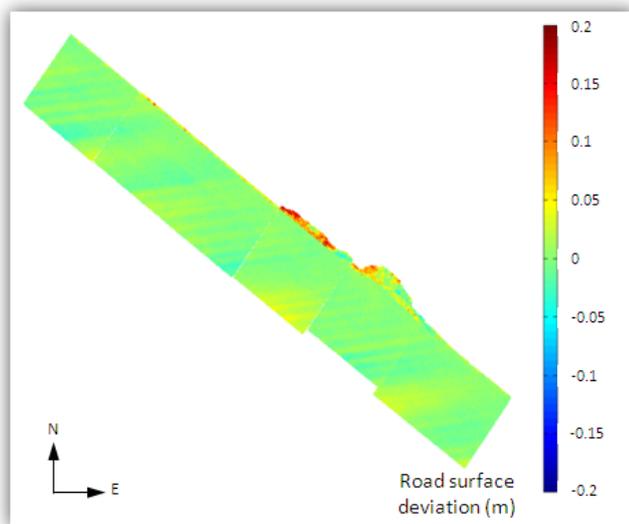


(b)

Figure 6.15: Road surface deviation maps for the third urban road section: (a) 3D and (b) 2D.

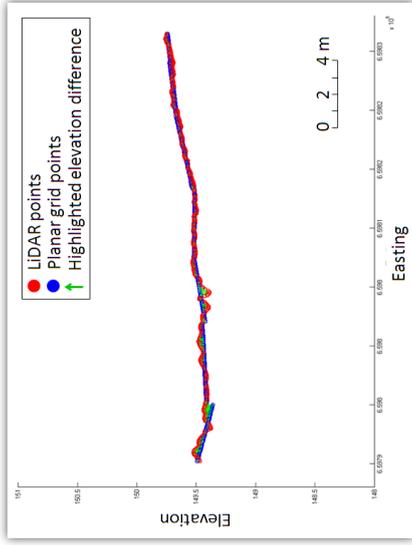


(a)

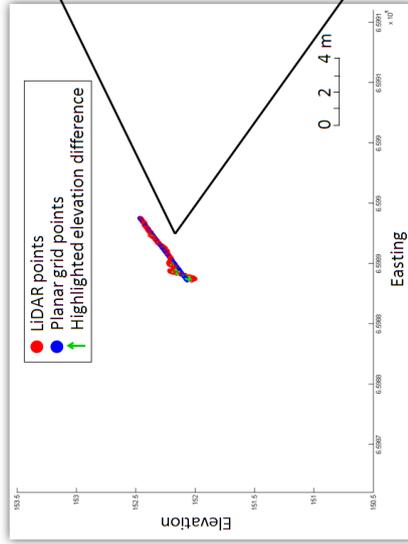


(b)

Figure 6.16: Road surface deviation maps for the national primary road section: (a) 3D and (b) 2D.



(a)



(b)

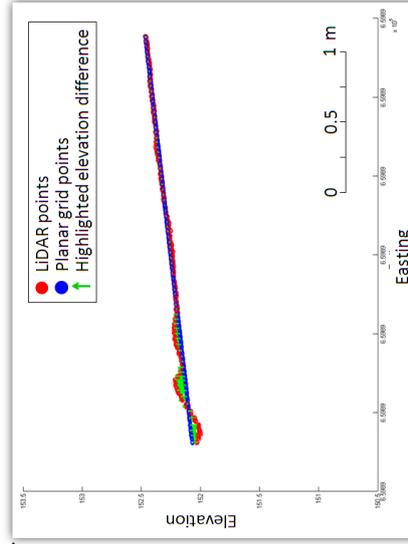
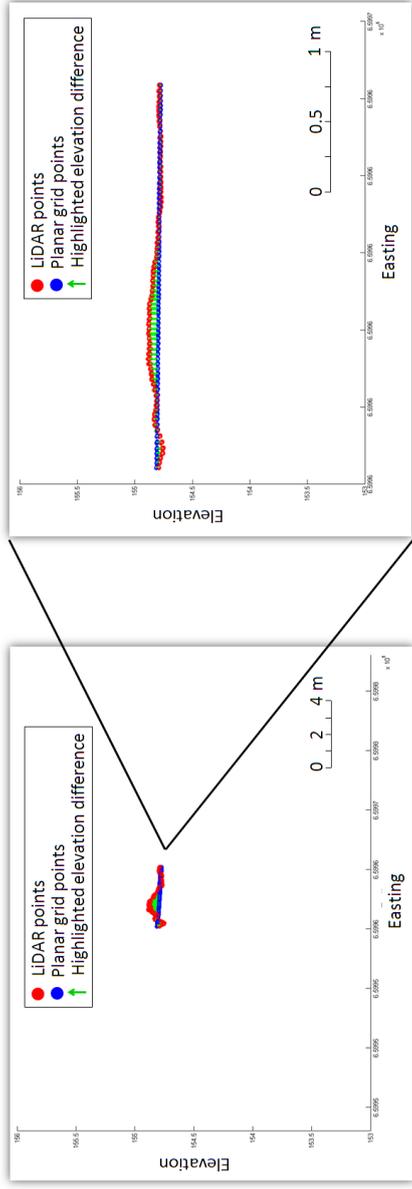
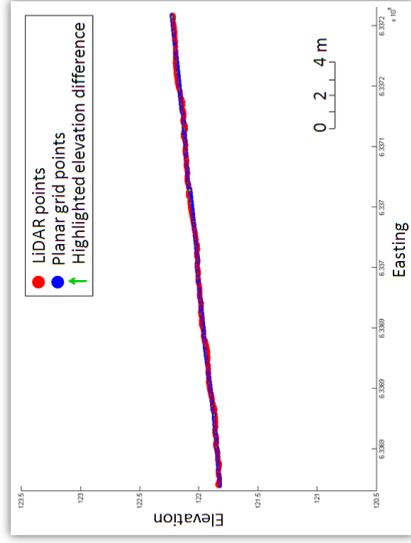


Figure 6.17: Plot of the LiDAR points and the surface grid points along the navigation track in (a) the first and (b) the second urban road section.

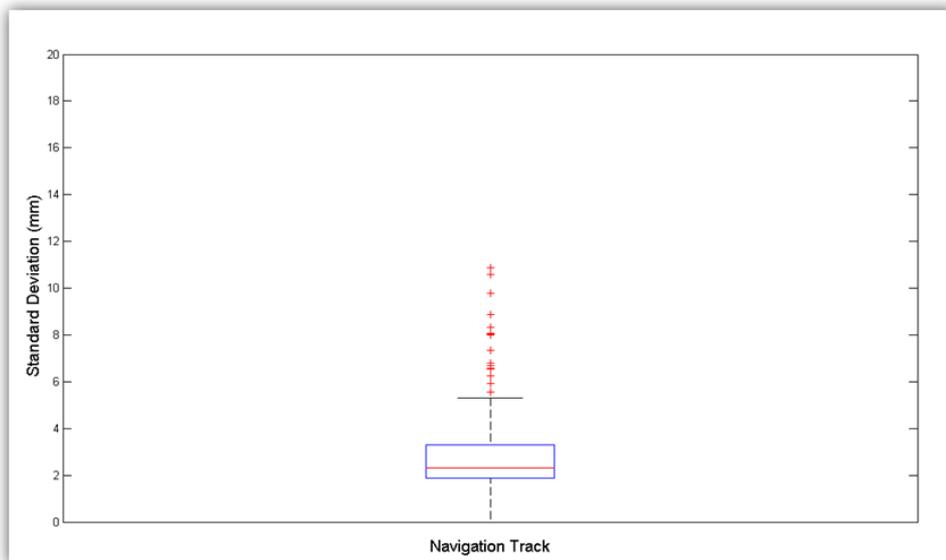


(a)

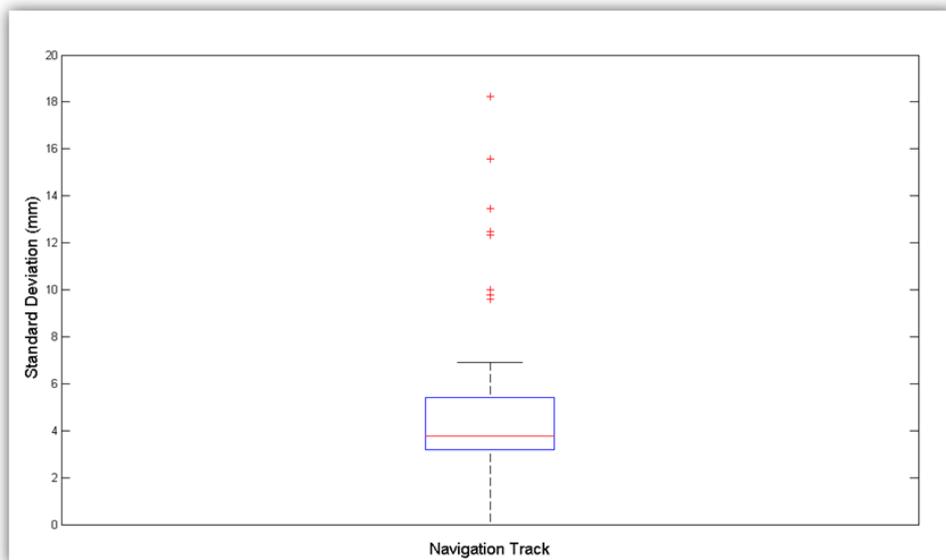


(b)

Figure 6.18: Plot of the LiDAR points and the surface grid points along the navigation track in (a) the third urban and (b) the national primary road section.

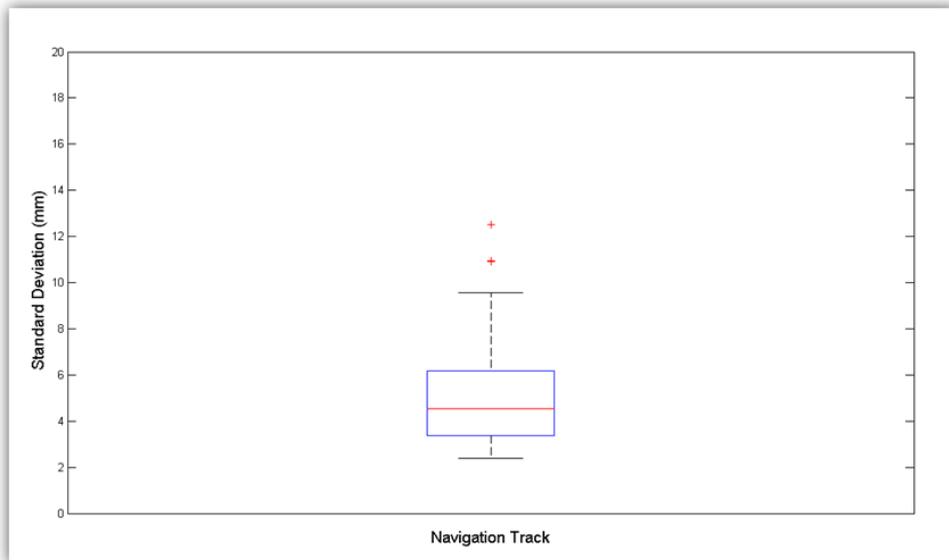


(a)

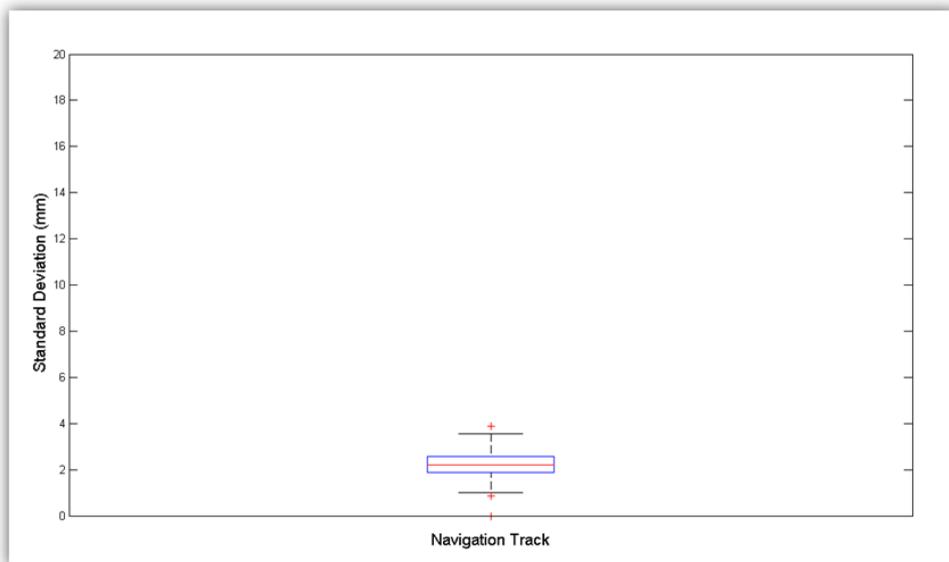


(b)

Figure 6.19: Box plot of the standard deviation of the elevation residual points along the navigation track in (a) the first and (b) the second urban road section.



(a)



(b)

Figure 6.20: Box plot of the standard deviation of the elevation residual points along the navigation track in (a) the third urban and (b) the national primary road section.

	First Urban	Second Urban	Third Urban	National Primary
minimum (mm)	0	0	2.38	0
maximum (mm)	20.14	18.22	12.53	3.87
lower adjacent (mm)	0	0	2.38	1.01
upper adjacent (mm)	5.32	6.92	9.56	3.55
25th percentile (mm)	1.88	3.17	3.38	1.88
75th percentile (mm)	3.29	5.41	6.18	2.56
mean (mm)	2.88	5.23	5.25	2.18
median (mm)	2.31	3.76	4.54	2.19
outliers (%)	7.83	17.39	8.11	3.91

Table 6.4: Statistical analysis of the standard deviation of the elevation residual points along the navigation track in the first, second, third urban and the national primary road section.

6.3 Discussion

We presented our algorithm for estimating road roughness from terrestrial mobile LiDAR data. Our algorithm was based on the assumption that by fitting a surface grid to the LiDAR points belonging to the road surface and computing the elevation difference, that road roughness could be estimated. We considered two challenges while developing this algorithm. The first challenge was to know the road surface extent, as was the case for the road marking extraction algorithm discussed in Chapter 5. This information was required to identify the LiDAR points that are relevant for estimating road roughness. We used our road edge extraction algorithm to determine the boundary of the road surface in the form of a snake curve allowing us to estimate the LiDAR points that belong to the road surface.

The second challenge was to provide a robustly fitted surface grid to the LiDAR points. In order to assist this process, the LiDAR points were rotated around the elevation axis and towards the Easting axis based on the average heading information of the mobile van along the road section under investi-

gation. We first divide the road surface into two sections, left and right. The RANSAC algorithm was applied to fit a surface grid to the LiDAR points that belong to the left side of the road section. As shown in Figure 6.4, this process was chosen due to the non-planar nature of full road surfaces. Attempting to fit one plane to both the left and right side of the road results in a surface that does not correctly fit either. We selected the surface grid cell size based on the surface area of the foot-print of the vehicle's wheel that comes into a contact with the road surface. The reason for choosing this dimension was to measure the roughness at each discrete instant of the vehicle's movement over the road surface. The fitted surface grid is a representation of the ideal road surface. The parameters used to apply the RANSAC were selected experimentally. However, we need to investigate their optimal values which can be used in our algorithm. Using the fitted surface grid, we determined the standard deviation value of the elevation residuals in each cell along the navigation track. These values provided a discrete estimation of the longitudinal road roughness. Our algorithm through the use of LiDAR data has an additional benefit of estimating the spatially referenced and localised roughness information along the road surface. It can also provide an estimate of the roughness across any track on the road section. The roughness information can be quantified into their standard reference scale values by finding a correlation between the calculated standard deviation and the IRI values along the road surface. We did not estimate the roughness along the right side of the road surface due to a lower density of the LiDAR points along that side.

We tested our algorithm on various sample sections of road. We selected four sections that covered 120 m of urban and national primary roads. In

the 50 m road sections, we used six LiDAR sections with an overlap of 2 m between each of them. This overlap was taken into account to avoid estimating the road roughness on these areas twice. The algorithm was able to provide standard deviation values that indicated the longitudinal roughness present along the navigation track. We generated surface deviation maps for the tested road sections. These deviations were computed as the elevation residuals between the LiDAR points and the fitted surface grid points. We plotted and highlighted the difference between the LiDAR points and the surface grid points by summing the mean and standard deviation of their elevation residuals. We carried out a statistical analysis of the calculated standard deviation values. In the national primary road section, the mean and median values of the standard deviation were 2.18 mm and 2.19 mm respectively, which indicate a smoother road surface. The values for the urban road sections were comparatively higher than the national primary roads. This indicates an uneven road surface in the former. The estimated road roughness could not be quantified in their standard reference scale as the IRI values were not available. In the next chapter, we discuss in more detail the contributions we have made to the field of road features extraction from LiDAR data.

Chapter 7

Conclusion

We presented a detailed review of terrestrial MMSs including imaging, laser scanning, navigation and data acquisition technology. This review provides an in-depth understanding of the operating principles of an MMS in terms of spatial data acquisition, direct georeferencing, synchronisation and data storage. We presented the XP-1 system, developed at NUI Maynooth, and described its components. We reviewed various methods developed for extracting road features from LiDAR data and identified short-comings in the current research. We described the suitability of terrestrial MMSs to acquire 3D information of road networks and route corridor environments that can, in turn, assist road authorities in identifying potential risk elements that may affect road safety. In this Chapter, we discuss the three main contributions we have made to the field of road feature extraction from LiDAR data.

7.1 Road Edge Extraction

We developed an automated algorithm for extracting road edges from terrestrial mobile LiDAR data. Our algorithm was based on the assumption

that LiDAR data provides elevation, intensity and pulse width information that can be used to differentiate the road surface from grass-soil edges in rural regions and kerb edges in urban regions. The algorithm is based on the novel use of a combination of two modified versions of the parametric active contour model to extract the road edges. We convert the LiDAR point cloud attributes into 2D raster surfaces in order to make the process of road edge extraction computationally efficient.

Our algorithm begins with the generation of 2D raster surfaces from LiDAR data, which is usually accompanied with high frequency noise. This inherent noise is filtered using a point thinning process in which multi-level terrain pyramids are generated from the LiDAR attributes. Smooth 2D raster surfaces are generated from the first level terrain pyramids using natural neighbourhood interpolation. To estimate the GVF external energy terms, we determine object boundaries from the raster surfaces with the consecutive use of hierarchical thresholding and Canny edge detection. The GVF external energy terms are estimated by iteratively diffusing the gradient vector values of the object boundaries determined from the raster surfaces. The balloon external energy is generated by providing a weight to the normal unit vector of the snake curve. In combination with the external energy, the balloon energy helps the snake grow to the road edges. The internal energy is input to the snake curve by adjusting its elasticity and stiffness properties. We developed a novel approach in which the snake curve is initialised as a parametric ellipse over 2D raster surface based on the navigation track of the mobile van. The navigation points and their associated headings are used to compute the semi-major and semi-minor axis of the snake curve. The snake curve is initialised at the centre of the road based on the angle between the

major axis and the slant height from the first or last navigation point. This angle is found empirically and fixed for the road sections with similar width. We use batch processing to extract the left and right edges from the final position of the closed snake curve. A series of overlapping snake curves are combined in their final positions and then non road edge points are removed. We have also developed an automated algorithm for validating the extracted left and right road edges with respect to their manual digitisation. In this validation algorithm, we consider navigation points as reference in order to compute the euclidean distances between the extracted and digitised road edge points.

We automate our road edge extraction algorithm through experimentation and the recommendation of optimal parameters and methods involved in it. We demonstrated that choosing a small cell size for computing the raster surfaces increases the computational cost without providing any considerable improvement in the extraction results and choosing a large cell size will affect the completeness and quality of the extracted road edges. We recommended a cell size of 0.06 m². We investigated the computational cost of the length of the road section to be processed. We found 10 m length of road section to be optimum for which the algorithm took a reasonable amount of time for processing. We analysed the importance and impact of the internal energy weight parameters in the algorithm to find their optimal values. A higher value of the α weight parameter resulted in the snake curve not being able to reach the road edges while a lower value led to the snake curve being jagged at the road edges. Similarly, a higher value of the β parameter caused extreme bending in the snake curve at some of its points while a lower value did not have any noticeable impact. We found optimal values for the α and

β parameters empirically which were applied successfully to all road sections. We also investigated the importance of the GVF and balloon external energy weight parameters in the algorithm. We analysed different weight values for the slope, reflectance and pulse width in the GVF energy and balloon energy terms to find the best combination of their values in the algorithm. We decided to assign the highest weight parameter to the slope based GVF energy term due to the imperfect normalised intensity values in our data, the inability of pulse width values to properly distinguish the kerb edges in urban road sections and primarily to the importance of height change in detecting edges. We expect correctly normalised intensity values would increase the accuracy of our algorithm. The pulse width values are more useful in rural road sections where they are able to distinguish grass-soil edges. The use of an increased value of the balloon energy weight parameter can be useful in noisy data as it can provide an additional inflation energy to the snake curve to overcome the noise.

To test our road edges extraction algorithm, we selected three different road sections. These sections covered 150 m of rural, urban and national primary roads that consisted of three different types of edges. In the first test, we applied our algorithm using the best manually selected hierarchical thresholding parameters for each road section. In the second test, the automated algorithm was applied with a similar set of hierarchical thresholding parameters for each road section. We validated both the manually and the automatically extracted road edges with respect to their manual digitisation. In the rural road section, the automated left and right edge displayed best results having mean accuracy values of 0.126 m and 0.038 m respectively. The median values, 0.096 m and 0.029 m, were best for the manual left edge and

the automated right edge respectively. The left edge accuracy results were poor compared to the right edge. This was due to the weak cells along the left edge which caused the snake curve to extend incorrectly into the grass and soil area. The difference in accuracy between manual and automated road edges were minimal. This was due to the similar values of the empirically and the median selected, T_{slope} and T_{ref} , parameters used for this road section. In the urban road section, the best mean and median values were -0.033 m and -0.034 m respectively for the automated left edge, and 0.013 m and -0.078 m for the manual right edge. The right edge accuracy results, in most cases, were poorer than the left edge due to a lower LiDAR point density along the right side of the road section. The automated right edge results were poor due to the use of T_{ref} and T_{pw} median parameters which were higher than their empirically selected values. For the national primary road section, the best mean and median values were 0.165 m and 0.082 m for the manual left edge, and -0.03 m and -0.001 m for the manual right edge. The left edge accuracy results were poor due to the increased balloon energy pushing the snake curve beyond the weak left edge points. The automated road edge results were affected due to the use of T_{slope} median parameter which was higher than its empirically selected value.

As expected, the manual road edges displayed better results than the automated process. But in both the processes, our algorithm was able to successfully extract the road edges. A decision on the selection of manual or automated extraction of the road edges can be made on the basis of accuracy or degree of automation required for extracting the road edges.

7.2 Road Marking Extraction

We developed an automated algorithm for extracting road markings from terrestrial mobile LiDAR data. The intensity returns of road markings from the LiDAR data have higher values than other road surface elements. We expected to extract these road markings by applying a threshold to the data.

We use our automated road edge extraction algorithm to determine the boundaries of the road in the form of a snake curve. We can then identify LiDAR points that belong to the road surface. A priori knowledge of the road surface reduces false positives and the amount of processing in the road markings extraction process. LiDAR intensity values depend upon the distance from the laser scanner to the illuminated surface, the incidence angle of the laser pulse and the illuminated surface's characteristics. We developed a range dependent thresholding function which attempts to deal with all of these three factors that affect the intensity values. The use of a threshold itself clusters data based on the reflectivity from the illuminated surface while a range dependent threshold takes account of the distance and incidence angle issues. The range dependent threshold is applied by converting the LiDAR intensity and range attributes into 2D rasters surfaces. To complete road markings shapes and to account for noise, we apply morphological operations. The raster surface is first converted into a binary image and then the dilation operation is applied. The binary cells are dilated using a linear shaped structuring elements such that any gap in between them is filled. The length of the structuring elements is selected empirically while its angle is calculated from average heading of the mobile van. The noise is removed based on a priori knowledge of the dimensions of the dilated road markings. The binary dilated cells are then eroded using the same linear

shaped structuring element.

Our algorithm was automated through the selection of the most applicable values of input parameters. We recommended a cell size of 0.06 m^2 for creating the raster surfaces. We demonstrated that the selection of the cell size dimension is a choice between accuracy and completeness. A cell size that is too small reduces the completeness while a cell size that is too large produces output road markings whose dimensions are greater than actual road markings. We selected a single optimal threshold value empirically and then developed a formula for applying the threshold values as a function of range along the road surface. We demonstrated the importance of choosing an optimal threshold value for extracting road markings. If the threshold value is too small, the extracted road marking will be accompanied with large areas of noise and if the threshold value is too large, it will remove the noise at the expense of extracting all the road markings.

We tested our automated road markings extraction algorithm on seven different road sections. These sections covered 170 m of rural, urban and national primary roads that consisted six different types of the road markings. Our algorithm was able to extract road markings, with an 86% detection rate. The majority of the undetected road markings were due to a lower point density of our LiDAR data along the right side of the road section. The algorithm also falsely identified 13 road markings. Most of these identified road markings were in rural and national primary road sections where the extracted road edges extended incorrectly to the nearby grass and soil area. This extension was primarily attributed to the use of increased balloon energy in them.

7.3 Road Roughness Estimation

We developed an automated algorithm for estimating road roughness from terrestrial mobile LiDAR data. Our algorithm was based on the assumption that fitting a surface grid to the LiDAR points belonging to the road surface provides elevation residual values that can be used for estimating road roughness. This road surface fitting process is an attempt to automatically reconstruct an ideal road surface.

A priori knowledge of the road surface in the LiDAR data allows roughness along the road surface to be estimated. We rotate the LiDAR points around the elevation axis and towards the Easting axis based on the average heading information of the mobile van. This step is carried out to assist the process of fitting the surface grid to the LiDAR points. We divide the road surface into left and right sections. We use a RANSAC algorithm to fit a surface grid to the LiDAR points belonging to the left side of the road section. We do not fit a surface grid to the LiDAR points belonging to the whole road surface as it provides an improper representation of the road surface. A cell size in the surface grid is provided based on the surface area of the vehicle's wheel foot-print that comes into a contact with the road surface at any instant. This size is selected to measure the roughness experienced at each instant of the vehicle's movement over the road surface. We calculate the standard deviation value of elevation residuals in each cell along the navigation track of the mobile van. These values provide a discrete estimation of the longitudinal road roughness. We do not find the roughness along the right side of the road surface due to a lower point density of our LiDAR data along that side. Applying our algorithm to LiDAR data has the advantage of estimating spatially referenced and localised roughness information across

any track on the road section.

We tested our automated road roughness estimation algorithm on four different road sections. These sections included 120 m of urban and national primary roads. We removed duplicate points in the LiDAR sections to avoid the dual estimation of the roughness during batch processing. Our algorithm was able to estimate standard deviation values that indicated the longitudinal roughness along the navigation track. We generated road surface deviation maps and plotted graphs to highlight the difference between the LiDAR points and the fitted surface grid points. We also carried out a statistical analysis of the calculated standard deviation values. The best mean and median values of the standard deviation were 2.18 mm and 2.19 mm respectively. These values were generated from the national primary road section which indicate a smoother road surface. The statistical values in the urban road sections were comparatively higher which indicate an uneven surface. We were not able to quantify the estimated road roughness in their standard reference scale as the IRI values were not available. In the next chapter, we present future work.

Chapter 8

Future Work

We have made three main contributions to the field of road feature extraction from LiDAR data as outlined in this thesis. In this chapter, we detail future research directions that are enabled by our work. We provide some recommendations for improvements that can be made to the research presented in this thesis.

8.1 Road Edge Extraction

Quantifying the effect of point density

The performance of the road edge extraction algorithm developed in the course of this research can be affected by point density. Lower LiDAR point density along the right side of the road section due to single LiDAR scanner located on left side of the mobile platform gives rise to noisy cells in the resulting 2D raster surfaces. Present day terrestrial mobile laser scanning systems used in professional road surveys are generally equipped with more than one laser scanner. This solution produces LiDAR data that has a more

uniform point density along both sides of the road section. We intend to test our road edge extraction algorithm using such LiDAR data as part of our follow-up research programme. The use of these LiDAR data will lead to the generation of smoother 2D raster surfaces, which in turn, will improve the quality of extracted road edges. This will allow us to examine in more detail the affect of point density on our algorithm, in terms of accuracy and completeness. It will also allow us to determine the lower limits of point density for extracting road edges at varying distances. This should also remove the necessity to change the balloon energy weighting for road sections based on varying point density.

Improving the quality of LiDAR reflectance attribute

The normalisation of intensity values to estimate reflectance is a relatively new addition to laser scanners and has yet to be fully tested and implemented. Our Riegl VQ-250 laser scanner does not provide reliable reflectance values. We intend to develop a method to normalise intensity values with respect to the illuminated surface characteristics, the distance from the laser scanner to the illuminated surface and the incidence angle of the laser pulse. Their use in the algorithm will lead to the generation of a smoother reflectance raster surfaces. This will allow us to assign a higher weight to the more reliable reflectance GVF energy parameter in the algorithm. This will be useful in rural and national primary road sections consisting of grass-soil edges where a slope difference between the road surface and grass-soil surface is negligible.

Automated selection of intersection points for snake curve merging

During batch processing of the consecutive road sections, we select intersection points with the highest and lowest Northing values between the overlapping snake curves. We need to investigate alternative options for selecting intersection points. The aim would be to identify the optimum approach to remove points in the most irregular snake curve.

Road edge extraction and the geometric active contour model

In our algorithm, we use a parametric active model since its implementation is less computationally expensive when compared to the geometric active contour model. Further research is required to investigate the applicability of the latter for extracting road edges. This could be advantageous as it will remove the requirement for weighting various input parameters.

Algorithm extensions

Two other extensions we intend to develop are error correction and larger scale implementation. In our mobile mapping systems group at NUIM, algorithms have been developed to remove false road edges which can be caused by false positives or by occlusions. Large scale implementation of these algorithms will be enabled through the ongoing construction of a data management system to handle 100s of kms of LiDAR data at NUIM.

8.2 Road Marking Extraction

Point density and accuracy

We intend to extend the testing of our road marking extraction algorithm using high density LiDAR data that is regularly spaced. This should enable complete detection of road markings along both sides of the road surface. We will also be able to examine the accuracy of our road marking detection and the accuracy of the object shape extraction.

Improving thresholding using improved reflectance values

The use of normalised values of intensity which represent reflectance attribute will allow us to select a single threshold for extracting road markings. This will remove the requirement of range dependent threshold values in our algorithm which could improve accuracy.

Noise removal

Morphological operations present a useful approach in terms of completing the shapes of road markings and in removing noise. These operations are applied using the linear shaped structuring elements. Further research is required to investigate alternative structuring elements which could be useful in removing noise introduced through the extracted road edges, extending incorrectly into grass and soil area.

Road marking classification

After we have a more robustly extracted road markings we can then implement a road markings recognition and classification tool. The output of our algorithm is a set of LiDAR points and objects. By examining each object, its shape, dimension and position on the road we intend to construct more effective algorithm that can recognise and classify each road marking type.

8.3 Road Roughness Estimation

Increasing sensitivity to road roughness

Our road roughness estimation algorithm provides a discrete estimation of the roughness in the form of the standard deviation of elevation residuals in each cell of the surface grid. One short-coming of the discrete measurement is that it may not produce a continuous measure of roughness similar to conventional IRI approach. One solution might be to use a sliding surface grid technique in which the grid could be simulated to slide along the road surface. The standard deviation values could be estimated at each instant. This could provide a dynamic estimation of the roughness along the road surface.

Surface grid fitting

We use the RANSAC algorithm to fit a surface grid to the LiDAR points. The fitted surface grid consists of slightly skewed cells. This step is intended to provide an accurate representation of the ideal road surface. An example of the fitted surface grid consisting of skewed cells is shown in Figure 8.1. The

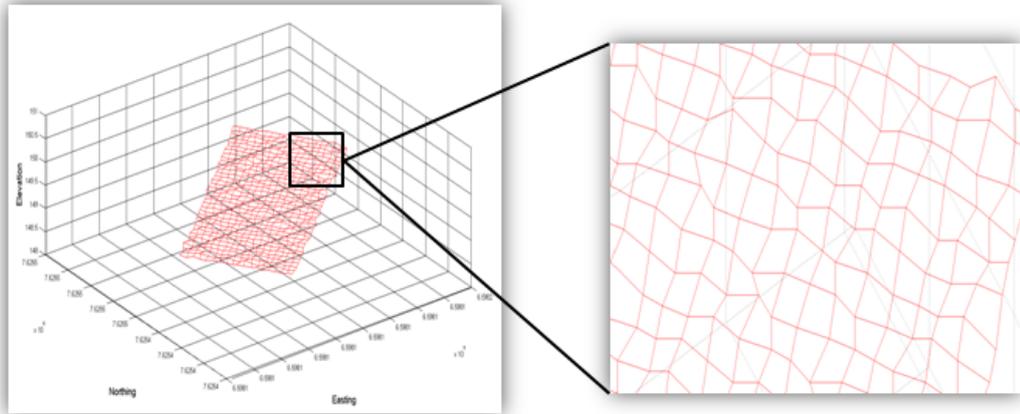


Figure 8.1: Fitted surface grid with an inset picture of its skewed cells.

parameters used to apply the RANSAC are selected experimentally. Further research is required to investigate their optimal values in the algorithm in order to generate a fitted surface grid that would best represent an ideal road surface.

Correlating roughness estimation with IRI scale

Direct comparison with the international standard IRI scale is required to aid interpretation of our road roughness estimation. Additional research is required to find a correlation between the calculated standard deviation and the IRI values along the road surface. This will allow us to quantify the estimated road roughness against this globally accepted IRI scale. This also requires us to process larger road sections as has been described in Section 8.1.

Point density effect on roughness estimation

The underlying point density of the LiDAR will affect the surface grid fitting. We need to investigate the limits of point density on our surface grid fitting approach and recommend a point density for different road surface grades.

Bibliography

- [AKF⁺05] N A Akel, K Kremeike, S Filin, M Sester, and Y Doytsher. Dense DTM generalization aided by roads extracted from LiDAR data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, pages 54–59, Enschede, The Netherlands, 2005. ISPRS.
- [BA98] W Benning and T Aussems. Mobile mapping by a car-driven survey system (CDSS). In *Symposium on Geodesy for Geotechnical and Structural Engineering*, pages 1–8, Eisenstadt, Austria, 1998. International Association of Geodesy (IAG).
- [BEF⁺96] J Borenstein, H R Everett, L Feng, S W Lee, and R H Byrne. Where am i? Systems and methods for mobile robot positioning. pages 1–282, <http://www-personal.umich.edu/~johannb/Papers/pos96rep.pdf>. Date Accessed: 12-01-2012, 1996.
- [BEY⁺10] C Brandstaetter, P Evgenikos, G Yannis, P Papantoniou, E Argyropoulou, J Broughton, J Knowles, M Reurings, M Vis, J F Pace, E Lopez-de Cozar, P Perez-Fuster, J Sanmartin, and M Haddak. Annual statistical report, Deliverable D3.1 of the

- EC FP7 project DaCoTA. pages 1–95, <http://www.dacota-project.eu>. Date Accessed: 18-12-2011, 2010.
- [BH09] S Becker and N Haala. Grammar supported facade reconstruction from mobile lidar mapping. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 229–234, Paris, France, 2009. ISPRS.
- [BI98] A Blake and M Isard. *Active Contours*. Springer, London, UK, 1998.
- [BLGTK07] H Boulaassal, T Landes, P Grussenmeyer, and F Tarsha-Kurdi. Automatic segmentation of building facades using terrestrial laser data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, pages 65–70, Espoo, Finland, 2007. ISPRS.
- [BMS06] D Barber, J Mills, and S Smithvoysey. Geometric validation of a ground-based mobile laser scanning system. *ISPRS Journal of Photogrammetry and Remote Sensing*, pages 128–141, January 2006.
- [Bur02] R Burtch. Lidar principles and applications. In *Imagine Conference*, pages 1–13, Traverse City, Michigan, US, 2002.
- [Can86] J Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), 1986.

- [CAR07] CARE. Road accident statistics in Europe. pages 1–10, http://ec.europa.eu/sverige/documents/traffic_press_stats.pdf. Date Accessed: 15-12-2011, 2007.
- [CCCD93] V Caselles, F Catt, T Coll, and F Dibos. A geometric model for active contours in image processing. *Numerische Mathematik*, 66:1–31, 1993.
- [CKR04] S Clode, P Kootsookos, and F Rottensteiner. The automatic extraction of roads from lidar data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004.
- [Coh91] L D Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, 1991.
- [Col07] R Collins. Robust estimation: RANSAC. pages 1–46, http://www.cse.psu.edu/~rcollins/CSE486/lecture15_6pp.pdf. Date Accessed: 10-3-2012, 2007.
- [Cra09] C Crawford. Minimising noise from LiDAR for contouring and slope analysis. <http://blogs.esri.com/~Minimizing-noise-from-lidar-for-contouring-and-slope-analysis.aspx>. Date Accessed: 10-1-2012, 2009.
- [CSW⁺09] X Chen, M Stroila, R Wang, B Kohlmeyer, N Alwar, and J Bach. Next generation map making : geo-referenced ground-level LiDAR point clouds for automatic retro-reflective road feature extraction. In *17th International Conference on Advances*

- in Geographic Information Systems (ACM SIGSPATIAL GIS)*, Seattle, US, 2009. ACM Press.
- [CTMF08] M Cavalli, P Tarolli, L Marchi, and G D Fontana. The effectiveness of airborne LiDAR data in the recognition of channel-bed morphology. *Catena*, 73:249–260, May 2008.
- [Der10] K G Derpanis. Overview of the RANSAC algorithm. pages 1–2, http://www.cse.yorku.ca/~kosta/CompVis_Notes/ransac.pdf. Date Accessed: 12-2-2012, 2010.
- [dFdS09] M M de Farias and R O de Souza. Correlations and analyses of longitudinal roughness indices. *Road Materials and Pavement Design*, 10(2):399–415, 2009.
- [DJS⁺10] J C F Diaz, J Judge, K C Slatton, R Shrestha, W E Carter, and D Bloomquist. Characterization of full surface roughness in agricultural soils using groundbased lidar. In *International Geoscience And Remote Sensing Symposium (IGARSS)*, pages 4442–4445, Honolulu, US, 2010. IEEE.
- [EAH08] J Engels, H Arefi, and M Hahn. Generation of roof topologies using plane fitting with RANSAC. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVII, pages 119–126, Beijing, China, 2008. ISPRS.
- [EES02] C Ellum and N El-Sheimy. Land-based mobile mapping systems. *Photogrammetric Engineering & Remote Sensing*, 68(1):13–17, 2002.

- [ERE⁺08] R Eenink, M Reurings, R Elvik, J Cardoso, S Wichert, and C Stefan. Accident prediction models and road safety impact assessment: recommendations for using these tools. pages 1–20, <http://ec.europa.eu>. Date Accessed: 17-12-2011, 2008.
- [ERS06] ERSO. Roads. pages 1–41, <http://ec.europa.eu>. Date Accessed: 8-12-2011, 2006.
- [Es05] N El-sheimy. An overview of mobile mapping systems. In *From Pharaohs to Geoinformatics*, pages 1–24, Cairo, Egypt, 2005.
- [ESR10] ESRI. Terrain pyramids. <http://help.arcgis.com>. Date Accessed: 28-12-2011, 2010.
- [ESR11] ESRI. Terrain to raster (spatial analyst). <http://help.arcgis.com>. Date Accessed: 01-02-2012, 2011.
- [ETS97] ETSC. Road safety audit and safety impact assessment. pages 1–30, <http://www.etsc.eu/oldsite/roadaudit.pdf>. Date Accessed: 15-12-2011, 1997.
- [FB81] M A Fischler and R C Bolles. Random Sample Consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 1981.
- [FNSZ03] G Forlani, C Nardinocchi, M Scaioni, and P Zingaretti. Building reconstruction and visualization from LiDAR data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXIV, pages 151–156, Portonovo, Italy, 2003. ISPRS.

- [FPWW00] R Fisher, S Perkins, A Walker, and E Wolfart. Gaussian smoothing. <http://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm>. Date Accessed: 23-2-2012, 2000.
- [GNA⁺06] F Goulette, F Nashashibi, I Abuhadrous, S Ammoun, and C Lourceau. An integrated on-board laser range sensing system for on-the way city and road modelling. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information*, volume XXXVI, Paris, France, 2006. ISPRS.
- [Goa91] C C Goad. The Ohio state university highway mapping project : the positioning component. In *47th Annual Meeting. The Institute of Navigation (ION)*, pages 117–120, Williamsburg, VA, 1991. The Institute of Navigation.
- [GPG⁺07] G Gatti, C Polidori, I Galvez, K Mallschutzke, R Jorna, M Leur, M Dietze, D Ebersbach, C Lippold, B Schlag, G Weller, A Wyczynski, F Iman, and C Aydin. Safety handbook for secondary roads. pages 1–227, <http://ec.europa.eu>. Date Accessed: 5-1-2012, 2007.
- [GQS06] X Geng, L Qingquan, and Z C Shi. GPS/IMU/DMI integrated high-precision positioning approach. *Geoinformatics: GNSS and Integrated Geospatial Applications*, 6418:1–8, 2006.
- [GR09] J Goepfert and F Rottensteiner. Adaption of roads to ALS data by means of network snakes. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information*, volume XXXVIII, pages 24–29, Paris, France, 2009. ISPRS.

- [GR10] J Goepfert and F Rottensteiner. Using building and bridge information for adapting roads to ALS data by means of network snakes. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 163–168, Saint-Mande, France, 2010. ISPRS.
- [Gra08] G Grafe. Kinematic 3D laser scanning for road or railway construction surveys. In *1st International Conference on Machine Control & Guidance*, pages 1–10, Zurich, Switzerland, 2008.
- [GSG03] H Gontran, J Skaloud, and P Y Gillieron. A mobile mapping system for road data capture via a single camera. In *Optical 3D Measurement Techniques VI*, Zurich, Switzerland, 2003.
- [GWA07] M S Grewal, L R Weill, and A P Andrews. *Global positioning systems, inertial navigation and integration*. John Wiley & Sons, New Jersey, US, 2007.
- [HC10] G Hunter and C Cox. StreetMapper Brochure. Technical report, 3D Laser Mapping, Nottingham, UK, 2010.
- [HCH⁺95] C Hock, W Caspary, H Heister, J Klemm, and H Sternberg. A kinematic surveying system for automatic GIS-data acquisition. In *SPIE, Digital Photogrammetry and Remote Sensing*, volume 2646, pages 124–130. SPIE, 1995.
- [HDP09] K Hammoudi, F Dornaika, and N Paparoditis. Extracting building footprints from 3D point clouds using terrestrial laser scanning at street level. In *International Archives of Photogramme-*

- try, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 65–70, Paris, France, 2009. ISPRS.
- [HG92] R M Haralick and Shapiro L G. *Computer and robot vision*. Addison-Wesley, Reading, MA, 1992.
- [HH10] M Hollaus and B Hofle. Terrain roughness parameters from full-waveform airborne lidar data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information*, volume XXXVIII, pages 287–292, Vienna, Austria, 2010. ISPRS.
- [HP07] B Hofle and N Pfeifer. Correction of laser scanning data: data and model-driven approaches. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62:415–433, 2007.
- [HPKH08] N Haala, M Peter, J Kremer, and G Hunter. Mobile LiDAR mapping for 3D point cloud collection in urban areas - a performance test. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVII, pages 1119–1124, Beijing, China, 2008. ISPRS.
- [HTH04] X Hu, C V Tao, and Y Hu. Automatic road extraction from dense urban area by integrated processing of high resolution imagery and lidar data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004. ISPRS.
- [IRF03] IRF. Safe mobility: an IRF discussion paper. pages 1–16, <http://www.irfnet.org>. Date Accessed: 29-11-2011, 2003.

- [IXS09] IXSEA. LANDINS user guide: introduction. Technical report, IXSEA, 2009.
- [JG09] B Jutzi and H Gross. Normalization of lidar intensity data based on range and surface incidence angle. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, Paris, France, 2009. ISPRS.
- [JHHK08] A Jaakkola, J Hyypä, H Hyypä, and A Kukko. Retrieval algorithms for road surface modelling using laser based mobile mapping. *Sensors*, 8:5238–5249, September 2008.
- [KAS⁺07] A Kukko, C O Andrei, V M Salminen, H Kaartinen, Y Chen, P Rönnholm, H Hyypä, J Hyypä, R Chen, H Haggren, I Kosonen, and K Capek. Road environment mapping system of the Finnish geodetic institute-FGI ROAMER. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, pages 241–247, Espoo, Finland, 2007. ISPRS.
- [KEA10] M Kabolizade, H Ebadi, and S Ahmadi. An improved snake model for automatic extraction of buildings from urban aerial images and LiDAR data. *Computers, Environment and Urban Systems*, 34(5):435–441, August 2010.
- [Ker98] M Kerschner. Homologous twin snakes integrated in a bundle block adjustment. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXII, pages 244–249, Columbus, Ohio, US, 1998. ISPRS.

- [KfB103] Z Ke-fei and X Ben-lin. GPS and mobile multisensor mapping system. *Wuhan University Journal of Natural Sciences*, 8(2B):557–565, June 2003.
- [KWT88] M Kass, A Witkin, and D Terzopoulos. Snakes: Active Contour Models. *International Journal of Computer Vision*, 1(4):321–331, 1988.
- [LC08] J Li and M A Chapman. Terrestrial mobile mapping towards real time geospatial data collection. In S Zlatanova and J Li, editors, *Geospatial Information Technology for Emergency Responses*, pages 103–119. Taylor & Francis Group, London, UK, 2008.
- [Lit01] D Litwiller. CCD vs. CMOS : facts and fiction. *Photonics Spectra*, (1), 2001.
- [Lit05] D Litwiller. CMOS vs. CCD: maturing technologies, maturing markets. *Photonics Spectra*, (8), 2005.
- [LKC08] T M Lillesand, R W Kiefer, and J W Chipman. *Remote sensing and image interpretation*. John Wiley & Sons, New Jersey, US, 6th edition, 2008.
- [LM10] Y C Lin and J P Mills. Factors influencing pulse width of small footprint , full waveform airborne laser scanning data. *Photogrammetric Engineering & Remote Sensing*, 76(1):49–59, 2010.

- [LRKH06] T Luhmann, S Robson, S Kyle, and I Harley. *Close range photogrammetry: principles, methods and applications*. Whittles Publishing, Caithness, Scotland, UK, 2006.
- [LS95] A Lundervold and G Storvik. Segmentation of brain parenchyma and cerebrospinal fluid in multispectral magnetic resonance images. *IEEE transactions on medical imaging*, 14(2):339–349, January 1995.
- [MEs10] A Moussa and N El-sheimy. Automatic classification and 3D modeling of LiDAR data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 155–159, Saint-Mande, France, 2010. ISPRS.
- [MK06] T V Mathew and K V KrishnaRao. Road markings. <http://www.cdeep.iitb.ac.in/nptel>. Date Accessed: 20-7-2012, 2006.
- [MKCM10] C P McElhinney, P Kumar, C Cahalane, and T McCarthy. Initial results from european road safety inspection (EURSI) mobile mapping project. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, Newcastle, UK, 2010. ISPRS.
- [MM09] S A Mumtaz and K Mooney. A semi-automatic approach to object extraction from a combination of image and laser data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 53–58, Paris, France, 2009. ISPRS.

- [MM10] T McCarthy and C P McElhinney. European road safety inspection (EuRSI) research project. In *European Transport Conference*, Glasgow, UK, 2010. Association for European Transport.
- [MMHN08] K M Molony, C P McElhinney, B M Hennelly, and T J Naughton. Segmentation of three-dimensional scenes encoded in digital holograms. *Optics and Photonics for Information Processing II*, 7072:1–11, 2008.
- [Moe08] T B Moeslund. *Image and video processing*. Computer Vision and Media Technology, Aalborg University, Aalborg, 2 edition, 2008.
- [MSV95] R Malladi, J A Sethian, and B C Vemuri. Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175, 1995.
- [MZC05] S D Mayunga, Y Zhang, and D J Coleman. Semi-automatic building extraction utilizing quickbird imagery. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, pages 131–136, Vienna, Austria, 2005. ISPRS.
- [Nov93] K Novak. Mobile mapping systems: new tools for the fast collection of GIS information. In *State-of-the Art Mapping*, volume 1943, pages 188–198, Orlando, Florida, US, 1993. SPIE.
- [NRA10] NRA. Traffic signs manual - chapter 7. Technical report, National Road Authority (NRA), Dublin, Ireland, 2010.

- [Opt10] Optech. LYNX mobile mapper. Technical report, Optech, Vaughan, Canada, 2010.
- [OTDS04] K Oda, T Takano, T Doihara, and R Shibasaki. Automatic building extraction and 3D city modeling from lidar data based on hough transformation. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004. ISPRS.
- [OV09] S J Oude Elberink and G Vosselman. 3D information extraction from laser point clouds covering complex road junctions. *The Photogrammetric Record*, 24(125):23–36, March 2009.
- [Pas10] R Paschotta. Encyclopedia of laser physics and technology. <http://www.rp-photonics.com>. Date Accessed: 16-4-2012, 2010.
- [Pat87] W D O Paterson. *Road deterioration and maintenance effects. Models for planning and management*. The John Hopkins University Press, Maryland, US, 1987.
- [Pet10] G Petrie. An introduction to the technology Mobile Mapping Systems. *Geoinformatics*, 13(1), 2010.
- [PHS03] S B Pattnaik, S Hallmark, and R Souleyrette. Collecting road inventory using lidar surface models. In *Map India Conference*, pages 1–7, New Delhi, India, 2003.
- [PJAA11] I Puente, H G Jorge, P Arias, and J Armesto. Land based mobile laser scanning systems: a review. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information*, volume XXXVIII, Calgary, Canada, 2011. ISPRS.

- [PME⁺09] C Pecharda, C Mariotta, R Eggen, S Toivonen, S Brown, and S Matena. Safety at the heart of road design. pages 1–10, <http://www.eranetroad.org>. Date Accessed:5-12-2011, 2009.
- [PT08] G Petrie and C K Toth. Introduction to laser ranging, profiling and scanning. In J Shan and C K Toth, editors, *Topographic laser ranging and scanning: principles and processing*. Taylor & Francis Group, Boca Raton, US, 2008.
- [PV06] S Pu and G Vosselman. Automatic extraction of building features from terrestrial laser scanning. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, Dresden, Germany, 2006. ISPRS.
- [RF06] F Remondino and C Fraser. Digital camera calibration methods: considerations and comparisons. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVI, pages 266–272, Dresden, Germany, 2006. ISPRS.
- [RFFP04] M R D Rocca, M Fiani, A Fortunato, and P Pistillo. Active contour model to detect linear features in satellite images. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004. ISPRS.
- [Rie09] Riegl. Data sheet Riegl VQ-250. Technical report, Riegl, Horn, Austria, 2009.

- [Rio11] R Rio. 3D laser scanning systems. <http://www.arcweb.com/market-studies/Pages/3d-Laser-Scanning.aspx>. Date Accessed: 13-4-2012, 2011.
- [RLW96] M D Reed, C E Landry, and K C Werther. The application of air and ground based laser mapping systems to transmission line corridor surveys. In *Position, Location and Navigation Symposium - PLANS '96*, pages 444–451, Atlanta, US, 1996. IEEE.
- [ROPV09] M Rutzinger, S J Oude Elberink, S Pu, and G Vosselman. Automatic extraction of vertical walls from mobile and airborne laser scanning data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, Paris, France, 2009. ISPRS.
- [RRP09] M Rutzinger, F Rottensteiner, and N Pfeifer. A comparison of evaluation techniques for building extraction from airborne laser scanning. *IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing*, 2(1):11–20, 2009.
- [RSPU10] P Rieger, N Studnicka, M Pfennigbauer, and A Ullrich. Advances in mobile laser scanning data acquisition. In *FIG Congress Facing Challenges - Building the Capacity*, pages 1–15, Sydney, Australia, 2010.
- [Saf09] SafetyNet. Road safety management. pages 1–41, <http://ec.europa.eu>. Date Accessed: 9-11-2011, 2009.

- [SBH09] F Samadzadegan, B Bigdeli, and M Hahn. Automatic road extraction from LIDAR data based on classifier fusion in urban area. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, Paris, France, May 2009. ISPRS.
- [SEs04] K P Schwarz and N El-sheimy. Mobile mapping systems-state of the art and future trends. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, pages 759–768, Istanbul, Turkey, 2004. ISPRS.
- [SGQ86] M W Sayers, T D Gillespie, and C A V Queiroz. The international road roughness experiment. Technical report, World Bank, Washington, US, 1986.
- [SHB08] M Sonka, V Hlavac, and R Boyle. *Image processing, analysis and machine vision*. Thomson, 2008.
- [SK98] M W Sayers and S M Karamihas. The little book of profiling. Technical report, University of Michigan Transportation Research Institute (UMTRI), Ann Arbor, US, 1998.
- [SMES⁺93] K P Schwarz, H E Martell, N El-Sheimy, R Li, M A Chapman, and D Cosandier. VISAT - A mobile highway survey system of high accuracy. In *Vehicle Navigation and Information Systems*, pages 476–481, Ottawa, Canada, 1993. IEEE.
- [SNG10] L Smadja, J Ninot, and T Gavrilovic. Road extraction and environment interpretation from LiDAR sensors. In *Interna-*

- tional Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, pages 281–286, Saint-Mande, France, 2010. ISPRS.
- [TAB⁺04] J Talaya, R Alamus, E Bosch, A Serra, W Kornus, and A Baron. Integration of a terrestrial laser scanner with GPS/IMU orientation sensors. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004.
- [TBA⁺04] J Talaya, E Bosch, R Alamus, A Serra, and A Baron. Geovan: the mobile mapping system from the ICC. In *International Symposium on Mobile Mapping Technology (MMT'04)*, Kunming, China, 2004.
- [TL07] C V Tao and J Li. *Advances in mobile mapping technology*. Taylor and Francis, London, 2007.
- [Top10] Topcon. IP-S2 3D mobile mapping system. Technical report, Livermore, US, 2010.
- [Tot09] C K Toth. R&D of mobile lidar mapping and future trends. In *ASPRS Annual Conference*, Baltimore, Maryland, US., 2009.
- [TPB08] C Toth, E Paska, and D Brzezinska. Using road pavement markings as ground control for LiDAR data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVII, pages 189–196, Beijing, China, 2008. ISPRS.

- [TTC07] Y H Tseng, K P Tang, and F C Chou. Surface reconstruction from LiDAR data with extended snake theory. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, volume 4679, pages 479–492, Ezhou, China, 2007. Springer.
- [TTM⁺79] J R Treat, N S Tumbas, S T McDonald, D Shinar, R D Hume, R E Mayer, R L Stansifer, and N J Castellan. Tri-Level study of the causes of traffic accidents. Technical report, Institute for Research in Public Safety, Indiana University, Bloomington, Indiana, US, 1979.
- [UNE08] UNECE. Transport review: road safety. Technical report, United Nations, New York and Geneva, 2008.
- [VL09] G Vosselman and Z Liang. Detection of curbstones in airborne laser scanning data. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, Paris, France, 2009. ISPRS.
- [Vos09] G Vosselman. Advanced point cloud processing. In D Fritsch, editor, *Photogrammetric Week*, pages 137–146, Heidelberg, Germany, 2009.
- [Wei12] E W Weisstein. Level set. <http://mathworld.wolfram.com/LevelSet.html>. Date Accessed: 25-7-2012, 2012.

- [WHO09] WHO. Global status report on road safety : time for action. Technical report, World Health Organisation, Geneva, Switzerland, 2009.
- [WHO10] WHO. Data systems - a road safety manual for decision makers and practitioners. Technical report, World Health Organisation, Geneva, Switzerland, 2010.
- [WHO11] WHO. The flyer on 2nd global status report on road safety. Technical report, World Health Organisation, Geneva, Switzerland, 2011.
- [Wil07] M Williams. Google street view - behind the scenes. <http://maps.google.ie/intl/en.ie/help/maps/streetview/behind-the-scenes.html>. Date Accessed: 11-1-2012, 2007.
- [XP97] C Xu and J L Prince. Gradient vector flow : a new external force for snakes. In *IEEE Conference. on Computer Vision Pattern Recognition (CVPR)*, volume 2, pages 66–71, Los Alamitos, California, US, 1997. IEEE.
- [XP98] C Xu and J L Prince. Snakes, shapes, and gradient vector flow. *IEEE transactions on image processing*, 7(3):359–369, January 1998.
- [XYP00] C Xu, A Yezzi, and J L Prince. On the relationship between parametric and geometric active contours. In *34th Annual Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 483–489, Pacific Grove, California, US, 2000. IEEE.

- [YAL⁺10] K S Yen, K Akin, A Lofton, B Ravani, and T A Lasky. Using mobile laser scanning to produce digital terrain models of pavement surfaces. Technical report, California Department of Transportation, California, US, 2010.
- [YB04] J Youn and J S Bethel. Adaptive snakes for urban road extraction. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXV, Istanbul, Turkey, 2004. ISPRS.
- [YBKY05] Y Yagi, M Brady, Y Kawasaki, and M Yachida. Active contour road model for road tracking and 3D road shape reconstruction. *Electronics and Communications in Japan, Part3*, 88(9):42–52, September 2005.
- [YC09] J Yoon and C D Crane. Evaluation of terrain using LADAR data in urban environment for autonomous vehicles and its application in the DARPA urban challenge. In *ICROS-SICE International Joint Conference*, pages 641–646, Fukuoka, Japan, 2009.
- [YZC⁺08] X Yuan, C X Zhao, Y F Cai, H Zhang, and D B Chen. Road-surface Abstraction using Ladar Sensing. In *10th International Conference on Control, Automation, Robotics and Vision*, pages 1097–1102, Hanoi, Vietnam, 2008.
- [ZF05] K Zhang and C Frey. Road grade estimation for on-road vehicle emissions modeling using lidar data. In *Annual Meeting of the Air & Waste Management Association*, number x, pages 1–22, Minneapolis, US, 2005.

- [Zha10] W Zhang. LIDAR based road and road-edge detection. In *Intelligent Vehicles Symposium*, pages 845–848, San Diego, US, 2010. IEEE.
- [ZR04] A M Zhang and R A Russell. Surface roughness measurement for outdoor mobile robotic applications. In *Australasian conference on robotics and automation (ACRA)*, pages 1–6, Canberra, Australia, 2004.
- [ZS03] H Zhao and R Shibasaki. A vehicle-borne urban 3-D acquisition system using single-row laser range scanners. *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics*, 33(4):658–666, January 2003.
- [ZXZ08] H Zhang, Z Xiao, and Q Zhou. Research on road extraction semi-automatically from high resolution remote sensing images. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVII, pages 535–538, Beijing, China, 2008. ISPRS.