POSITION ESTIMATION OF MOBILE LASER SCANNER USING AERIAL IMAGERY

Syed Zille Hussnain

POSITION ESTIMATION OF MOBILE LASER SCANNER USING AERIAL IMAGERY

DISSERTATION

to obtain the degree of doctor at the Universiteit Twente, on the authority of the rector magnificus, prof.dr. T.T.M. Palstra, on account of the decision of the graduation committee to be publicly defended on Friday 24 January 2020 at 12.45 uur

by

Syed Zille Hussnain born on August 02, 1982 in Gujrat, Pakistan This dissertation has been approved by:

Supervisor Prof. dr. M.G. Vosselman

co-supervisor Dr. S.J. Oude Elberink

ISBN: 978-90-365-4935-6 DOI: 10.3990/1.9789036549356 ITC dissertation number: 375

Printed by: ITC Printing Department Address: ITC, P.O. Box 217, 7500 AE Enschede, The Netherlands

© 2020 Zille Hussnain, The Netherlands. All rights reserved. No parts of this thesis may be reproduced, stored in a retrieval system or transmitted in any form or by any means without permission of the author. Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd, in enige vorm of op enige wijze, zonder voorafgaande schriftelijke toestemming van de auteur.



Graduation committee:

Chairman/secretary Prof.dr.ir. A. Veldkamp

Supervisor Prof.dr.ir. M.G. Vosselman

Co-supervisor Dr.ir. S.J. Oude Elberink

Members

Prof.dr. R. Zurita Milla Prof.dr.ir. S. Stramigioli Prof.dr.ing. M. Gerke Prof.dsc.tech.res. H. Kaartinen

University of Twente/ITC University of Twente/EEMCS TU Braunschweig, Germany Finnish Geospatial Research Institute

Acknowledgements

I would like to express my heartiest gratitude to my supervisor prof. George Vosselman for his keen interest and involvement in this research project. His direct assistance improved my ability to analyse what matters the most and stimulated the development of critical thinking. I am as well grateful to his help in the modelling of the trajectory adjustment method. His guidance and help throughout this research project were the keys to accomplish important goals.

I would like to present my special thanks to my co-supervisor dr. Sander Oude Elberink for his support and the fruitful discussions throughout the duration of my research work. His invaluable advice helped me to develop the ability to explain difficult concepts in easy and simple ways. He always emphasized the need for the smooth transfer of knowledge to the reader, which I think is one of the most important parts in scientific literature writing.

I wish to present my sincere appreciation to prof. Markus Gerke for his help and encouragement especially at the beginning of this project. His suggestions and discussions about implementing the photogrammetric techniques were useful, while most of his suggestions would work straight out of the box.

My sincere gratitude to dr. Phillipp Fanta-Jende for helping me to familiarize with the terms related to the study of geoinformatics. I feel lucky that together we would discuss many technical topics for hours, which helped us to clarify many issues. Apart from the research, he is a great friend as well.

I have enjoyed the company of many amazing friends and colleagues at ITC. Their help was very important for me, and I would like to wholeheartedly thank them for their continuous support.

I am greatly thankful to my family for providing me with the support I needed to complete my study. Especially, I would like to thank my wife Tehreem Ali, who took well care of me at home by preparing very good food and supported me so I can spend more time at my studies. Her help was essential for me in achieving important milestones. During my character-building, my father showed me how to consistently work hard, while my mother stood as a symbol of great patience. I believe these learnings were the keys which prepared me to achieve difficult goals. Finally, I would also like to thank all my other family members and friends for their uninterrupted and unconditional support.

Table of Contents

Acknowledgements	i
List of figures	v
List of tablesix	х
1 – Introduction	1
1.1 Background	2
1.2 Manual correction	5
1.3 Automatic direct 3D/2D registration	7
1.4 Automatic 2D feature matching12	2
1.5 Orientation update or trajectory adjustment15	5
1.6 Our research aim18	8
1.7 Structure of the thesis20	0
1.8 Context and contributions22	2
2 – Low-level Tie Feature Extraction of Mobile Mapping Data (MLS/Images))
and Aerial Imagery29	9
2.1 Introduction	1
2.2 Project overview	1
2.3 Related work	2
2.3.1 Previous approaches	2
2.3.2 Low-Level Feature Extraction	2
2.4 Low-level feature extraction	3
2.4.1 Mobile Laser Scanning Images	4
2.4.2 Mobile Mapping Images35	5
2.5 Results	5
2.5.1 Feature detection	7
2.5.2 Feature descriptor matching	9
2.6 Discussion	9
2.6.1 Conclusions49	9
2.6.2 Outlook	0
3 – Automatic Feature Detection, Description and Matching from Mobile	
Laser Scanning Data and Aerial Imagery53	3
3.1 Introduction55	5
3.2 Feature extraction57	7
3.2.1 Selection of test area57	7
3.2.2 Pre-processing58	8
3.2.3 Feature detection60	0
3.2.4 Feature description63	3
3.3 Descriptor matching65	5
3.4 Outliers filtering	5
3.5 Results	8
3.5.1 Feature detection68	8
3.5.2 Feature matching70	D
3.5.3 Discussions	2

	Evaluation of estimated shift	73
3.6 Co	nclusion	74
4 – Auto	omatic Extraction of Accurate 3D Tie Points for Trajectory	
Adjustmen	t of Mobile Laser Scanners using Aerial Imagery	77
4.1 Int	roduction	79
4.2 Re	lated Work	81
4.3 De	eveloped method	
4.3.1	Pre-processing mobile laser scanning point clouds	
	and aerial images	87
4.3.2	2D-2D registration	91
4.3.3	A2A 3D tie points extraction	92
4.3.4	A2P 3D points extraction	93
4.3.5	3D-3D correspondence search	93
4.4 Im	plementation, Experiments and Results	94
4.4.1	Mobile Laser Scanning Point Cloud datasets	95
4.4.2	Aerial nadir imagery	97
4.4.3	Pre-processing results of datasets	97
4.4.4	2D-2D correspondence between image patches	101
4.4.5	A2A 3D tie points	107
4.4.6	A2P 3D tie points	110
4.4.7	3D-3D correspondence	112
4.4.8	Reasons and implications for areas without 3D-3D	
	correspondence	120
4.5 Co	correspondencenclusions	120 124
4.5 Co 5 – Enh	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin	120 124 g Aerial
4.5 Co 5 – Enh Images	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin	120 124 g Aerial 131
4.5 Co 5 – Enh Images 5.1 Int	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin	120 124 g Aerial 131 133
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work	120 124 g Aerial 131 133 134
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-3	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment	120 124 g Aerial 131 133 134 136
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-t 5.3.1	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization	120 124 g Aerial 131 133 134 136 137
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.1 5.3.2	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation	120 124 g Aerial 131 133 134 136 137 137
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.3	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation	120 124 g Aerial 131 133 134 136 137 137 140
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-5 5.3.1 5.3.2 5.3.2 5.3.3 5.3.4	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation	120 124 g Aerial 131 133 134 136 137 137 140 141
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-s 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Soft constraint observation	120 124 g Aerial 131 133 134 136 137 137 140 141 143
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-9 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation	120 124 g Aerial 131 133 134 136 137 137 140 141 143 144
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.6 5.3.7	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation Trajectory update	120 124 g Aerial 131 133 134 136 137 140 141 143 144 145
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Acceleration observation Soft constraint observation Fixed pose observation Trajectory update	120 124 g Aerial 131 133 134 136 137 137 140 141 144 145 146
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-5 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation Trajectory update guantitative analysis for the tie point observation	120 124 g Aerial 131 133 134 136 137 137 140 141 144 144 144 146 148
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-5 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1 5.4.2	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation Trajectory update guantitative analysis for the tie point observation Qualitative analysis for the tie point observation	120 124 g Aerial 131 133 134 136 137 137 140 141 143 144 145 146 148 149
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1 5.4.2 5.4.3	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Acceleration observation Soft constraint observation Fixed pose observation Fixed pose observation Trajectory update Sugn of the experiments Quantitative analysis for the tie point observation Qualitative analysis for the tie point observation Assessment of trajectory constructed only by IMU	120 124 g Aerial 131 133 134 136 137 137 140 141 143 144 145 146 148 149
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1 5.4.2 5.4.3	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation Trajectory update guantitative analysis for the tie point observation Qualitative analysis for the tie point observation Assessment of trajectory constructed only by IMU and soft constraints observations	120 124 g Aerial 131 133 134 136 137 140 141 143 144 144 145 146 149 149
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B- 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1 5.4.2 5.4.3 5.4.4	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Acceleration observation Soft constraint observation Fixed pose observation Trajectory update Sign of the experiments Quantitative analysis for the tie point observation Qualitative analysis for the tie point observation Assessment of trajectory constructed only by IMU and soft constraints observations Impact of soft constraints	120 124 g Aerial 131 133 134 136 137 137 140 141 144 144 145 146 148 149 149 150
4.5 Co 5 – Enh Images 5.1 Int 5.2 Re 5.3 B-5 5.3.1 5.3.2 5.3.3 5.3.4 5.3.5 5.3.6 5.3.7 5.4 De 5.4.1 5.4.2 5.4.3 5.4.4 5.5 Re	correspondence nclusions anced Trajectory Estimation of Mobile Laser Scanners Usin troduction lated work spline based 6dof trajectory adjustment B-spline order and knot interval optimization 3D-3D correspondence observation Acceleration observation Angular velocity observation Soft constraint observation Fixed pose observation Trajectory update sign of the experiments Quantitative analysis for the tie point observation Assessment of trajectory constructed only by IMU and soft constraints observations Impact of soft constraints	120 124 g Aerial 131 133 134 136 137 137 140 141 143 144 145 146 148 149 150 150

5.5.2	Optimal knot interval and B-spline order	151
5.5.3	Regeneration of point cloud and A2P points	153
5.5.4	Experiments and evaluation	156
5.6 Co	nclusions	163
6 – Synt	hesis	169
6.1 Co	nclusions per objective	
6.2 Ret	flections and outlook	178
Summary.		
Samenvatt	ing	

List of figures

Figure 1.1: GNSS signals blockage and reflection in urban canyons.2 Figure 1.2: a) Tall buildings in the urban area promotes poor satellite Geometric Dilution of Precision, b) while a road without tall buildings is free Figure 1.3: (Left), acquisition of GCPs outside of an urban canyon, (right) manual selection of GCPs correspondences in MLS point cloud by a human Figure 1.4: Workflow for manual correction of MLS point cloud.6 Figure 1.5: Workflow of fully automatic correction of MLS point cloud using automatic extraction of tie points and trajectory adjustment......7 Figure 1.6: (Left) green 3D line segments extracted and projected onto the aerial image plane. (Right) image shows the lines after the pose estimation. Red colour lines are the line segments extracted from the aerial image, a figure Figure 1.7: (Top left) 2D orthogonal corners detected in DSM, (top right) 2D orthogonal corners detected in an oblique image, (bottom middle) matches after Hough transform, green corners are projected from DSM, blue are image's original corner and red lines are correspondences, a figure from Ding et al. (2008)......9 Figure 1.8: (a) point cloud before correction, (b) point cloud after correction, notice that the point belongs to the same structure are aligned, figure from Harrison et al. (2008).....11 Figure 1.9: (Left) 3D point clouds from two sensors before alignment, (Right) 3D point clouds after alignment, figure from Levinson et al. (2007).11 Figure 1.10: Comparison of two examples of road marking in the point cloud and aerial images.....14 Figure 1.11: The subpixel features detected on the corners of a zebra-crossing. a) A perspective projection of a point cloud onto an image plane. b) The corresponding aerial image patch.15 Figure 1.12: Yellow observations are global constraints extracted from the aerial images, where other observations are local constraints acquired with SLAM, figure from Kümmerle et al. (2011).....17 Figure 1.14: All 7 partners involved in this NWO project and their main contributions in terms of input datasets; MLS point cloud and aerial imagery. Figure 2.2: Mobile mapping panoramic image in equirectangular projection 35 Figure 2.3: Panoramic image projected onto an artificial ground plane.36 Figure 2.4: Four subsets of a typical urban scene (coloured tiles from scene 1

Figure 2.5: SIFT keypoints detected in aerial image (left), panoramic image Figure 2.6: KAZE keypoints detected in aerial image (left), panoramic image Figure 2.7 AKAZE keypoints detected in aerial image (left), panoramic image Figure 2.8 Förstner keypoints detected in aerial image (left), panoramic image Figure 2.9 Comparison of SIFT (top) and KAZE (bottom) in 4th run on the 1st Figure 2.10 Matching results of AKAZE (top) and KAZE (bottom) in 4th run on Figure 2.11: Matched LATCH keypoints in the first scene and first iteration.45 Figure 2.12 Comparison of matching results of AKAZE (top), KAZE (centre) and SIFT (bottom) in 3rd run of the 1st scene46 Figure 2.13 Matched SIFT keypoints in the second scene and first iteration (correct correspondence is light purple)......47 Figure 2.14 Matched SIFT keypoints in the second scene and second iteration Figure 2.15 Matched KAZE keypoints in the second scene and third iteration Figure 2.16 Comparison of matching results of AKAZE (top), KAZE (centre) and Figure 3.1: Workflow diagram of the developed method for MLSPC to aerial image registration......57 Figure 3.2: Visualization of the MLSPC of the test area and cropped tiles....58 Figure 3.3: Visualization of the aerial image of the test area and cropped tiles. Figure 3.4: Point cloud patch on the left is converted to an orthoimage on the right......59 Figure 3.6: Example of adaptive Harris corner keypoint detection of a road mark, multiple keypoints (red dots) are detected over two observable corners. Figure 3.7: Adaptive Harris keypoint detection of a whole tile, with the total Figure 3.8: Adaptive approach for different aerial images of the same scene. Though look similar, the underlying image differences can be realized by Figure 3.9: The computation of a particular LATCH descriptor used in this Figure 3.10: Illustration of the matched descriptors of image 1 and image 2.

```
Figure 3.11: Hamming distance based descriptor matching of an example tile
with five nearest neighbours (k=5)......66
Figure 3.12: In image 1 and Image 2, (P2 \Leftrightarrow P2') and (P4 \Leftrightarrow P4') are correct
matches with respect to the seed point P1 and P2', whereas (P3 \Leftrightarrow P3') is an
outlier due to the difference between \theta_3 and \theta_3', though d_3 and d_3' are equal.
Figure 3.13: Correspondences computed without \theta n' constraint. Blue arrows
Figure 3.14: Homography based outliers filtering......68
Figure 3.15: Adaptive Harris corner feature detection from aerial (left) and
MLSPC (right) image patch......69
Figure 3.16: Different visualizations of the matched features......71
Figure 3.17: Manual selection of a road mark corner point for manual
evaluation of inaccurate point cloud. .....73
Figure 3.18: Overlap of PDFs of translation error in X and Y coordinates, error
estimated by the Developed Method (DM) and Manually Measured (MM), tile 4
(top) and tile 8 (bottom)......74
Figure 4.2: Workflow automatic 3D tie point extraction.......86
Figure 4.4: A 3D point cloud tile and its projection to multiple perspective
Figure 4.5: Distributions of undesired points and road points along a pixel
Figure 4.6: Pixel and subpixel-level correspondence of a same corner feature.
Figure 4.7: Extracted missing links retrieved using multiview matching. Dotted
lines are missing links while the solid colour lines are well-established
correspondences.......94
Figure 4.8: MLS_DATA-I trajectory (a). MLS_DATA-II trajectory (b), both
Figure 4.9: The arrangement of aerial image extents over test area. .......97
Figure 4.11: Left, six MLS images generated from tile 18 of MLS_DATA-I. Right,
projections of same point cloud on three different perspective planes. .... 100
Figure 4.12: AIPs for the tile 18 of MLS_DATA-I......101
Figure 4.13: For MLS image-to-AIP matching, yellow lines are
correspondences, keypoint in aerial images are represented by green +'
symbol and keypoints in point cloud images are red circles. Notice that the
pixel-level corresponding keypoints are always in the middle of the pixel while
the subpixel-level keypoints are not necessarily in the middle of the pixel. It is
Figure 4.14: For AIP-to-AIP matching, yellow lines are correspondences,
keypoints in right patches are represented by green '+' symbol and keypoints
```

in left patches are red circles. Notice that the pixel-level corresponding keypoints are always in the middle of the pixel while the subpixel-level keypoints are not necessarily in the middle of the pixel. It is evident by Figure 4.15: Top, few matches due to shifted and repainted road marks. Figure 4.16: No matches due to the alternating areas occluded from the laser Figure 4.20: A2A tie points for tile 18 from MLS DATA-I, Amersfoort / RD New. Figure 4.21: Matches between MLS images and AIPs related to tile 18 of Figure 4.22: All obtained 3D-3D correspondences for MLS DATA-I and MLS_DATA-II, a) and b) respectively. A2A tie points are shown as blue dots. The A2P points come underneath the A2A tie points due to the level of scale. Figure 4.23: The number of 3D-3D correspondences obtained for each point cloud tile of MLS_DATA-I, the results are divided into three sub-plots, a), b) Figure 4.24: The number of 3D-3D correspondences obtained for each point cloud tile of MLS_DATA-II, the results are divided into two subplots, a) and b), showing results of 44 tiles in ascending order......116 Figure 4.25: 3D-3D correspondences of the tile 2 of MLS_DATA-II. a) The whole aerial patch overlaid by the original 3D point cloud points. b) The close Figure 4.26: Probability Density Functions (PDF) of the ΔX (m), ΔY (m) and ΔZ Figure 4.27: Longest consecutive areas without 3D-3D correspondence. The AIPs and MLS images from the MLS_DATA-I are in (a) and (b) respectively. The AIPs and MLS images from the MLS_DATA-II are in (c) and (d) Figure 4.28: Tiles failed to produce 3D-3D correspondence. (a) AIP-to-AIP Figure 5.3: 3D-3D correspondence observation based on A2A and A2P 3D tie Figure 5.4: The relationship between positions and direction of the car. ... 143 Figure 5.5: Trajectory-I dataset.147

```
Figure 5.7: Illustration of trajectory segments used to assess the reliability of
Figure.5.8: Example of 3D A2A tie points (blue dots) and A2P tie points (red
dots) along with the Kalman filtering result (red curve) and the trajectory after
the adjustment with our method (blue curve), Amersfoort / RD New coordinate
Figure 5.9: The evaluation of regenerated A2P points using checkpoint A47.
Figure 5.10: The evaluation of regenerated point cloud using checkpoint A50.
Figure 5.11: Plot of the Trajectory-IA used for experiments 1, 2 and 3. Here
red '*' are the locations of the checkpoints, Amersfoort / RD New coordinate
Figure 5.12: Plot of the Trajectory-IB used for experiments 4. Here red '*' are
the locations of the checkpoints, Amersfoort / RD New coordinate system.157
Figure 5.13: Plot of the Trajectory-II, red '*' are the locations of the
checkpoints, Amersfoort / RD New coordinate system.......161
```

List of tables

Table 2-1 Number of combined keypoints over all subsets per detection method
Table 2-2 Matching results of scene 1 between aerial and MLS orthoimage of the 1st and 2nd iteration 41
Table 2-3 Matching results of scene 1 between aerial and MLS orthoimage of the 3rd and 4th iteration 41
Table 2-4 Matching results of scene 2 between aerial and MLS orthoimage of the 1st and 2nd iteration. 43
Table 2-5 Matching results of scene 2 between aerial and MLS orthoimage of the 3rd and 4th iteration 43
Table 2-6 Matching results of scene 1 between aerial and panoramic image of the 1st and 2nd iteration 45
Table 2-7 Matching results of scene 1 between aerial and panoramic image of the 3rd and 4th iteration
Table 2-8 Matching results of scene 2 between aerial and panoramic image of the 1st and 2nd iteration 47
Table 2-9 Matching results of scene 2 between aerial and panoramic image of the 3rd and 4th iteration. 48
Table 3-1: Results of the adaptive Harris keypoint detection 70 Table 3-2: Number of Matched keypoints and the normal probability function
parameters of translation error between the point cloud and aerial images (all units are in meters)72

Table 4-1: AIP-to-AIP matches for MLS_DATA-I tile 18 and MLS_DATA-II tile 2. The number of corresponding keypoints before and after the mapping of Table 4-2: MLS image-to-AIP matches of MLS_DATA-I tile 18 and MLS_DATA-II tile 2, number and percentage of corresponding keypoints before and after Table 4-3: Comparison of the number of tie points and obtained 3D-3D correspondences......116 Table 4-4: Differentiation of tiles based on the number of tie points and the Table 5-1: Categorization of experiments and their related observations. . 146 Table 5-2: Results for B-spline fitting to Trajectory-I dataset using combinations of knot interval and curve order.152 Table 5-3: Results for B-spline fitting to Trajectory-II dataset using Table 5-5: The example of residuals measured using checkpoints on the point cloud regenerated from the trajectory enhanced using all observations, these checkpoints can also be located in Figure 5.12 of Trajectory-IB...... 155 Table 5-6: Results of the experiments categorised in Table 1 and conducted on Trajectory-I.....158

х

1.1 Background

Over the past 10 years, interest in the applications of Mobile Laser Scanning (MLS) point clouds has been increasing continuously. An MLS point cloud can provide accurate geometrical information of urban structures at the comfort of the office environment. The main advantage of MLS is the collection of accurate and dense 3D information in less time compared to terrestrial laser scanning, while advancements in the MLS technology have been making data collection cost-effective. However, the benefit of the time saving by MLS technique comes at the cost of an error-prone data acquisition procedure. As the 3D points are measured in bulk by laser scanners mounted on a moving car, the measurements are georeferenced by the Global Navigation Satellite System (GNSS) in combination with Inertial Navigation System (INS). With this setup, the acquired point cloud can achieve sufficient accuracy in areas without GNSS signal disturbances. However, blockage and reflection of GNSS signals in urban areas lead to poor positioning of the MLS platform (Gu et al. 2016; Hsu 2017). Depiction of such interference of GNSS signals by urban structure is illustrated in Figure 1.1. Another crucial issue is the elevation angle of satellites in the direct line of sight with the GNSS receiver. A constellation of satellites making a narrow-angle with a GNSS receiver causes poor Geometric Dilution of Precision (GDOP) as depicted in Figure 1.2.



Figure 1.1: GNSS signals blockage and reflection in urban canyons.

The poor GDOP can also be the result of masking reflected and unreliable satellite signals to avoid the multipath effect. Therefore, even when the

reflected satellite signals are eliminated (Hsu 2018; Lesouple et al. 2018), the estimated positioning is still not reliable in urban areas. In an ideal condition at the outside of urban canyons, without any GNSS signal outage and multipath effect, the state-of-the-art Mobile Mapping (MM) platforms can reach 2-3 cm accuracy (Haala et al. 2008; Kaartinen et al. 2012). In a GNSS troubled area, the GNSS accuracy could quickly get worse than 50 cm during a complete outage of GNSS signals Kukko (2013). Wang et al. (2016) fused GPS, Inertial Measurement Unit (IMU) and dead-reckoning data by grid constraints. Mostly the accuracy remained above half a meter with RMSE X=0.79 m, Y=0.32 m, Z=0.86 m. A localization setup developed to fuse the GNSS/IMU/Distance Measurement Instrument (DMI)/lidar sensors information proposed by Meng et al. (2017) exhibited the error of 1 meter at some occasions only for a mapping distance of 140 meters.





Figure 1.2: a) Tall buildings in the urban area promotes poor satellite Geometric Dilution of Precision, b) while a road without tall buildings is free from this problem.

The inaccurate positioning of the MLS platforms can be refined by the Kalman filtering (Ding et al. 2007; Mohamed et al. 1999; Qi et al. 2002; Zhao 2011). However, according to the literature, the Kalman filtering based approaches cannot handle the GNSS signal outage longer than 10 seconds and lead to error

propagation in positioning (Chiang et al. 2008; Gao et al. 2006; Taylor et al. 2006). Xiong et al. (2018) proposed a strong tracking filtering approach but the achieved accuracy was near or above meters. Regardless of various efforts to filter the inaccurate positions, the problem remains unsolved for the urban canyons. Therefore, during the mapping, in the presence of no other accurate positioning reference, the data is acquired as incorporated with erroneous GNSS positions, leaving the liability of point cloud correction on the postprocessing step. Later in the post-processing, the point cloud correction software first try to correct the trajectory by automatic registration of overlapping areas, consecutive scan lines matching or even matching with a prior accurate point cloud of aerial or terrestrial origins (Hsu et al. 2016). However, the point cloud self-registration based correction has a similar problem as faced by the Simultaneous Localization and Mapping (SLAM) approach, the error remains above 2 to 3 meters (Moosmann et al. 2011). The main problem still remains the propagation of error, but gets smaller compared to when no self-registration is done. These techniques can increase the relative accuracy and absolute accuracy to some extent but need data that has multiple scans/passes of the same area. However, due to the high cost of mobile laser scanning per hour, it is highly desirable to scan an area once and as quickly as possible.

Some techniques use an already acquired reference (accurate) point cloud of the same area for the correction of the newly acquired point cloud. The registration with an accurate reference point cloud is also problematic because the reference point cloud was already corrected perhaps using manual correction. Moreover, Iterative Closest Point (ICP) and similar registration techniques also have a limitation. These techniques consider that a solution is achieved when a local optimum is reached. The registration between the MLS point cloud and aerial lidar point cloud could be an interesting possibility because the aerial point cloud can achieve a desirable accuracy without correction. Unfortunately, aerial point clouds have low point density especially on the building facades. Unbalanced coverage and unequal spatial distribution of points between MLS point cloud and aerial lidar point cloud makes the ICP based registration even more unstable. The point cloud generated by aerial image dense matching will also have insufficient quality for the registration of the MLS point cloud.

With no automatic solution for decimetre level correction (the required accuracy level), the correction of the MLS point cloud highly depends on the Ground Control Points (GCPs), which are acquired manually in an independent survey. This correction procedure hinders the data acquisition of up-to-date urban structures. Still, accuracy cannot be improved in areas where GCPs themselves are either inaccurate or impossible to measure because of the unavailability of any proper landmarks. Moreover, the accuracy of the GCPs

can be similarly unreliable in urban canyons. As a consequence, many users, e.g. the city planning institutions, are forced to work with outdated and expensive datasets.

In the following sections, we discuss the currently available manual and automatic MLS data correction procedures in detail and show that how their inability to achieve decimetre accuracy necessitated the development of our automatic correction procedure, which is discussed afterwards.

1.2 Manual correction

For the manual correction of MLS point clouds, two main further postprocessing steps are needed. To minimize the destined manual effort, the first post-processing step tries to correct the MLS point cloud by available but inaccurate automatic means, similar to the technique proposed by (Ding et al. 2007; Levinson et al. 2007; Zhao 2011). This step performs registration between overlapping point cloud patches and filters inaccurate position of MLS platform in the trajectory. Normally, this step is performed without any external reference. Therefore, the inaccurate data can only be corrected if there are enough data overlapping or discrepancies are found between IMU and GNSS estimations. The sensors discrepancies become apparent because IMU estimations are reliable during small intervals, whereas the inaccurate GNSS readings may show uncertainty among consecutive measurements. However, utilizing these techniques, the final accuracy remains near meter as expressed by Chiang et al. (2008), especially during long-term GNSS outages in urban canyons. These automatic techniques are further discussed in the next section.

Second post-processing step involves manually acquired GCPs for manual data correction. Although some latest software provide assistance to help human operator in handpicking the exact features by automatic detection of landmarks corresponding to GCPs, the manual intervention of a human operator is still needed for the final decision. Firstly, the ground control points are collected by surveying the target area as shown in Figure *1.3* (left). Secondly, the manual selection of the corresponding 3D points in the point cloud is performed carefully as shown in Figure *1.3* (right).



Figure 1.3: (Left), acquisition of GCPs outside of an urban canyon, (right) manual selection of GCPs correspondences in MLS point cloud by a human operator.

Overall, this complete step is very labour intensive and hinders the automatic acquisition of high-quality MLS point cloud. Moreover, the GCP measurements could still be uncertain and the handpicking of the GCPs in the MLS data sets could as well be imprecise. Furthermore, to increase the reliability and validity of GCPs' accuracy, the same GCPs are acquired multiple times, which makes the correction procedure even more expensive. If undetected inaccurate GCPs get mixed in the final reference set, then there is no third reference to verify the inconsistent GCPs. The main issue is that accurate GCPs can only be acquired where there are no tall buildings. Nevertheless, after painstaking manual efforts and corrections, the second step could improve the MLS point cloud accuracy to the desired level while keeping final product costly and error prone. The workflow of the manual post-processing for the correction of the MLS data is presented in Figure *1.4*.



Figure 1.4: Workflow for manual correction of MLS point cloud.

Because of all the problems mentioned earlier, it is desirable to have an automatic procedure which can perform the correction while improving the accuracy at low cost and in less time. In this project, we will use aerial imagery as the source for georeferencing MLS point clouds. The exterior orientation of aerial images is known accurately from GNSS-supported aerial triangulation.

High resolution aerial imagery can therefore provide precise reference information. Therefore, the main goal of our research project is twofold; first is to determine the correspondences between well-oriented aerial photographs and the MLS point cloud by automatic matching between the point cloud and aerial images. The determined correspondences then lead to the computation of 3D tie points. The second sub-goal is to correct the MLS dataset better or equal to the results achieved by the manual correction. The workflow of such type of automatic correction procedure is presented in Figure 1.5.



Figure 1.5: Workflow of fully automatic correction of MLS point cloud using automatic extraction of tie points and trajectory adjustment.

1.3 Automatic direct 3D/2D registration

The registration between the inaccurate dataset (point cloud) and reference dataset (aerial images) can determine the correspondences necessary for the adjustment and correction of the inaccurate dataset. In this section, we will discuss techniques which can register the inaccurate data with reference data by directly minimizing the offset error. It is a 3D-to-2D or 2D-to-3D registration problem with a lot of available literature. However, we only focus on studies which are useful and related to the application at hand. One of the favoured approaches for the 2D/3D registration is to focus on the structural information. Even with the varying perspectives and dissimilar sensors, the geometry of objects is always perceived and preserved in both point cloud and images. For texture mapping application, Frueh et al. (2004) preferred to extract structural lines from 3D models of the city for the registration with the edges extracted from oblique images as shown in Figure 1.6. The GNSS and INS based exterior orientation of the oblique image was an initial guess for registration. The lines in the point cloud were based on depth map edge points while the oblique image edges were detected by canny edge detection, both 2D and 3D line segments were estimated by the recursive endpoint subdivision algorithm Lowe (1987). The 3D model's lines were projected to the oblique image plane and instead of one line, three connected lines were grouped to rate a particular

camera pose similar to Lee et al. (2002). Frueh et al. (2004) considered a point cloud as the reference dataset. It is the other way around in our project.



Figure 1.6: (Left) green 3D line segments extracted and projected onto the aerial image plane. (Right) image shows the lines after the pose estimation. Red colour lines are the line segments extracted from the aerial image, a figure from Frueh et al. (2004).

Ding et al. (2008) proposed a method to extract 2D orthogonal corners from digital surface models (DSM) obtained from an ALS point cloud. The 2D orthogonal corners were constructed from the oblique images which in turn used for vanishing points detection. Then the corners from the DSM were projected using the exterior orientation of the oblique camera as shown in Figure 1.7. For the matching, the distance between the corners and the similarity of the corner description criteria was used. Wrong matches were filtered out using Hough transform with generalized M-estimator sample consensus. For the registration, the camera recovery method was used on the corresponding corners Lowe (1987). However, the DSM used was already constructed by merging the aerial and ground laser scanning point cloud views Frueh et al. (2003). That's why the obtained 3D model was already quite accurate on the boundaries of the building. Due to this reason, it was possible to extract features confidently to match them directly with the oblique image. Ding et al. (2008) assumed that the more reliable corner features were the building top edges. Their technique detected too many line segments as features in oblique images where most of them had no viable correspondence in the point cloud. The line segment detection from noisy and occluded point cloud images can also produce many false and unmatchable features. The

validation and accuracy of the camera poses were assumed accurate on the basis of visual inspection.



Figure 1.7: (Top left) 2D orthogonal corners detected in DSM, (top right) 2D orthogonal corners detected in an oblique image, (bottom middle) matches after Hough transform, green corners are projected from DSM, blue are image's original corner and red lines are correspondences, a figure from Ding et al. (2008).

Fruh et al. (2001) proposed a method for vehicle position estimation by registration of a 3D model derived from laser scans with 2D aerial images and roadmaps. However, the accuracy of this method was limited to the width of the road which was being scanned. The matching between the point cloud and aerial images was realized by the maximum cross-correlation, while the positioning was maintained by Monte Carlo Localization (MCL). The mounting point of the laser scanner was increased in height to avoid the occlusion by road traffic. This method was not designed to handle the occlusion caused by roadside fixed objects e.g. trees. It was assumed that the features like building edges could be extracted clearly from the MLS point cloud. Moreover, the aerial images were merged together with roadmaps to assist localization. The accuracy determined to be the width of the road also does not meet the requirement of our application. Chen et al. (2009) developed a method to automatically detect geo-referenced lane markings from MLS point cloud. This work exploited the point cloud reflectance information to detect road markings. The line segments of the road markings were acquired by the Hough transform. However, the geometrical accuracy of the detected line segment was not discussed and this technique was tested in the areas without tall buildings. The

images were acquired from the same mobile platform and the colour information from the images was assigned to the corresponding detected road signs based on the well-known relative orientation between camera and laser scanner but a fusion of the datasets was not discussed.

A method proposed by Wang et al. (2009) performs registration of aerial lidar data to oblique images. Instead of using the camera pose as proposed by Ding et al. (2008), the orientation of a laser scanner estimated by GNSS/INS was used. A feature constructed from 3-connected line segments was proposed and extracted from ALS point cloud and oblique images. The authors claimed that the 3-connected line segments increase the number of matches compared to using single line segments matching, discussed earlier. For outlier match removal, first-level processing of RANSAC removed global outlier matches, while the second level of RANSAC removed the local-level outliers. For edge detection in the aerial images, a curve was fitted on the detected edges and then a breakpoint algorithm was used to obtain line segments. This type of edge detection can introduce small errors in the detection of original edges which is not suitable for accuracy requirements we needed for our application.

A possibility for the improvement of MLS platform localization is to perform distortion removal in the point cloud, which in turn can also provide the correction to the point cloud and position. A point cloud can be distorted if there is continuously changing error in the position of moving lidar sensor. Harrison et al. (2008) implemented an approach to use an LMS200 laser scanner with an odometer. The approach mainly relies on vertical objects in the scene. The Bayesian filtering technique was used to estimate an accurate position. The point cloud before and after the distortion removal is shown in Figure 1.8. However, only the data from a single laser scanner was used, whereas, in our research, we will utilize the data from two laser scanners. Another disadvantage is that they mainly rely on the fact that the single scanner will always scan the same surface more than once, which is not necessary for the dataset at hand. Bosse et al. (2009) described a scanmatching method based on iterative closest points (ICP) to recover an accurate trajectory. Like the previous technique, this technique is also based on another single laser range sensor SICK LMS291. The ICP algorithm is used without the initial guess from IMU or odometer sensors, which can lead to an unreliable convergence of the ICP process. Moreover, the data of the INS is not used at all, which otherwise could provide an initial guess to the ICP algorithm.



Figure 1.8: (a) point cloud before correction, (b) point cloud after correction, notice that the point belongs to the same structure are aligned, figure from Harrison et al. (2008).

As mentioned earlier, a shift error can occur between two point clouds acquired from two different lidar sensors due to the poor estimation of the relative orientation between the two lidar sensors. Levinson et al. (2007) described a simultaneous localization and mapping approach to tackle this problem, the point cloud before and after the adjustment is shown in Figure 1.9. Moreover, if the consecutive strips in the MLS data are not aligned, then the point cloud strip adjustment approaches proposed by Haala et al. (2008) and Bornaz et al. (2003) can be used. However, these types of techniques can only be used to increase the relative accuracy in the MLS point cloud.



Figure 1.9: (Left) 3D point clouds from two sensors before alignment, (Right) 3D point clouds after alignment, figure from Levinson et al. (2007).

1.4 Automatic 2D feature matching

Another well-known strategy to register the point cloud with the aerial images is to first generate perspective images of the point cloud and then extract and match 2D features. The aerial images with the accurate interior and exterior orientation can provide accurate georeferenced locations of corresponding features in the point cloud. Then the matched features between the aerial images and the MLS point cloud can be used for the orientation update of the point cloud as discussed in the next section.

The 2D matching with multiple images has many advantages over the direct 3D/3D registration approach. One of them is the possibility to detect geometrically very accurate features. An accurate 2D feature can represent the geometrical property of a landmark up to pixel or subpixel level, depending on the keypoint detection technique used. If the subpixel-level accurate features are detected and matched correctly, the point cloud improved by subpixel-level correspondences can achieve near aerial image accuracy.

Apart from the accuracy-related advantages, the 2D feature-based approach also faces difficulties, notably among them are the detection and matching of dissimilar features perceived from largely distinct perspectives. Furthermore, an image comprises information from white light reflected from the scene by cameras, whereas the lidar sensor measures the geometrical information and surface reflection of the 3D points in the scene with an active infrared sensor. Moreover, an optical image has a regular grid of pixels over the image space, whereas the MLS point density depends on the distance from the lidar sensor to the object and on the speed of MLS car. Furthermore, the laser reflection intensity of each 3D point is not the same as the white light pixel intensity. The matching would be easier if at least the data was captured from the same platform. Then the individual sensor can perceive the geometrical information from the same perspective and the landmarks or features appear similar and thus are easier to match. For this reason, most of the work has been done towards matching the point cloud and images captured from the same platform, e.g. the matching techniques discussed in (Abayowa et al. 2015; Parmehr et al. 2014; Yang et al. 2015; Zhang et al. 2015).

For 3D/2D registration, a 2D matching technique needs to first convert a point cloud into a perspective raster image and determining the pixel values from the reflectivity information of the laser points. The similar 3D/2D registration by matching rasterized laser scans and the camera images using the Scale-invariant feature transform (SIFT) based approach is proposed by Meierhold et al. (2010). Both camera image and point cloud were acquired from the ground, the point cloud was used as a reference. Therefore, the accuracy of improved aerial images was evaluated and instead of measuring the error in feature

locations, the error was measured in terms of improved camera exterior orientation parameters. In our case, the images are used as a reference dataset, while the accuracy of the improved point cloud needs to be evaluated in the absolute coordinate system and not in terms of camera orientation. Another method utilizing the SIFT-based matching approach on the point cloud and aerial imagery both captured from an aerial platform is investigated in Abedini et al. (2008). However, in this preliminary study, the registration was just estimated approximately without accuracy verification. In another similar study, Gao et al. (2015) performed the matching of a rasterized point cloud with UAV imagery. They reported RMSE of $\Delta X = 8.6$ cm, $\Delta Y = 6.3$ cm, and ΔZ =10.6 cm in the corrected point cloud. However, they evaluated only the relative accuracy using control points handpicked from UAV images and checkpoints handpicked from the adjusted point cloud. The evaluation quantified the error introduced by the adjustment and not absolute accuracy. In contrast, our goal is to automatically extract tie points which can be used by the trajectory adjustment method and achieve decimetre-level absolute accuracy.

Road markings are the prominent features detectable in both aerial images and point clouds as shown in Figure *1.10*. Because road markings and Zebracrossings are printed on the road surface with highly reflective white paint. In the point cloud, they are extractable by differentiating the strength of the laser beam reflectance. In aerial imagery, road markings are represented by pixels with highest intensity. Like many 3D to 2D matching techniques discussed earlier, the 3D point cloud points can be projected onto 2D perspective planes to generate the 2D views. Then the 2D features can be detected from both datasets for 2D matching. Examples of the feasibility of 2D feature detection from the road markings from both datasets are shown in Figure *1.11*. Sometimes, these features are also called low-level features.

Zebra crossing in point cloud







Road marking in nadir image



Figure 1.10: Comparison of two examples of road marking in the point cloud and aerial images.

Chapter 1



Figure 1.11: The subpixel features detected on the corners of a zebra-crossing. a) A perspective projection of a point cloud onto an image plane. b) The corresponding aerial image patch.

1.5 Orientation update or trajectory adjustment

The automatic 3D/2D registration and 2D matching techniques discussed earlier only facilitate the establishment of correspondences between the datasets. After the matching, another step is needed to update the orientation

of the MM platform. Almost all techniques such as SLAM, (Extended) Kalman Filter or Particle Filter (PF) can incorporate the retrieved constraints. For example, the SLAM approach can be used to compute the relative orientation, where every node of the graph contains the current pose and the observations. An edge in the graph represents the relative transformation between two connected nodes.

Using a Particle Filter or Monte Carlo Localization technique, (Kümmerle et al. 2009) estimated the global orientation of laser scanner by considering aerial image features as external references to the point cloud features, where point cloud features are the local observations. The technique tries to maximize the likelihood of all the observations including the prior, which yields the globally consistent estimation of the trajectory. In later research work, Kümmerle et al. (2011) implemented a SLAM procedure for a mobile laser scanning platform while using aerial images as a refined map. The concept of constraint extracted from aerial images is shown in Figure 1.12. This procedure achieved an overall accuracy of 20 cm. For localization of a laser scanner based vehicle in urban areas, Levinson et al. (2010) also used a SLAM approach. They claimed to achieve a 9 cm lateral error and 12 cm longitudinal error. However, they had to generate a probabilistic-map using an already existed accurate point cloud. Choi (2014) proposed a hybrid map-based SLAM using Rao-Blackwellized particle filters. They improved the trajectory of laser scanning setup for a 1 km long trajectory. Though the approach performs way better than the other approaches with residual above many meters, their approach also cannot suppress the residual lower than 2 meters near the end of the trajectory. (Im et al. 2016) used vertical corners features extracted from the point cloud and registered them with prebuild corner map using the ICP algorithm. They reported near decimetre accuracy on the horizontal plane, but the vertical accuracy was still near half a meter.



Figure 1.12: Yellow observations are global constraints extracted from the aerial images, where other observations are local constraints acquired with SLAM, figure from Kümmerle et al. (2011).

Wolcott et al. (2014) developed image-based navigation for self-driving systems. In their method, the mobile mapping camera images were registered with a previously obtained 3D point cloud by maximizing the normalized mutual information. Their developed approach reported a longitudinal RMS of 19.1~45.4 cm and a lateral RMS of 14.3~20.5 cm. Recently, Javanmardi et al. (2018) proposed a technique for MLS platform localization based on the `abstract maps'. However, this technique utilizes accurate maps already generated from an accurate prior point cloud. In our case, we do not consider that a (prior) accurate MLS point cloud is already available. Their multilayer 2D vector map-based localization achieved a mean 2D error of 20 cm, while the planar surface map-based localization achieved about 43 cm of error.

One of the earliest accounts of the B-splines based trajectory design and control of wheeled mobile robots was reported in the work of Komoriya et al. (1989). Most of the work related to B-spline based trajectory is dedicated towards the path planning. For the computation of the B-spline coefficients, Jauch et al. (2017) utilized the Kalman filter while in our case we directly estimate the coefficients using a system of linearized equations. Many researchers used the B-splines based trajectory for path planning. Elbanhawi et al. (2016) implemented randomized B-splines for robotic car navigation, where B-splines were used for the continuous motion and to accommodate the online constraints. Many researchers choose to represent the 6DOF trajectory for mobile mapping systems with the B-splines. For a visual odometry application, Patron-Perez et al. (2015) combined the discrete camera poses with continuous unsynchronized IMU observations, which lead to the estimation of a continuous camera trajectory. They reported an RMSE of 1.96 meters for a trajectory of 792 meters. It is convenient to update the B-spline

locally from the improvements in the local control points. For micro aerial vehicles, Usenko et al. (2017) modify the local intervals of the B-spline trajectory for an unmodelled obstacle in the pre-processed global trajectory. The improvement of the mobile mapping platform's trajectory can lead to the correction of MLS data, likewise, Gao et al. (2015) used shape-preserving piecewise cubic Hermite interpolating method for the adjustment of trajectory parameters. They reported the achieved accuracy of RMS - ΔX =8.6 cm - ΔY =6.3 cm - ΔZ =10.6 cm in the improved point cloud. However, the reported accuracy was estimated by the checkpoints from the same reference aerial imagery. The research work towards trajectory correction has shown that the achievement of the near decimetre accuracy in the MLS dataset is still a challenge.

1.6 Our research aim

The main aim of our research is to eliminate the dependency on the manual intervention otherwise necessary for the correction of the MLS point cloud. As discussed earlier, we believe that the utilization of the aerial imagery as a reference data set is essential in achieving the reliable and automatic registration of the point cloud.

The major steps of the automatic workflow are coarsely divided into the following four parts.

- 1) Feature detection to represent the same corresponding geometrical position with pixel-level accuracy in both lidar and image datasets.
- 2) Feature matching to find the subpixel-level feature correspondences.
- Estimation of the decimetre-level accurate 3D tie points leading to 3D-3D correspondence observation.
- 4) Adjustment of the 6DOF platform trajectory using mainly the 3D-3D correspondence and IMU observations.

The complete workflow of the proposed method is depicted in Figure 1.13. This schematic flow visualizes the first part as feature detection from both data sets, followed by a feature matching part, the features correspondences lead to extraction of 3D tie points. The last part refers to the orientation update that mainly utilizes 3D tie points and IMU observations. We aimed to reach an improvement in the absolute accuracy of the mobile mapping platform in urban canyons from about 50 cm to 10 cm. The improved dataset e.g. can be used for large-scale topographic mapping purposes.



Figure 1.13: Schematic flow of the developed research method.

The chapters in this thesis associated with research publications are as following:

Chapter 2.

 Jende, P., Z. Hussnain, M. Peter, S. Oude Elberink, M. Gerke and G. Vosselman, 2016. Low-level tie feature extraction of mobile mapping data (MLS/images) and aerial imagery. Int. Arch. of Photogramm. and Remote Sens. pp. 19-26.

Chapter 3.

 Hussnain, Z., S. Oude Elberink and G. Vosselman (2016). Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLI-B1: 609-616.

Chapter 4.

- Hussnain, Z., Oude Elberink, S., Vosselman, G., 2019. Automatic extraction of accurate 3D tie points for trajectory adjustment of mobile laser scanners using aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing 154, 41-58.
- Hussnain, Z., Oude Elberink, S., Vosselman, G., 2018. An Automatic Procedure for Mobile Laser Scanning Platform 6DOF Trajectory Adjustment. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-1, 203-209.

1.7 Structure of the thesis

This thesis is structured into seven chapters. The first and last chapters are introduction and synthesis respectively, the remaining chapters are independent scientific writings holding individual research objectives, methodology, results, discussions and conclusions. The broader topics covered in each chapter are remarked in Figure 1.13 for convenience. The chapters of this thesis are based on the research publications, therefore, the background, importance and motivation to develop an automated MLS point cloud registration with aerial imagery for correction are shared among most of the chapters.

Chapter 1. Introduction: This chapter presents the background and importance of our research project, research scope and the contributions and briefly describe the place of each chapters' contribution to the main research.

Chapter 2. Low-level Tie Feature Extraction of Mobile Mapping Data (MLS/Images) and Aerial Imagery: The first task towards the automatic feature extraction is the identification or development of a suitable 2D feature matching technique. We examine out of the box automatic feature matching techniques and then determine a feasible technique useful for our dataset. Due to the challenging task of matching dissimilar datasets and based on the literature review, it is expected that the mainstream feature matching techniques are not directly suitable. Most of the techniques or combination of feature detector and descriptor that are tailor-made for specific tasks. These approaches include e.g. matching of the images from the similar sensors, though captured from varying orientation. In this chapter, we aim to enable a reliable matching pipeline for MM data obtained in urban areas and verify existing data sets according to their localisation accuracy. The experiments conducted are based on joint work. The preparation and preprocessing of the MLS dataset and corresponding aerial image dataset together with the description of associated image matching methods and experiments are conducted by Zille Hussnain. The experiments on Mobile Mapping Imaging (MMI) dataset and the description of the related methods are conducted by Phillipp Fanta-Jende. The common features between the ground data and aerial nadir imagery are extracted based on the imprecise but approximate exterior orientation of the MM data. The technique yields adequate 2D correspondences is further developed in the next chapter.

Chapter 3. Automatic Feature Detection, Description and Matching from Mobile Laser Scanning Data and Aerial Imagery: Many of the currently available solutions are either semi-automatic or unable to achieve pixel-level accuracy. We aim to further advance the feature matching technique determined in chapter 2 to achieve pixel-level accuracy. A normal
2D matching technique does not necessarily produce pixel-level accurate result unless adopted properly. The technique comprises three steps; image feature detection, description, and matching between corresponding patches of a point cloud and aerial orthoimages created by the projection of point cloud on the virtual ground plane. For the feature detection, we developed an adaptive Harris-operator keypoint detection to detect clusters of feature points on the vertices of road markings. For the feature description phase, we use the LATCH binary descriptor, which is robust to data from dissimilar sensors. For outlier correspondences filtering, we developed a technique by exploiting the arrangements of relative Euclidean-distances and angles between corresponding sets of feature points.

Chapter 4. Automatic Extraction of Accurate 3D Tie Points for Trajectory Adjustment of Mobile Laser Scanners using Aerial Imagery: The feature matching technique developed in chapter 3 can retrieve the pixellevel accurate 2D correspondences. One of the main aims in this chapter is to achieve the subpixel accurate 2D correspondences. However, only 2D correspondences are not enough for the orientation update. Further improvements involve the projection of the point cloud into the aerial image planes to generate perspective images for each view. This is more reliable than projecting on a ground plane. If the image plane is not parallel to the ground plane then the ground projection will vary from the information in the aerial image. Another main advancement in this chapter is the computation of the 3D tie points from 2D-2D correspondences, which involves further steps of multiview matching and 3D triangulation of multiview matches. The reliability of the 3D tie points is also assessed.

Chapter 5. Enhanced Trajectory Estimation of Mobile Laser Scanners Using Aerial Images: The automatic 3D tie point extraction method described in the previous chapter provides the 3D feature correspondences for the orientation update of the 6DOF trajectory, but does not describe the adjustment. The method described in this chapter utilizes multiple observations to improve the MLS platform trajectory. The observations equations are linearized to adjust the B-spline based 6DOF trajectory. The first type of observation is derived from the 3D tie points computed automatically in the previous chapter. The second set of observation equations is based on IMU readings; covering accelerations and angular rotation. As a third type of observation soft-constraints on the related pose parameters are formulated. These observations provide updates to B-spline coefficients and lead to improved sensor orientations. Later in this chapter, we analyse the accuracy of the trajectory adjustment procedure. In real-world road scenario road marks, tie points and checkpoints are not available everywhere. Therefore, we perform detailed analysis of the adjustment procedure to confirm that the same level of accuracy has been achieved in the areas where no checkpoints are

Introduction

available. The main objective of the research work is to generate the improved point cloud from the adjusted trajectory, which is also realized in this chapter.

Chapter 6. Synthesis: This chapter discusses the research contributions, conclusions and recommendations to further extend and improve the methods covered in this research project. It compares the achievement of the research goal with state-of-the-art methods. Moreover, it provides advice to further improve and enhance the accuracy of the mobile laser scanning point cloud by an automated procedure.

1.8 Context and contributions

This PhD project is a part of the NWO project titled 'Position Estimation of Mobile Mapping Sensors using Airborne Imagery'. Among two parts in total, one part of this project is related to the registration of MLS point cloud with the airborne images for laser scanner position estimation, which is covered in this thesis of Zille Hussnain. The other part of the project covers the registration between ground panoramic images and airborne images for accurate positioning, which is investigated by PhD candidate Phillipp Fanta-Jende. The aim of both sub-projects is to estimate the position of a mobile mapping sensor for absolute and accurate mapping. A research group in the consortium consists of George Vosselman, Sander Oude Elberink, Zille Hussnain, Phillipp Fanta-Jende and Francesco Nex of the University of Twente, Enschede and Markus Gerke (Instituts für Geodäsie und Photogrammetrie, Braunschweig). Moreover, the user group comprises of CycloMedia®, Fugro Geospatial®, Slagboom en Peeters®, Topcon Europe Positioning®, Kadaster®, Het Waterschapshuis®. The formation of the research interaction is illustrated in Figure 1.14. The user group provided the input, test and validation data sets for the whole research project and evaluated results. The input datasets for this thesis consist of MLS dataset and aerial imagery of Rotterdam. The aerial images were provided with the accurate interior and exterior orientations. The GNSS receiver mounted at high altitude also cannot suffer from poor GDOP. Therefore, the positioning accuracy of the features in the input aerial images is high. We assume that features in the aerial image can be used as the external reference for the features in the MLS data. Therefore, the feature correspondence between MLS data and images can be used as constraints in orientation updating techniques.



Figure 1.14: All 7 partners involved in this NWO project and their main contributions in terms of input datasets; MLS point cloud and aerial imagery.

References for chapter 1

- Abayowa, B., Yilmaz, A., Hardie, R., 2015. Automatic registration of optical aerial imagery to a LiDAR point cloud for generation of city models. ISPRS Journal of Photogrammetry and Remote Sensing 106, 68-81.
- Abedini, A., Hahn, M., Samadzadegana, F., 2008. An investigation into the registration of LIDAR intensity data and aerial images using the SIFT approach, The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Beijing, p. 6.
- Bornaz, L., Lingua, A., Rinaudo, F., 2003. Multiple scanner registration in LIDAR close-range applications. The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 34, 72-77.
- Bosse, M., Zlot, R., 2009. Continuous 3D scan-matching with a spinning 2D laser, Robotics and Automation, 2009. ICRA'09. IEEE International Conference on. IEEE, pp. 4312-4319.
- Chen, X., Kohlmeyer, B., Stroila, M., Alwar, N., Wang, R., Bach, J., 2009. Next generation map making: geo-referenced ground-level LIDAR point clouds for automatic retro-reflective road feature extraction, Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, pp. 488-491.
- Chiang, K.-W., Huang, Y.-W., 2008. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. Applied Soft Computing 8, 722-733.

23

- Choi, J., 2014. Hybrid map-based SLAM using a Velodyne laser scanner, 17th International IEEE Conference on Intelligent Transportation Systems (ITSC). IEEE, pp. 3082-3087.
- Ding, M., Lyngbaek, K., Zakhor, A., 2008. Automatic registration of aerial imagery with untextured 3d lidar models, Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, pp. 1-8.
- Ding, W., Wang, J., Rizos, C., Kinlyside, D., 2007. Improving adaptive Kalman estimation in GPS/INS integration. Journal of Navigation 60, 517-529.
- Elbanhawi, M., Simic, M., Jazar, R., 2016. Randomized bidirectional B-Spline parameterization motion planning. IEEE Transactions on intelligent transportation systems 17, 406-419.
- Frueh, C., Sammon, R., Zakhor, A., 2004. Automated texture mapping of 3D city models with oblique aerial imagery, 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Proceedings. 2nd International Symposium on. IEEE, pp. 396-403.
- Frueh, C., Zakhor, A., 2003. Constructing 3d city models by merging groundbased and airborne views. IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2, 562-569.
- Fruh, C., Zakhor, A., 2001. 3D model generation for cities using aerial photographs and ground level laser scans, Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. IEEE, pp. II-II.
- Gao, B., Coifman, B., 2006. Vehicle identification and GPS error detection from a LIDAR equipped probe vehicle, Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE. IEEE, pp. 1537-1542.
- Gao, Y., Huang, X., Zhang, F., Fu, Z., Yang, C., 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 40, 41.
- Gu, Y., Hsu, L.-T., Kamijo, S., 2016. GNSS/Onboard inertial sensor integration with the aid of 3-d building map for lane-level vehicle self-localization in urban canyon. IEEE Transactions on Vehicular Technology 65, 4274-4287.
- Haala, N., Peter, M., Kremer, J., Hunter, G., 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 37, 1119-1127.
- Harrison, A., Newman, P., 2008. High quality 3D laser ranging under general vehicle motion, Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on. IEEE, pp. 7-12.
- Hsu, L.-T., 2017. GNSS multipath detection using a machine learning approach, 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). IEEE, pp. 1-6.

- Hsu, L.-T., 2018. Analysis and modeling GPS NLOS effect in highly urbanized area. GPS solutions 22, 7.
- Hsu, L.-T., Gu, Y., Kamijo, S., 2016. Autonomous driving positioning using building model and DGNSS, 2016 European Navigation Conference (ENC). IEEE, pp. 1-7.
- Im, J.H., Im, S.H., Jee, G.I., 2016. Vertical corner feature based precise vehicle localization using 3D LIDAR in urban area. Sensors 16, 1268.
- Jauch, J., Bleimund, F., Rhode, S., Gauterin, F., 2017. Recursive B-spline approximation using the Kalman filter. Engineering Science and Technology, an International Journal 20, 28-34.
- Javanmardi, E., Gu, Y., Javanmardi, M., Kamijo, S., 2018. Autonomous vehicle self-localization based on abstract map and multi-channel LiDAR in urban area. IATSS Research.
- Kaartinen, H., Hyyppä, J., Kukko, A., Jaakkola, A., Hyyppä, H., 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. Sensors 12, 12814-12835.
- Komoriya, K., Tanie, K., 1989. Trajectory design and control of a wheel-type mobile robot using B-spline curve, Intelligent Robots and Systems' 89.
 The Autonomous Mobile Robots and Its Applications. IROS'89.
 Proceedings., IEEE/RSJ International Workshop on. IEEE, pp. 398-405.
- Kukko, A., 2013. Mobile Laser Scanning–System development, performance and applications. Finnish Geodetic Institute.
- Kümmerle, R., Steder, B., Dornhege, C., Kleiner, A., Grisetti, G., Burgard, W., 2011. Large scale graph-based SLAM using aerial images as prior information. Autonomous Robots 30, 25-39.
- Kümmerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C., Kleiner, A., 2009. On measuring the accuracy of SLAM algorithms. Autonomous Robots 27, 387.
- Lee, S.C., Jung, S.K., Nevatia, R., 2002. Automatic pose estimation of complex 3D building models, Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on. IEEE, pp. 148-152.
- Lesouple, J., Robert, T., Sahmoudi, M., Tourneret, J.Y., Vigneau, W., 2018. Multipath mitigation for GNSS positioning in an urban environment using sparse estimation. IEEE Transactions on Intelligent Transportation Systems, 1-13.
- Levinson, J., Montemerlo, M., Thrun, S., 2007. Map-Based Precision Vehicle Localization in Urban Environments, Robotics: Science and Systems. Citeseer, p. 1.
- Levinson, J., Thrun, S., 2010. Robust vehicle localization in urban environments using probabilistic maps, 2010 IEEE International Conference on Robotics and Automation. IEEE, pp. 4372-4378.
- Lowe, D.G., 1987. Three-dimensional object recognition from single twodimensional images. Artificial intelligence 31, 355-395.

- Meierhold, N., Spehr, M., Schilling, A., Gumhold, S., Maas, H., 2010. Automatic feature matching between digital images and 2D representations of a 3D laser scanner point cloud, Proceedings of the ISPRS Commission, pp. 446-451.
- Meng, X., Wang, H., Liu, B., 2017. A Robust Vehicle Localization Approach Based on GNSS/IMU/DMI/LiDAR Sensor Fusion for Autonomous Vehicles. Sensors 17, 2140.
- Mohamed, A.H., Schwarz, K.P., 1999. Adaptive Kalman filtering for INS/GPS. Journal of geodesy 73, 193-203.
- Moosmann, F., Stiller, C., 2011. Velodyne slam, 2011 ieee intelligent vehicles symposium (iv). IEEE, pp. 393-398.
- Parmehr, E.G., Fraser, C.S., Zhang, C., Leach, J., 2014. Automatic registration of optical imagery with 3D LiDAR data using statistical similarity. ISPRS Journal of Photogrammetry and Remote Sensing 88, 28-40.
- Patron-Perez, A., Lovegrove, S., Sibley, G., 2015. A spline-based trajectory representation for sensor fusion and rolling shutter cameras. International Journal of Computer Vision 113, 208-219.
- Qi, H., Moore, J.B., 2002. Direct Kalman filtering approach for GPS/INS integration. IEEE Transactions on Aerospace Electronic Systems 38, 687-693.
- Taylor, G., Brunsdon, C., Li, J., Olden, A., Steup, D., Winter, M., systems, u., 2006. GPS accuracy estimation using map matching techniques: Applied to vehicle positioning and odometer calibration. Computers, environment 30, 757-772.
- Usenko, V., von Stumberg, L., Pangercic, A., Cremers, D., 2017. Real-Time Trajectory Replanning for MAVs using Uniform B-splines and 3D Circular Buffer. arXiv preprint arXiv:1703.01416.
- Wang, L., Neumann, U., 2009. A robust approach for automatic registration of aerial images with untextured aerial lidar data, Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, pp. 2623-2630.
- Wang, S., Deng, Z., Yin, G.J.S., 2016. An accurate GPS-IMU/DR data fusion method for driverless car based on a set of predictive models and grid constraints. 16, 280.
- Wolcott, R.W., Eustice, R.M., 2014. Visual localization within lidar maps for automated urban driving, Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on. IEEE, pp. 176-183.
- Xiong, H., Tang, J., Xu, H., Zhang, W., Du, Z., 2018. A robust single GPS navigation and positioning algorithm based on strong tracking filtering. IEEE Sensors Journal 18, 290-298.
- Yang, B., Chen, C., 2015. Automatic registration of UAV-borne sequent images and LiDAR data. ISPRS Journal of Photogrammetry and Remote Sensing 101, 262-274.

- Zhang, W., Zhao, J., Chen, M., Chen, Y., Yan, K., Li, L., Qi, J., Wang, X., Luo, J., Chu, Q., 2015. Registration of optical imagery and LiDAR data using an inherent geometrical constraint. Optics express 23, 7694-7702.
- Zhao, Y., 2011. GPS/IMU integrated system for land vehicle navigation based on MEMS. Stockholm, Sweden, KTH Royal Institute of Technology, p. 85.

Introduction

2 – Low-level Tie Feature Extraction of Mobile Mapping Data (MLS/Images) and Aerial Imagery¹

¹This chapter is based on:

Jende, P., Z. Hussnain, M. Peter, S. Oude Elberink, M. Gerke and G. Vosselman, 2016. Low-level tie feature extraction of mobile mapping data (MLS/images) and aerial imagery. Int. Arch. of Photogramm. and Remote Sens. pp. 19-26.

Abstract

Mobile Mapping (MM) is a technique to obtain geo-information using sensors mounted on a mobile platform or vehicle. The mobile platform's position is provided by the integration of Global Navigation Satellite Systems (GNSS) and Inertial Navigation Systems (INS). However, especially in urban areas, building structures can obstruct a direct line-of-sight between the GNSS receiver and navigation satellites resulting in an erroneous position estimation. Therefore, derived MM data products, such as laser point clouds or images, lack the expected positioning reliability and accuracy. This issue has been addressed by many researchers, whose aim to mitigate these effects mainly concentrates on utilising tertiary reference data. However, current approaches do not consider errors in height, cannot achieve sub-decimetre accuracy and are often not designed to work in a fully automatic fashion. We propose an automatic pipeline to rectify MM data products by employing high resolution aerial nadir and oblique imagery as horizontal and vertical reference, respectively. By exploiting the MM platform's defective, and therefore imprecise but approximate orientation parameters, accurate feature matching techniques can be realised as a pre-processing step to minimise the MM platform's threedimensional positioning error. Subsequently, identified correspondences serve as constraints for an orientation update, which is conducted by an estimation or adjustment technique. Since not all MM systems employ laser scanners and imaging sensors simultaneously, and each system and data demand different approaches, two independent workflows are developed in parallel.

Still under development, both workflows will be presented and preliminary results will be shown. The workflows comprise of three steps; feature extraction, feature matching and the orientation update. In this paper, initial results of low-level image and point cloud feature extraction methods will be discussed as well as an outline of the project and its framework will be given.

2.1 Introduction

Mobile Mapping is on the verge of becoming a substantial addition to the family of geo-data acquisition techniques. Airborne or satellite data cover large areas, but have limited capabilities when it comes to the density of data postings and high accuracy, whereas classical terrestrial techniques are expensive and often impractical. Particularly in urban areas, MM shapes up to be an extraordinarily useful technique not just to complement airborne or satellite coverage, but to enable a completely new array of possibilities. MM imaging systems and laser scanners collect high-resolution data, but have to rely on external georeferencing by GNSS. As a system being intermittently available, INS provides relative measures between position fixes and compensates for measurement noise and errors. Although GNSS carrier-phase measurements allow highly accurate positioning, urban areas remain problematic regarding the measurement reliability due to multipath effects and occlusions. When these phenomena persist over longer periods, accurate positioning cannot be maintained, and consequently data accuracy will be diminished (Godha et al. 2005). This paper presents a method to detect and extract low-level image and point cloud features as a prerequisite for the rectification of MM data using aerial imagery. First, a brief outline of the project will be given. In section 2.2, a literature overview on similar work will be presented, and applied feature detection and extraction methods will be shortly introduced, followed by section 2.4. addressing low-level feature extraction for point clouds as well as for camera images. Section 2.5 discusses initial results of low-level feature extraction methods of both aerial and MM images as well as point cloud data. Lastly, section 2.6 concludes the work presented in this paper as well as gives an outlook on future developments.

2.2 Project overview

The aim of our research project is to enable a reliable localisation pipeline for MM data obtained in urban areas, and to verify existing data sets according to their localisation accuracy in order to economise the acquisition of ground control. Due to apparent differences in the sensor setup and data, two workflows for Mobile Laser Scanning (MLS) and Mobile Mapping Imaging (MMI) are being developed. The common basis is the utilisation of high-resolution aerial nadir and oblique imagery as an external reference to compensate for vertical as well as for horizontal errors. In a first stage, common features between the ground data and aerial nadir imagery are sought. Based on the imprecise, but approximate exterior orientation of the MM data, more reliable and efficient matching techniques can be employed. For instance, a confined search for correspondences and their verification in the other image can be inferred even from coarse orientation parameters. The next stage will be the integration of oblique images into the pipeline to yield common features on the vertical axis in order to better detect errors in height, and will enable to

increase the overall number of tie features considerably. Façades and other vertical objects, such as streetlights and traffic signs, are potential objects which can be used for that purpose in future work. In a last step, this tie information allows for either a re-computation of the trajectory or, alternatively, an adjustment of the data as such.

2.3 Related work

2.3.1 Previous approaches

Coping with poor localisation of mobile platforms in urban areas has been addressed by many authors. Mostly by employing tertiary data as an external reference, either the data itself (Tournaire et al. (2006); Jaud et al. (2013); Ji et al. (2015)) or the platform's trajectory (Kümmerle et al. (2011); Levinson et al. (2007); Leung et al. (2008)) has been corrected. Depending on the data input and type (e.g. aerial imagery, digital maps or ground control points), different registration methods were utilised to impose unaffected, reliable and precise orientation information from external data on MM data sets. Subsequently, yielded correspondences were used as a constraint within a filter or adjustment solution. Even though many authors achieved a successful localisation based on an external reference, errors in height were not corrected, and a consistent sub-decimetre accuracy could not be reached.

2.3.2 Low-Level Feature Extraction

Both, low- and high-level feature extraction methods, are relevant for this research project. Whereas low-level features allow a great flexibility towards the selection of suitable correspondences, the registration of data originating from different sensors (i.e. Mobile Laser Scanning and aerial imagery) may demand an extension of that concept. Although MLS intensity information enables the derivation of corner features, thorough and reliable transformation parameters can only be determined on an abstract representation by identifying common objects in both data sets. Hence, high-level feature extraction methods will be highlighted in the future. In this paper, however, emphasis will be placed on low-level feature extraction which is still an active field of research as real-time applications have been gaining more attention in the last few years. Classic feature detection algorithms, such as the Förstner-Operator (Förstner et al. 1987) or the Harris Corner Detector (Harris et al. 1988) are accompanied by state-of-the-art approaches like AKAZE (Alcanterilla et al. 2013) or FAST (Rosten et al. 2006). Although many improvements have been made in this field, the most important property of a feature detector remains to identify the same keypoints over a set of images.

Once features have been detected in the image, they have to be described unambiguously to increase their distinctiveness among other features in order

to match them correctly. Low-level feature description approaches can be divided into two categories – binary and float description. Whereas float descriptors, such as SIFT (Lowe 2004), are based on a Histogram of Oriented Gradients (HoG), binary descriptors (e.g. BRIEF (Calonder et al. 2010)) are analysing the neighbourhood of a feature keypoint with a binary comparison of intensities according to a specific sampling pattern. Float descriptors are typically more expensive to compute, and need more memory to store their output than binary descriptors. However, depending on the application, robustness of these two categories varies (Heinly et al. (2012); Miksik et al. (2012)).

In this paper, different feature detection as well as float and binary description methods will be compared taking the example of aerial nadir, MM panoramic imagery and intensity images derived from MLS data. Feature keypoints across the data sets will be computed with SIFT (Lowe 2004), KAZE (Alcantarilla et al. 2012), AKAZE (Alcanterilla et al. 2013) and the Förstner Operator (Förstner et al. 1987).

SIFT detects blobs with a Difference-of-Gaussian method at gradually scaled instances of the image. KAZE computes a non-linear scale space using an additive operator splitting technique, where keypoints are detected at locations with a maximum response of the determinant of the Hessian matrix. Similarly, AKAZE also relies on keypoint detection based on the Hessian matrix, but computes a non-linear scale space with fast explicit diffusion. Förstner detects corners based on the search for local minima of eigenvalues of a covariance matrix of image gradients. Except for Förstner, all aforementioned procedures allow an additional feature description. SIFT utilises a histogram of oriented gradients in a local neighbourhood to describe a keypoint. KAZE's keypoints are described with the SURF descriptor (Bay et al. 2008) modified to be compatible with the detector's non-linear scale space. AKAZE uses a binary description based on an adapted version of Local Difference Binary (Yang et al. 2012) where sample patches around the keypoint are averaged and then compared in a binary manner. For Förstner keypoints, LATCH (Levi et al. 2015) has been used for a binary feature description. LATCH compares sampletriplets around a keypoint, where the sampling arrangement is learnt. Respective results will be discussed in section 2.5.

2.4 Low-level feature extraction

Aerial nadir ortho-images with a ground sampling distance of approximately 10 centimetres serve as the reference data set in this project. In order to successfully use the aerial images' exterior orientation for the rectification of MM data, respective tie information has to be reliable and accurate. Although ground and aerial nadir data have a different perspective on the scene, lowlevel feature correspondences can be identified in all data sets. For example, corners of road markings, centres of manholes and building corners resemble each other across all sensors.

The only overlapping area for feature detection induced by different original perspectives between aerial ortho-images and ground views is the road surface and its immediate vicinity. Therefore, road markings, such as zebra crossings or centre lines are being targeted on for feature detection. In the process of projecting the mobile mapping data onto the ground, not just the projection but also the approximate scale and rotation of the aerial image have been retrieved simultaneously. In particular, this circumstance simplifies the matching process considerably, but also renders to be useful for the step of feature description as less invariances and therefore fewer ambiguities have to be considered by the descriptor; i.e. the descriptor does not have to account for scale and rotational invariance since the panoramic image is north oriented and has the same resolution.

On the other hand, the aerial images have not been acquired at the same time as the mobile mapping dataset. Consequently, this fact is resulting in another category of a description problem. For instance, changes in illumination and contrast may affect the computation of the descriptor. Moreover, repetitive patterns of road markings (e.g. zebra crossings) cannot be ignored as they may result in false feature matches. Either this issue has to be tackled on the descriptor level or during the matching stage. Introducing rules, such as ordering constraints (Egels et al. (2001), p. 198) or perceptual grouping (Lowe (1985), p. 4), to describe a chain or group of adjacent features may prevent misassignment. Alternatively, the approximate camera parameters can be exploited within the matching procedure. By back-projecting identified keypoints into the other image, a window can be defined to constrain the search for correspondences. These methods are currently under development or labelled future work. Aforementioned feature detection and description procedures will be applied to our data sets and results will be discussed in section 2.5.

2.4.1 Mobile Laser Scanning Images

The Mobile Laser Scanning Point Cloud (MLSPC) is acquired from two lidar sensors(s) mounted on a car. The car's trajectory is estimated by GNSS and IMU, where a GNSS based position is retrieved after one second interval and the IMU observations are used for the interpolation of intermediate positions at 100 Hz. The mobile laser scanning data is acquired at the speed of 36km/h, approximately covering an area of 10m in 1 second. The IMU provides relatively accurate positions during 1 second interval which favour to patchwise crop point cloud tiles of each size 10x10 m. State of the art laser scanning systems claim to achieve relative accuracy of 10 mm within the distance of

100 m at normal speed (Riegl, 2015). So, even if the mobile laser scanning is conducted at a slower speed, the 10x10m patch would not be effected by (IMU based) distortions to an extent that would hamper the feature extraction.

In order to simplify and optimise feature matching, the mobile laser scanning points are projected onto an artificial ground plane to increase the resemblance to the aerial images. The ground plane is computed based on the location of the mobile laser scanning system. In areas where the actual ground is not exactly flat, this approximation can lead to spatial distortion in the projected information. In the future, the rather reliable relative orientation between two recording locations will be used to compute a more accurate plane. Since this paper focuses solely on feature detection and description, and the aerial images used are ortho-projected, this fact can be neglected for now. Thereafter, from each cropped point cloud tile, an orthoimage patch is generated. The image grey values are calculated by the barycentric interpolation of laser reflection intensities of neighbouring points. An example point cloud tile and the generated point cloud orthoimage patch is shown in *Figure 2.1*.



Figure 2.1: Point cloud patch (left) to an orthoimage (right).

2.4.2 Mobile Mapping Images

The MM images used in this research project are 360x180 degrees panoramic images (*Figure 2.2*) acquired every 5 metres along the trajectory. For more details and specifications, please see (Beers 2011).



Figure 2.2: Mobile mapping panoramic image in equirectangular projection

MM panoramic images are stored in an equirectangular projection, encoding directly spherical coordinates for every image pixel. Therefore, no projection matrix or other intrinsic parameters are needed to reproject the panoramic image. The quadratic ground plane is centred at the dropped perpendicular foot of the respective recording location. Analogue to the aerial imagery's resolution of 10 centimetres, the ground plane is rasterised holding a world coordinate for every cell. Subsequently, each raster cell's coordinate is backprojected into the panoramic image in order to extract the respective RGB value, and transfer the information back onto the ground plane. An example of a panoramic image projection on the ground plane is shown in *Figure 2.3*.

Since every back-projected ray will pierce the image plane of the panoramic image, and thus every raster cell will contain an RGB value, an interpolation of the resulting projected image seems dispensable. However, the geometric representation of the pixels of both grids varies, leading to multiple assignments of the same RGB value especially at the edge of the projected image appearing as blur. Hence, a bilinear interpolation of the extracted value according to the pixel neighbourhood of the panoramic image is conducted. Consequently, every pixel in the projected image is composed of an individual set of grey values.

Resulting from atmospheric conditions and motion blur (esp. cameras without forward motion compensation), the image quality of the aerial photographs can be affected. To compensate for these effects, the projected panoramic images need to be blurred.



Figure 2.3: Panoramic image projected onto an artificial ground plane.

2.5 Results

In this section, feature detection and description methods will be compared according to their potential for deriving significant tie features and correspondences between aerial nadir and mobile mapping panoramic images as well as between aerial nadir and MLS intensity images. First, a comparison between SIFT², KAZE, AKAZE and Förstner³ on each of the three data sets will be conducted. Subsequently, acquired keypoints will be described with their corresponding method except for Förstner where a LATCH description will be used. Although still under development, feature matching will be utilised to compare the quality of each descriptor. To this end, simple descriptor matching to yield correspondences and a homography estimation to detect outliers will be used. As the focus of this project is on urban areas, four subsets with each 15 m side length of a typical road scene between two intersections have been selected for this experiment (Figure 2.4).



Figure 2.4: Four subsets of a typical urban scene (coloured tiles from scene 1 on the left to scene 4 on the right)

2.5.1 Feature detection

In urban areas, road markings and other prominent objects, such as kerbstones or manholes, identifiable among all data sets are favoured for feature detection. However, due to noise and different original perspectives, it is considered to be a challenging task for the step of feature detection to maintain a comparable detection rate over the entire data set.

Depending on the scene, this detection rate varies. The number of road markings and the detector itself, highly influence the results. For instance, due to its scale invariance, SIFT detects keypoints on different blurred instances of the same image, and thus yields a lot more potential features than a corner detector, such as Förstner. As it will be shown in section 2.5.2, a potent feature detection alone is not sufficient for a successful registration.

² For SIFT, KAZE, AKAZE and LATCH, their respective OpenCV implementation has been used

³ Implementation of the Förstner-Operator by Marc Luxen, University of Bonn

2.5.1.1 SIFT

SIFT yields more keypoints than any other method used in this paper (Table 2-1). It detects 60% keypoints more than KAZE and even 5 times more keypoints than AKAZE or Förstner. Being very sensitive to image noise and detecting keypoints on different image scales, the detected features are not always useful. In particular, this comes into effect for both types of MM images as they have a higher original resolution and therefore a higher entropy (Figure 2.5).

Table 2-1 Number of combined keypoints over all subsets per detection method

	Aerial Image	Panoramic Image	MLS Intensity	Total
SIFT	234	379	810	1423
KAZE	119	304	458	881
AKAZE	29	68	175	272
Förstner	40	75	153	268



Figure 2.5: SIFT keypoints detected in aerial image (left), panoramic image (centre) and MLS intensity image (right)

2.5.1.2 KAZE

KAZE detects fewer keypoints than SIFT, but still considerably more than AKAZE or Förstner. However, road markings are very well preserved, and especially their corner features, which are the most important image entity in our case, were mostly detected (Figure 2.6).



Figure 2.6: KAZE keypoints detected in aerial image (left), panoramic image (centre) and MLS intensity image (right)

2.5.1.3 AKAZE

Although, AKAZE and KAZE are quite similar in the way how a feature is detected (determinant of Hessian), their main difference lies in the computation of image pyramids to detect keypoints at different image scales. AKAZE detects fewer keypoints than KAZE, but these keypoints are most often important corners of road markings (Figure 2.7). Nonetheless, in two of four aerial images, AKAZE only detected one single keypoint which turns out to be too few for matching purposes.



Figure 2.7 AKAZE keypoints detected in aerial image (left), panoramic image (centre) and MLS intensity image (right)

2.5.1.4 Förstner operator

The Förstner Operator is the only feature detector without the consideration of scale. This, and the fact that Förstner detects features solely at corners and centres of small image objects, leads to a very deliberate keypoint detection (see *Figure 2.8*). However, almost every detected feature can be regarded as significant for the registration process. Due to its capability of sub-pixel localisation of keypoints, the same object point can be represented slightly shifted among different image sources which shapes up as a challenge for feature description.



Figure 2.8 Förstner keypoints detected in aerial image (left), panoramic image (centre) and MLS intensity image (right)

2.5.2 Feature descriptor matching

Identified keypoints need to be described unambiguously to enable feature matching between two images. In general, difficulties arise if there is a change

in perspective, illumination, coverage, or scale between the images. In order to tackle parent difficulties for registering the images, the MM data has been projected onto the ground to increase the resemblance to aerial imagery. As a consequence, scale and perspective are more similar among the data sets, but differences in illumination and coverage cannot be mitigated easily. Thus, different description methods are evaluated with regard to their ability to cope not just with the aforementioned changes but also to their performance to bridge sensor-induced differences.

SIFT, KAZE, AKAZE and LATCH will be used for feature description. As mentioned earlier, LATCH will be used for keypoints detected with the Förstner-Operator. To measure the descriptor quality of each method, the images have to be matched. The number of matches, inliers classified by RANSAC as well as the actual number of correct correspondences will be compared among different descriptors. Two out of four scenes (scene 1 and scene 2) will be discussed in detail. Moreover, for every test scene, four different settings have been tested. To this end, MM data has been blurred with a Gaussian filter to increase the resemblance to the aerial data set. Moreover, a resampling of all data sets has been conducted as it has been shown that increasing the sampling size can facilitate a feature's distinctiveness considerably (Köthe 2003b).

- 1st run No modification of source images
- 2nd run Gaussian blurring of source images
- 3rd run Resampling to 150% of original size
- 4th run Blurring and subsequent resampling of source images

2.5.2.1 Aerial images and MLSPC intensity images

Scene 1:

The first scene comprises of a zebra crossing and dotted road markings aggravating correct matching due to possible descriptor ambiguity. If enough correct correspondences are found, RANSAC converges to a correct solution. In the 1st run, KAZE yielded the best results with only few mismatches (Table 2-2), and mismatches occurred due to descriptor ambiguity. AKAZE has a competitive result, however obtained fewer matches than KAZE. SIFT and LATCH both equally failed to achieve a reliable number of matches. In the 2nd run, blurring of the images dramatically improved correct matches from SIFT descriptor. Similarly, KAZE also moderately improved the results. While, the AKAZE has performed consistently and did not improve the results. The blurring did not affect the poor results of LATCH descriptor. In the 3rd run (Table 2-3), resizing the image to 150%, the results of SIFT and KAZE improved. Interestingly, all calculated matches are correct and there is no mismatch. The results from AKAZE are also improved slightly.

The 4th run, blurring and resizing the images, slightly improved the number of inliers from KAZE, while there is no mismatch as shown in Figure 2.9. The total number of matches are even decreased in case of AKAZE.

Table 2-2 Matching results of scene 1 between aerial and MLS orthoimage of the 1st and 2nd iteration

		1st r	un		2nd	run
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	61	15	1	61	14	12
KAZE	41	12	8	41	13	12
AKAZE	14	7	5	14	7	4
LATCH	9	4	0	9	4	0

Table 2-3 Matching results of scene 1 between aerial and MLS orthoimage of the 3rd and 4th iteration

	3rd	run		4th run		
	Match	Inliers	Correc t	Match	Inliers	Correc t
SIFT	59	14	14	59	14	14
KAZE	78	19	19	78	20	20
AKAZE	24	9	6	24	8	3
LATCH	7	4	0	7	0	0

Low-level Tie Feature Extraction of Mobile Mapping Data



Figure 2.9 Comparison of SIFT (top) and KAZE (bottom) in 4th run on the 1st scene.

Scene 2:

In the 1st run (Table 2-4), on this difficult scene, all descriptors totally failed except KAZE which also performed poorly due to descriptor ambiguity. SIFT also seems to struggle with the descriptor ambiguity and therefore yielded no match at all. The 2nd run did not change anything, except that the KAZE result improved slightly. The 3rd run (Table 2-5) did not lead to any significant improvements. Contrarily to the 1st scene, the total number of matches by KAZE even decreased due to the descriptor ambiguity. Although AKAZE is able to derive some matches, it cannot be considered as a significant improvement. SIFT and LATCH also failed to achieve a single match in 3rd run. In the 4th run, the number of matches from KAZE improved significantly, interestingly there was not a single mismatch. These results are plotted in Figure 2.10. SIFT could not improve here and yielded only a single match.

		1st ru	n		2nd r	un
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	54	10	0	54	11	0
KAZE	42	10	4	42	11	7
AKAZE	1	0	0	1	0	0
LATCH	1	0	0	1	0	0

Table 2-4 Matching results of scene 2 between aerial and MLS orthoimage of the 1st and 2nd iteration.

Table 2-5 Matching results of scene 2 between aerial and MLS orthoimage of the 3rd and 4th iteration

		3rd r	un		4th	run
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	51	9	0	51	9	1
KAZE	79	17	5	79	21	21
AKAZE	15	6	3	15	8	5
LATCH	3	0	0	3	0	0

Low-level Tie Feature Extraction of Mobile Mapping Data



Figure 2.10 Matching results of AKAZE (top) and KAZE (bottom) in 4th run on scene 2.

2.5.2.2 Aerial images and panoramic images

Scene 1:

In the first run, however, none of the methods was able to achieve a good result (see e.g. Figure 2.11). The derived keypoint descriptors were not distinct enough to be matched accordingly. By blurring the images with a Gaussian low-pass filter in the second iteration, results slightly improved for KAZE (see Table 2-6). Yet, by resampling the source images to 150% of their original size, results get significantly better especially for KAZE, but also a bit for SIFT and AKAZE (see Figure 2.12). In the fourth run, a Gaussian blurring followed by a resampling did not have an impact on the matching quality of this scene (see Table 2-7). Furthermore, LATCH yielded very poor results regardless of the iteration.



Figure 2.11: Matched LATCH keypoints in the first scene and first iteration

Table 2-6 Matching results of scene 1 between aerial and panoramic image of the 1st and 2nd iteration

		1st r	un		2nd	run
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	61	15	1	61	13	0
KAZE	41	12	1	41	12	2
AKAZE	14	6	0	14	6	0
LATCH	9	4	0	9	0	0

Table 2-7 Matching results of scene 1 between aerial and panoramic image of the 3rd and 4th iteration

		3rd r	un		4th	run
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	59	15	2	59	16	0
KAZE	78	17	10	78	16	10
AKAZE	24	9	2	24	9	0
LATCH	7	4	0	7	0	0

Low-level Tie Feature Extraction of Mobile Mapping Data



Figure 2.12 Comparison of matching results of AKAZE (top), KAZE (centre) and SIFT (bottom) in 3rd run of the 1st scene

Scene 2:

The second scene shows linear road markings and parts of a zebra crossing. Whereas major parts of the zebra crossing and the dotted road markings were visible in the first scene, large parts of the road markings are covered by the mobile mapping car itself in the second scene which may impede the matching process. Similarly to the first scene, unmodified imagery was difficult to match and LATCH nor AKAZE found a single correspondence. With SIFT, however, a couple of keypoints could be matched, even though just one correct correspondence has been identified (see Figure *2.13* and Table 2-8).



Figure 2.13 Matched SIFT keypoints in the second scene and first iteration (correct correspondence is light purple)

Table 2-8 Matching results of scene 2 between aerial and panoramic image of the 1st and 2nd iteration

		1st r	un		2nd	run
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	54	12	1	54	11	2
KAZE	42	8	1	42	12	0
AKAZE	1	0	0	1	6	0
LATCH	1	0	0	1	0	0

By blurring the images with a Gaussian filter, especially SIFT returns a better result. Albeit, only two correct correspondences have been identified, results got considerably better (see Figure 2.14). Apparently, RANSAC removed a couple of outliers, and was able to stabilise the estimation of the homography. Without ground truth, the matched bars of the zebra crossing might appear as correct correspondences. These descriptor ambiguities have to be tackled on another processing level.



Figure 2.14 Matched SIFT keypoints in the second scene and second iteration

Now, by resampling the images to 150% of their original size in the 3rd iteration, KAZE benefitted the most, although yielding only 4 correct matches out of 14 matches classified as inliers (see Figure 2.15).



Figure 2.15 Matched KAZE keypoints in the second scene and third iteration

Blurring the images prior to resampling them further improves the results for KAZE and AKAZE while decreasing the quality of SIFT's output (see Figure 2.16 and Table 2-9). Again, LATCH did not show any improvement.

Table 2-9 Matching results of scene 2 between aerial and panoramic image of the 3rd and 4th iteration.

		3rd r	un		4th r	un
	Matches	Inliers	Correct Matches	Matches	Inliers	Correct Matches
SIFT	51	12	0	51	11	1
KAZE	79	14	4	79	20	6
AKAZE	15	6	0	15	8	2
LATCH	3	0	0	3	0	0



Figure 2.16 Comparison of matching results of AKAZE (top), KAZE (centre) and SIFT (bottom) in 4th run of the 2nd scene.

2.6 Discussion

2.6.1 Conclusions

This paper addressed the topic of tie feature extraction within the framework of the registration of aerial nadir images, mobile mapping panoramic images and MLS data. The aim of the overall project is to develop an automatic pipeline to correct the trajectory of mobile mapping platforms, especially in urban areas where reliable GNSS localisation is scarce. As a prerequisite for an orientation update of the platform's trajectory, precise tie information is needed. In this paper, feature-based extraction techniques have been evaluated. It could be shown that the outcome highly depends on the algorithm itself and data preprocessing. KAZE seems to be the most reliable feature extraction method in both cases – mobile laser scanning intensity and panoramic imagery. SIFT and AKAZE only yield mediocre results, and do not benefit from resampling and blurring the images to the same extent as KAZE. Although the FörstnerOperator detects good and significant features, LATCH failed to describe them accordingly to allow for a successful matching. In our scenario, binary descriptors are not as powerful coping with changes in illumination as float descriptors, and cannot manage to handle features originating from different sensors that well. However, due to this specific setup, and a very generic feature matching, further tests have to be conducted to draw a thorough conclusion on binary descriptors' performance. Interestingly, the discussed techniques show a better performance in conjunction with MLS and aerial data than with MMI data.

2.6.2 Outlook

Evidently, repetitive patterns of road markings are the biggest obstacle for a successful registration of the data sets. To efficiently tackle this issue, spatial information has to be introduced. Although the positioning accuracy of mobile platforms may be diminished in urban areas, their exterior orientation could support feature matching. In other words, they can be utilised to introduce search constraints as they allow for the localisation of individual keypoints. Besides that, using contextual information and shape knowledge can augment feature description to prevent mismatches.

As far as MLS data is concerned, utilising high-level feature extraction methods could further facilitate the registration process also in areas with a lower point density and fewer distinct keypoints. Therefore, entities, such as kerbstones or entire zebra crossings, can be used to accomplish this task.

Acknowledgements

This work is part of the research programme OTP with project number 13589, which is financed by the Dutch Research Council (NWO). We would like to thank Topcon© for providing the mobile laser scanning datasets and Cyclomedia© for the aerial imagery.

References of chapter 2

Alcantarilla, P.F., Bartoli, A., Davison, A.J., 2012. KAZE Features.

- Alcanterilla, P.F., Nuevo, J., Bartoli, A., 2013. Fast Explicit Diffusion for Accelerated
- Features in Nonlinear Scale Spaces.
- Bay, H., Ess, A., Tuytelaars, T., Van Gool, L., 2008. Speeded-up robust features (SURF). Computer vision and image understanding 110, 346-359.
- Beers, B., 2011. Collection and Application of 2D and 3D Panoramic Imagery.
- Calonder, M., Lepetit, V., Strecha, C., Fua, P., 2010. Brief: Binary robust independent elementary features. Computer Vision–ECCV 2010, 778-792.

Egels, Y., Kasser, M., 2001. Digital Photogrammetry. Taylor & Francis, Inc.

- Förstner, W., Gülch, E., 1987. A fast operator for detection and precise location of distinct points, corners and circular features.
- Godha, S., Petovello, M.G., Lachapelle, G., 2005. Performance Analysis of MEMS IMU/HSGPS/Magnetic Sensor Integrated System in Urban Canyons. ION GNSS 2005.
- Harris, C., Stephens, M., 1988. A Combined Corner and Edge Detector. Proceedings of Fourth Alvey Vision Conference, 147-151.
- Heinly, J., Dunn, E., Frahm, J.-M., 2012. Comparative evaluation of binary features, Computer Vision–ECCV 2012. Springer, pp. 759-773.
- Jaud, M., Rouveure, R., Faure, P., Monod, M.-O., 2013. Methods for FMCW radar map georeferencing. Isprs J Photogramm 84, 33-42.
- Ji, S., Shi, Y., Shan, J., Shao, X., Shi, Z., Yuan, X., Yang, P., Wu, W., Tang, H., Shibasaki, R., 2015. Particle filtering methods for georeferencing panoramic image sequence in complex urban scenes. ISPRS Journal of Photogrammetry and Remote Sensing 105, 1-12.
- Köthe, U., 2003a. Edge and junction detection with an improved structure tensor. Pattern Recognition, Proceedings 2781, 25-32.
- Köthe, U., 2003b. Gradient-Based Segmentation Require Doubling of the Sampling Rate.
- Kümmerle, R., Steder, B., Dornhege, C., Kleiner, A., Grisetti, G., Burgard, W., 2011. Large scale graph-based SLAM using aerial images as prior information. Auton. Robots 30, 25-39.
- Leung, K.Y.K., Clark, C.M., Huissoon, J.P., 2008. Localization in urban environments by matching ground level video images with an aerial image, Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, pp. 551-556.
- Levi, G., Hassner, T., 2015. LATCH: Learned Arrangments of Three Patch Codes.
- Levinson, J., Thrun, S., 2007. Map-Based Precision Vehicle Localization in Urban Environments. Robotics: Science and Systems.
- Lowe, D.G., 1985. Perceptual Organization and Visual Recognition. Kluwer Academic Publishers.
- Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91-110.
- Miksik, O., Mikolajczyk, K., 2012. Evaluation of Local Detectors and Descriptors for Fast Feature Matching. Int C Patt Recog, 2681-2684.
- Rosten, E., Drummond, T., 2006. Machine learning for high-speed corner detection, Computer Vision–ECCV 2006. Springer, pp. 430-443.
- Tournaire, O., Soheilian, B., Paparoditis, N., 2006. Towards a Sub-Decimetric Georeferencing of Ground-Based Mobile Mapping Systems in Urban Areas: Matching Ground-Based and Aerial-based Imagery Using Roadmarks. ISPRS Commission I Symposium "From Sensors to Imagery".

Yang, X., Cheng, K.-T., 2012. LDB: An ultra-fast feature for scalable augmented reality on mobile devices, Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on. IEEE, pp. 49-57.

3 – Automatic Feature Detection, Description and Matching from Mobile Laser Scanning Data and Aerial Imagery⁴

⁴ This chapter is based on:

Hussnain, Z., S. Oude Elberink and G. Vosselman (2016). Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. **XLI-B1**: 609-616.

Abstract

In mobile laser scanning systems, the platform's position measured by GNSS and IMU is often not reliable in urban areas. Consequently, derived Mobile Laser Scanning Point Cloud (MLSPC) lacks expected positioning reliability and accuracy. Many of the current solutions are either semi-automatic or unable to achieve pixel-level accuracy. We propose an automatic feature extraction method which involves utilizing corresponding aerial images as a reference dataset. The proposed method comprises three steps; image feature detection, description and matching between corresponding patches of nadir aerial and MLSPC orthoimages. In the data pre-processing step, the MLSPC is patch-wise cropped and converted to orthoimages. Furthermore, each aerial image patch covering the area corresponding to MLSPC patch is cropped from the aerial image. For feature detection, we implemented an adaptive variant of Harrisoperator to automatically detect corner feature points on the vertices of road markings. In feature description phase, we used the LATCH binary descriptor, which is robust to data from different sensors. For descriptor matching, we developed an outlier filtering technique, which exploits the topological relation; relative Euclidean-distances and angles between corresponding sets of feature points. We found that the positioning accuracy of the computed correspondences have achieved the pixel level accuracy, where the image resolution is 10cm. The developed approach is reliable when enough road markings are available in the data sets. We conclude that, in urban areas, the developed approach can reliably extract features necessary to improve the MLSPC accuracy to near pixel-level.

3.1 Introduction

Over the past few years, the use of mobile mapping data products has been growing constantly. Data providers want to produce highly accurate data products and to generate them more frequently at lower costs. However, it is always necessary to utilize manually measured Ground Control Points (GCPs) for the data correction and adjustment due to the low positioning accuracy of the GNSS in urban areas. Manually measuring the GCPs and handpicking their corresponding locations in the point cloud still may not lead to the pixel-level accuracy. Moreover, acquiring the ground control point is very labour intensive and tedious work and hinders the acquisition of high-quality data in an automatic fashion. There are many techniques available, which can try to minimize the number of ground control points needed to correct a point cloud product. However, even acquisition of fewer ground control points and their selection requires manual interventions. This manual post processing step of data correction forces surveyors to survey a city site less frequently at the cost of more manual effort, while as a consequence, customers use the outdated, imprecise and expensive datasets.

Automatic feature extraction between Mobile Laser Scanning Point Cloud (MLSPC) and Aerial imagery can be very advantageous for maintaining MLSPC product quality. Especially, in the urban canyons, where GNSS based positioning inaccuracies are common.

Previous research has shown that the automatic image feature extraction can be used for the registration between two datasets from different sensors. The obtained transformation can be used to correct a point cloud when the reference data already has a reliable accuracy. Jende et al. (2016) have presented the preliminary results of registration between aerial and ground datasets; i.e. MLSPC orthoimages and terrestrial orthoimages. Similarly, Gao et al. (2015) have improved the Mobile Laser Scanning (MLS) data accuracy by its automatic registration with high -resolution -accurate UAV's imagery. They performed Harris corner keypoint detection and edge-based template matching between UAV imagery and rasterized MLSPC orthoimage patches, where registration is achieved using bundle adjustment. This work reported the RMS - ΔX =0.086m - ΔY =0.063m - ΔZ =0.106m in the corrected dataset. A preliminary SIFT based feature detection and matching approach for registration of aerial lidar to the aerial image was described by Abedini et al. (2008), however, the achieved accuracy of feature matching was not described.

Fruch et al. (2003) proposed to utilize the aerial imagery as a reference global edge map for the lidar observations on the ground and achieved an absolute coordinate difference of ± 5 meters using Monte-Carlo-Localization (MCL).

Automatic Feature Detection, Description and Matching from Mobile Laser Scanning

Kümmerle et al. (2011) developed a simultaneous localization and mapping (SLAM) approach that computes its global position for every node, where MLSPC points are the observations and the aerial image is the reference. In this approach, the accuracy is defined by the distance from the true orientation of building edge/wall to the estimated orientation of edge/wall, and it is 0.2m for five test data sets.

Due to the difference in sensor characteristics, the automatic feature detection and matching of the data from two sensors is a challenging task. The aerial image captures the projection of reflected white light, while the MLSPC represents the geometrical information of the structures in scene as 3D positions and the reflection intensity of each surface point. Moreover, the optical images have a regular grid of pixels over the image space, whereas the MLSPC's point density is variable and depends upon the distance of the lidar sensor to the object and the speed of MLS car. Furthermore, the laser reflection intensity of each point is not same as the white light surface reflection.

In this scenario, designing an automatic feature extraction technique which can detect common features is a difficult but crucial task. Once achieved automatically, the feature extraction can save lots of extra effort, cost and time. It can improve the accuracy of the already refined or raw datasets. Moreover, the data can be acquired more frequently because no manual adjustment would be needed. Furthermore, the extracted features can be used for automatic data evaluation and quality control.

The purpose of our research work is to develop an automatic pipeline, which can be used to correct the MLSPC using the aerial images of the corresponding area. Our technique is developed to automatically extract the accurate low-level (2D) features from MLSPC and aerial imagery.

In the given situation and the data sets, it is apparent that the road surface is the biggest part common between the aerial images and the MLSPC. Apart from the different characteristics of the involved information, the road surface area can be used for the 2D features extraction. Mainly, road markings are the objects distinguishable based on information in both aerial images and MLSPC. In case, the road markings are not present in both datasets at all or completely repainted to different positions, then this technique could not yield reliable correspondences. Usually, it is not required to have road marking in each tile because state of the art position filtering methods and MLS system need accurate correspondences only after every 100 meters.

The organization of sections in this paper is presented according to the developed method's workflow diagram in Figure *3.1*. In the end, section 3.5 describes results followed by an evaluation and conclusions.


Figure 3.1: Workflow diagram of the developed method for MLSPC to aerial image registration.

3.2 Feature extraction

As described earlier that the feature extraction will be performed on the 2D images, therefore first the datasets need to be cropped and projected on the ground to generate the 2D images. In this section, we will describe the selection of the test area and pre-processing technique followed by the feature extraction from both datasets.

3.2.1 Selection of test area

In this project, we used the datasets from Rotterdam city. The MLSPC is acquired by the Topcon® IP-S3 mobile mapping system, it has a built-in 360-degrees lidar sensor which captures 700,000 pulses per second. For the experiment, a consecutive part of MLSPC of size 527 m is selected, which is visualized in Figure 3.2. Moreover, the aerial images of the same area are also obtained, where each original aerial image has 20010x13080 pixels and the resolution of 10 cm. The visualization of aerial imagery mosaic is also shown in Figure 3.3. The test area is selected such that the following disturbances are included:

- i) GNSS error (MLSPC)
- ii) different types of road markings (both)
- iii) occlusions (both)
- iv) traffic (both)
- v) trees (both)
- vi) shadows (aerial)

vii) variable positioning error (MLSPC)

3.2.2 Pre-processing

Each selected dataset is cropped into 14 tiles, where each tile has a size of 38x38 meters approximately. The size of a tile is mainly influenced by the relative accuracy of the MLSPC and the total computational time required to detect and match features. The 14 tiles of MLSPC and aerial images are shown in Figure 3.2 and Figure 3.3 respectively.



Figure 3.2: Visualization of the MLSPC of the test area and cropped tiles.



Figure 3.3: Visualization of the aerial image of the test area and cropped tiles.

3.2.2.1 MLSPC orthoimage generation

The MLSPC consist of long consecutive series of points and due to the limitation of the computational power, it is useful to crop the point cloud into small pieces. Therefore, first the MLSPC is patch-wise cropped into small tiles. The MLSPC is automatically cropped based on the coordinate bounds of X, Y and Z-axis. The bounds originate from the MLS platform's 3D position in the trajectory and extended outward to crop a tile of size 38x38 meters. Now, each cropped tile is still a small point cloud as shown in Figure *3.4* (left). Then, to convert it to a 2D image, the obtained point cloud is projected on the ground plane and an orthoimage is obtained as shown in Figure *3.4* (right). The laser reflectance property of each projected 3D point is used to calculate the grey values of the corresponding pixels in raster image. For occlusion of missing data, the interpolation of the grey values is estimated by the bilinear interpolation. The points above 4 m ground level are also removed before the projection. The 4 m height can easily remove the unnecessary information like trees, poles,

building etc., while at the same time preserves the parts of road which are relatively higher.



Figure 3.4: Point cloud patch on the left is converted to an orthoimage on the right.

3.2.2.2 Aerial orthoimage generation

As the MLS platform's 3D position is used to generate the MLSPC orthoimage, the same position is also backprojected to the original nadir aerial image and a patch of 380x380 pixels is cropped as illustrated in Figure 3.5. At this moment, it is assumed that the interior and exterior orientations of the aerial images are accurate and it is considered that there is no image distortion.



Figure 3.5: Acquisition of the aerial image patch.

3.2.3 Feature detection

The main focus in this step is to automatically detect features in both images, where features' positioning accuracy should be reliable. Corners are the most reliable 2D features which could be detected from the intersection of image gradients. Therefore, a corner feature detection algorithm by Harris et al. (1988) is used. Harris corner detector is further improved to automatically adapt according to our feature detection requirements. An automatic adaptation of parameters is necessary because, for example, when Harris detector is applied with fixed parameters to detect weak gradient changes, then it becomes too sensitive to the sensor noise and detects many false keypoints. Moreover, it may not detect the same features from the different images of even same sensor. To overcome this problem, the threshold parameter is estimated dynamically by an iterative approach. So, the computation of the threshold is achieved dynamically for every image. This type of automatic parameterization is necessary for the automatic feature extraction pipeline and because it can compensate for inter and cross sensor characteristics and noise.

As shown in Figure 3.6 (top, middle), the gradient values of the same piece of road marking in both datasets are quite different, where, the adaptive approach detects keypoints reliably as shown in Figure 3.6 (bottom). Usually, a corner feature is detected at the intersections of two edges, however, sometimes due to the large difference in description of the feature, a single corner point failed to match in descriptor space. That's why instead of relying on a single corner, the adaptive approach also detects multiple corners, which are more reliable while matching descriptors. Thereby, the corner features are detected as clusters of corner keypoints.

The adaptive approach requires a constant input parameter, which in our case is the number of required keypoints. If the scene is same, we assume that the number of keypoints should also be same in all images. The number of keypoints needed is set to 5000 ± 100 (100=tolerance), where iterations are incremented by the gradient threshold on normalized resultant gradient image. The number of required keypoints is calculated from the size of a tile. As, a tile covers an area of 38x38m, which means roughly two to three keypoints for $1 m^2$. The required number of keypoints is also influenced by the computational costs for matching maximum keypoints. The different steps of detection of 5000 keypoints from an MLSPC orthoimage and from an aerial orthoimage patch are shown together in Figure 3.7. An automatic adaptive approach is not only useful for the data from different sensors but also for the images captured with the same camera, this type of inter sensor keypoint detection and differences are shown in Figure 3.8.

Chapter 3



Figure 3.6: Example of adaptive Harris corner keypoint detection of a road mark, multiple keypoints (red dots) are detected over two observable corners.



Figure 3.7: Adaptive Harris keypoint detection of a whole tile, with the total number of keypoints, threshold and required iterations.



Figure 3.8: Adaptive approach for different aerial images of the same scene. Though look similar, the underlying image differences can be realized by comparing the threshold, iterations and the obtained keypoints.

3.2.4 Feature description

For description of the feature points, we use binary descriptor called Learned Arrangements of Three Patch Codes (LATCH) proposed by Levi et al. (2015). LATCH is robust to noise in both the MLSPC and aerial orthoimages because it utilizes patches of pixels instead of a single pixel to establish a binary relation. LATCH is useful in our problem for two reasons, firstly, it is a binary descriptor, and thereby the descriptor matching is very fast, as the computational time is an important issue because we would like to process a lot of keypoints. Secondly, the grey values of images from sensors of different characteristics cannot be compared directly but patterns of higher/lower grey values are comparable. Moreover, patch triplet approach is robust to noise and it is better than directly comparing a single pixel value as practised in pixel pairs approach. Due to patch triplet based approach, LATCH can ignore small occlusions and cluttered locations in the orthoimage of the MLSPC. Also, it can compensate for shadows and occlusions in aerial images. The MLSPC laser reflection intensity has different behaviour of noise than grey values of white light camera images, and LATCH descriptor avoid sensitivity to individual sensor noise by not sampling and comparing individual pixels. In our project, the arrangement of the all patch triplet is already learned from the data sets

provided in Brown et al. (2011). The dataset used for the training is of generic type, thereby, it is not required to learn the arrangement from our data. Performance results of the LATCH descriptor over benchmark data set and comparison with other binary descriptors and with float descriptors are also described in Levi et al. (2015).

The LATCH descriptor returns a binary string of 32 bytes or 256-bits for each keypoint (Figure 3.9), so, a total of 256 triplets are computed for each descriptor. The size of each mini patch is 7x7 pixels. There are total three patches in a triplet, one of the middle one is called 'Anchor' and the other two are 'companions' or C1 and C2 Levi et al. (2015). A descriptor's binary string is calculated based on the comparison between SSD1 and SSD2, here, SSD stands for the sum-of-squared difference of two (7x7 pixels) patches, where SSD1 is computed between the anchor and C1 patches and SSD2 is computed between the anchor and C1 patches and SSD2 then the returned binary value is 0 and otherwise it is 1. The descriptors are computed around all keypoints in both images.



Figure 3.9: The computation of a particular LATCH descriptor used in this project.

3.3 Descriptor matching

Like other binary descriptors, LATCH descriptor matching is also based on the similarity of corresponding binary strings. The distance of each descriptor from a list of length n to all descriptor in a list of length m is calculated by Hamming distance, which is the number of total different bits occurred in two binary strings. Moreover, for Lowe's ratio test, the threshold of 0.99 is used as proposed by Levi et al. (2015), which was proposed as 0.66 in the original setting by Lowe (2004).

Sometimes, in descriptor matching, the correct corresponding descriptors are not near in descriptor space and may even the inaccurate corresponding descriptors are near in the descriptor space. For this reason, instead of the one closest descriptor, first five nearest descriptors are obtained by k-NN based descriptor matching. Therefore, for each query descriptor in the first image, up to five nearest neighbours in the descriptor space of the second image are retrieved. The query descriptors linked with up to 5 descriptors are illustrated in Figure 3.10 and same keypoint correspondences are also shown in Figure 3.11. In the experiment, the inlier ratio of obtained correspondences is small, however, more importantly there are potential correct correspondences. So, during inevitable situations, when there are noise and occlusion, the flexibility of descriptor matching can reliably include correct matches which otherwise could be rejected. Later, a great number of outliers can be removed by our developed filtering algorithm.



Figure 3.10: Illustration of the matched descriptors of image 1 and image 2.

Automatic Feature Detection, Description and Matching from Mobile Laser Scanning



Figure 3.11: Hamming distance based descriptor matching of an example tile with five nearest neighbours (k=5).

3.4 Outliers filtering

The outliers filtering approach is developed to remove outlier correspondences, which are introduced due to relaxed descriptor matching. Most of the outliers are included due to the descriptor matching with five nearest neighbours. The Homography (computed with RANSAC) based outlier filtering could not be used directly because the inliers ratio is very low, therefore, first a large number of outliers should be removed. As, the given data sets have a large error in translation and small error in scale and rotation, we have developed a technique to remove the outliers based on their relative Euclidean distances and the angles. The developed technique initiates the correspondence search starting from a seed feature point. This is a brute-force technique and iterates through each keypoint in any one of the lists (in Figure 3.10). A seed feature point P₁ in image 1 is illustrated in Figure 3.12, and its corresponding candidate points are P_2 , P_3 and P_4 . In image 2, the seed point P'_1 has corresponding candidate points P'_2 , P'_3 and P'_4 , where d_n is a relative distance of nth correspondence P_n to a seed feature point and θ_n is a relative angle. Due to the distortion in datasets, these parameters are relaxed by a tolerance value, so, $abs(d'_n - d_n) < tolerance_d$ and $abs(\theta'_n - \theta_n) < tolerance_{\theta}$. The relaxation by tolerance can be considered as the maximum accuracy that can be achieved by this technique. In other words, a very small value of tolerance could yield very accurate matches but if the level of distortion is high then many potential correct correspondences will be missed as well. Therefore, it is important to not neglect correct matches and increase the tolerance to the maximum and optimal value. In the experiment, we set $tolerance_d$ to 0.9 pixel and $tolerance_{\theta}$ to 0.15 degree, these constratints togather maintain the final accuracy to near pixel-level.



Figure 3.12: In image 1 and Image 2, $(P_2 \Leftrightarrow P'_2)$ and $(P_4 \Leftrightarrow P'_4)$ are correct matches with respect to the seed point P_1 and P'_2 , whereas $(P_3 \Leftrightarrow P'_3)$ is an outlier due to the difference between θ_3 and θ'_3 , though d_3 and d'_3 are equal.

However, the computation of both the distances and angles in the brute-force technique can be computationally expensive. Moreover, if the error in rotation is more than 0.15 degree then we can only use the d_n constraint. An example result obtained by only applying d_n constraint is shown in Figure 3.13. Without the θ_n constraint, there are a small amount of outliers marked with blue arrows.

Once a large number of outliers are removed by d_n constraint alone, we can then estimate an accurate homography matrix as now the inlier ratio is large. Now, with the obtained homography, the remaining outlier correspondences can be removed directly after the descriptor matching. The result of same example after the outlier removal using Homography is shown in Figure 3.14.

Automatic Feature Detection, Description and Matching from Mobile Laser Scanning



Figure 3.13: Correspondences computed without θ'_n constraint. Blue arrows are pointing toward some visible outlier correspondences.



Figure 3.14: Homography based outliers filtering.

3.5 Results

In this section, we will stepwise discuss the results according to the feature extraction workflow.

3.5.1 Feature detection

In the feature detection, clusters of keypoints are detected at the corners of road markings, an example is shown in Figure *3.15*. Many keypoints are detected at the edges of a traffic light passing over the zebra crossing in the aerial image patches, which imitate intersections. The two different zoom levels are provided for the visualization of the feature detection, one is a normal view (Figure *3.15*, top) as well as a zoomed view (Figure *3.15*, bottom).

Comparatively, more keypoints are detected at the corners of the road markings in the aerial images than in the MLSPC orthoimage. It can be noticed with a close observation that the correct corresponding keypoints are present inside the clusters of keypoints in both images. For detailed inspection, it is possible to further zoom-in and analyse and compare the grey values with the detected keypoints (in a digital copy of this paper).

For the keypoint detection from all tiles, the threshold and the total iterations required to obtain the keypoints are provided in Table *3-1*. The extremely different requirements of threshold and the number of iterations needed to detect features in the tile 2 and tile 6 shows the differences faced by feature detection from the images of even the same sensor.



Figure 3.15: Adaptive Harris corner feature detection from aerial (left) and MLSPC (right) image $patch^{5}$.

⁵ This image can be zoomed to 6400% for a very close observation using the digital copy of this paper.

Tile no.	Iterations [MLSPC/Aerial image]	Threshold [MLSPC/Aerial image]	Keypoints [MLSPC/Aerial image]	
1	782/85	7.8/0.83	5098/5091	
2	1151/179	11.50/1.78	5096/5091	
3	1250/346	12.49/3.45	5100/5100	
4	885/147	8.84/1.46	5095/5099	
5	369/25	3.68/0.24	5090/5074	
6	160/24	1.59/0.23	5088/5078	
7	359/62	3.58/0.61	5092/5060	
8	483/121	4.82/1.2	5091/5100	
9	256/167	2.55/1.66	5094/5100	
10	568/13	5.67/0.12	5097/5100	
11	529/7	5.28/0.06	5097/5100	
12	310/35	3.09/0.34	5088/5053	
13	707/88	7.06/0.86	5098/5098	
14	778/273	7.77/2.72	5098/5092	

Table 3-1: Results of the adaptive Harris keypoint detection.

3.5.2 Feature matching⁶

In this section, we will discuss the results of the developed filtering approach. In first glance, matched features appear as random clusters of the keypoints in the image pair (Figure 3.16, top). A zoomed visualization of the matched keypoints with connecting lines is also crowded due to many matches (Figure 3.16, middle). Therefore, a comprehensible visualization of the results with block patterns is produced in (Figure 3.16, bottom). By close observation of the block patterns, the background intensity information can be compared across images, which shows that the patterns of intensity information are also similar in corresponding images. Moreover, the grey values of edges in context of the road markings are also similar with respect to each image. This also shows that the features are matched to the pixel-level.

For estimation of error found by our matching method in the point cloud relative to aerial images, we compute the normal probability function of the error in translation. The obtained correspondences, the total matched descriptors, together with mean (μ) and sigma (σ) of X and Y axis for all tiles are presented in Table *3-2*.

⁶ Visualizations of all matching results are provided at: <u>https://www.researchgate.net/profile/Zille Hussnain</u>

Chapter 3



Figure 3.16: Different visualizations of the matched features.

Tile no.	Matches from descriptor	Matches from developed	(ш) xrl	(ա) հո	ax(m)	a۷(m)
1	4393	140	-2.16	0.87	0.03	0.08
2	7016	930	-2.07	0.95	0.06	0.06
3	12961	1703	-2.02	0.81	0.07	0.05
4	8159	702	-1.99	0.93	0.08	0.05
5	3702	275	-2.16	1.01	0.07	0.08
6	5090	316	-1.93	1.00	0.05	0.09
7	5318	456	-1.82	0.87	0.09	0.07
8	9403	495	-1.74	0.85	0.04	0.08
9	6050	438	-1.74	0.90	0.06	0.08
10	4109	65	-1.90	0.87	0.07	0.02
11	6043	748	-1.78	1.00	0.05	0.08
12	4849	185	-1.91	0.93	0.08	0.04
13	6460	561	-1.78	1.05	0.06	0.07
14	8824	750	-1.81	1.02	0.07	0.05

Table 3-2: Number of Matched keypoints and the normal probability function parameters of translation error <u>between the point cloud and aerial images (all units are in meters)</u>.

3.5.3 Discussions

The matching results have shown that the developed method has achieved the pixel-level accuracy on the given data sets. The developed filtering approach can work only when the error in rotation and scale is small. When the error in the rotation is large, then without the rotation check in the outliers filtering part, only the distance constraint can be imposed. The resultant correspondences together with a small subset of outliers can be used to compute the accurate homography matrix and the remaining outliers can be removed from the initial correspondence set.

Even though the aerial imagery was captured in winter and the roads were comparatively well visible, the occlusion due to the branches of the tree was the biggest problem. We have noticed that the low point density and small occlusions in the point cloud do not have much effect on the results. So, the developed method can give better results when the road surface is clearly visible in the aerial imagery.

Overall, distortion in the aerial imagery due to the atmospheric dispersion, different level of contrast/illumination and variable exposure could hinder the keypoint detection, however, these problems were handled by the adaptive Harris keypoint detection. The false keypoint detection caused by shadows, trees and poles, and matching was handled by the outlier filtering approach.

3.5.4 Evaluation of estimated shift

The actual error in the inaccurate MLSPC relative to the aerial imagery could be measured roughly by handpicked correspondences. It is important to note here that this is not the evaluation of accuracy achieved by our matching technique, as it is the future work. This evaluation method only considers the error in translation and does not consider the error in rotation. Though not very accurate, still this method can provide a rough estimation of the error presents in the inaccurate MLSPC.

An example of manual selection of corner point is shown in Figure 3.17. In this figure, it has been shown that the top left corner point of a road marking (bounded by a green window) is selected as the same corresponding point in both images. This process was repeated for each tile and few well distributed points were obtained. The distribution of the obtained error is shown in Figure 3.18. As here it is not feasible due to limited space to show the results from all tiles, only accurate and least accurate results from two different tiles are shown in Figure 3.18, top and bottom respectively.



Figure 3.17: Manual selection of a road mark corner point for manual evaluation of inaccurate point cloud.



Figure 3.18: Overlap of PDFs of translation error in X and Y coordinates, error estimated by the Developed Method (DM) and Manually Measured (MM), tile 4 (top) and tile 8 (bottom).

3.6 Conclusion

In this paper, we have implemented an automatic and reliable feature extraction procedure for the MLSPC and aerial orthoimages, which can compute correspondences with up to pixel-level accuracy. The feature matching results have shown that in future the MLSPC can be corrected reliably to the pixel level accuracy.

The filtering technique proposed is feasible for the real world problem of MLSPC correction. The developed filtering technique cannot compensate for the scale variations and large rotations. Moreover, the filtering technique is a brute-force method as it is developed for the post processing purpose, it may not be feasible for real time applications. The advanced searching algorithms can be

introduced to decrease the total time needed for feature matching and to compensate rotation and scale variations.

Image filtering methods can be used to minimize the time spent on the computation of feature detection parameters, it can avoid the need to adapt the parameters to images from same sensor. The evaluation of error present the inaccurate point cloud is a rough estimation because manual correspondence selection is error prone. A more accurate evaluation method will be used in future.

Acknowledgements

This work is part of the research programme OTP with project number 13589, which is financed by the Dutch Research Council (NWO). We would like to thank Topcon© for providing the mobile laser scanning datasets and Cyclomedia© for the aerial imagery.

References of chapter 3

- Abedini, A., M. Hahn and F. Samadzadegana, 2008. An investigation into the registration of LIDAR intensity data and aerial images using the SIFT approach. The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 2, p. 6.
- Brown, M., G. Hua and S. Winder, 2011. Discriminative learning of local image descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(1), pp. 43-57.
- Frueh, C. and A. Zakhor, 2003. Constructing 3d city models by merging ground-based and airborne views. IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 562-569.
- Gao, Y., X. Huang, F. Zhang, Z. Fu and C. Yang, 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 40(1), p. 41.
- Harris, C. and M. Stephens, 1988. A combined corner and edge detector. Alvey vision conference, Vol. 15, p. 50.
- Jende, P., Z. Hussnain, M. Peter, S. Oude Elberink, M. Gerke and G. Vosselman, 2016. Low-Level Tie Feature Extraction of Mobile Mapping Data (mls/images) and Aerial Imagery. Int. Arch. of Photogramm. and Remote Sens. pp. 19-26.
- Kümmerle, R., B. Steder, C. Dornhege, A. Kleiner, G. Grisetti and W. Burgard, 2011. Large scale graph-based SLAM using aerial images as prior information. Autonomous Robots, 30(1), pp. 25-39.
- Levi, G. and T. Hassner, 2015. LATCH: Learned Arrangements of Three Patch Codes. arXiv preprint :1501.03719.
- Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), pp. 91-110.

Automatic Feature Detection, Description and Matching from Mobile Laser Scanning

4 – Automatic Extraction of Accurate 3D Tie Points for Trajectory Adjustment of Mobile Laser Scanners using Aerial Imagery⁷

⁷ This chapter is based on:

Hussnain, Z., Oude Elberink, S., Vosselman, G., 2019. Automatic extraction of accurate 3D tie points for trajectory adjustment of mobile laser scanners using aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing 154, 41-58.

Abstract

Poor GNSS measurements in urban areas caused by blocked GNSS signals and multi-path is a well-known problem, which leads to an inaccurate trajectory estimation of Mobile Laser Scanning (MLS) platforms. Consequently, the MLS point cloud contains positioning errors. This paper presents a new method for the automatic extraction of accurate 3D tie points for the trajectory adjustment of MLS platforms in GNSS denied or troubled areas. The new method relies on aerial imagery as a reliable external source of reference provided that accurate exterior orientation parameters are available. Accordingly, one of the main objectives is to register the mobile laser scanning point cloud with corresponding aerial images. The matches between aerial images are used to obtain 3D tie points by forward intersection. By also determining the corresponding locations in the point cloud we obtain a 3D-3D correspondence between the MLS point cloud and the aerial images. In the future, the obtained 3D-3D correspondences will be used for trajectory adjustment.

Our automatic tie point extraction procedure is tested on two independent MLS point clouds. The point clouds were acquired by two different platforms in Rotterdam. The aerial imagery of the same area was acquired at a different time. We evaluated the matching results for both datasets and concluded that the new procedure reliably extracted the 3D tie points for 55% of the tiles of the size of 90 metres from the first MLS dataset. In the second dataset, 60% of the tiles of size 74 metres yielded reliable 3D tie points. It is not necessary to successfully register all tiles because the results of this work will be used for the trajectory adjustment and the IMU can reliably support the positioning for small intervals.

4.1 Introduction

Accuracy improvement has become one of the priorities in turning a Mobile Laser Scanning (MLS) point cloud into a reliable and accurate commercial product. In a mobile laser scanning system, it is crucial to have an accurate position estimation of the mobile mapping platform, since a small error in the estimated sensor position directly affects the positional accuracy of the acquired data points. This problem becomes worse in urban canyons, where GNSS signals are either obstructed or reflected from multiple surfaces as mentioned by Cui et al. (2003), Godha et al. (2005) and Haala et al. (2008). In urban canyons, Kukko (2013) demonstrated that the accuracy of the estimated trajectory of a mobile mapping platform could be decreased to more than 50 cm due to the outage of GNSS signals. However, without any GNSS signal outage, the state-of-the-art Mobile mapping (MM) platform can achieve 2-3cm accuracy Haala et al. (2008) and Kaartinen et al. (2012).

There are two major contributors to the erroneous position estimation; one is the long-term GPS signal outage, which causes the continuous accumulation of error in trajectory over time because the positioning is estimated by the IMU observations prone to drift error. The second factor is the multipath effect, which causes trajectory interpolation between accurate and inaccurate positions resulting in an inaccurate trajectory. Without other accurate positioning reference, corrections for the erroneous GNSS measurements are needed. This typically depends upon manually surveyed GCPs and their manually selected corresponding locations in the point cloud.

Over the last decade, interest in the application of mobile mapping data products has continued to grow. Data providers want to produce highly accurate data products and to acquire data frequently at lower costs. However, it is necessary to utilize the manually acquired GCPs for the MLS data correction and adjustment if high accuracy is requested. In commercial procedures, two main post-processing steps are usually involved. The first post-processing step is to correct the data only by automatic means, similar to the technique reported by Levinson et al. (2007), Ding et al. (2007) and Zhao (2011); this procedure performs registration between overlapping point cloud patches and adjusts MLS platform positions in the trajectory. This step is accomplished without any external reference. The inaccurate data can be corrected if there is enough data overlapping. However, even by utilizing these techniques, the final achieved accuracy is at the meter level as expressed in Chiang et al. (2008), especially during long-term GNSS outages in urban canyons. This accuracy is insufficient to customers.

The manual post-processing step involves the collection of the GCPs by surveying the target area and then the manual and careful selection of the

corresponding points in the MLS dataset. Overall, this complete step is labour intensive and mainly hinders the automatic acquisition of high-quality MLS point cloud. Since the post-processing step is both expensive and time-consuming, it forces most customers to use outdated dataset. Moreover, the GCPs' measurements could still be uncertain and the handpicking of the GCPs in the MLS datasets could as well be imprecise. Furthermore, to increase the reliability and validity of GCPs' accuracy, the same GCPs are acquired multiple times, which makes the correction procedure even more expensive. Additionally, if undetected inaccurate GCPs mix in the final reference set, then there is no third reference to verify the inconsistent GCPs. Nevertheless, after the cumbersome manual efforts and corrections, the second step could improve the data accuracy to the desired level, but the final product will still be costly.

Therefore, it is desirable to have an automatic procedure that can extract 3D tie points automatically while improving the accuracy at low cost and in less time. Our workflow replaces the manual effort with an automatic procedure. Our work aims to achieve similar or better accuracy than the manual post-processing method. The main objective of the developed method is twofold; first is to register both MLS images and aerial patches automatically; second is to compute the subdecimeter-level accurate 3D tie points.

For the registration, the aerial images are considered a reliable and accurate reference, which can improve MLS point cloud inconsistencies cost-effectively, especially when combined with the automatic procedure. However, the automatic matching of images from two different sensors and 3D feature extraction is a challenging task. The aerial image presents the white light reflection captured from the scene by a camera, whereas the lidar sensor measures the geometrical information and laser light reflection of the surface of the scene. The laser reflection intensity of each 3D point is not the same as the white light pixel intensity. Moreover, an optical image has a regular grid of pixels over the image space, whereas the MLS point density strongly depends on the distance from the lidar sensor to the objects and the speed of MLS platform. Furthermore, all pixels in an image are captured at once, whereas each 3D point is measured at a different time.

In this paper, the description of our developed automatic procedure is organised as follows: The second chapter investigates the literature related to our work. The third chapter starts with the introduction of the developed method and explains the concept of the 3D tie points. The remainder of the third chapter describes the pre-processing of the datasets following by the 2D-2D image registration and lastly the calculation of the 3D tie points from image correspondence. The fourth chapter provides all results obtained by 2D-2D

image registration and presents the quantitative statistics of 2D matches and 3D tie points.

4.2 Related Work

One of the main areas of research work is the registration of 3D data to 2D data or vice versa. This is a common topic across many research fields. Much of the literature has been published for 3D-2D registration. However, every research application has some specific assumptions and difficulties, which are not directly applicable to our case. For example, 3D to 2D registration for medical images in Markelj et al. (2012) and Tomazevic et al. (2003) or facial feature recognition in Zhou et al. (2018) may assume an accurate calibration between two sensors and maybe benefit from the stable artificial ambient light condition, and the task would be to refine the calibration error with epipolar geometry or dense matching. Moreover, the small scale and the different demand of accuracy influence the development of those registration algorithms. On the other hand, the registration process involves surfaces from deformable and texture-less objects.

However, the problem at hand relates to 3D-2D registration problems involving at least a 3D point cloud from a lidar sensor and one camera image. Especially, it involves a large-scale outdoor environment, where the position of at least one sensor is not known reliably.

In this vein, Fruh et al. (2001) proposed a method for position estimation by maximizing the cross-correlation and Monte Carlo Localization (MCL) of a 3D model (derived from laser scanners) with 2D aerial images and roadmaps. The accuracy of this method was limited by the size of the road width. Moreover, this technique was implemented in an area without tall buildings. Ding et al. (2008) proposed a method to extract 2D orthogonal corners from a 3D lidar model obtained from an ALS point cloud and oblique images. The correspondence between the 2D orthogonal corners is established using a Hough transform and generalized M-estimator sample consensus. The correspondence is then used to refine camera pose by the algorithm of Lowe (1987). However, the validation and accuracy of the camera pose were assumed based on the visually correct mapping of the texture on the buildings, and no further accuracy measurement was provided. Another method was proposed by Wang et al. (2009) for the registration of aerial lidar data by oblique images. Triples of line segments were defined and matched, and an overall correct pose recovery rate of 98.5% was reported. However, the accuracy of the pose estimation was not mentioned. A registration between range image of a terrestrial laser scanner and ground images by implementing the epipolar constraint-assisted robust matching approach was described by González-Aguilera et al. (2009), where the goodness-of-fit in the robust camera resection was 0.03 m for the first case and 0.01 m for the second case.

The datasets are matched by applying traditional 2D feature extraction techniques. Research efforts have been devoted to the 3D-2D registration problem in the context of automatic tie point computation for the correction of mobile laser scanning datasets. In a preliminary investigation, Jende et al. (2016) described available methods for the multi-sensor data registration and provided the results of registration between aerial and ground (MLS point cloud and terrestrial orthoimages) datasets. In our earlier work (Hussnain et al. (2016)), we developed a method for the registration of point cloud orthoimages to aerial images. The experimental results have shown that the image feature extraction can be used for the registration between the two datasets (originating from various sensors). Moreover, the results demonstrated that the pixel-level accurate transformation could be calculated for the correction of MLS dataset (when the aerial image already has a reliable accuracy). The matching between the rasterized laser scans and the camera images using SIFT feature extraction was implemented in the work of Meierhold et al. (2010). However, these techniques do not estimate the 3D tie points, which are necessary for the adjustment of the MLS point cloud or its trajectory.

Another approach for the 3D-2D registration is to first convert the 2D images to a 3D point cloud and then register that with laser scanning point cloud. The registration of the terrestrial laser scan by an SFM point cloud generated from ground images was demonstrated in the work of Stamos et al. (2008).

For the image-based navigation of self-driving systems, the MM camera images are registered using prior 3D lidar by maximizing the normalized mutual information described in Wolcott et al. (2014). Sometimes, the new MLS data is registered using previously referenced data for corrective purposes. Sheehan et al. (2013) showed the correction of the trajectory of newly acquired MLS data using the previously surveyed MLS dataset. Similar projects are implemented without the use of prior 3D point cloud dataset. For example, 2D to 2D registration is developed for a UAV-based navigation system, by the registration of UAV-borne images to previously captured satellite images (as reference data), this work is reported in Conte et al. (2008). Feature-based 3D to 2D registration is also popular, a novel feature-based 2D to 3D registration method for the automatic extraction of control points in optical and lidar images was proposed by Palenichka et al. (2010). Recently, Javanmardi et al. (2017) have performed the correction of the MLS point cloud by using multiple reference datasets, including aerial images. However, their approach constrained the registration problem to only 2 DOF.

Methods have been developed to utilize mutual-information based registration for 3D/2D registration. The registration between the camera and laser data using mutual information is covered in the method developed by Wang et al. (2012). However, that method is developed on a setup where both laser scanner and camera sensors were mounted on a common mobile platform. Thereby, the accurate calibration was already known with only 2-3 pixel of error before registration, which was reduced to one pixel after the registration. Another mutual-information based method proposed by Parmehr et al. (2014) performs registration between aerial images and aerial lidar using statistical dependency between same- and multi-modal datasets; however, it requires a prior (similar) training dataset. The accuracy is calculated based on the difference with updated exterior orientation parameters.

Line segments are extracted from the 3D data and 2D image datasets for the automatic registration purposed by Frueh et al. (2004). The GNSS-based exterior orientations of the oblique images are used as the initial guess for registration. Lines of a 3D model were projected to the oblique image plane and instead of one line, groups of three lines rated a particular camera pose as proposed by Lee et al. (2002). However, this technique was implemented for texture mapping, for which high accuracy was not important, in contrast to the application at hand. A linear feature-based 2D and 3D registration scheme was developed by Liu et al. (2012), in which the goal was to estimate the camera orientation by matching 2D ground images and 3D ground range data (3D is considered as reference). An average relative error of around 1–2 pixels was reported and verified visually.

MLS point cloud/image registration methods have been implemented to address the problem of relative accuracy that does not require high absolute accuracy. Some of the methods' evaluations were based on visual verification and some methods required previously constructed 3D models, whereas the highly accurate 3D models were generated by combining MLS and ALS data as well as by combining ground and aerial views. Moreover, the input 3D models were created with manual editing. In some cases, feature extraction processes were based on the geometrically incorrect edges in the point cloud for texture mapping; the edge extraction process; thus, the extracted lines could lead to geometrically inaccurate correspondences. A great deal of research has been focused on the registration of building façades and building boundary mapping. This registration is only possible between the MLS point cloud and oblique images. This technique does not extract features from roads because, in some urban areas, road features are not consistently available in oblique images.

Gao et al. (2015)have performed registration a rasterized point cloud with UAV imagery. They reported RMS values of $\Delta X=8.6$ cm, $\Delta Y=6.3$ cm, and $\Delta Z=10.6$

cm in the corrected point cloud. However, they have calculated the relative accuracy using control points handpicked from UAV images and checkpoints handpicked from the adjusted point cloud. The evaluation quantified the error introduced by the adjustment and not absolute accuracy. In contrast, our goal in this paper is to extract tie points which can be used by the method developed in Hussnain et al. (2018) and achieve decimetre level absolute accuracy. Moreover, in this paper we present two further advancements to our previous work Hussnain et al. (2016); one is the improvement in the accuracy of the 2D-2D registration by improving pixel-level correspondences to subpixel-level. Second is the development of a reliable method to compute 3D tie points from 2D-2D correspondences. Moreover, 3D-3D correspondence between tie points will be established, such that they can be used for the 6DOF trajectory adjustment introduced by Hussnain et al. (2018).

4.3 Developed method

To extract the tie points between aerial imagery and MLS point clouds, common landmarks or features have to be identified. Therefore, it is necessary to choose a common part between both aerial images and the MLS point cloud. The most suitable area is the road surface, where road markings are well visible in both aerial images and the MLS point cloud. The previous method (Hussnain et al. (2016)) projects the point cloud to the horizontal ground plane. However, if the road is not parallel to the horizontal plane, the projection introduces error in the positioning of the outcome. To avoid this type of inaccuracy, in this paper, we project the point cloud onto the aerial image plane.

The idea of common feature identification in the point cloud and the aerial imagery is illustrated in Figure 4.1. The common feature identified in the point cloud is called A2P tie point and its corresponding feature in aerial images is A2A tie point. The A2A tie points are considered accurate because the orientation of the aerial images is accurately known, whereas the A2P points are known to be inaccurate because the laser scanner positioning is unreliable.

The workflow of our method is presented in Figure 4.2, It starts with the preprocessing of both point cloud and images. Then a 2D image registration technique performs registration between the rasterized point cloud and aerial images. The inter-registration of aerial images produces A2A tie points, while the cross-registration between aerial images and the rasterized point cloud will determine A2P tie points. In the end, 3D-3D correspondences are established by searching the correspondences between A2A and A2P tie points sets.

The remainder of this paper uses the following definitions for the tie points.



- i) A2A tie point: 3D point calculated from aerial images correspondence and multiview triangulation.
- ii) A2P tie point: 3D point cloud point determined based on correspondence between the rasterized point cloud and an aerial image.

Figure 4.1: The concept of A2A and A2P 3D tie points.



Figure 4.2: Workflow automatic 3D tie point extraction.

4.3.1 Pre-processing mobile laser scanning point clouds and aerial images

In the pre-processing step, we crop small parts of the datasets for registration. There are two main reasons that the data pre-processing is needed: first, the positioning error will vary over time. Therefore, the MLS point cloud should not be registered with one set of transformation parameters (e.g. affine). Secondly, the large MLS datasets cannot be processed in memory at once. The pre-processing divides the datasets into subsets, which are compatible and practical for fast registration.

4.3.1.1 Mobile laser scanning point cloud perspective image generation

The MLS point cloud is a set of 3D points acquired continuously over time. Each 3D point is observed in a local vehicle-based coordinate system and then transformed to the global coordinate system making use of the GNSS- and INS-based positioning. The mobile laser scanning operation needs to be conducted while mapping car moves in normal traffic. Some areas are scanned faster than the others, due to the variable car speed and curvature of the road. Moreover, the laser scanner configuration and road traffic play a role in the point cloud occlusions and density. To overcome occlusions and data gaps, the same area is scanned with two laser scanners mounted on a car with different angles w.r.t. the driving direction and aiming at an as complete as possible acquisition of the road environment. The two laser scanners may scan the same surface, but because of occlusion, many other surfaces will only be captured by one of the scanners. In the end, points from all passes and scanners are combined to obtain a single point cloud with as few as possible occlusions and data gaps. The points related to the same object surface can originate from data collection at very different GNSS times in case of multiple passes over the same trajectory. Therefore, when the positioning error changes over time, then the point cloud points suffer from alternating errors in estimated position. Therefore, the 3D points belong to the same world objects get misaligned in the merged point cloud. The misaligned point cloud cannot simply be projected into an image since 2D features belonging to different GNSS time should not be put together in one 2D image. This is essential for the 2D image registration and for utilizing the tie points in the later step of trajectory adjustment. Our pre-processing procedure starts with ordering the points in a time chronological order per laser scanner. Eventually making the road marking features also chronologically feasible and spatially consistent for matching.

Next, these point clouds are divided into smaller parts or tiles. One of the issues in the tiling process is to maintain the contextual information of the road markings. Since our registration method is based on the road markings, it is desirable to properly include the complete road marks at the edges and provide enough overlap between the point cloud tiles, such that an MLS image has a sufficiently large border around the road marks to calculate the feature descriptor. For the context-based tiling, the pre-processing procedure generates polygons derived from the shape of the trajectory and includes the road marks based on the information retrieved from OpenStreetMap, which is easier than detecting road marks directly from the point cloud. This important landmark information on the map is freely available for the potential mapping areas. The tile length is influenced by the points set that can be handled in memory at once and the average length of consecutive positioning that can be reliably supported by the IMU alone.

Once the point cloud tiles are created, the next step is to filter 3D points representing moving traffic. The side view of this situation is shown in the top part of Figure 4.3. The top-down view on the same area with a moving car before and after filtering is shown in the left bottom and right bottom of the figure respectively. These points on cars need to be removed because projection onto the aerial images can mask features on the road surface, which is a nuisance for the keypoint detection and matching.



Figure 4.3: A moving vehicle in MLS point cloud.

Points belonging to moving vehicles can be removed by the segmentation of point cloud into smooth surfaces and only keeping points on such surfaces. However, this extra effort can be avoided by using the results of the point cloud back-projection, which is needed for the point cloud rasterization, regardless of the point cloud filtering. To use back-projection for both purposes, first, the point cloud is back-projected onto the relevant aerial image planes. For example, the back-projection of an MLS point cloud tile onto three aerial image planes is illustrated in Figure 4.4. This process generates multiple 2D point clouds as shown at the top of Figure 4.4.



Figure 4.4: A 3D point cloud tile and its projection to multiple perspective planes, for the generation of point cloud images.

Now, to filter the undesired points based on the back-projection results, we first determine the distance to the image plane for each 3D MLS point which will be projected into the same pixel. This provides the distribution of the points along each pixel frustum as depicted in Figure 4.5. Then we only keep the points which lie inside the furthest peak of this distribution. Finally, to determine the grey value of the image pixel, we calculate the median of the laser reflectance intensities of the selected MLS points. We will refer to the images created this way as the MLS images.



Figure 4.5: Distributions of undesired points and road points along a pixel frustum.

4.3.1.2 Aerial Image Patch cropping

As the projection of the point cloud provides the extent of the MLS image on an aerial image plane, the same extent can be used to crop an Aerial Image Patch (AIP) from a large aerial image. Sometimes, not all area of an AIP extent overlaps with the extent of a large aerial image. In this situation, the AIP can differ in size from the corresponding MLS images, which is resolved by imagepadding to the AIP. However, the generated AIP should have enough nonpadded image area for keypoint detection and descriptor computation, otherwise, the patch is discarded. We use the matching technique from (Hussnain et al. 2016) for descriptor calculation. This technique requires at least 48x48 pixels around the keypoint.

4.3.2 2D-2D registration

In order to extract A2P or A2A tie points, the preliminary step is the registration of the 2D image patches. Accordingly, the registration is needed for two cases; first, is the AIP-to-AIP registration and second is the MLS image-to-AIP registration. In this paper, we used the 2D-2D image registration technique developed in our earlier publication Hussnain et al. (2016). This registration technique detects the keypoints using adaptive Harris keypoints on the road marking corners. Initially, the non-adaptive version was proposed by Harris et al. (1988). Our earlier method adapted the keypoint detection threshold iteratively depending on the image contents. However, the iterative process is computationally expensive. In this paper, iterations are reduced by predicting the target threshold from a non-linear approximation fit over the outcomes of a few initial iterations. Then, the keypoints are matched based on the LATCH descriptor as described in Levi et al. (2016) and an outlier filtering technique.

The main improvement compared to Hussnain et al. (2016) is the method to obtain subpixel-level correspondence. The registration developed earlier can only provide the pixel-level 2D-2D correspondences. In this paper, we first detect the Förstner keypoints (Förstner et al. (1987)) in both corresponding images. Then, we search the nearest Harris keypoints, which are already matched. This search eventually establishes the correspondence between Förstner keypoints in both images. The Förstner keypoint detection cannot be used instead of the adaptive Harris keypoint detection since it can only provide the single keypoint per corner feature. Multiple keypoints per corner procures robustness in matching for data from different sensors. The comparison of pixel and subpixel correspondences of the same 2D feature is illustrated in Figure 4.6. In the figure, the green dots represent new subpixel-level keypoints establishing correspondence that is derived from the red dots representing the pixel-level keypoints correspondence. A result from the real dataset, for the mapping from pixel to subpixel-level correspondence is discussed in section 4.4.4.

We use an outlier filtering technique described in Hussnain et al. (2016) because the orientation of the point cloud images is not accurately known. The features in point cloud images in relation to aerial images can have meters of error, therefore, it is not practical to search the corresponding features e.g. along the epipolar lines. Though, we can use the epipolar search constraint for only aerial-to-aerial correspondence matching because their orientation is very accurate. However, once we choose to use the outlier filtering technique for the point cloud image to aerial image feature matching, which can also match the aerial-to-aerial images, then the implementation of the epipolar based matching technique will be redundant.



Figure 4.6: Pixel and subpixel-level correspondence of a same corner feature.

4.3.3 A2A 3D tie points extraction

For the A2A tie point calculation, the registration is performed between the image pairs of AIPs. Image pairs are formed based on all permutations in the set of available AIPs. For example, if 5 AIPs are used for the calculation of A2A tie points for an area, then the number of AIP-to-AIP matching pairs will be 10. The 2D registration method can only perform independent registrations in one image pair at a time. However, a feature needs to be visible in at least three aerial images and one MLS image to establish a reliable multi-view correspondence. An A2A tie point computed from multiple aerial images is more reliable, therefore, common correspondences in more than two aerial images are desirable. To achieve this, we perform multi-view matching. The 3D tie points calculated from stereoviews are also retained. Note that the multiview matching is used for two purposes in our workflow; the first is to provide multi-view correspondences for the computation of the A2A tie points, the second is to search for common A2P points to establish 3D-3D correspondences. The latter is described in section 4.3.5. Finally, A2A tie points are calculated using multi-view triangulation based on the non-linear method described by Hartley et al. (2003). The iterative process starts with a linear
estimate of 3D point as an initial approximation and then a non-linear solution is optimized by the Gauss-Newton algorithm.

4.3.4 A2P 3D points extraction

In this step, we extract A2P tie points by matching MLS images and AIPs. The same 2D registration method described in section 4.3.2 has been used. Similarly, for registration, the first step is to prepare image pairs for matching. Contrary to A2A tie point extraction, the total image pairs are equal to the number of aerial images involved. The 2D registration just provides the correspondences between the MLS images and AIPs but any matched 2D pixel of MLS image can be traced back to its predecessor 3D MLS point cloud point. While multiple point cloud points contributed to the formation of an MLS image pixel, the 3D point whose back-projection is nearest to the centre of an MLS image's pixel is defined as the A2P 3D tie point. So, the A2P tie points are basically the point cloud points which have indirect correspondence with aerial images through MLS images.

4.3.5 3D-3D correspondence search

Not all A2A and A2P tie points can establish a 3D-3D correspondence. The final 3D-3D relation depends on the A2P tie points themselves since these points provide the link between MLS image and AIP features. So, a 3D-3D relation can be established if an MLS image point has correspondences to at least two AIPs points or if the MLS image has a match to one AIP point, which, in turn, has a match to a point of another AIP.

To search the common correspondences, multi-view matching is implemented using a robust graph-based search. Then a common correspondence is traversed through all nodes of the related registered image pairs. This search includes correspondences from both A2A and A2P tie points. An example of these search flows is illustrated in Figure 4.7. Unlike 4.3.3, which only deals with AIP-to-AIP search flows, for 3D-3D correspondence search, there are two types of search flows; one is from AIP-to-AIP and other is from AIP-to-MLS image. So, the total possible search pairs are equal to the permutations of all involved images. To comprehend the benefit of this technique, suppose that there are 4 AIPs and an MLS image, which are related to a test area. Then there will be one to one registrations between 4 MLS images and 4 AIPs. Moreover, 6 further registrations between 4 AIPs. This leads to a total of 10 possible correspondences for the same feature across all views. Now, assume that among 10 potential correspondences, only registrations between AIP1-AIP2, A3-A4, AIP2-MLS image and AIP3-MLS image are successful with correspondences C_1 between X_1 and X_2 , C_2 between X_3 and X_4 , C_3 between X_2 and X_5 and C_4 between C_3 and X_5 respectively. With the given information, it is not possible to determine a corresponding A2A tie point to an A2P point with multiviews correspondence. However, with multi-view matching, we can establish the unknown links, plotted as blue dotted curves in Figure 4.7. This technique also introduces robustness to our method in difficult cases when only a few stereo pairs are successfully matched.



Figure 4.7: Extracted missing links retrieved using multiview matching. Dotted lines are missing links while the solid colour lines are well-established correspondences.

4.4 Implementation, Experiments and Results

We perform experiments on two MLS datasets. These point clouds are acquired in the urban areas of Rotterdam. The extraction of A2A tie points by registration between AIPs and A2P points by registration of AIPs MLS images are described respectively. In the end, obtained 3D-3D correspondences together with quantitative and qualitative analysis are discussed.

The input datasets are mainly acquired in the GNSS troubled areas and consist of roads surrounded by buildings with three or more stories. Moreover, the test area is intentionally selected to contain the following properties in MLS point cloud and aerial images:

- viii) various types of road markings (both)
- ix) occlusions (both)
- x) traffic (both)
- xi) trees (both)
- xii) shadows (aerial images)

- xiii) GNSS troubled areas (MLS point cloud)
- xiv) atmospheric and image sensor noise (aerial images)
- xv) variable 3D point density and data gaps (MLS point cloud)
- xvi) multiple sensors (MLS point cloud)
- xvii) multiple passes (MLS point cloud)

4.4.1 Mobile Laser Scanning Point Cloud datasets

The MLS datasets were acquired using two independent laser-scanning systems, where both systems utilized different laser scanners and position estimation sensors. The first dataset, MLS_DATA-I, is acquired with a mobile mapping system based on a single laser scanner, the Topcon® IP-S3. This system has a built-in 360-degree lidar sensor that measures 700,000 ranges per second. Its positioning is estimated with an industrial grade IMU (KVH® CG-5100). The overall mapped area in this dataset is 13 km and the total continuous scanning time is 56 minutes. The MLS_DATA-I dataset consists of the single one-go acquisition but has multiple passes of the same roads. Its trajectory is plotted and superimposed on the aerial orthoimage and shown on the left of Figure *4.8*.

The second point cloud is acquired with Topcon® IP-S2 system that is based on two 360-degree laser scanners mounted in a cross-configuration. The position estimation of this system is based on a built-in MEMS gyroscope. The MLS_DATA-II dataset was acquired in 13 minutes of continuous laser scanning on a 3.5 km road. It consists of the single one-go acquisition and there are no overlapping passes. The trajectory of MLS_DATA-II is plotted on the right side of Figure 4.8.





Figure 4.8: MLS_DATA-I trajectory (a). MLS_DATA-II trajectory (b), both datasets are in Amersfoort / RD New coordinate system.

4.4.2 Aerial nadir imagery

Fifteen aerial images are captured with the UltraCam® camera on a manned flight, where each aerial image has 20010x13080 pixels with a GSD of 10 cm. The aerial image acquisition positions and pseudo flight path are shown in Figure 4.9. The aerial image extents (green rectangles) are plotted on the orthoimage. These aerial images have 60% forward overlap and 40% across track overlap. The exterior orientation parameters of the images are known in the Dutch reference system RD-NAP.



Figure 4.9: The arrangement of aerial image extents over test area.

4.4.3 Pre-processing results of datasets

The MLS_DATA-I is pre-processed and shapes of the resulting tiles are plotted in Figure 4.10. The first tile is numbered when the trajectory enters the coverage of two or more aerial images. The pre-processing has generated 155 tiles from the MLS_DATA-I, each tile is approximately 90 metres in length towards driving direction and 60 metres in width across the road. Note that each point cloud tile causes the generation of multiple MLS images and AIPs.

The pre-processing of the MLS_DATA-II has generated 47 tiles, while each tile is approximately 74 metres towards driving direction and 22 meters across the road as shown at the right side of Figure *4.10*. To study the reliability of our

method in producing the tie points regardless of the tile size, different tile size parameters are used for both datasets.



a)



Figure 4.10: a) MLS_DATA-I tiles. b) MLS_DATA-II tiles.

4.4.3.1 Mobile Laser Scanning Images

The tile number 18 generated from the MLS_DATA-I (as in the previous section) produces six MLS images, which are obtained by the projection onto aerial image 1, 2, 3, 6, 7, and 8. Note that the number of MLS images in Figure 4.11 is equivalent to the number of corresponding AIPs in section 4.4.3.2. The MLS images shown in Figure 4.11 do not show cars on the road, as they have been successfully removed (based on the point cloud filtering method described in section 4.3.1.1) unlike the aerial images in Figure 4.12. The image

area without any projected point is coloured grey and excluded from the matching. Moreover, the arbitrary variation in the pixel information is noticeable among three magnified patches at the right side of Figure 4.11, this is caused by the projection on different perspective planes. This could lead to a disparity in the description of features and it is one of the reasons that the robust descriptor matching method has been used.



Figure 4.11: Left, six MLS images generated from tile 18 of MLS_DATA-I. Right, projections of same point cloud on three different perspective planes.

4.4.3.2 Aerial Image Patches

The corresponding AIPs of tile 18 from MLS_DATA-I are shown in Figure 4.12. The aerial images 1, 2 and 3 were captured when the airplane approximately headed east, whereas the aerial images 6, 7 and 8 were captured while the airplane headed approximately west as airplane camera orientation described

in section 4.4.2. The arbitrary appearance of the traffic is evident in the AIPs in Figure 4.12. This is a problem for matching, and unfortunately, this problem cannot be avoided for both AIP-to-AIP and AIP-to-MLPCPI matching cases. The techniques similar to Javanmardi et al. (2017) to remove traffic are also not useful because artificial image information introduced to mask traffic cannot be of help in the matching.



Figure 4.12: AIPs for the tile 18 of MLS_DATA-I.

4.4.4 2D-2D correspondence between image patches

The registration described in section 4.3.2 produces 2D-2D correspondence. This section describes in general, the results of registration, especially the update to our previous technique that is a transition from the pixel-level to subpixel correspondences. In the next section, the results of registration regarding A2P and A2A tie points are discussed separately. For the example of registration between MLS image and AIP, a result related to tile 18 is shown in Figure 4.13. The left and right side of Figure 4.13 shows pixel-level and subpixel correspondences respectively. For visual comparison, matched features are gradually zoomed-in. At the top of the figure, corresponding regions are marked by blue windows across matched pairs. To recognize the road marks by context, marked regions are further magnified in the middle level. Positions of the keypoints within the pixel are of the main interest and

can be observed at the bottom of the figure. Notice keypoints in the bottomleft images are always in the centre of the pixel while on right side keypoints can have arbitrary locations, reflecting the subpixel accuracy of the Förstner keypoints. To validate the procedure for the aerial-to-aerial matching scenario, a similar example is replicated for the AIP-to-AIP matching in Figure 4.14. Onwards, the number of matched keypoints is only mentioned after the mapping to the subpixel.

The initial guess of the positional error of the MLS point cloud is deduced from an error distribution obtained in earlier matching experiments. The search window size for the MLS_DATA-II is set to ± 3 metres. In the earlier experiments, the search window size was larger than the actual error.



Figure 4.13: For MLS image-to-AIP matching, yellow lines are correspondences, keypoint in aerial images are represented by green `+' symbol and keypoints in point cloud images are red circles. Notice that the pixel-level corresponding keypoints are always in the middle of the pixel while the subpixel-level keypoints are not necessarily in the middle of the pixel. It is evident by comparing the matched keypoints in the bottom row.



Figure 4.14: For AIP-to-AIP matching, yellow lines are correspondences, keypoints in right patches are represented by green '+' symbol and keypoints in left patches are red circles. Notice that the pixel-level corresponding keypoints are always in the middle of the pixel while the subpixel-level keypoints are not necessarily in the middle of the pixel. It is evident by comparing the matched keypoints in the bottom row.

Sometimes the road marks are removed and repainted on a different location. So, if one dataset was acquired before and the other was acquired after the repainting then similar road marks appear relocated as an image pair at the left side of Figure 4.15. The matching procedure establishes only correct correspondences because the outlier filtering technique in (Hussnain et al. 2018) favours parameters of sets with majority correspondences. These parameters are the relative distances and angles between the keypoints of the same image. The group of displaced features will not be matched because their relative arrangement differs from the other features across the image pair.

Similarly, road traffic can also alter the description of the feature and lead to mismatches or no match as shown on the right side of Figure 4.15. Most of the road marks are covered by traffic. Only some are completely visible, among them only a few have the description still intact. Another problem with similar

consequences is the point cloud occlusion caused when the mobile mapping vehicle drives on the other side of the road as shown in Figure 4.16. Interestingly, the same area was scanned twice from the alternating side of the road as plotted in Figure 4.16. Consequently, features are only matched according to alternating non-occluded areas as marked in the figure.

Moreover, it is noticeable that 2D features other than road marks can be matched. Some road landmarks usually exhibit the same properties that define the corner features and fulfil the criteria of detection based on the corner of the gradients. One example of such corner feature detection is shown in Figure 4.17. However, in the case of this specific feature, the feature and its description are reasonably similar in both datasets. Unfortunately, this is not the case with most of the similar features. Therefore the chance of successfully matching similar features is low using our 2D registration technique.

One benefit of the matching technique is the invariance to the differences caused by shadows. Even though the intensity differences inside and outside shadow areas are quite different, the features keep the same intensity gradient patterns and can be matched. In Figure 4.18, the road surface is partially covered by the shadows in aerial images, but still able to match for case (b) of MLS image.



Figure 4.15: Top, few matches due to shifted and repainted road marks. Bottom, matches missed due to traffic cover.





Figure 4.16: No matches due to the alternating areas occluded from the laser scanner.



Figure 4.17: Matched feature other than road marks.



Figure 4.18: Matched features despite building shadow.

4.4.5 A2A 3D tie points

In this section, we discuss the results obtained regarding the A2A tie points extraction. The tile 18 of MLS_DATA-I has projections on six aerial image planes. Therefore, the number of total permutations for matchable pairs is 15. The correspondences of each pair are plotted in Figure 4.19. Each patch inherits the original orientation according to the airborne camera as depicted in Figure 4.9. Moreover, each image pair is further rotated to 90 degrees, to fit all fifteen matching results together in Figure 4.19. Some image pairs show fewer matches compare to the other ones. However, all matches contribute to the computation of the A2A tie points plotted in Figure 4.20, this means that not all tie points are computed with the same group of views. Moreover, it is observable that the A2A tie points are confined to the road area. However, Figure 4.19 shows matches outside of the road area. Once the A2A tie points are computed, these points are filtered by utilizing the trajectory information.

Automatic Extraction of Accurate 3D Tie Points for Trajectory



Figure 4.19: AIP to AIP matches for tile 18 from MLS_DATA-I.



For comparison to pixel-to-subpixel correspondence mapping, Table 4-1 provides example results from both MLS_DATA-I and -II. Usually, the number of subpixel correspondences is reduced enormously because only a single keypoint is detectable on a corner feature, whereas multiple pixel-level keypoints are detected on a corner to overcome the descriptor-matching problem. After this step, the total number of matches is reduced but the accuracy of the remaining matches is increased.

Table 4-1:	AIP-to-AIP	matches for	· MLS	DATA-I	tile 1	L8 and	MLS	DATA-II	tile 2.	The
number of	correspondi	ng keypoints	befor	e and af	ter th	e mapp	ing of	f pixel-lev	el mat	ches
to Förstner	r keypoints.									

DATASE	T AIP-TO-AIP PAIR	PIXEL-LEVEL	SUBPIXEL	PERCENTAGE
	1&2	10100	126	1.50%
	1&3	7328	63	1.70%
	1&6	734	35	1.90%
	1&7	815	29	1.70%
	1&8	754	25	1.60%
	2&3	9169	106	1.80%
	2&6	690	27	1.30%
I	2&7	817	24	3.0%
	2&8	843	27	3.30%
	3&6	1072	34	3.20%
	3&7	1165	26	2.20%
	3&8	1132	26	2.30%
	6&7	8583	103	1.20%
	6&8	5344	79	1.50%
	7&8	7248	119	1.70%
п	1&2	1494	84	5.70%
	187	266	21	7.90%
	1&8	272	25	9.20%
	2&7	240	23	9.60%
	2&8	344	32	9.40%
	7&8	1049	113	10.80%

4.4.6 A2P 3D tie points

In this section, we discuss results related to the computation of A2P points. An example of matched MLS images of tile 18 with AIPs is shown in Figure 4.21. In this example, each pair has a slightly different arrangement of the matched correspondences, it is noticeable by comparing correspondences across all image pairs. The orientation of every MLS image is always the same as the corresponding AIP, this was independent in the case of the AIP-to-AIP matching.



Figure 4.21: Matches between MLS images and AIPs related to tile 18 of MLS_DATA-I.

Percentages of subpixel keypoints related to the same datasets are comparable across Table 4-1 and Table 4-2. Similar to the previous section, the keypoints are reduced significantly after mapping to subpixel correspondences. This shows that the mapping process delivers similar results for both AIP-to-AIP and MLS image-to-AIP matching. The percentage of subpixel correspondences after the mapping is higher for MLS_DATA-II, which indicates that there are fewer pixel-level keypoints per feature in MLS_DATA-II compares to MLS_DATA-I.

pixel-level matches to Förstner keypoints.						
DATASET	MLS IMAGE-TO-AIP PAIR	PIXEL-LEVEL	SUBPIXEL	PERCENTAGE		
I	Tile 18 AIP 1	2479	35	1.50%		
	Tile 18 AIP 2	2165	36	1.70%		
	Tile 18 AIP 3	1944	36	1.90%		
	Tile 18 AIP 6	1419	24	1.70%		
	Tile 18 AIP 7	2042	31	1.60%		
	Tile 18 AIP 8	2086	36	1.80%		
п	Tile 2 AIP 1	318	32	10.10%		
	Tile 2 AIP 2	289	33	11.50%		
	Tile 2 AIP 7	265	33	12.50%		

246

35

14.30%

Table 4-2: MLS image-to-AIP matches of MLS_DATA-I tile 18 and MLS_DATA-II tile 2, number and percentage of corresponding keypoints before and after the mapping of pixel-level matches to Förstner keypoints.

4.4.7 3D-3D correspondence

Tile 2 AIP 8

The final 3D-3D correspondences are shown in Figure 4.22 for both datasets. Both plots show the start and end of the MLS platform's trajectory. The areas with tie points are recognizable where the trajectory (in green colour) is not visible, while the trajectory is only visible in the areas without the tie points. The areas without 3D-3D correspondence do not necessarily mean that matching in all image pairs failed. It could just imply that there were insufficient matches that could be traversed for multiview matching. To quantify the availability of 3D-3D correspondences per tile, the number of tie points for each tile is plotted in Figure 4.23 and Figure 4.24 for MLS_DATA-I and MLS_DATA-II respectively.







Figure 4.22: All obtained 3D-3D correspondences for MLS_DATA-I and MLS_DATA-II, a) and b) respectively. A2A tie points are shown as blue dots. The A2P points come underneath the A2A tie points due to the level of scale. MLS trajectory is plotted as a green curve in Amersfoort / RD New.



Figure 4.23: The number of 3D-3D correspondences obtained for each point cloud tile of MLS_DATA-I, the results are divided into three sub-plots, a), b) and c), showing results of 155 tiles in ascending order.



Automatic Extraction of Accurate 3D Tie Points for Trajectory

Figure 4.24: The number of 3D-3D correspondences obtained for each point cloud tile of MLS_DATA-II, the results are divided into two subplots, a) and b), showing results of 44 tiles in ascending order.

To relate the number of 3D-3D correspondences to the A2A and A2P tie points, the results of tile 18 and tile 2 are provided in Table 4-3. It confirms that not all A2A tie point end up in the 3D-3D relation with A2P points because the number of tie points is much larger than the obtained 3D-3D correspondences. Moreover, it shows that the bottleneck is the establishment of A2P point because the number of 3D-3D correspondences is not more than the A2P points.

Table 4-3: Comparison of the number of tie points and obtained 3D-3D correspondences.

Dalasel	me no.	AZP LIE POINL	AZA LIE POINL	3D-3D correspondence
Ι	18	124	865	120
II	2	80	298	50

To understand the reliability of the tie point extraction, tiles are grouped based on the number of tie points and the number of views. For MLS_DATA-I, and II this grouping is given in Table 4-4. There are three groups in this table, the first group contains the number of tiles with more than 10 tie points and 2D keypoints matched across more than two views. If a tile fits this criterion, the tie points produced are considered reliable. The second group contains tiles with tie points only generated from a stereo view, these tie points are considered less reliable compared to the first group. The last group contains tiles with less than 10 tie points, most of the tiles in this group have no tie point at all.

Table 4-4: Differentiation of tiles based on the number of tie points and the number of views involved in both the test datasets.

DATASET	TOTAL TILES	>10 TIE POINTS AND >2 VIEWS	>10 TIE POINTS AND 2 VIEWS	<10 A2P TIE POINTS
I	154 (100%)	84 (55%)	112 (73%)	41 (27%)
11	47 (100%)	28 (60%)	44 (93%)	9 (20%)

The 3D-3D correspondences of tile 2 of MLS_DATA-II are plotted in Figure 4.25. In this figure, each tie relation is plotted as a yellow vector from each corresponding A2P tie point towards A2A tie point. At the left side of this figure, point cloud tile is superimposed on the aerial orthoimage. The road marks of red and white colour are from the point cloud and aerial orthoimage respectively. A magnified view of a subpart of this result is provided at the right side of the figure.

To realize the upper bound of error in MLS_DATA-II, the probability density functions of the error distribution based on each 3D-3D correspondence, are plotted in Figure 4.26, the parameters of three distributions are; $\mu_{\Delta x} = -0.46$ m, $\sigma_{\Delta x} = 0.39$ m, $\mu_{\Delta y} = 0.23$ m, $\sigma_{\Delta y} = 0.79$ m and $\mu_{\Delta z} = 0.73$ m, $\sigma_{\Delta z} = 0.74$ m.





Amersfoort / RD New.



Figure 4.26: Probability Density Functions (PDF) of the ΔX (m), ΔY (m) and ΔZ (m) of 3D-3D correspondences of MLS_DATA-II.

4.4.8 Reasons and implications for areas without 3D-3D correspondence

For the trajectory adjustment, the 3D-3D correspondences are needed consistently. During an interval of no 3D-3D correspondences, the trajectory adjustment procedure will solely rely on the INS based navigation. This can be a problem for reliable position estimation. Therefore, it is important to examine consecutive tiles without 3D-3D correspondences. Two such cases are marked on the plots in Figure 4.23 and Figure 4.24. In MLS_DATA-I, the longest interval without any 3D-3D correspondence is between tile 22 and 27 as related patches are shown in Figure 4.27 (a) and (b). This interval lasts for 48 seconds and roughly 556 metres in distance on the road surface. This interval is on the Erasmusbrug bridge, which is almost featureless in terms of road marks, except for tile 22, where the mobile laser-scanning car has travelled on the other side of the road without road markings. Normally, this area should not have a GNSS signal problem because there are no tall surrounding buildings. For MLS DATA-II, the time interval without any 3D-3D correspondence consists of tile 7 and 8. It lasts 13 seconds in which the car drove 140 metres. It is noticeable in Figure 4.27 (c) and (d) that both AIPs and MLS images have no road marks and no matchable features.



Figure 4.27: Longest consecutive areas without 3D-3D correspondence. The AIPs and MLS images from the MLS_DATA-I are in (a) and (b) respectively. The AIPs and MLS images from the MLS_DATA-II are in (c) and (d) respectively.

To analyse the reliability of the developed method, we also examine individual problematic tiles from the MLS_DATA-I. This is different from the 2D-2D matching problems discussed at the end of section 4.4.4. Instead, here we only discuss cases when A2A or A2P tie points may have been extracted but together failed to produce 3D-3D correspondences. For this analysis, we only discuss tiles having road marks, contrary to the tiles shown in Figure 4.27. Thereby, we can comprehend the expectation of our method concerning the extraction of 3D-3D correspondence by providing reasons for unsuccessful matching scenarios. These cases are shown in Figure 4.28 and discussed in the following.

One of the main problems faced by matching are shadows of buildings or other objects. Shadows of traffic and building can modify the description of the feature to a level that cannot be handled by the descriptor matcher. The features effected by shadows are not always matchable, unlike the successful case shown earlier in Figure 4.18. The shadows cast by building(s) can darken entire road in aerial images as shown by tile 38 (b), but the laser scanning is not affected by the shadows. Interestingly, the case tile 38 (a) is able to produce matches because, relatively, the feature descriptions are similar. However, the case of tile 38 (b) failed, because the laser scanner showed the road surface details that are not equally captured in aerial images. If the situation is not as extreme as in tile 38, the tiles with shadows can still be matched as shown in Figure 4.18. The small shadows can arbitrarily change the image information and in some cases make it impossible to match. We expected at least some matches on the right side of the patches of tile 75 (b). However, the right side of the image patches have shadows of tree branches and therefore did not produce any matches. The occlusion by tree branches in aerial images is a significant problem for both AIP-to-AIP and MLS image-to-AIP matching because many roads have trees. The worst-case scenario is when road marks are partially occluded by tree branches and there is a shadow of building, e.g. in tile 66. In this situation, both (a) and (b) failed, especially (a) because tree branches appeared relocated in aerial images. Detection and matching of static shadows in aerial images is another problem. If a shadow stays at the same place in two consecutive AIPs, then it could be matched, as the shadow of a pole intersected middle of the zebra crossing in tile 70 (a). However, these types of matches are rejected during the search of 3D-3D correspondence because there are no shadow-based features in the MLS images.

Another scenario causing problems is when completely new road marks appear in the images, e.g. tile 33 and 149. Matches are still established for case (a) in both tiles, however, the case (b) was not successful in both tiles because the point cloud was acquired after the aerial images were captured. Other physical changes in the scene can also affect the matching, as in tile 75 (b). Here, a new construction site was scanned in the point cloud making this area unmatchable in MLS images.

Tiles 70 and 55 are two examples where we expected to match successfully. However, matching failed in case of tile 70 (b). The variation in the laser reflectance intensity is small and features are not clearly distinguishable. In the case tile 70 (a), the road marking near a car parking, the intersection of the road marking with the pavement, and the edge of pavement are successfully matched. However, in the MLS image, the parked cars occluded many road markings. Moreover, the pavement edges are not detectable in the laser scans and neither the intersection of pavement with the road markings. In tile 55, point cloud density is low due to rapid change in the elevation of the scanned road, in this and other similar cases sparse points' projection can produce pixelation effect in the MLS images. Moreover, the matching failed because the aerial images have shadows of multiple thin poles over the main road mark.

Road marks in tiles 66 and 70 are not properly detectable in both MLS image and AIP. The road in these images is a brick-road and white bricks are used to imitate the painted road markings. Therefore, the size of these road marks is smaller and the reflected white colour is dissimilar and rather faded compared to painted road marks.

Our research particularly focuses on the crucial step of MLS image-to-AIP matching due to the difficulty involved. The AIP-to-AIP matching is indeed necessary, but it is not the focus of our research. The importance of the MLS image-to-AIP matching is also apparent from the matching results presented in Figure 4.28, where the tie points could not be extracted because the MLS image-to-AIP matching cases were unsuccessful while most of the AIP-to-AIP cases were successful.





Figure 4.28: Tiles failed to produce 3D-3D correspondence. (a) AIP-to-AIP matches. (b) MLS image-to-AIP matches.

4.5 Conclusions

This paper has described a fully automatic method to extract 3D tie points for the MLS platform trajectory adjustment in GNSS denied urban areas. The developed method is highly reliable in extracting 3D points when there are road marks in the datasets. Moreover, no manual intervention is needed, and our implementation can be set up to run as a one-click solution. Our procedure process the whole of MLS_DATA-I in roughly a day on a normal personal computer. However, further optimization of the procedure is out of the scope of the research project for mainly two reasons; first that the developed procedure is a post-processing solution only designed as an offline application, second that the current performance of the procedure is already acceptable compared to the manual correction procedure, which can easily takes up to weeks or more. Moreover, the processing time of our procedure can be further decreased by running it on the cloud-computing setup, which is a normal industrial practice.

The main contribution in this paper comprises two improvements to our earlier developed 2D matching technique Hussnain et al. (2016);

- 1) We improved the 2D registration from pixel-level to subpixel-level.
- We developed a reliable method to compute subdecimetre-level accurate 3D tie points from 2D-2D correspondences between the point cloud images and aerial images.

As the newly developed 2D registration technique provides subpixel accurate correspondences. Considering the aerial image GSD of 10 cm, we expect to obtain subdecimetre-level accurate 3D tie points. To our knowledge, we did not come across any similar method which can match point cloud with aerial imagery and provide subpixel-level 2D correspondences. The accuracy of the obtained 3D tie point is not tested in this paper because according to our broader project workflow, it is feasible after the adjustment of the trajectory from the tie points, this needs more research work and a publication of its own. We have already published the results of trajectory evaluation from obtained tie points. The 3D tie points extracted from the MLS_DATA-I has already been utilized for the trajectory adjustment and evaluated in Hussnain et al. (2018). It is important to note that the verification of the adjusted point cloud is difficult in the areas without any road marks since the checkpoint based evaluation also requires road markings. However, verification is still possible by intentionally removing tie points in areas well registered and verified. However, that is future work related to Hussnain et al. (2018).

The reliability of the developed method is assessed by analysing the robustness of the matching on highly dissimilar datasets. The method was tested with two datasets MLS_DATA-I and -II, and various experiments are performed. The reliability of the tie points is defined by the number of views involved. For MLS_DATA-I, our method produced reliable 3D tie points for 55% of the tiles. In the case of MLS_DATA-II, 60% of the tiles yielded reliable 3D tie points. In MLS_DATA-I, 27% of the tiles provide less reliable or no tie points, whereas in MLS_DATA-II these tiles account for 20%. Comparatively, the MLS_DATA-II has more road markings. This shows that datasets with consistently available road markings can lead to an improvement in the number of tiles successfully matched. Interestingly, the overlapping area where both datasets produced

plentiful tie points is near the same landmarks, which proves the consistency of the developed method in obtaining the tie points. Similarly, successful registration of tiles in overlapping areas also confirms that the different tile size does not affect the tie point extraction. Effectively, the small tiles can avoid processing unnecessary point cloud points.

The experiments for the trajectory adjustment in Hussnain et al. (2018) have shown that the distribution of the tie points within a tile is not important, also it is not essential to have many tie points per tile. The more important factor for the trajectory adjustment is to have consistent availability of the tie points within a given time interval. The time interval of no tie point is important to consider because in this interval trajectory is based on the IMU only, so we can expect the positioning quality to deteriorate. The earlier experiments on trajectory adjustment have shown that the time interval of 9 seconds consisting of 100 metres of trajectory can be constructed without losing the desired positioning accuracy. It confirms that successful matching is not necessary for every consecutive tile and certainly not for every tile.

Problematic scenarios in urban areas are places without any road marks. Fortunately, most of these areas have plentiful building-façades information in the MLS point cloud. Moreover, oblique images can also increase the accuracy and reliability of the positioning due to the better image intersection geometry. Similar to the road mark feature matching, building-façade features in MLS point cloud can be matched with aerial images. However, due to the minimum visibility of façade in nadir aerial imagery, we believe that the oblique imagery can cover the façade detail necessary for the registration. The main workflow for the tie point extraction remains the same, only the feature detection should be adopted for the point cloud to oblique image matching.

Acknowledgements

This work is part of the research programme OTP with project number 13589, which is financed by the Dutch Research Council (NWO). We would like to thank Topcon© for providing the mobile laser scanning datasets and Cyclomedia© for the aerial imagery.

Reference for chapter 4

- Chiang, K.-W. and Y.-W. Huang, 2008. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. Applied Soft Computing. 8: 722-733.
- Conte, G. and P. Doherty, 2008. An integrated UAV navigation system based on aerial image matching. 2008 IEEE Aerospace Conference, IEEE: 1-10.

- Cui, Y. and S. S. Ge, 2003. Autonomous vehicle positioning with GPS in urban canyon environments. IEEE transactions on robotics and automation. **19:** 15-25.
- Ding, M., K. Lyngbaek and A. Zakhor, 2008. Automatic registration of aerial imagery with untextured 3d lidar models. Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE: 1-8.
- Ding, W., J. Wang, C. Rizos and D. Kinlyside, 2007. Improving adaptive Kalman estimation in GPS/INS integration. Journal of Navigation. **60:** 517-529.
- Förstner, W. and E. Gülch, 1987. A fast operator for detection and precise location of distinct points, corners and centres of circular features. Proc. ISPRS intercommission conference on fast processing of photogrammetric data: 281-305.
- Frueh, C., R. Sammon and A. Zakhor, 2004. Automated texture mapping of 3D city models with oblique aerial imagery. 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Proceedings. 2nd International Symposium on, IEEE: 396-403.
- Fruh, C. and A. Zakhor, 2001. 3D model generation for cities using aerial photographs and ground level laser scans. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, IEEE. 2: II-II.
- Gao, Y., X. Huang, F. Zhang, Z. Fu and C. Yang, 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences. **40:** 41.
- Godha, S., M. Petovello and G. Lachapelle, 2005. Performance analysis of MEMS IMU/HSGPS/magnetic sensor integrated system in urban canyons. Proceedings of the ION GNSS: 1977-1990.
- González-Aguilera, D., P. Rodríguez-Gonzálvez and J. Gómez-Lahoz, 2009. An automatic procedure for co-registration of terrestrial laser scanners and digital cameras. ISPRS Journal of Photogrammetry and Remote Sensing. 64: 308-316.
- Haala, N., M. Peter, J. Kremer and G. Hunter, 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. The international archives of the photogrammetry, remote sensing and spatial information sciences. **37**: 1119-1127.
- Harris, C. and M. Stephens, 1988. A combined corner and edge detector. Alvey vision conference. **15:** 50.
- Hartley, R. and A. Zisserman, 2003. Multiple view geometry in computer vision, Cambridge university press.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2016. Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLI-B1: 609-616.

- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2018. AN AUTOMATIC PROCEDURE FOR MOBILE LASER SCANNING PLATFORM 6DOF TRAJECTORY ADJUSTMENT. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-1: 203-209.
- Javanmardi, M., E. Javanmardi, Y. Gu and S. Kamijo, 2017. Towards High-Definition 3D Urban Mapping: Road Feature-Based Registration of Mobile Mapping Systems and Aerial Imagery. Remote Sensing. **9:** 975.
- Jende, P., Z. Hussnain, M. Peter, S. Oude Elberink, M. Gerke and G. Vosselman, 2016. Low-Level Tie Feature Extraction of Mobile Mapping Data (mls/images) and Aerial Imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XL-3/W4: 19-26.
- Kaartinen, H., J. Hyyppä, A. Kukko, A. Jaakkola and H. Hyyppä, 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. Sensors. **12**: 12814-12835.
- Kukko, A., 2013. Mobile laser scanning–system development, performance and applications. Publications of the Finnish Geodetic Institute, Veröffentlichungen des Finnischen Geodätischen Institutes. Finland. 153.
- Lee, S. C., S. K. Jung and R. Nevatia, 2002. Automatic pose estimation of complex 3D building models. Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on, IEEE: 148-152.
- Levi, G. and T. Hassner, 2016. LATCH: learned arrangements of three patch codes. Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on, IEEE: 1-9.
- Levinson, J., M. Montemerlo and S. Thrun, 2007. Map-Based Precision Vehicle Localization in Urban Environments. Robotics: Science and Systems, Citeseer. 4: 1.
- Liu, L. and I. Stamos, 2012. A systematic approach for 2D-image to 3D-range registration in urban environments. Computer Vision and Image Understanding. **116**: 25-37.
- Lowe, D. G., 1987. Three-dimensional object recognition from single twodimensional images. Artificial intelligence. **31:** 355-395.
- Markelj, P., D. Tomaževič, B. Likar and F. Pernuš, 2012. A review of 3D/2D registration methods for image-guided interventions. Medical image analysis. **16:** 642-661.
- Meierhold, N., M. Spehr, A. Schilling, S. Gumhold and H. Maas, 2010. Automatic feature matching between digital images and 2D representations of a 3D laser scanner point cloud. Proceedings of the ISPRS Commission. **1501**: 446-451.
- Palenichka, R. M. and M. B. Zaremba, 2010. Automatic extraction of control points for the registration of optical satellite and LiDAR images. IEEE Transactions on Geoscience and Remote sensing. **48**: 2864-2879.
- Parmehr, E. G., C. S. Fraser, C. Zhang and J. Leach, 2014. Automatic registration of optical imagery with 3D LiDAR data using statistical similarity. ISPRS Journal of Photogrammetry and Remote Sensing. 88: 28-40.
- Sheehan, M., A. Harrison and P. Newman, 2013. Continuous vehicle localisation using sparse 3D sensing, kernelised Rényi distance and fast Gauss transforms. Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, IEEE: 398-405.
- Stamos, I., L. Liu, C. Chen, G. Wolberg, G. Yu and S. Zokai, 2008. Integrating automated range registration with multiview geometry for the photorealistic modeling of large-scale scenes. International Journal of Computer Vision. **78**: 237-260.
- Tomazevic, D., B. Likar, T. Slivnik and F. Pernus, 2003. 3-D/2-D registration of CT and MR to X-ray images. IEEE transactions on medical imaging. **22:** 1407-1416.
- Wang, L. and U. Neumann, 2009. A robust approach for automatic registration of aerial images with untextured aerial lidar data. Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE: 2623-2630.
- Wang, R., F. P. Ferrie and J. Macfarlane, 2012. Automatic registration of mobile LiDAR and spherical panoramas. Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, IEEE: 33-40.
- Wolcott, R. W. and R. M. Eustice, 2014. Visual localization within lidar maps for automated urban driving. Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, IEEE: 176-183.
- Zhao, Y., 2011. GPS/IMU integrated system for land vehicle navigation based on MEMS. KTH Royal Institute of Technology, Stockholm, Sweden. Doctoral dissertation: 85.
- Zhou, S. and S. Xiao, 2018. 3D face recognition: a survey. Human-centric Computing Information Sciences. **8:** 35.

5 – Enhanced Trajectory Estimation of Mobile Laser Scanners Using Aerial Images

Abstract

Multipath effects and signal obstruction by buildings in the urban canyon can lead to inaccurate GNSS measurements and therefore errors in the estimated trajectory of Mobile Laser Scanning (MLS) systems. Consequently, the point cloud gets distorted and loses spatial consistency, making it unusable for numerous applications. We aim to enhance the trajectory accuracy from meter level to decimetre-level.

We propose a method for the enhancement of the mobile laser scanning platform trajectory in areas with inaccurate GNSS measurement, by matching the MLS data with aerial images with well-known exterior orientations. The registration between the point cloud and aerial images determine the corresponding locations of features in the point cloud. The trajectory is first represented by B-spline functions and enhanced by updating the spline coefficients using three types of observations. The main observations of our trajectory enhancement method are the correspondences between MLS point cloud and aerial images: it contains high local and global consistency. The second type of observation is based on IMU data, this type provides local consistency. The third type of observation is based on constraints applied to the pitch and yaw rotations of the vehicle. This reduces the degrees of freedom. In this paper, we analyse the quality of the trajectory enhancement under several conditions. Experiments are designed to test the influence of the number and quality of corresponding points and to test different settings for the B-spline representation.

We tested our method on two independently acquired MLS datasets at a test area in Rotterdam. We enhanced the trajectory with all observations and evaluated the enhanced trajectory using checkpoints. Moreover, we performed experiments to find the minimum quantity and quality of the observations needed to achieve the decimetre-level accurate trajectory. The best results are achieved when only the high-quality correspondences are used, which constituted 55% and 60% of all correspondences. The RMSE values of the original Kalman filter results at the checkpoints were 0.26 m, 0.30 m and 0.47 m for the X-, Y-, and Z-coordinates in the first dataset and 1.10 m, 1.51 m, and 1.81 m in the second dataset. After the trajectory adjustment these RMSE values were reduced to 0.09 m, 0.11 m, and 0.16 m for the first dataset and 0.12 m, 0.14 m and 0.18 m for the second dataset. The results confirmed that if there are consistent road marks available in the MLS dataset and the IMU measurements are sufficiently accurate, our method can successfully achieve the near decimetre-level accuracy in the MLS dataset.

5.1 Introduction

Multipath effects and signal obstruction by buildings in the urban canyon can lead to inaccurate GNSS measurements and therefore errors in the estimated trajectory of Mobile Laser Scanning (MLS) systems. Kukko (2013) demonstrated that the GNSS measurement accuracy can worsen to more than 50 cm during an outage of GNSS signals. In this case, the acquired point cloud quality suffers from an inaccurate trajectory and the 3D data becomes less useful for mapping applications. Under ideal circumstances, without any GNSS signal outage and multipath effect, the state-of-the-art Mobile Mapping (MM) platforms can achieve 2-3 cm accuracy, estimated by Haala et al. (2008); Kaartinen et al. (2012). However, this is not possible in urban canyons. Accordingly, the task of trajectory correction is very crucial to ensure the quality of the mobile laser scanning point clouds. Commercial software first tries to correct the trajectory by automatic registration of multiple passes, similar to the techniques reported by Levinson et al. (2007), Ding et al. (2007); Zhao (2011). In this vein, Hunter et al. (2006) and Bornaz et al. (2003) proposed a consecutive strip adjustment to improve the misalignments in the MLS data sets. Similarly, Bosse et al. (2009) described a scan-matching method based on iterative closest points (ICP) to recover an accurate trajectory. However, during long-term GNSS signal outages, the final achieved accuracy still remains in metres, as expressed by Chiang et al. (2008). Furthermore, all of these techniques can only be used to increase the relative accuracy and requires data that has multiple scans/passes of the same scene. Due to the high cost of mobile laser scanning per hour, it is, however, highly desirable to scan an area once and as quickly as possible.

The second step in practice is to measure GCPs in the target area and then carefully handpick their correspondences in the MLS point cloud. Although some latest software provide assistant to automatically detect such landmarks, the final decision and effort remain with a human operator. The last step adjusts and improves the erroneous trajectory based on the established correspondences and subsequently regenerates the MLS point cloud. However, manual acquisition and handpicking of the GCPs is very labour intensive, costly and error-prone. Therefore, an automatic method is desirable to replace the manual correction, especially to improve the MLS platform's trajectory.

This paper describes an automatic 6 degree of freedom (DOF) trajectory adjustment technique. We make use of well-oriented aerial imagery as the source for georeferencing the MLS data. The workflow comprises two major steps. The first step is the registration of the MLS point cloud with aerial images to determine correspondences. In the second step, we utilize three types of observation equations for the trajectory enhancement: (1) those resulting from the established correspondences, (2) IMU observations, and (3) soft

constraints to the pitch and yaw rotations of the vehicle. We design various experiments to analyse the accuracy of the trajectory in the GNSS denied areas. Specifically, we analyse how many observations are needed to reach a minimum level of accuracy, and what should be the maximum time of maintaining positioning accuracy on the basis of IMU data only. We model the six pose parameters of the trajectory with B-spline functions. For modelling with sufficient accuracy, the optimal values of the curve order and knot interval are determined.

In the remainder of this paper, Section 5.2 first describes the literature, which aims at MLS point correction, then discusses the literature, which is related to the trajectory estimation for the MM applications. Section 5.3 describes our method of the trajectory adjustment by introducing the various observation equations. Section 5.4 prepares the design of the experiments and section 5.5 provides results of the experiments over the datasets acquired from two different sources.

5.2 Related work

It is convenient to update the local interval of the B-spline based on the changes in the local control points since each B-spline is nonzero only over a certain interval and the updated control points only change coefficients of related B-splines. In this vein, for micro aerial vehicles, Usenko et al. (2017) updated the local part of the B-spline trajectory, when an unmodelled obstacle arrived in the pre-processed global trajectory. However, the main purpose of this study was to show that the developed system could accommodate the real-world dynamics into the trajectory and not the assessment of the achieved accuracy of the positioning.

The improvement of the mobile mapping platform localisation can lead to the correction of MLS data. Gao et al. (2015) have reported RMS values of $\Delta X=8.6$ cm, $\Delta Y=6.3$ cm, and $\Delta Z=10.6$ cm in the corrected point cloud. However, they evaluated the accuracy of point cloud before and after the correction using checkpoint from the aerial imagery. Moreover, the adjustment of the point cloud was limited to 3DOF.

The B-spline representation of 6DOF trajectory for mobile mapping systems is a common tool for trajectory correction procedures. For example, for a visual odometry application, Patron-Perez et al. (2015) unified the discrete camera poses with continuous unsynchronized IMU observations to estimate the continuous camera trajectory. They used a rolling shutter camera model, which introduced individual time stamp for each pixel in continuous pose over time trajectory. However, they only presented results for simulated datasets and the accuracy and reliability of the developed system was not discussed. Wolcott et al. (2014) developed image-based navigation for self-driving systems. The mobile mapping camera images are registered with previously acquired 3D lidar data by maximizing the normalized mutual information method. Their developed approach achieved an RMS error, longitudinal 19.1~45.4 cm, and lateral 14.3~20.5 cm. Kümmerle et al. (2011) developed a SLAM procedure for mobile laser scanning point cloud while using aerial images as refine map, they have an overall accuracy of 20 cm.

Recently, Karam et al. (2018) proposed an evaluation method for indoor laser scanning points clouds. They developed an indoor mapping system that generates a 3D point cloud based on a feature-based 6DOF SLAM method. The evaluation method proposed is influenced by the configuration of the laser scanner and rely on the indoor building structural constraints. To utilize these constraints the same and preferably flat surfaces needs to be scan multiple times which is not the case with our application.

Vosselman (2014) designed an indoor laser scanner system to estimate the 6DOF B-spline based trajectory using SLAM. The developed technique showed that the constraint derived from the indoor wall structures (or planes) of simulated indoor environment can be used to estimate the 6DOF B-spline based trajectory. Likewise, in our case, the constraints come from the 3D A2P tie points, which are forward intersected from aerial image correspondences.

In our previous publication (Hussnain et al. (2018), we showed preliminary results of the trajectory adjustment method, however, the accuracy of the obtained trajectory was not assessed in areas without 3D-3D correspondences. In this paper, we analyse the accuracy of the trajectory in the areas where our earlier developed method Hussnain et al. (2019) did not extract 3D tie points.

Recently, Javanmardi et al. (2018) proposed a technique for MLS platform localization based on the 'abstract maps'. However, this technique utilizes accurate maps generated from an accurate prior point cloud. Moreover, the abstract maps are an estimation of the structures in the prior point cloud, which can introduce errors. Furthermore, the localization accuracy will be always lower than the prior point cloud accuracy. In our case, we do not consider that a (prior) accurate MLSPC is already available. In a previous publication, Javanmardi et al. (2017) have shown the correction of the MLSPC by using multiple reference data sets, including aerial images. However, their approach constrained the registration problem to the only 2DOF, which only works in an ideal scenario when the error does not occur in remaining coordinates.

5.3 B-spline based 6dof trajectory adjustment

In this paper, we further develop and analyse the trajectory adjustment procedure initially formulated in Hussnain et al. (2018). Here, we provide the observation equations leading to the system of matrix equations. This representation helps to elaborate on the core working of B-spline adjustment from the computational point of view.

The workflow of the adjustment procedure is presented in Figure 5.1. The developed workflow employs correspondences between the point cloud and aerial images, IMU measurements, and soft constraints on the trajectory. The initial approximate estimate of the trajectory is obtained from the standard Kalman filtering solution which comprises the errors due to signal multi-path and signal occlusion. The workflow incrementally improves the trajectory estimates until it converges.



Figure 5.1: Workflow of trajectory adjustment procedure.

In the remainder of this section, we first start with the estimation of optimal parameters for the representation of trajectory with B-splines. Then we describe the observation equations obtained from 3D-3D correspondences derived from Hussnain et al. (2019), IMU measurements of angular velocities and accelerations, and soft constraints.

5.3.1 B-spline order and knot interval optimization

To model the MLS trajectory with B-splines of sufficient accuracy, we need to estimate the optimal values of the knot interval and the curve order. The B-spline with optimal parameters only needs a minimum number of coefficients. Otherwise, for every additional knot interval, an extra spline coefficient needs to be estimated for each of the six pose parameters. Because we want to achieve the decimetre-level accuracy in the improved MLS point cloud we should ensure that the errors introduced by modelling the trajectory with splines are significantly smaller. We set the maximum position error introduced by the spline modelling to 4 cm.

For the rotational error, the criterion is also based on the maximum positional error allowed. Suppose a mobile mapping car observes a 3D point 20 m away (based on an average road width in the datasets), then the maximum error allowed in any angle is $\theta_{\varepsilon} = 0.12^{\circ}$ if the effect of the error on the coordinate calculation should remain within 4 cm, this situation is shown in Figure 5.2.



Figure 5.2: Maximum rotational error allowed.

Once the trajectory is modelled with sufficient accuracy, the step is to define the observation equations for the B-splines adjustment. In section 5.5.2 we analyse different combinations of spline order and knot intervals to find the splines with the minimum number of parameters which still fulfil the accuracy requirement.

5.3.2 3D-3D correspondence observation

The correspondence between aerial images and the MLS point cloud is the most important input to our adjustment procedure because the aerial images provide the georeferenced capability. The correspondences can be derived automatically. This observation is obtained from the result of automatic matching between the rasterized point cloud and aerial images and was described in Hussnain et al. (2016) and Hussnain et al. (2019). The correspondences determined by the matching technique lead to the computation of 3D tie points. Points in common between multiple aerial images are used to estimate 3D points with a multiview triangulation. These 3D points will be referred to as A2A tie points. The corresponding locations of these points in the MLS point cloud are determined by matching rasterized point clouds and aerial images and will be referred to as A2P tie points. The 3D-3D correspondences are then established by searching association between A2A and A2P tie point sets.

In order to linearize the 3D-3D correspondences observation for B-spline based adjustment, we start with the point cloud point observation. A laser scanner observes a 3D point X_{PC}^{C} in the car coordinate system, which can be rotated to the world coordinate system with a rotation matrix R_{C}^{W} representing the attitude of the car in the world coordinate system. The point X_{PC}^{C} can then be translated to the world coordinate system. This relation is represented in Eq. 1. The upper indices *C* and *W* indicate the coordinate system of the car and world respectively. The lower indexing specifies the source of the coordinate vector, e.g. *AI* for aerial image and *PC* for the point cloud.

$$X_{PC}^{W} = R_{C}^{W}(t) X_{PC}^{C} + T_{C}^{W}(t)$$
(1)

Here X_{PC}^{W} is the point cloud point as observed in the world coordinate system. The time dependency of $R_{C}^{W}(t)$ and $T_{C}^{W}(t)$ represents rotation and translation that change over time t.

Because the GNSS data was unreliable, we want to re-estimate the rotation $R_c^W(t)$ and translation $T_c^W(t)$ based on the A2A 3D tie point X_{AI}^W , this relation is represented in Eq. 2.

$$X_{AI}^{W} = R_{C}^{W}(t) X_{PC}^{C} + T_{C}^{W}(t)$$
(2)

The right-hand side of the Eq. 2 represents the A2P tie point corresponding to A2A tie point. Figure *5.3* depicts that the correct location of the A2P (point cloud) point w.r.t the aerial imagery is, in fact, the location of the A2A point. Therefore, when the trajectory is accurate then the difference between A2A and A2P tie points diminishes.



Figure 5.3: 3D-3D correspondence observation based on A2A and A2P 3D tie points.

The original GNSS and IMU data have been processed by a Kalman filter to estimate the rotation $R_{C,Kalman}^{W}(t)$ and translation $T_{C,Kalman}^{W}(t)$ between the car and the world coordinate system. The Kalman filter results have been used to reconstruct the original point cloud points X_{PC}^{C} in the car coordinate system as in Eq. 3.

$$X_{PC}^{C} = R_{C,Kalman}^{W^{T}}(t) \left[X_{PC}^{W} - T_{C,Kalman}^{W}(t) \right]$$
(3)

The result of the Kalman filtering is used to obtain the approximate spline coefficients of the six pose parameters over time. The six pose parameters comprise three angles $\omega(t), \varphi(t), \kappa(t)$ and a translation vector described by translations along the three axes, $T_X(t), T_Y(t), T_Z(t)$. The modelling of e.g. $\omega(t)$ by a B-spline function is given in Eq. 4.

$$\omega(t) = \sum_{i} \alpha_{\omega,i} B_i(t) \tag{4}$$

Where B_i is *i*'th B-spline function and $\alpha_{\omega,i}$ its coefficient to be estimated. Eq. 2 is linearized with the upper index ⁰ denoting the approximate value. In the first iteration, the output of the Kalman filter is used to obtain the spline coefficients for the approximate rotation and translation. The time dependency (*t*) is omitted below to shorten the expression.

$$X_{AI}^{W} - R_{C}^{W^{0}} X_{PC}^{C} - T_{C}^{W^{0}} = \begin{pmatrix} \sum_{i}^{L} \Delta \alpha_{T_{X},i} B_{i} \\ \sum_{i}^{L} \Delta \alpha_{T_{Y},i} B_{i} \\ \sum_{i}^{L} \Delta \alpha_{T_{Z},i} B_{i} \end{pmatrix}$$

$$+ \frac{\partial R_{C}^{W}}{\partial \omega} X_{PC}^{C} \sum_{i} \Delta \alpha_{\omega,i} B_{i} + \frac{\partial R_{C}^{W}}{\partial \varphi} X_{PC}^{C} \sum_{i} \Delta \alpha_{\varphi,i} B_{i}$$

$$+ \frac{\partial R_{C}^{W}}{\partial \kappa} X_{PC}^{C} \sum_{i} \Delta \alpha_{\kappa,i} B_{i}$$
(5)

At the left-hand side is the observed misclosure between the 3D tie point obtained from the aerial images and the 3D point obtained in the point cloud expressed in the world coordinate system. On the right-hand side are the increments to the spline coefficients multiplied with the elements of the Jacobian.

The 3D-3D correspondences are not available consistently over the whole trajectory. Instead, their availability is dependent on the presence of the road markings. Moreover, even if available, tie points are sparse and cannot cover every spline interval. Therefore, the positioning has to be supported by inertial navigation.

5.3.3 Acceleration observation

The IMU observes accelerations for the car's position in the sensor coordinate system S. These are denoted \ddot{X}^S_{IMU} . After rotation to the car coordinate system by the rotation matrix R^C_s and then to the world coordinate system by R^W_c , these accelerations should correspond to the second derivatives of the car's location in the world coordinate system, i.e. \ddot{T}^W_c . Hence,

$$\ddot{T}_C^W = R_C^W R_S^C \, \ddot{X}_{IMU}^S \tag{6}$$

The angles of the rotation matrix R_s^c are calibrated and remain constant during the laser scanning operation.

To model the second derivative of the car location T_C^W , it is straightforward to take the second derivative of spline as it is a polynomial function, like for $T_X(t)$:

$$\ddot{T}_X(t) = \sum_i \alpha_{T_X,i} \, \ddot{B}_i(t) \tag{7}$$

The B-spline coefficients $\alpha_{T_X,i}$ to be estimated remain exactly the same. It's only the B-splines functions themselves that need to be differentiated twice.

Also, note that the differentiation only applies to the B-splines of the translation and not to those of the rotation angles in R_c^W .

With this the linearized equation becomes:

$$\ddot{T}_{C}^{W^{0}} - R_{C}^{W} R_{S}^{C} \ddot{X}_{IMU}^{S} = \begin{pmatrix} \sum_{i}^{i} \Delta \alpha_{T_{X},i} \ddot{B}_{i} \\ \sum_{i}^{i} \Delta \alpha_{T_{Y},i} \ddot{B}_{i} \\ \sum_{i}^{i} \Delta \alpha_{T_{Z},i} \ddot{B}_{i} \end{pmatrix}$$
$$- \frac{\partial R_{C}^{W}}{\partial \omega} R_{S}^{C} \ddot{X}_{IMU}^{S} \sum_{i} \Delta \alpha_{\omega,i} B_{i} - \frac{\partial R_{C}^{W}}{\partial \varphi} R_{S}^{C} \ddot{X}_{IMU}^{S} \sum_{i} \Delta \alpha_{\varphi,i} B_{i}$$
$$- \frac{\partial R_{C}^{W}}{\partial \kappa} R_{S}^{C} \ddot{X}_{IMU}^{S} \sum_{i} \Delta \alpha_{\kappa,i} B_{i}$$
(8)

For the first iteration, approximate values of acceleration $\ddot{T}_{C}^{W^{0}}$ are obtained from the Kalman filter solution.

5.3.4 Angular velocity observation

The IMU also observe angular velocities in the sensor coordinate system denoted $\dot{\boldsymbol{\omega}}_{IMU}^{S} = [\dot{\omega}_{IMU}^{C} \ \dot{\varphi}_{IMU}^{C} \ \dot{\kappa}_{IMU}^{C}]^{T}$. The first derivatives of the angles describing the rotation from the car to world coordinate systems are denoted $\dot{\boldsymbol{\omega}}_{C}^{W} = [\dot{\omega}_{C}^{W} \ \dot{\varphi}_{C}^{W} \ \dot{\kappa}_{C}^{W}]^{T}$. To determine the relationship between the observed angular velocities, we first need to define the order and direction of rotation R_{C}^{W} from the car coordinate system to the world coordinate system,

$$R_C^W = R_3(\kappa_C^W)R_2(\varphi_C^W)R_1(\omega_C^W)$$

where components of the rotation matrix R_c^W are defined as;

$$R_{3}(\kappa) = \begin{pmatrix} \cos \kappa_{C}^{W} & -\sin \kappa_{C}^{W} & 0\\ \sin \kappa_{C}^{W} & \cos \kappa_{C}^{W} & 0\\ 0 & 0 & 1 \end{pmatrix}; R_{2}(\varphi) = \begin{pmatrix} \cos \varphi_{C}^{W} & 0 & \sin \varphi_{C}^{W}\\ 0 & 1 & 0\\ -\sin \varphi_{C}^{W} & 0 & \cos \varphi_{C}^{W} \end{pmatrix}; R_{1}(\omega)$$
$$= \begin{pmatrix} 1 & 0 & 0\\ 0 & \cos \omega_{C}^{W} & -\sin \omega_{C}^{W}\\ 0 & \sin \omega_{C}^{W} & \cos \omega_{C}^{W} \end{pmatrix}$$

The rotation from the world coordinate system to the car coordinate system is therefore defined by:

$$R_W^C = R_1(-\omega_C^W)R_2(-\varphi_C^W)R_3(-\kappa_C^W)$$

The ω_c^W is the first rotation applied when rotating from the car to the world coordinate system, therefore the angular velocity $\dot{\omega}_c^W$ of the car in the world coordinate system is only observed around the X-axis of the IMU. The first derivative of car's rotation around the Y-axis in the world coordinate system does not directly correspond to the rotational velocity around Y-axis in car coordinate system because Y-axis of the car has been rotated by $-\omega_c^W$ around the X-axis. Therefore, the derivative of φ_c^W should be rotated to the car coordinate system by the rotation matrix $R_1(-\omega_c^W)$. Similarly, the derivative of κ_c^W needs to be rotated around the Y-axis by $R_2(-\varphi_c^W)$ and then X-axis by $R_1(-\omega_c^W)$ before it corresponds to the angular velocity vector in the car coordinate system. Hence,

$$\begin{pmatrix} \dot{\omega}_{IMU}^{S} \\ \dot{\varphi}_{IMU}^{S} \\ \dot{\kappa}_{IMU}^{S} \end{pmatrix} = R_{C}^{S} \begin{pmatrix} \dot{\omega}_{C}^{W} \\ 0 \\ 0 \end{pmatrix} + R_{C}^{S} R_{1} (-\omega_{C}^{W}) \begin{pmatrix} 0 \\ \dot{\varphi}_{C}^{W} \\ 0 \end{pmatrix} + R_{C}^{S} R_{1} (-\omega_{C}^{W}) R_{2} (-\varphi_{C}^{W}) \begin{pmatrix} 0 \\ 0 \\ \dot{\kappa}_{C}^{W} \end{pmatrix}$$

$$\begin{pmatrix} \dot{\omega}_{IMU}^{S} \\ \dot{\varphi}_{IMU}^{S} \\ \dot{\kappa}_{IMU}^{S} \end{pmatrix} = R_{C}^{S} \begin{pmatrix} 1 & 0 & -\sin \varphi_{C}^{W} \\ 0 & \cos \omega_{C}^{W} & \sin \omega_{C}^{W} \cos \varphi_{C}^{W} \\ 0 & -\sin \omega_{C}^{W} & \cos \varphi_{C}^{W} \end{pmatrix} \begin{pmatrix} \dot{\omega}_{C}^{W} \\ \dot{\varphi}_{C}^{W} \\ \dot{\kappa}_{C}^{W} \end{pmatrix}$$

which can be written to define the angular velocity observation equation,

$$\begin{pmatrix} \dot{\omega}_{IMU}^{S} \\ \dot{\phi}_{IMU}^{S} \\ \dot{\kappa}_{IMU}^{S} \end{pmatrix} = R_{c}^{S} S_{W}^{C} \begin{pmatrix} \dot{\omega}_{c}^{W} \\ \dot{\phi}_{C}^{W} \\ \dot{\kappa}_{C}^{W} \end{pmatrix}, \quad S_{W}^{C} = \begin{pmatrix} 1 & 0 & -\sin\varphi_{c}^{W} \\ 0 & \cos\omega_{c}^{W} & \sin\omega_{c}^{W}\cos\varphi_{c}^{W} \\ 0 & -\sin\omega_{c}^{W} & \cos\omega_{c}^{W}\cos\varphi_{c}^{W} \end{pmatrix}$$

or

or

$$\dot{\boldsymbol{\omega}}_{IMU}^{S}(t) = R_{C}^{S} S_{W}^{C} \dot{\boldsymbol{\omega}}_{C}^{W}(t)$$
(9)

where $\dot{\omega}$ is a vector of angular velocities around X-, Y- and Z- coordinates.

In order to model the angular velocities, we use the first derivatives of the B-splines function, e.g. for $\dot{\omega}(t)$

$$\dot{\omega}(t) = \sum_{i} \alpha_{\omega,i} \, \dot{B}_{i}(t) \tag{10}$$

Linearization of Eq. 9 leads to

$$\dot{\boldsymbol{\omega}}_{IMU}^{S} - R_{C}^{S} S_{W}^{C^{0}} \dot{\boldsymbol{\omega}}_{C}^{W^{0}} = R_{C}^{S} S_{W}^{C^{0}} \begin{pmatrix} \sum_{i}^{L} \Delta \alpha_{\omega,i} \dot{B}_{i}(t) \\ \sum_{i}^{L} \Delta \alpha_{\varphi,i} \dot{B}_{i}(t) \\ \sum_{i}^{L} \Delta \alpha_{\kappa,i} \dot{B}_{i}(t) \end{pmatrix}$$
$$- R_{C}^{S} \frac{\partial S_{W}^{C^{0}}}{\partial \omega} \dot{\boldsymbol{\omega}}_{C}^{W^{0}} \sum_{i}^{L} \Delta \alpha_{\omega,i} B_{i} - R_{C}^{S} \frac{\partial S_{W}^{C^{0}}}{\partial \varphi} \dot{\boldsymbol{\omega}}_{C}^{W^{0}} \sum_{i}^{L} \Delta \alpha_{\varphi,i} B_{i}$$
(11)

where the partial derivate of $S_W^{\mathcal{C}}$ w.r.t. ω and φ are

$$\frac{\partial S_W^{C^0}}{\partial \omega} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -\sin \omega_C^W & \cos \omega_C^W \cos \varphi_C^W \\ 0 & -\cos \omega_C^W & -\sin \omega_C^W \cos \varphi_C^W \end{pmatrix}, \qquad \frac{\partial S_W^{C^0}}{\partial \varphi} = \begin{pmatrix} 0 & 0 & -\cos \varphi_C^W \\ 0 & 0 & -\sin \omega_C^W \sin \varphi_C^W \\ 0 & 0 & -\cos \omega_C^W \sin \varphi_C^W \end{pmatrix}$$

5.3.5 Soft constraint observation

The direction of the car is described by the heading κ_c^W and pitch φ_c^W , but these two rotations can also be inferred from the first derivatives of the car's positions T_c^W as shown in Figure 5.4. We can therefore add two constraints to ensure that the rotations are consistent with the trajectory, effectively reducing the degrees of freedom in the transformation from the car to the world coordinate system from six to four.



Figure 5.4: The relationship between positions and direction of the car.

The heading can be inferred from the first derivatives of the X- and Ycoordinates of the car trajectory. As, the first rotation from the world coordinate system to the car coordinate system was defined $-\kappa$. Hence, the soft constraint observation related to the heading of car is,

$$-\kappa_{C}^{W} = \tan^{-1} \left(\frac{\dot{T}_{C,X}^{W}}{\dot{T}_{C,Y}^{W}} \right)$$
(12)

To shorten the notation below, first derivatives are denoted $\dot{T}_{C,X}^W = \dot{T}_X$, $\dot{T}_{C,Y}^W = \dot{T}_Y$ and $\dot{T}_{C,Z}^W = \dot{T}_Z$.

The soft constraint observation related to car heading is linearized to

$$-\kappa_{c}^{W^{0}} - \tan^{-1}\left(\frac{\dot{T}_{X}^{0}}{\dot{T}_{Y}^{0}}\right) = \sum_{i} \Delta \alpha_{\kappa,i} B_{i} + \frac{\dot{T}_{Y}^{0}}{\dot{T}_{X}^{0^{2}} + \dot{T}_{Y}^{0^{2}}} \sum_{i} \Delta \alpha_{T_{X},i} \dot{B}_{i} + \frac{\dot{T}_{X}^{0}}{\dot{T}_{X}^{0^{2}} + \dot{T}_{Y}^{0^{2}}} \sum_{i} \Delta \alpha_{T_{Y},i} \dot{B}_{i}$$
(13)

where \dot{T}_X^0 and \dot{T}_Y^0 are the first derivatives of the X- and Y- coordinates derived from the Kalman filtered trajectory for the first iteration of the adjustment process.

Similarly, the pitch angle of the car in the world coordinate system can be inferred from

$$-\varphi_C^W = \tan^{-1} \left(\frac{\dot{T}_Z}{\sqrt{\dot{T}_X^2 + \dot{T}_Y^2}} \right)$$
(14)

which linearized to

$$-\varphi_{C}^{W^{0}} - \tan^{-1} \left(\frac{\dot{T}_{Z}}{\sqrt{\dot{T}_{X}^{2} + \dot{T}_{Y}^{2}}} \right) = \sum_{i} \Delta \alpha_{\varphi,i} B_{i}$$
$$+ \frac{-\dot{T}_{X}^{0} \dot{T}_{Z}^{0} \sum_{i} \Delta \alpha_{T_{X},i} \dot{B}_{i} - \dot{T}_{Y}^{0} \dot{T}_{Z}^{0} \sum_{i} \Delta \alpha_{T_{Y},i} \dot{B}_{i} + \left(\dot{T}_{X}^{0^{2}} + \dot{T}_{Y}^{0^{2}} \right) \sum_{i} \Delta \alpha_{T_{Z},i} \dot{B}_{i}}{\left(\dot{T}_{X}^{0^{2}} + \dot{T}_{Y}^{0^{2}} + \dot{T}_{Z}^{0^{2}} \right) \sqrt{\dot{T}_{X}^{0^{2}} + \dot{T}_{Y}^{0^{2}}}}$$
(15)

5.3.6 Fixed pose observation

In a normal situation, the mobile mapping car starts from a location with reliable GNSS signals reception. Similarly, we also assume that a reliable start and end poses are known. We fix the starting and end pose of the trajectory by adding two more observation equations; Eq. 17 and Eq. 19. So, when the GNSS signal reception is reliable, we can use the reliable poses from the Kalman filtering. Thus, both the position being estimated and position from the Kalman filtering are equal only at the start and end of the trajectory.

If the reliable start and end poses of the car are unknown, we can rely on the observations of A2P points that are also globally consistent. However, if no

road marks exist at the exact start or end of the trajectory as well, then the small parts of the trajectory at the start and end can have residuals.

$$T_{C,Kalman}^{W} = T_{C}^{W}$$
(16)

This linearizes to,

$$T_{C,Kalman}^{W} - T_{C}^{W} = \begin{pmatrix} \sum_{i} \Delta \alpha_{T_{X},i} B_{i} \\ \sum_{i} \Delta \alpha_{T_{Y},i} B_{i} \\ \sum_{i} \Delta \alpha_{T_{Z},i} B_{i} \end{pmatrix}$$
(17)

Similarly, we can consider that the pose angles are also equivalent.

$$\begin{bmatrix} \omega_{C,Kalman}^{W} \\ \varphi_{C,Kalman}^{W} \\ \kappa_{C,Kalman}^{W} \end{bmatrix} = \begin{bmatrix} \omega_{C}^{W} \\ \varphi_{C}^{W} \\ \kappa_{C}^{W} \end{bmatrix}$$
(18)

Which linearizes to,

$$\begin{bmatrix} \omega_{C,Kalman}^{W} \\ \varphi_{C,Kalman}^{W} \\ \kappa_{C,Kalman}^{W} \end{bmatrix} - \begin{bmatrix} \omega_{C}^{W} \\ \varphi_{C}^{W} \\ \kappa_{C}^{W} \end{bmatrix} = \begin{pmatrix} \sum_{i} \Delta \alpha_{\omega,i} B_{i}(t) \\ \sum_{i} \Delta \alpha_{\varphi,i} B_{i}(t) \\ \sum_{i} \Delta \alpha_{\kappa,i} B_{i}(t) \end{pmatrix}$$
(19)

5.3.7 Trajectory update

We represent pose parameters with 6 individual B-spline functions over time. As B-splines are polynomial functions and are based on coefficients, updating the B-splines involves estimating the changes to the coefficients, e.g. example, estimating $\Delta \alpha_{\omega,i}$ for the roll angle. The linearized observations equations represent these changes to the coefficients. For numerical computation of delta coefficients, we add each observation to a normal matrix and solve the system to obtain these increments to coefficients. The adjustments to the coefficients are performed in an iterative process, to update the trajectory, computed increments are then added to the B-spline coefficients after each iteration. The iterative process converges when new observations yield residuals very close to the previous observations, which means that the trajectory cannot be improved further. It requires 10-20 iterations to converge to the final solution in the experiments.

5.4 Design of the experiments

In this section, we lay out the design of experiments aiming to assess the trajectory after the adjustment. As multiple inputs take part in the adjustment process, various experiments are needed to analyse the effect of individual input on the accuracy. This section proposes several experiments to determine the minimum amount and quality of the tie points and the maximum time without tie points which still allows an accurate estimation of the trajectory. The proposed set of experiments are summarised in Table *5-1*. The description of these experiments is provided in the next sections. Note that the experiment 0 is the original trajectory as it resulted from the Kalman filtering. This is used as baseline for the adjusted trajectories making use of the different types of observation equations.

		Observations for			
Experiment	Experiment	po		Trajactory	
type	No.	Tie points	IMU	Soft constraints	Гајестогу
Result of Kalman filter	0	No	No	No	All
	1	All	Yes	Yes	All
Tie point quantitative	2	Combination of trajectory parts with and without tie points	Yes	Yes	All
anaiysis	3	Reduced but well distributed in all parts	Yes	Yes	All
	4a	Low quality standard deviation >8 cm	Yes	Yes	All
Tie point qualitative analysis	4b	Medium quality standard deviation >5 cm and ≤8 cm	Yes	Yes	All
	4c	High quality standard deviation ≤5 cm	Yes	Yes	All
IMU and soft	5a	No	Yes	Yes	Time- based
constraints- based	5b	No	Yes	Yes	Time- based
trajectory	5c	No	Yes	Yes	Time- based
Implication without Soft constraints	6	All	Yes	No	All

Table 5-1: Categorization of experiments and their related observations.

For the experiments, we use two independently acquired trajectory datasets; Trajectory-I and Trajectory-II as plotted in Figure 5.5 and Figure 5.6

respectively. Both systems are based on different laser scanning and position estimation systems. The Trajectory-I dataset is acquired by Topcon® IP-S3 mobile mapping system, which is a single 360-degree lidar sensor. The poses are estimated by an industrial grade IMU (KVH® CG-5100). The length of Trajectory-I on the road surface is 13 km with the total scanning time of 56 minutes. Only considering the points from the single pass, the mean density of points on the road surface is 320 pts/dm².



Figure 5.5: Trajectory-I dataset.

Trajectory-II is acquired by Topcon® IP-S2 system that is based on two 360degree laser scanners mounted in a cross-configuration. The inertial navigation is based on a built-in MEMS gyroscope. The length of the trajectory on the road surface is 10 km with the scanning operation lasting 45 minutes. Only considering the points from the single pass and single scanner, the mean density of points on the road surface is 74 pts/dm².

Enhanced Trajectory Estimation of Mobile Laser Scanners Using Aerial Images



X (m)

Fifteen aerial images of 20010x13080 pixels were captured with the UltraCam® camera on a manned flight at the approx. height of 4500 m. The quality of the orientation of the aerial images has been assessed by the Pix4Dmapper software using checkpoints, which is accurate to the level of 10 cm GSD. The aerial images have 60% forward overlap and 40% across track overlap. The orientation parameters of both aerial images and MLS datasets are known in the Dutch reference system RD-NAP.

5.4.1 Quantitative analysis for the tie point observation

Experiments 1, 2, and 3 are used to determine the minimum number of tie points needed to accurately correct the trajectory. This is achieved by performing the adjustment with gradually decreasing quantities of the tie points and then evaluating the accuracy of each adjustment with checkpoints, which have been surveyed in the terrain by GNSS and corresponding locations are hand-picked in the MLS point cloud.

The 2nd experiment reduces the tie points by completely removing them from arbitrary locations. The arbitrary reduction can therefore also eliminate the tie points in areas where they are already scarce. Thereby, this experiment analyses the situation when some segments provide plentiful tie points while the others provide none. This leads to a situation where some locations in the trajectory may not be improved as the trajectory reconstruction in these parts will completely depend on the IMU.

In the 3rd experiment, we remove the tie points such that the remaining points are equally distributed as much as possible along the trajectory. To fulfil these criteria, we first calculate the distribution of the tie points along the trajectory over time. Then the tie points are numbered in bins w.r.t. the interval size that

Figure 5.6: Trajectory-II dataset.

is optimized in section 5.5.2, and then the bins containing more tie points are reduced first to get a more even distribution. This strategy ensures that the reduced tie points remain well distributed along the trajectory because otherwise emptying bins with few or single tie point can create a scarcity of tie points at arbitrary locations.

5.4.2 Qualitative analysis for the tie point observation

Experiments 4a, 4b, and 4c are conducted to analyse the relationship between the quality of the tie points and that of the reconstructed trajectory. The quality of the tie point is defined by the standard deviation of the error in the multiview triangulation. By only using the tie points with a certain range of standard deviations, we can analyse the relationship between the quality of the tie points and the accuracy achieved in the trajectory after the enhancement.

The experiments 4a, 4b and 4c require a trajectory containing tie point sets of varying standard deviations. Based on the standard deviation, we can divide tie point sets into low, medium and high quality. Normally, the high-quality tie points are computed from more than two aerial images and entail small backprojection error. Moreover, the high-quality tie points are the result of triangulation of subpixel accurate 2D keypoints which are the best estimation of 2D corner features in the aerial image, whereas the low-quality tie points have the opposite characteristics. As indicated in Table *5-1*, we separately enhance the trajectory from the tie points of high-quality (4c) with standard deviation ≤ 5 cm, medium-quality (4b) with standard deviation ≥ 5 cm and ≤ 8 cm and low-quality (4a) with standard deviation ≥ 8 cm, and then compare the results to establish the minimum level of quality needed.

5.4.3 Assessment of trajectory constructed only by IMU and soft constraints observations

The IMU observations provide inertial navigation, where the 3D-3D correspondence observation provides a certain correction. However, in locations where 3D-3D correspondences are not available, the IMU-based positioning suffers from the accumulation of noise over time and drift errors in the inertial measurements. Even when 3D-3D correspondences are available, in the construction of the entire trajectory, there are small segments of various lengths that are reconstructed based on IMU observations only. In the 5th experiment, we prepare segments of stepwise increasing lengths from the same trajectory. Moreover, we fix the start and end pose of the trajectory segments because not fixing the start end pose results in a singular equation system. The fixed pose observation is explained in Hussnain et al. (2018). An illustration of the start and end poses of three trajectory segments and their unified formation is shown in Figure *5.7*. Between each start and end pose, continuously, the trajectory is constructed only based on the IMU and soft

constraint observations. Fixing the pose at both ends is important because it forces the trajectory to pass through the start and end poses, this situation is compatible with the real world scenario where an interval of no tie points is always adjoined at both ends with the segments containing tie points. Otherwise, if we only fix pose at the start, it can increase the drift error near the end pose. In the experiments 5a, 5b, and 5c, we use both the IMU and soft constraint observations.



Figure 5.7: Illustration of trajectory segments used to assess the reliability of the IMU observations.

5.4.4 Impact of soft constraints

Experiment 6 has been used to distinguish the improvement ascribed to the soft constraints by not applying these constraints and then comparing the result with the same trajectory which has been already improved by the soft-constraints as in experiment 1. All of the above defined experiments employ the soft constraints because it is not necessary to remove them while analysing the influence of the other observations on the adjustment. Therefore, we also use soft constraints for the IMU based trajectory in section 5.4.3 because it focuses on exploiting other available observations except for 3D-3D correspondence.

5.5 Results

In this section, we present the results of the experiments described in the previous section starting from the criteria of the trajectory segment selection, followed by the optimization of the B-spline knot interval and order. Then an important procedure to evaluate an updated trajectory using checkpoints is described. In the end, the main evaluations of the experiments are presented, where each experiment is associated with the analysis of individual observations.

5.5.1 Trajectory selection criteria

To conduct the experiments, we need to select a part of a trajectory where most subparts are supplied with the road marks and thereby the checkpoints are attainable. The trajectory can be considered as a collection of subparts or segments, where each segment is related to a point cloud tile and its associated tie points. The extraction of tie points related to a point cloud tile and its registration to aerial images is described in Hussnain et al. (2019). When all consecutive segments of a trajectory contain tie points, it is feasible to evaluate and ensure that a certain accuracy persists in the whole trajectory. Moreover, it provides a reliable and accurate start and end poses, which are also needed for the IMU-based observation. For the experiments in this section, we select parts of the trajectories, which fulfil these criteria.

5.5.2 Optimal knot interval and B-spline order

We need to understand how well a trajectory can be modelled by B-splines. Therefore, we take a trajectory as processed by the Kalman filter, sample it densely by extracting points at every 10 ms and use those points to reconstruct the trajectory with B-splines. These reconstructed trajectories are then compared against the original trajectory to analyse the errors introduced by the approximation of the trajectory by B-splines. For the optimization, various combinations of knot interval and curve order parameters have been used. Based on preliminary experiments on the spline fitting, we selected testable ranges of knot interval and curve order. As a result, three curve orders 2, 3 and 4 in combination with the five knot intervals 0.1, 0.2, 0.5, 1.0 and 2.0 have been prepared. Each combination of parameters sets is used separately to fit B-splines over the six parameters X, Y, Z, ω , ϕ and ϕ . We calculate the RMSE and the minimum and maximum errors.

The result of optimization on the Trajectory-I dataset is given in Table 5-2. For the selection of the optimal parameters, it is important to consider that we are not seeking the best results, the main rule is to choose the combination of largest knot interval with a small curve order, which narrowly satisfies accuracy criteria. The largest errors are caused by the large knot intervals in combination with all curve orders, in any case, we are not interested in the small knot intervals either. The knot interval of 2.0 second is too large to model the trajectory retaining enough accuracy. Comparing with the accuracy criteria of 4 cm for position and 0.12° for rotation as mentioned in section 5.3.1, the knot interval of 1 second and curve orders 3 has achieved RMSE X=0.60 cm, Y=0.30 cm, Z=0.61 cm, ω =0.08, ϕ =0.12 cm and =0.05 cm, this is comparable with the knot interval of 1 second and curve order 4 with RMSE X=0.53 cm, Y=0.31 cm, Z=0.58 cm, ω =0.07 cm, φ =0.12 cm and =0.04 cm. Both combinations can be determined as the optimal values. The RMSE in position and rotation for both combinations [3,1.0] and [4, 1.0] is within the

allowed error of 4 cm. However, comparatively the knot interval of 1 second and curve of order 3 produced large errors, so we prefer the curve of order 4.

We also perform the same parameter optimization procedure on the second dataset and record the results that are shown in Table 5-3. For the Trajectory-II in Table 5-3, knot interval of 1.0 second and curve of order 4 are clearly optimal with RMSE X=0.52 cm, Y=0.53 cm, Z=0.46 cm, ω =0.16 cm, ϕ =0.12 and =0.03 when compared with (second to best) knot interval of 1.0 second and curve of order 3 with RMSE X=0.62 cm, Y=0.53 cm, Z=0.53 cm, ω =0.20 cm, ϕ =0.15 cm and =0.03 cm. Moreover, the minimum and maximum errors of rotation for the curve of order 3 exceeded little above the threshold of θ_{ϵ} = 0.12° in comparison with the curve of order 4. The results show that the increase in the knot interval time also reciprocates the error because fewer coefficients cannot accurately represent the rapid variation in the poses.

B-spline order	Knot interval (sec)	X B-spli	ne (cm)	Y B-spline (cm)			Z B-spline (cm)			ω	ω B-spline (deg)			φ B-spline (deg)			к B-spline (deg)		
		RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max
2	0.1	0.04	-0.16	0.16	0.03	-0.12	0.13	0.03	-0.18	0.16	0.02	-0.14	0.14	0.02	-0.12	0.10	0.00	-0.03	0.03
2	0.2	0.08	-0.36	0.32	0.06	-0.22	0.26	0.12	-0.55	0.64	0.03	-0.15	0.13	0.03	-0.12	0.16	0.01	-0.03	0.03
2	0.5	0.45	-1.86	1.74	0.22	-0.93	1.19	0.49	-1.80	1.55	0.07	-0.21	0.19	0.08	-0.39	0.26	0.02	-0.08	0.10
2	1.0	1.49	-7.83	5.96 20.27	0.73	-2.13	3.76 11.77	0.64	-2.31	1.88	0.09	-0.25	0.26	0.12	-0.50	0.35	0.07	-0.33	0.27
3	0.1	0.03	-0.14	0.15	0.02	-0.12	0.12	0.02	-0.12	0.12	0.02	-0.13	0.14	0.02	-0.11	0.12	0.00	-0.03	0.03
3	0.2	0.05	-0.20	0.17	0.02	-0.25	0.12	0.02	-0.38	0.12	0.02	-0.16	0.14	0.02	-0.13	0.12	0.00	-0.03	0.03
3	0.5	0.14	0.45	0.54	0.17	0.55	0.40	0.24	0.07	1.10	0.00	0.21	0.10	0.08	0.27	0.26	0.01	0.04	0.05
3	1.0	0.60	-2.41	2.52	0.30	-0.90	1.24	0.61	-2.31	1.62	0.08	-0.21	0.26	0.12	-0.27	0.38	0.01	-0.17	0.03
3	2.0	1 07	-5 17	6 15	0.88	-2.03	2 01	0.90	-2.84	3 25	0.14	-0.45	0.30	0.16	-0.49	0.56	0.17	-0.40	0.45
4	0.1	0.03	-0.14	0.17	0.02	-0.12	0.14	0.02	-0.08	0.09	0.02	-0.14	0.15	0.02	-0.12	0.11	0.00	-0.03	0.03
4	0.2	0.05	-0.21	0.16	0.05	-0.20	0.17	0.10	-0.42	0.48	0.03	-0.16	0.14	0.03	-0.12	0.12	0.01	-0.03	0.03
4	0.5	0.27	-0.92	0.99	0.16	-0.51	0.64	0.43	-1.44	1.40	0.06	-0.21	0.18	0.08	-0.32	0.26	0.01	-0.05	0.04
4	1.0	0.53	-1.77	1.84	0.31	-0.87	0.82	0.58	-2.18	1.82	0.07	-0.21	0.26	0.12	-0.41	0.35	0.04	-0.12	0.12
4	2.0	2.07	-5.59	5.07	1.37	-3.69	3.40	0.76	-3.12	1.95	0.11	-0.34	0.38	0.14	-0.48	0.55	0.11	-0.39	0.27

Table 5-2: Results for B-spline fitting to Trajectory-I dataset using combinations of knot interval and curve order.

B-spline order	Knot interval (sec)	X B-spli	ne (cm)		Y B-spli	ne (cm)		Z B-spli	ne (cm)		ω B-spline (deg)			φ B-spline (deg)			к B-spline (deg)		
		RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max
2	0.1	0.05	-0.22	0.22	0.04	-0.20	0.23	0.07	-0.28	0.28	0.14	-0.61	0.46	0.09	-0.70	0.34	0.00	-0.02	0.02
2	0.2	0.12	-0.58	0.64	0.10	-0.44	0.78	0.17	-0.64	1.03	0.16	-0.66	0.56	0.11	-0.63	0.58	0.01	-0.03	0.03
2	0.5	0.44	-2.24	2.53	0.32	-1.28	2.13	0.42	-1.18	1.69	0.20	-0.79	0.81	0.15	-0.85	0.93	0.02	-0.10	0.09
2 2	1.0 2.0	1.18 4.62	-4.49 -15.62	6.45 23.48	1.11 4.29	-4.28 -16.60	6.34 24.27	0.61 1.27	-2.17 -5.04	3.67 5.41	0.21 0.22	-0.77 -0.79	0.81 0.82	0.16 0.20	-0.88 -1.14	0.94 1.13	0.04 0.15	-0.17 -0.61	0.18 0.70
3	0.1	0.05	-0.24	0.18	0.04	-0.18	0.29	0.06	-0.26	0.31	0.14	-0.53	0.45	0.10	-0.62	0.41	0.00	-0.03	0.03
3	0.2	0.10	-0.39	0.30	0.07	-0.22	0.39	0.15	-0.46	0.48	0.15	-0.66	0.55	0.11	-0.75	0.44	0.01	-0.02	0.02
3	0.5	0.25	-0.79	1.03	0.20	-0.59	0.65	0.33	-1.02	1.08	0.19	-0.71	0.78	0.13	-0.49	0.63	0.01	-0.06	0.05
3	1.0	0.62	-2.53	1.82	0.53	-1.64	2.04	0.53	-1.47	2.22	0.20	-1.07	0.80	0.15	-1.04	0.68	0.03	-0.11	0.11
3	2.0	2.81	-7.90	6.38	1.90	-6.48	5.25	0.92	-3.47	4.86	0.22	-1.20	0.87	0.16	-1.33	0.83	0.08	-0.25	0.21
4	0.1	0.05	-0.21	0.17	0.04	-0.19	0.28	0.06	-0.22	0.27	0.14	-0.30	0.26	0.10	-0.68	0.41	0.00	-0.03	0.02
4	0.2	0.09	-0.33	0.31	0.07	-0.30	0.28	0.14	-0.45	0.49	0.14	-0.34	0.29	0.11	-0.81	0.43	0.01	-0.02	0.03
4	0.5	0.25	-1.10	0.97	0.20	-0.62	0.69	0.39	-1.18	1.50	0.15	-0.43	0.35	0.11	-0.52	0.51	0.01	-0.06	0.06
4	1.0	0.52	-2.34	1.50	0.53	-1.47	1.57	0.46	-1.27	1.80	0.16	-0.46	0.38	0.12	-0.87	0.60	0.03	-0.11	0.08
4	2.0	2.11	-7.42	6.23	1.46	-3.85	3.99	0.75	-2.64	2.65	0.20	-0.57	0.44	0.15	-0.84	0.59	0.05	-0.18	0.15

Table 5-3: Results for B-spline fitting to Trajectory-II dataset using combinations of knot interval and curve order.

5.5.3 Regeneration of point cloud and A2P points

The main motivation for the trajectory adjustment is to reconstruct an accurate MLS point cloud. The point cloud in which the A2P tie points have been measured has originally been computed with the trajectory resulting from the Kalman filtering. Hence, a point in the world coordinate system X_{PC}^{W} was computed based on the observed point X_{PC}^{C} by the laser scanner in the car coordinate system with Eq. 16.

$$X_{PC}^{W} = R_{C,Kalman}^{W^{T}}(t) X_{PC}^{C} + T_{C,Kalman}^{W}(t)$$
(16)

where the rotation matrix $R_{C,Kalman}^{W^T}$ and translation vector $T_{C,Kalman}^W$ are the inaccurate pose parameters from the Kalman filter solution. So, the computed point X_{PC}^W is the inaccurate point cloud point, which needs to be recalculated based on the updated pose parameters after the trajectory enhancement.

Assuming that the trajectory pose after the adjustment corresponds to $[R_c^{W'}, T_c^{W'}]$, then the new position of the point cloud point in world coordinate systems can be computed by Eq. 17;

$$X_{PC}^{W'} = R_C^{W'}(t) X_{PC}^C + T_C^{W'}(t)$$
(17)

where $X_{PC}^{W'}$ is a 3D point of the improved point cloud in the world coordinate system. It is also how the improved position of the A2P tie points will be computed because $X_{PC}^{W'} = A2P'$. Notice the local laser scanner observation X_{PC}^{C}

remains the same comparing Eq. 16 with Eq. 17. It is only the pose parameters $R_C^{W'}(t)$ and $T_C^{W'}(t)$ that are updated. For every experiment, we first select a trajectory and then adjust it using the combination of observations stated in section 5.3.

An example of the corresponding A2A and A2P points obtained for Trajectory-I dataset as well as the trajectory before and after the adjustment is plotted in Figure.5.8. In this figure, the red dots represent the A2P points and blue dots are their corresponding A2A points. Similarly, the red curve in same figure is trajectory before the adjustment while the blue curve represents the trajectory after the adjustment.



Figure.5.8: Example of 3D A2A tie points (blue dots) and A2P tie points (red dots) along with the Kalman filtering result (red curve) and the trajectory after the adjustment with our method (blue curve), Amersfoort / RD New coordinate system.

We verify the accuracy, by regenerating the A2P points from the adjusted trajectory and measuring the improvement compared to the checkpoints for which two sets of corresponding checkpoints are acquired and labelled in Table 5-4. Set A contains 38 checkpoints and set B contains 12 checkpoints. As an evaluation example, the adjustment of Trajectory-I using all observation is conducted and the residuals between the regenerated A2P points and checkpoint A47 and checkpoint A50 are provided in Table 5-5. Plots showing the checkpoints before and after the adjustment are presented in Figure 5.9 and Figure 5.10, for the checkpoint A47 and A50 respectively. The residual of A47 is comparatively larger than A50 on XY plane, however, considering the Z

coordinate, there is more error in A50 than in A47. Moreover, calculating the norm, e.g. ||A47 - A2P|| = 23 cm and ||A50 - A2P|| = 0.18 cm shows that the trajectory has been adjusted comparatively better for A50 than for the A47.

Table 5-4: Two sets of checkpoints used for evaluation.

Checkpoint set	Label
A	A23,A24,A25,A26,A27,A37,A38,A39,A42,A43,A44,A45,A46,A47,A50,A62,A63,A71,A72,A73, A74,A77,A78,A79,A80,A81,A92,A93,A94,A95,A96,A97,A104,A105,A106,A107,A108,A109
В	B39,B40,B41,B42,B44,B43,B45,B46,B47,B48,B49,B50

Table 5-5: The example of residuals measured using checkpoints on the point cloud regenerated from the trajectory enhanced using all observations, these checkpoints can also be located in Figure 5.12 of Trajectory-IB.

Obs.	Dataset	Check point	A2P point (XYZ) after adjustment, Amersfoort / RD new (m)	$\begin{array}{l} \Delta X_{checkpoint},\\ \Delta Y_{checkpoint},\\ \Delta Z_{checkpoint} \left(m\right)\end{array}$	Image of point cloud after adjustment
	Trail	A47	93448.54 437020.68 4.880	0.13 0.14 -0.13	Figure 5.9
All	11 aj1	A50	92871.03 436411.46 5.31	0.09 -0.04 -0.16	Figure <i>5.10</i>



Figure 5.9: The evaluation of regenerated A2P points using checkpoint A47.



Enhanced Trajectory Estimation of Mobile Laser Scanners Using Aerial Images

Figure 5.10: The evaluation of regenerated point cloud using checkpoint A50.

5.5.4 Experiments and evaluation

We perform the experiments according to the guidelines prepared in Table 5-1. We conducted the experiment on Trajectory-I and Trajectory-II datasets.

We selected two subparts Trajectory-IA and Trajectory-IB from the Trajectory-I, which are plotted in Figure *5.11* and Figure *5.12* respectively. These subparts fulfil the criteria for the experiments as mention in section 5.5.1. The first subpart is used for experiments 1, 2, 3 and 4 because these experiments can be conducted with infrequent checkpoints. As these experiments are based on the tie points, we need to select the trajectory part which has road marks at some places. For experiment 5 we need to select a trajectory with abundant and consistent availability of checkpoints, regardless of the road marks. For this purpose, the second subpart is feasible because it has more checkpoints to test the IMU-based trajectory over small intervals without the involvement of the tie points.



Figure 5.11: Plot of the Trajectory-IA used for experiments 1, 2 and 3. Here red `*' are the locations of the checkpoints, Amersfoort / RD New coordinate system.



Figure 5.12: Plot of the Trajectory-IB used for experiments 4. Here red `*' are the locations of the checkpoints, Amersfoort / RD New coordinate system.

To conduct an experiment, first, we perform the adjustment of a particular trajectory using the observations mentioned in Table *5-1*. Then the evaluation is performed using the regenerated A2P points and corresponding checkpoints. The combination of checkpoints alongside the trajectory dataset used for each experiment is indicated in Table *5-6*. This table also presents the RMSEs of six pose parameters, including the minimum and maximum error.

Type	Data	Checkpoints	No.	Criteria		X (m)			Y (m)				
Type	Data	CHECKPOINTS	110.	Cinteria	RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max
Kalman	I	All	0	No tie points	0.26	-1.28	0.96	0.30	-0.94	1.13	0.47	-0.58	0.59
	Ι	All	1	100%	0.09	-0.20	0.14	0.14	-0.17	0.22	0.17	-0.33	0.35
Тe				90%	0.11	-0.24	0.23	0.16	-0.25	0.27	0.28	-0.36	0.39
poin		۵23	2	80%	0.12	-0.30	0.28	0.19	-0.43	0.30	0.30	-0.42	0.50
t qua			2	50%	0.16	-0.81	0.35	0.22	-0.60	0.31	0.40	-0.49	0.55
Intita		A24 A25		30%	0.16	-0.85	0.62	0.28	-0.69	1.12	0.40	-0.60	0.58
tive a		A26 A27		90%	0.10	-0.20	0.15	0.14	-0.19	0.25	0.17	-0.33	0.35
analy		A37 A38	3	80%	0.10	-0.22	0.15	0.17	-0.26	0.30	0.17	-0.36	0.38
sis	IA	A39 A42	5	50%	0.10	-0.35	0.25	0.18	-0.30	0.64	0.19	-0.38	0.40
		A43 A44		30%	0.19	-0.54	0.30	0.21	-0.32	0.69	0.26	-0.40	0.42
립그		A45 A62 A63	4a	Low quality, 20%	0.24	-0.94	0.50	0.20	-0.87	0.97	0.34	-0.53	0.57
e point alitative			4b	Medium quality, 25%	0.17	-0.53	0.36	0.17	-0.42	0.55	0.21	-0.50	0.44
(D			4c	High quality, 55%	0.09	-0.15	0.12	0.11	-0.14	0.16	0.16	-0.30	0.32
		A46 A47 B39 B40	5a	30 secs 396 m	0.06	-0.15	0.10	0.07	-0.15	0.16	0.11	-0.13	0.19
IMU and soft constraints based trajectory	IB	A46 A47 A50 B39 B40 B41 B42 B44 B43 B45 B45 B46 B47 B48	5b	60 secs 792 m	0.10	-0.20	0.17	0.12	-0.20	0.18	0.18	-0.21	0.27
		A46 A47 A50 B39 B40 B41 B42 B44 B43 B45 B46 B47 B48 B49 B50	5c	90 secs 1101 m	0.21	-0.42	0.38	0.27	-0.33	0.39	0.23	-0.34	0.45
No soft constraint	I	All	6	No soft constraints, 100%	0.11	-0.20	0.16	0.15	-0.18	0.26	0.22	-0.33	0.37

Table 5-6: Results of the experiments categorised in Table 1 and conducted on Trajectory-I.

The experiment 0 in

Table 5-5 is conducted to analyse the accuracy of the Kalman trajectory before using any observation for the adjustment. This experiment provides an assessment of the current state of the Trajectory-I dataset. Before analysing the relation between the quantity of the tie points and the accuracy of the trajectory, experiment 1 apply all tie points and other observation to provide the reference for further analysis. It results in the accuracy of the trajectory without reducing the number tie points.

Staring from experiment 2 and 3 in Table *5-6*, we decrease the total number of tie points to test the effect of their unavailability. In experiment 2, we remove the tie points situated in consecutive order. This operation lengthens the trajectory intervals without tie points, where tie points are scarce. Dissimilarly in experiment 3, the total number of tie points are reduced from where they are already in majority. This removal technique circumvents the wide gaps of no tie points, therefore, we expect to obtain better results than for experiment 2. For evaluation of the trajectory produced by experiments 2 and 3, we use 14 checkpoints. By comparing the results of experiments 2 and 3, we notice that while reducing the tie points, the accuracy considerably suffers in experiment 2. As with the wide gaps of no tie points, the IMU is the only source to construct the trajectory. However, it is not directly comparable to experiment 5 where we only use IMU observation without any tie points to test the reliability of IMU for small intervals.

In experiment 3, the accuracy does not deteriorate when applying the 80% tie points because the remaining tie points nonetheless will provide observation for approximately all locations. However, when we use only 50% tie points in experiment 3, the accuracy approaches an undesirable level.

In the 4th experiment, it is clearly noticeable that only high-quality tie points alone can achieve decimetre-level accuracy. Conversely, only using the lowquality tie point leads to large errors in the trajectory. Yet, the utilization of medium or even low-quality tie points is better than using no tie points at all. It is apparent when comparing the results of experiment 4 with experiment 5. Even for a small interval of the 120 seconds, the accuracy is poorer than using the low-quality tie points alone. Note that the low-quality tie points are used for the whole trajectory sustaining 45 minutes but still, the accuracy is comparatively not as poor as the IMU-based trajectory exhibit for the small intervals. The high-quality tie points are only 55% of the total set, yet they are alone enough to correct the trajectory to the maximum accuracy. The result of experiment 3 with 80% tie points and high-quality tie points are comparable but the accuracy is little decreased because 20% removed tie points set comprise some high-quality tie points. Moreover, the remaining 80% tie points can comprise low-quality tie points, therefore, the accuracy achieved in experiment 3 with 80% tie points is still less than the experiment 4c. Furthermore, a comparison of the result of experiment 4c with experiment 1 confirms that the use of low-quality tie points introduces error in the trajectory. It is therefore advisable to only perform the trajectory adjustment with high-quality tie points.

In experiment 5, the trajectory has been constructed for the interval of 30, 60 and 90 seconds, which is scanning distance of 396, 792 and 1101 meters respectively. We only check the trajectory for the interval of up to 90 seconds because the resulted accuracy is already degraded more than the desired level. The main reason to conduct experiment 5 is to determine the maximum tolerable time of having no tie points. The largest interval of no tie points in the Trajectory-I lasts for 48 seconds, which is roughly 556 metres of distance on the road. This interval arrives between the range of 30 and 60 seconds so, we expect the accuracy to be better than indicated in 60 seconds interval, while the accuracy of 60 seconds interval is a little above the decimetre level. Therefore, we expect the accuracy to be below or near the decimetre level during the 48 seconds interval of no tie points.

In experiment 6, we evaluate the improvement in the trajectory without using the soft constraint observation. Comparing this result with experiment 1 confirms that the soft constraint observation has improved the accuracy RMSE X = 2 cm, RMSE Y = 1 cm and RMSE Z = 5 cm.

The Trajectory-II is in the areas with the tallest buildings in Rotterdam. The Trajectory-II alongside the checkpoints is plotted in Figure *5.13*. For the experiments, we do not divide Trajectory-II such as the Trajectory-I because there is an abundance of checkpoints available within short intervals. There is a total of 21 checkpoints available for this trajectory. We repeat the five experiments for this trajectory as well as the results are provided in Table *5-7*.



Figure 5.13: Plot of the Trajectory-II, red '*' are the locations of the checkpoints, Amersfoort / RD New coordinate system.

Туре	Data	checkpoints	No.	Criteria	X (m)			Y (m)	,		Z (m)		
					RMSE	Min	Max	RMSE	Min	Max	RMSE	Min	Max
Kalman		All	0	No tie points	1.10	-4.02	2.96	1.51	-1.10	4.05	1.81	-3.23	4.79
Tie			1	100%	0.12	-0.25	0.20	0.15	-0.16	0.27	0.20	-0.41	0.37
poin		A71 A72		90%	0.15	-0.60	0.38	0.21	-0.40	0.54	0.40	-0.50	0.63
t qua		A73 A74	2	80%	0.33	-1.01	0.95	0.62	-0.98	1.21	0.83	-1.55	1.64
ntitat		A77 A78	-	50%	0.54	-1.25	1.62	1.10	-0.98	1.62	1.30	-2.55	3.57
live a		A79 A80		30%	0.83	-2.02	0.92	1.41	-1.15	1.84	1.81	-2.23	2.79
nalys		A81 A92		90%	0.12	-0.37	0.21	0.14	-0.18	0.25	0.20	-0.41	0.38
SI.		A93 A94	3	80%	0.16	-0.75	0.28	0.30	-0.21	0.42	0.25	-0.61	0.83
		A95 A96 A97 A104 A105 A106 A107 A108 A107 A108 A109 A93 A93 A95		50%	0.26	-0.83	0.31	0.30	-0.53	0.62	0.55	-1.08	1.49
				30%	0.56	-1.75	0.60	0.34	-0.85	1.41	0.84	-1.35	2.28
Tie qua ana			4a	Low quality, 20%	0.64	-1.81	0.65	0.40	-0.78	1.63	0.73	-1.92	2.74
p litativ lysis	II		4b	Medium quality, 20%	0.27	-0.70	0.34	0.23	-0.45	0.94	0.27	-1.63	1.42
/e			4c	High quality, 60%	0.12	-0.21	0.16	0.14	-0.12	0.23	0.18	-0.36	0.34
IMU trajec			5a	30 secs, 219 m	0.16	-0.26	0.18	0.12	-0.16	0.23	0.14	-0.16	0.21
and soft tory		A92 A93 A95 A94	5b	60 secs, 555 m	0.34	-0.57	0.43	0.19	-0.34	0.67	0.25	-0.52	0.61
constraints-based		A92 A93 A95 A94 A77 A78 A78 A79 A80 A81	5c	90 secs, 839 m	0.48	-0.81	0.52	0.30	-0.49	0.74	0.43	-0.67	0.90
No soft constrain ts		All	6	No soft constraints, 100%	0.17	-0.31	0.24	0.15	-0.16	0.28	0.25	-0.42	0.40

Table 5-7: Results of the experiments conducted on Trajectory-II.

Overall the Kalman filter estimate of Trajectory-II is less accurate than that of Trajectory-I (cf. Table 5-6 and Table 5-7). Trajectory-II can only support the accurate positioning for up to 30 seconds while Trajectory-I positioning is accurate for up to 60 seconds. The IMU integrated with IPS-3 system utilizes fibre optic based inertial measurement, which is very accurate whereas the IPS-2 system is an older version and utilizes a MEMS gyroscope. However, the quality of the Trajectory-II after the tie point-based adjustment is similar to Trajectory-I. Obviously, the error still increases in the areas where there are no tie points. This is in particularly noticeable by comparing the results of low-

quality tie points in experiment 4a in both trajectories. This is true for both low-quality tie points and a small percentage of tie points.

In experiment 5, it is noticeable that the car speed of Trajectory-II is lower than Trajectory-I. Ignoring the distance travel by the laser scanning car and only considering the scanning time confirms that the IMU system used for Trajectory-II is comparatively less accurate than the IMU used for Trajectory-I. Therefore, the RMSE values increased by X=21 cm, Y=7 cm and Z=16 cm in the trajectory reconstructed by IMU observations alone for only 90 seconds in experiment 5c compared to the whole trajectory of 45 minutes reconstructed using only 20% tie points of medium-quality in experiment 4b. The trajectory in experiment 5c is also inaccurate by RMSE X=33 cm, Y=9 cm and Z=3 cm and RMSE X=36 cm, Y=16 cm and Z=23 cm even when 10% tie points are reduced in experiment 2 and 3 respectively.

In experiments 2 and 3, as expected, the accuracy of the trajectory-II decreases by reducing the percentage of tie points. Similar to our previous experience, the results obtained by the 2nd experiments are comparatively worse. Moreover, due to the comparatively low quality of the IMU observations, the results when using 80% of the tie points are not equivalent between the Trajectory-I and II. As the accuracy already became worse using the 80% tie points, to make the results from both datasets comparable, we started from the set of 90% tie points for Trajectory-II.

For trajectory-II, we can utilize 90% of the tie points by the removal method proposed in the third experiment. Removing 10% tie points and removing randomly cannot produce the near decimetre accurate results. The maximum time of having no tie points should not exceed the 30 seconds interval, while the maximum time without a tie point in Trajectory-II is 34 seconds. So, the trajectory may have an error of a few centimetres above the expected accuracy.

Similar to our experience with Trajectory-I, when comparing the result of experiment 6 conducted without soft constraints with experiment 1, it is also clear that soft constraints further improved 5 cm in the RMSE in both X and Z coordinates.

5.6 Conclusions

We developed and described an automatic method for the enhancement of 6DOF MLS platform trajectories. For this purpose, we described multiple observations related to automatically acquired 3D tie points matched to corresponding points extracted from aerial photographs, IMU measurements, and soft constraints, which were used collectively for the trajectory

adjustment. We proposed to model the trajectory parameters with B-Splines and also estimated the optimal values of the B-spline knot interval and curve order. To evaluate the developed method, we designed several experiments. Each experiment was focused on a specific aspect of the correction.

To evaluate the accuracy of the enhanced trajectory, the developed method is tested on two independent datasets. The adjustment with all observations brings a large improvement when results are compared with the original Kalman filter solution. The Kalman filter solution of Trajectory-I had RMSE values of 26 cm, 36 cm and 47 cm for the X-, Y- and Z-coordinates respectively. With the adjustment using all introduced observations these RMSE values were reduced to 9 cm, 14 cm. and 17 cm. When only high-quality tie points (55% with standard deviation \leq 5 cm) were used, RMSE values were further reduced to 9 cm, 11 cm, and 16 cm.

For Trajectory-II, the residuals in the Kalman filter trajectory are 1.10 m, 1.51 m, 1.81 m for the X-, Y- and Z-coordinates respectively. After the enhancement of the Trajectory-II using all observations, the remaining errors in the trajectory amount to 12 cm, 15 cm, and 20 cm. After the adjustment with all observations Trajectory-II shows a larger improvement than Trajectory-I because Trajectory-II was more erroneous due to the lower quality IMU. Similar to the first dataset, the use of only high-quality tie points (60%) leads to a further improvement in trajectory with RMSE values at 12 cm, 14 cm, and 18 cm. The evaluation of the developed method on both data sets confirms that our method can significantly improve the trajectory for applications demanding high accuracy in the urban canyons. It also shows that the significantly lower quality of the IMU in the second dataset has little to no effect on the quality of the reconstructed trajectory if sufficient tie points to aerial photographs are available.

It is preferred to only use high-quality tie points for Trajectory-I, and -II respectively, rather than using all tie points. Moreover, the results obtained by 80% or 90% inconsistently located tie points lead to RMSE values of 12 cm, 19 cm, and 30 cm for Trajectory-I and 15 cm, 21 cm, and 40 cm for Trajectory-II. However, the same amount of tie points when removed consistently can achieve RMSE values of 10 cm, 14 cm, and 17 for Trajectory-I and of 12 cm, 14 cm, and 20 cm for Trajectory-II dataset. Therefore, we conclude that the consistent availability of the tie point is crucial for the developed method. The consistent availability of the tie points is dependent on the existence of the road-marks in the dataset. The earlier developed image matching procedure (Hussnain et al. (2019)) is already designed to be robust enough to handle the difficult matching situations, since, we cannot afford long stretches of trajectory without tie points.
The quality of the tie points depends on the accuracy of the exterior and interior orientation parameters of the aerial images as well as the accuracy of the matching in the aerial images. The tie point quality is assessed by the standard deviation of the multiview triangulation. Moreover, the high aerial image resolution provides precise information needed to reliably detect and estimate the subpixel-level features, which can improve the estimation of high-quality tie points. In this way the use of aerial images can strongly improve the MLS trajectory estimation and thereby reduce the need for surveys on the ground.

Acknowledgements

This work is part of the research programme OTP with project number 13589, which is financed by the Dutch Research Council (NWO). We would like to thank Topcon© for providing the mobile laser scanning datasets and Cyclomedia© for the aerial imagery.

Reference of chapter 5

- Bornaz, L., Lingua, A., Rinaudo, F., 2003. Multiple scanner registration in LIDAR close-range applications. The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 34, 72-77.
- Bosse, M., Zlot, R., 2009. Continuous 3D scan-matching with a spinning 2D laser, Robotics and Automation, 2009. ICRA'09. IEEE International Conference on. IEEE, pp. 4312-4319.
- Chiang, K.-W., Huang, Y.-W., 2008. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. Applied Soft Computing 8, 722-733.
- Ding, W., Wang, J., Rizos, C., Kinlyside, D., 2007. Improving adaptive Kalman estimation in GPS/INS integration. Journal of Navigation 60, 517-529.
- Gao, Y., Huang, X., Zhang, F., Fu, Z., Yang, C., 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 40, 41.
- Haala, N., Peter, M., Kremer, J., Hunter, G., 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. The Internal Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 37, 1119-1127.
- Hunter, G., Cox, C., Kremer, J., 2006. Development of a commercial laser scanning mobile mapping system–StreetMapper. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci 36.
- Hussnain, Z., Oude Elberink, S., Vosselman, G., 2016. Automatic feature detection, description and matching from mobile laser scanning data and

aerial imagery. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLI-B1, 609-616.

- Hussnain, Z., Oude Elberink, S., Vosselman, G., 2018. An Automatic Procedure for Mobile Laser Scanning Platform 6DOF Trajectory Adjustment. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-1, 203-209.
- Hussnain, Z., Oude Elberink, S., Vosselman, G., 2019. Automatic extraction of accurate 3D tie points for trajectory adjustment of mobile laser scanners using aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing 154, 41-58.
- Javanmardi, E., Gu, Y., Javanmardi, M., Kamijo, S., 2018. Autonomous vehicle self-localization based on abstract map and multi-channel LiDAR in urban area. IATSS Research.
- Javanmardi, M., Javanmardi, E., Gu, Y., Kamijo, S., 2017. Towards High-Definition 3D Urban Mapping: Road Feature-Based Registration of Mobile Mapping Systems and Aerial Imagery. Remote Sensing 9, 975.
- Kaartinen, H., Hyyppä, J., Kukko, A., Jaakkola, A., Hyyppä, H., 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. Sensors 12, 12814-12835.
- Karam, S., Peter, M., Hosseinyalamdary, S., Vosselman, G.J.I.A.o.P., Remote Sensing, Sciences, S.I., 2018. AN EVALUATION PIPELINE FOR INDOOR LASER SCANNING POINT CLOUDS. 4.
- Kukko, A., 2013. Mobile laser scanning–system development, performance and applications, Finland.
- Kümmerle, R., Steder, B., Dornhege, C., Kleiner, A., Grisetti, G., Burgard, W., 2011. Large scale graph-based SLAM using aerial images as prior information. Autonomous Robots 30, 25-39.
- Levinson, J., Montemerlo, M., Thrun, S., 2007. Map-Based Precision Vehicle Localization in Urban Environments, Robotics: Science and Systems. Citeseer, p. 1.
- Patron-Perez, A., Lovegrove, S., Sibley, G., 2015. A spline-based trajectory representation for sensor fusion and rolling shutter cameras. International Journal of Computer Vision 113, 208-219.
- Usenko, V., von Stumberg, L., Pangercic, A., Cremers, D., 2017. Real-Time Trajectory Replanning for MAVs using Uniform B-splines and 3D Circular Buffer. arXiv preprint arXiv:1703.01416.
- Vosselman, G., 2014. Design of an indoor mapping system using three 2D laser scanners and 6 DOF SLAM. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2, 173.
- Wolcott, R.W., Eustice, R.M., 2014. Visual localization within lidar maps for automated urban driving, Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on. IEEE, pp. 176-183.
- Zhao, Y., 2011. GPS/IMU integrated system for land vehicle navigation based on MEMS. Stockholm, Sweden, KTH Royal Institute of Technology, p. 85.

- Bornaz, L., A. Lingua and F. Rinaudo, 2003. Multiple scanner registration in LIDAR close-range applications. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 34, 72-77.
- Bosse, M. and R. Zlot, 2009. Continuous 3D scan-matching with a spinning 2D laser. Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, IEEE: 4312-4319.
- Chiang, K.-W. and Y.-W. Huang, 2008. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. Applied Soft Computing 8, 722-733.
- Ding, W., J. Wang, C. Rizos and D. Kinlyside, 2007. Improving adaptive Kalman estimation in GPS/INS integration. Journal of Navigation, **60**, 517-529.
- Gao, Y., X. Huang, F. Zhang, Z. Fu and C. Yang, 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 40, 41.
- Haala, N., M. Peter, J. Kremer and G. Hunter, 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **37**, 1119-1127.
- Hunter, G., C. Cox and J. Kremer, 2006. Development of a commercial laser scanning mobile mapping system–StreetMapper. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2016. Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **XLI-B1**, 609-616.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2018. An automatic procedure for mobile laser scanning platform 6dof trajectory adjustment. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **XLII-1**, 203-209.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2019. Automatic extraction of accurate 3D tie points for trajectory adjustment of mobile laser scanners using aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing **154**, 41-58.
- Kaartinen, H., J. Hyyppä, A. Kukko, A. Jaakkola and H. Hyyppä, 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. Sensors **12**, 12814-12835.
- Karam, S., M. Peter, S. Hosseinyalamdary, G. Vosselman, Remote Sensing and S. I. Sciences, 2018. An evaluation pipeline for indoor laser scanning point clouds. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences 4, 1.

- Kukko, A., 2013. Mobile laser scanning–system development, performance and applications. Publications of the Finnish Geodetic Institute, Veröffentlichungen des Finnischen Geodätischen Institutes, Finland. 153.
- Kümmerle, R., B. Steder, C. Dornhege, A. Kleiner, G. Grisetti and W. Burgard, 2011. Large scale graph-based SLAM using aerial images as prior information. Autonomous Robots **30**, 25-39.
- Levinson, J., M. Montemerlo and S. Thrun, 2007. Map-Based Precision Vehicle Localization in Urban Environments. Robotics: Science and Systems **4**, 1.
- Patron-Perez, A., S. Lovegrove and G. Sibley, 2015. A spline-based trajectory representation for sensor fusion and rolling shutter cameras. International Journal of Computer Vision **113**, 208-219.
- Usenko, V., L. von Stumberg, A. Pangercic and D. Cremers, 2017. Real-Time Trajectory Replanning for MAVs using Uniform B-splines and 3D Circular Buffer. arXiv preprint arXiv:1703.01416.
- Wolcott, R. W. and R. M. Eustice, 2014. Visual localization within lidar maps for automated urban driving. Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, IEEE: 176-183.
- Zhao, Y., 2011. GPS/IMU integrated system for land vehicle navigation based on MEMS. KTH Royal Institute of Technology, Stockholm, Sweden. **Doctoral dissertation:** 85.

6 – Synthesis

Synthesis

In this research project, we have developed an automatic method for the enhancement of MLS datasets acquired in the urban canyons where trouble with GNSS signals is inevitable. Our developed method approached the solution with the utilization of highly-accurate aerial imagery. Most of the MLS data correction techniques can be categorised into two major groups; the ones using a reference dataset for the correction and the others that are not. Our method involving aerial imagery fits into the group needing a reference dataset. The second category of methods remains focused on the improvement of algorithms to efficiently employ the local observations of positioning devices and sensors while avoiding the use of any external reference datasets. These techniques are based on the idea that necessary information to estimate the correct positioning is known but the observations are not applied efficiently enough. We believe that there is a limit in improving the accuracy by the only deployment of efficient algorithms and these techniques cannot surpass the improvements that can be achieved by involving the reference datasets. Compared to many of these techniques, our goal was not to avoid the need for the reference dataset. Our research work was concentrated on the minimization of total effort to correct the MLS dataset by automatic means. Most of the recent comparable studies also preferred the use of the reference datasets. This approach promises reaching the accuracy of reference dataset in the input dataset if not more than that.

An emerging and well-known approach in the category as ours is to involve the same dataset acquired earlier as a reference instead of using another type of dataset. The main consideration is that a future MLS dataset correction pipeline can benefit from the point cloud already acquired and corrected for the same area. This is feasible for areas where a new point cloud is only needed to accommodate the latest changes in the road markings and buildings. However, if a point cloud of the same area is not already available or it is not accurate enough then this technique will not work. Moreover, if the area is altered at many locations in the new dataset then the registration would also not bring accurate results. In comparison, achieving e.g. 10 cm accuracy in reference aerial imagery is still a lot easier than achieving the same accuracy in the reference MLS point cloud even if the GCPs have been utilized. Due to the spatial consistency of information, aerial images as reference dataset provide the same accuracy of the features across the image. However, spatial consistency cannot be guaranteed in the manually corrected point cloud intended to use as a reference, especially in the areas where no GCPs have not been utilized.

6.1 Conclusions per objective

1) Feature detection to represent the same corresponding geometrical position with pixel-level accuracy in both lidar and image datasets.

A lot of recent literature preferred strategies other than 2D feature detection and matching. Instead of 2D features matching, most of the localization applications matched MLS dataset with a reference map generated from a reference point cloud (Wan et al. 2018; Wolcott et al. 2017). At a broader level, many of the reference utilization techniques come under the umbrella of map matching. Still there are enormous differences in the way maps were generated and utilized. Recently, quite a lot of research is invested in to reference maps. Nagy et al. (2018) used a high definition point cloud as a reference map for the localization of the self-driving car. Moreover, instead of 2D keypoints, 3D keypoints were detected in the objects extracted from the point cloud, and objects semantic information was derived for alignment. Similarly, Kim et al. (2019) generated binary maps of road markings from the input point cloud and a reference point cloud. Some studies, like Wang et al. (2017) detected curbs as feature together from the reference point cloud and reference digital map, while the method developed by Meng et al. (2017) was dependent upon only an offline digital map of the curbs. Many preferred features other than 2D and 3D. Hsu et al. (2019) extracted features as combined distances-angles measurements observed by a virtual 360degree laser scanner from each possible grid-location on the map. To match, these features were extracted from both the map and the live laser scans. This approach can be considered similar to Particle Filter, which also directly correlate the sensor observations with an environment map for localization. Most of the map-based techniques require maps very large in size and often matching them with online laser scan is computationally expensive. In order to overcome this problem, some techniques suggested to first downsize the information needed to carry by the mobile mapping platform for the online MLS operation. For example, some techniques derived vectors from the reference point cloud to represent the map, which is advantageous because it can only be the fraction of total size of whole reference point cloud. In this vein, Javanmardi et al. (2018) estimated few vectors representing the whole boundary of a building. They also estimated surface planes representing whole walls from the reference point cloud as features to be matched. This technique is interesting in areas where there are no road marks and areas which remain the same in an epoch of two acquisitions. Instead of representing planes with surface normal vectors, Im et al. (2018) directly extracted Hough line features, which is in line with many techniques discussed in chapter 1 focused on the extraction of linear features representing the geometry of the urban structures. The detection of the linear features from the aerial images can only work with the prominent edges of the buildings. As the lidar scans buildings from the road level, either reliable building edges are missing in the points cloud or if available the point density is very low at that distance. The other methods like ICP can only achieve high accuracy when an accurate reference point cloud with few or no updates and similar density is available. Even though the accuracy of 3D point cloud generated from the aerial imagery at hand can be the same as 3D tie points, the density of the point cloud reconstructed from images is not enough on the roadside structures and buildings surfaces for ICP based matching. By comparing our research work with other related studies, we found that not many of the studies involved 2D feature matching of the point cloud with aerial images for lidar sensor localization. The whole focus in the development of 2D feature detection and matching was to achieve decimetre-level accurate matches. The main advantage of our technique is that the positions of matched keypoints inherit accuracy of the features in aerial images. This type of correctness is hard to guarantee using methods other than 2D feature matching.

The methods described in chapter 2 and 3 were mainly focused on the detection of intensity-based 2D features from point cloud images and aerial images. Based on the feasibility, our feature detection technique concentrated on the road markings, which was easily available in the urban areas. Relying on the disparity of high-intensity information of the road marks with low intensity of the road was essential because this interrelationship persisted across observations from unalike sensors. Our feature extraction technique extracted geometrically accurate features because it relied on the estimation of the intersection of linear edges, while blob-based features e.g. SIFT may not be geometrically accurate, though they seem contextually correct. This type of feature rather gets displaced in images of the same object from dissimilar perspectives, while the corner-based features are immune from perspective distortion. Apart from the benefit of accuracy, the intersection of gradient edges can only provide a single keypoint per corner feature, which was not sufficient for the robust matching of feature descriptor due to large and varying noise in the image information. The developed method tackled this problem by detecting keypoints in a cluster for a single corner feature as described in Chapter 3. The feature-based techniques e.g. SIFT by Lowe (2004), AKAZE Alcantarilla et al. (2011) and KAZE by Alcantarilla et al. (2012) had not been robust enough in matching point cloud images with aerial images. Therefore, the Harris Corner Detector by Harris et al. (1988) was adapted by relaxing the keypoint detection threshold to detect multiple features surrounding the single corner keypoint. This feature-detection strategy increased the possibility and robustness of matching feature descriptor. However, the new feature detection technique alone was not enough for

the accuracy of features. To maintain the geometric consistency, it was necessary to generate the point cloud images from the chronologically ordered point clouds (as mentioned in Chapter 1, 2 and 3). Otherwise, the point cloud ghosting effect can lead to either the detection of multiple inaccurate features or misplaced features.

In chapter 2 and 3, the point cloud was projected on the ground to generate images of the point cloud, which introduced errors when Z coordinates of the road points were not homogenous. To overcome this problem and represent the point cloud information from perspective same as of aerial imagery, in later research work, the point cloud was projected onto the aerial image perspective planes as discussed in chapter 4. This was advantageous because the 2D perspective images of the point cloud were able to maintain the same perspective as well as the same sampling distance, which was essential for very accurate matching to aerial images. The other difference among the approach developed in chapter 2 and 3 from chapter 4 was the removal of the road traffic and other roadside objects e.g. road signs, trees. In some studies like Javanmardi et al. (2018), vehicles were removed as well from the aerial imagery, however, we did not remove the vehicles or elevated objects from the aerial images as it cannot improve the matching of descriptors.

2) Feature matching to find the subpixel-level feature correspondences.

The combinations of feature detection and the feature descriptor matching were experimented and developed concurrently in chapter 2 and 3 because without a suitable feature descriptor matching, a feature detection would fail and vice versa. Our descriptor matching technique tackled two problems: the detection of compliant features was solved by detecting multiple keypoints which provided diverse information around single feature as discussed earlier in this section and chapter 3 and 4. The second problem, the handling of the diverse information around features was resolved by using the LATCH feature descriptor, which can compensate variation in information by learning the optimal configuration of corresponding pixel-triplets. The LATCH feature descriptor was advantageous because it does not seek the exact information but only compares patterns of similarities. For the other descriptors, a technique to relax the descriptor matcher tolerance also failed and instead raised the number of outliers. The major difference between our technique and the other techniques is that when the descriptor matcher tolerance was relaxed, our approach promised to include correct matches as well as the outliers, while the other just introduce more outliers.

Synthesis

The initial experiments in chapter 2 showed that the LATCH descriptor performed poorly because in contrast to SIFT, AKAZE, and KAZE which have their own feature detection, LATCH is just a feature descriptor. Therefore, we have used the Förstner operator Förstner et al. (1987) in the initial experiments to detect the subpixel accurate keypoints. However, this strategy failed to provide satisfactory results because it provided only a single keypoint for a single corner, which we have already discussed is unsuitable for our purpose. The other techniques showed better result in chapter 2 for the specific dataset, however, they were also not very robust in difficult situations experimented later. Therefore, instead of subpixel keypoint, we have combined a pixel-level keypoint detection with LATCH descriptor in chapter 3. Later experiments in chapter 3 and 4 have shown that this combination of feature detector descriptor outperformed the previous results of SIFT, AKAZE, KAZE. Though this technique successfully introduced correct matches, it included lots of outliers as well. The outliers exceeding the inliers were very difficult to remove with the conventional outlier removal techniques e.g. estimation of homography. Therefore, we tailor-made an outlier filtering approach in chapter 3 by exploiting the relative topology of features in the corresponding images. The developed outlier filtering approach was beneficial in promising the correct solution by brute force algorithm. This was time-consuming but due to the automatic nature of correction pipeline, it was more important to know whether the achieved solution was correct or not, and brute the force approach guaranteed that by searching for every possible solution. Oppositely, e.g. homography estimated by RANSAC can converge to a solution still containing outliers. Moreover, when there were e.g. 80% or more outliers (which was true in our case) then the conventional technique would have failed to obtain the correct solution in specified iterations. Outlier filtering approach was also useful in assuring that the corresponding 2D-2D inliers were pixel-level accurate. The clusters of keypoints were helpful to avoid the problem of repeated patterns by providing asymmetrical configuration of feature clusters across an image, which is not the case when single features were detected on the corners of e.g. a zebra crossing.

The subpixel keypoints matching was unreliable as discussed in chapter 2. Therefore, afterwards in this research work, we have developed a pixellevel feature matching in chapter 3. However, our aim still was to achieve the subpixel-level correspondences to estimate the near decimetre accurate 3D tie points because pixel-level correspondences were not sufficiently accurate. Therefore, we developed a mapping technique to determine the subpixel-level correspondences from pixel-level correspondences. In the process of mapping, the number of matches reduced dramatically because it was not possible to estimate subpixel-level correspondence for each pixel-level correspondence.

While our technique was concentrated on descriptor matching, other studies adapted diverse matching approaches depending on their feature detection. It is important to note here that methods of direct adjustment do not require matching step. For direct determination of the similar shapes of road markings, Suhr et al. (2016) extracted road markings from the scanned point cloud and directly transformed them to the reference digital map obtained from a 3rd party. Similarly, Nagy et al. (2018) directly aligned objects between input MLS dataset and reference point cloud, where the semantic information about the objects was helpful. Many image matching methods preferred correlation-based matching, which involved including all image information to compute the correlation. However, Im et al. (2018) only performed correlation of the detected lines because lines were the most interesting matchable features between point cloud and images. Instead of features and objects, Kim et al. (2019) performed direct ICP based alignment of binary maps obtained from both input and reference point clouds. Javanmardi et al. (2018) also directly aligned the point cloud with the vectors representing reference maps, while Wang et al. (2017) used the ICP algorithm the curb registration of input point cloud with the accurate reference digital map. Hsu et al. (2019) extracted special features from both input and reference point clouds and matching of features was based on the similarity of contiguous distance-angle observations. Most of the methods developed in recent studies are similar to direct registration or alignment of dataset based on distance minimization, these methods are feasible in areas without intensity based features but cannot provide superior accuracy compared to our feature based techniques when intensity based features are available.

3) Estimation of the decimetre-level accurate 3D tie points leading to 3D-3D correspondence observation.

In chapter 4, subpixel-level correspondences were used for the computation of aerial 3D tie points. The developed 2D feature matching was well robust between individual pair of two images. However, a feature can only be matched in more than two images if it has exactly the same correspondence shared across more than two aerial images. To determine the multiview matches, the enhancement of the pairwise 2D correspondences with a graph-based search was a beneficial strategy. Considering the point cloud images as the aerial perspective views of the point clouds in the multiview matching was also helpful in processing all correspondences together. To our knowledge, there is no other automatic method which can utilize the aerial imagery and provide subpixel-level

correspondences between images and point cloud that in turn lead to decimetre accurate 3D tie points.

The decimetre-level accurate 3D-3D observations obtained in chapter 4 were applied in chapter 5 and 6 and allowed the adjustment to achieve a near decimetre-level accurate trajectory. For the estimation of the trajectory with B-splines, we found that splines of order 4 and interval 1 second can reconstruct the trajectory with a sufficient accuracy. For the enhancement of the trajectory, a B-spline based adjustment was helpful as it only updates intervals in the trajectory relevant to specific observations. Due to this property, B-spline based trajectory enhancement was preferred in many similar studies like Patron-Perez et al. (2015), Páleš et al. (2016) and Usenko et al. (2017). The same benefit can also save the computational time of an update e.g. when compared with Hermit spline or Catmull–Rom spline (Li et al. 2016).

A lot of effort was dedicated towards the robustness of matching technique, which ensured the extraction of 3D tie points wherever road markings existed. Yet, extracted 3D tie points were not equally distributed across the trajectory. Also, at some places road marks were nonexistent, which created complete gaps of no tie points as discussed at the end of chapter 4. In this scenario, to estimate the positions within these gaps, IMU observations were essential. We also introduced constraints on heading and pitch angles of the car coordinate system, which helped to reduce the degree of freedom and thereby improved the estimability of the trajectory.

4) Adjustment of the 6DOF platform trajectory using mainly the 3D-3D correspondence and IMU observations.

Unlike our method, many recent autonomous vehicle localization methods were focused on 2DOF problem and just improved lateral and longitudinal accuracy needed for the navigation like Wolcott et al. (2017). In contrast, the main purpose of our research was to enhance the positional accuracy of the MLS point cloud which cannot be achieved unless all six pose parameters of the trajectory are accurate as well. Some techniques were only focused on the fusion of the observations from multiple sensors including lidar to support the positioning of the car in an urban environment. In a similar application, the final solution of Wan et al. (2018) accomplished 5-10 cm positioning accuracy by using an already existed accurate map as input. Meng et al. (2017) also fused the GNSS/IMU/DMI sensors observations and smooth the trajectory to compensate the GNSS jumps or discrepancies. However, they mainly perform correction of lateral error as it was needed for autonomous vehicles. Their results showed a lot of improvement from the initial error but still, the error remained was in the meter range at some places. For the localization using low-cost IMU,

Chapter 6

Suhr et al. (2016) showed that it was possible to improve the positioning of the vehicle in the urban canyons from many meters to meter-level. We already defined the minimum accuracy requirements of the IMU sensors needed to achieve the near decimetre level accuracy in chapter 6. The filtration of the sensors signals and other algorithmic improvements can as well enhance the trajectory of even low-quality sensors, however, this topic was out of the scope of our research. In recent studies, Im et al. (2018) reported lateral and longitudinal RMS of 0.14 m and 0.23 m respectively in the improved point cloud, while Kim et al. (2019) only performed an adjustment to X, Y and heading angle, and reported positional error of less than 50 cm in both lateral and longitudinal direction. The evaluation of Nagy et al. (2018) was based on 25 locations in MLS dataset, which showed average errors from 0.15 to 0.5 m. However, the maximum error rose up to 1 to 2 meters at some places. The multilayer 2D vector map-based localization developed by Javanmardi et al. (2018) has achieved accuracy with a mean 2D error of 0.20m. Their planar surface map-based localization achieved accuracy with about 0.43 m of error. For the outdoor environment, a method developed by Hsu et al. (2019) has achieved lateral accuracy with RMSE 2.16 m and longitudinal accuracy with RMSE 1.25 m. These best results were achieved with window-based localization, but the trajectory estimated from localization results showed an increase in errors. Most of the studies dealt with the problem similar to ours but aiming different applications. Also, it is important to note that techniques developed by Wan et al. (2018), Im et al. (2018), and Javanmardi et al. (2018) show comparable accuracy because all made use of a Velodyne multi-channel scanner. The multi-channel scanner provides a higher rate of point acquisition, Kukko (2013) concluded that it improves the data usability and result by providing more even and complete point distribution on the object surfaces. Including the problem of low point density, the configuration of standard MLS systems is also more prone to the consequences of positioning accuracy. Decimetre-level accuracy was not the focus of many studies, though it was achieved in some studies. Often, the adjustment of only two degrees of freedom was sufficient for navigation as the car always moves on the road surface. In other research projects, use of already acquired point clouds or maps generated from point clouds were permitted because the companies investing in autonomous cars will be able to afford expensive datasets or maps. However, our target market is the companies who are producing accurate MLS datasets or maps in the first hand.

All phases of our methods were developed by keeping the focus on running the whole workflow without any manual intervention. The evaluation has shown that our method has achieved better accuracy than most of the map-based techniques. Our achievements in this project have verified that the initial assumption to use the aerial imagery as reference dataset was advantageous in urban canyons. We have tested our method on two datasets. The improved first point cloud exhibited RMSE of X=0.09 m, Y=0.11 m, and Z=0.16 m, while the second point cloud has achieved accuracy with RMSE X=0.12 m, Y=0.14 m and Z=0.18 m. Results from both datasets have confirmed that the developed system was consistent in the enhancement of the MLS point clouds.

6.2 Reflections and outlook

The aerial images used in our project were captured by a commercial camera with highly accurate intrinsic parameters and needed no camera calibration within our pipeline. The high-resolution aerial images provided sufficient detailed of the ground features which was essential in matching feature and establishing the correspondences to subpixel-level. Normally, road markings are abundant in urban areas. However, road marks matching entails some problems, one of them is the occlusion of roads in aerial images. More tall buildings in the dataset mean occlusions in more aerial views, reducing the number of multiviews per surface patch. Due to the low number of aerial images in our dataset, only around half of the generated tie points were considered reliable. To increase the quantity of highly accurate tie points, the total number of aerial images and the overlap between the images should be increased.

The key to obtaining sufficient features is the quality of intensity information in the aerial images. Usually, the aerial imagery is captured at the high elevation because every country and city have their own regulations depending on the locations of nearby airports, hospitals etc. So, it is not allowed to fly aeroplanes under a certain height. The aerial imagery captured at high elevation resulted in narrow intersection angle of multiview features which lead to low accuracy in the Z coordinate of tie points. To increase the accuracy in the Z coordinate of tie points, the aerial images should be captured by drones or airplanes flying at low heights. Even though the aerial imagery in our project was acquired with high-quality image sensors and optics, the noise in image intensities caused difficulties for feature detection. Though it was tackled by the automatic adaptation of the feature detection threshold, we expect to face more difficulties in the detection of features from the images of off-the-shelf cameras. For non-commercial cameras, the atmospheric diffraction of white light due to high elevation together with the low quality of the image sensors lead to poor quality of the pixel intensities in the aerial image, let alone the poor accuracy of features due to lens distortion.

We have noticed that the dataset from the Rotterdam was one of the extreme examples because nearly every road in the dataset has roadside trees blocking the view of surface from aerial cameras. Luckily, many wider roads in the dataset offered sufficient road marking information in aerial images which otherwise would have posed an extreme problem. Therefore, the road marks matching is not expected to perform equally well in all areas when a dataset consists of narrow roads with a lot of trees. In our datasets, occlusion due to trees was minimized by the acquisition of datasets in winter but it was still a problem wherever narrow roads exist. Similarly, narrow roads with lots of tall buildings can also block direct line of sight from road surface to aerial views. Road markings are very important for feature-based matching and missing them in one of the datasets will clearly lead to unreliable tie points or a lack of tie points. Normally, the mobile mapping datasets are acquired in winter to minimize the occlusion by trees. However, in winter, elevated objects cast elongated shadows on the road surface regardless of time of the day. We noticed that it was problematic for aerial-to-aerial image matching due to matched shadow edges.

The MLS dataset utilized in our project was collected during daytime because the car was also mounted with the white light panoramic camera which is normally the case with such a mobile mapping system. At daytime, due to long stopovers and slow movement in urban areas, the total time needed for mobile mapping gets prolonged which increases the number of total points for a particular area and needs more memory in the computer to handle extra points. More traffic at daytime leads to more occlusion as well. Therefore, when only point cloud registration is intended, it is better to scan the area at nighttime because limited traffic will reduce the total time for mapping the area due to short or no stop on the crossroads. This will decrease the total size of point cloud which in turn will reduce the MLS dataset processing time and acquisition cost. The MLS dataset used in this project was acquired at daytime, we had to remove the points observed on the road traffic. With limited traffic at nighttime, all those points would have been observed on the road surface. The disadvantage, however, will be to conduct two separate acquisitions, one for the MLS dataset and other for white light images.

In the agreement between the users' group (as mentioned in chapter 1), one of the requirements was the transfer of knowledge. Therefore, the proof of concept software implementation of the whole MLS dataset correction pipeline was implemented and delivered to the user group at the end of the research project. The software modules implemented in this research project are depicted in chapter 1. The companies involved were also responsible for conducting the evaluation of our results, therefore, the results of our method were evaluated and verified by the MLS dataset providers. Their independent verification also confirmed our findings.

References for chapter 6

- Bornaz, L., A. Lingua and F. Rinaudo, 2003. Multiple scanner registration in LIDAR close-range applications. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **34**, 72-77.
- Bosse, M. and R. Zlot, 2009. Continuous 3D scan-matching with a spinning 2D laser. Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, IEEE: 4312-4319.
- Chiang, K.-W. and Y.-W. Huang, 2008. An intelligent navigator for seamless INS/GPS integrated land vehicle navigation applications. Applied Soft Computing 8, 722-733.
- Ding, W., J. Wang, C. Rizos and D. Kinlyside, 2007. Improving adaptive Kalman estimation in GPS/INS integration. Journal of Navigation, **60**, 517-529.
- Gao, Y., X. Huang, F. Zhang, Z. Fu and C. Yang, 2015. Automatic Georeferencing Mobile Laser Scanning Data to UAV images. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 40, 41.
- Haala, N., M. Peter, J. Kremer and G. Hunter, 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **37**, 1119-1127.
- Hunter, G., C. Cox and J. Kremer, 2006. Development of a commercial laser scanning mobile mapping system–StreetMapper. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2016. Automatic feature detection, description and matching from mobile laser scanning data and aerial imagery. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **XLI-B1**, 609-616.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2018. An automatic procedure for mobile laser scanning platform 6dof trajectory adjustment. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **XLII-1**, 203-209.
- Hussnain, Z., S. Oude Elberink and G. Vosselman, 2019. Automatic extraction of accurate 3D tie points for trajectory adjustment of mobile laser scanners using aerial imagery. ISPRS Journal of Photogrammetry and Remote Sensing **154**, 41-58.
- Kaartinen, H., J. Hyyppä, A. Kukko, A. Jaakkola and H. Hyyppä, 2012. Benchmarking the performance of mobile laser scanning systems using a permanent test field. Sensors **12**, 12814-12835.
- Karam, S., M. Peter, S. Hosseinyalamdary, G. Vosselman, Remote Sensing and S. I. Sciences, 2018. An evaluation pipeline for indoor laser scanning

point clouds. ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences **4**, **1**.

- Kukko, A., 2013. Mobile laser scanning–system development, performance and applications. Publications of the Finnish Geodetic Institute, Veröffentlichungen des Finnischen Geodätischen Institutes, Finland. 153.
- Kümmerle, R., B. Steder, C. Dornhege, A. Kleiner, G. Grisetti and W. Burgard, 2011. Large scale graph-based SLAM using aerial images as prior information. Autonomous Robots **30**, 25-39.
- Levinson, J., M. Montemerlo and S. Thrun, 2007. Map-Based Precision Vehicle Localization in Urban Environments. Robotics: Science and Systems **4**, 1.
- Patron-Perez, A., S. Lovegrove and G. Sibley, 2015. A spline-based trajectory representation for sensor fusion and rolling shutter cameras. International Journal of Computer Vision **113**, 208-219.
- Usenko, V., L. von Stumberg, A. Pangercic and D. Cremers, 2017. Real-Time Trajectory Replanning for MAVs using Uniform B-splines and 3D Circular Buffer. arXiv preprint arXiv:1703.01416.
- Wolcott, R. W. and R. M. Eustice, 2014. Visual localization within lidar maps for automated urban driving. Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, IEEE: 176-183.
- Zhao, Y., 2011. GPS/IMU integrated system for land vehicle navigation based on MEMS. KTH Royal Institute of Technology, Stockholm, Sweden. Doctoral dissertation: 85.

Synthesis

Summary

Mobile Laser scanning is a technique to obtain geo-information for large areas using Laser scanners mounted on a car or another vehicle. The obtained product is called a mobile laser scanning point cloud. Under normal conditions, accurate positioning is provided by the integration of Global Navigation Satellite Systems (GNSS) and Inertial Navigation Systems (INS). However, especially in urban areas building structures impede a direct line-of-sight to navigation satellites and lead to multipath effects. In this situation the mobile mapping derived products, such as mobile laser scanning point clouds, lack the expected reliability and inherit an unknown positioning error.

This issue has been addressed by many researchers, whose aim to mitigate these effects mainly concentrates on utilizing datasets, such as digital maps, orthoimages or airborne lidar. These datasets serve as a reliable source of orientation and are being used subsidiarily or as the basis for adjustment.

This research aims to improve the position estimation of mobile laser scanning platforms by employing high-resolution multiview nadir aerial imagery as a reference dataset. The aerial images have highly accurate exterior orientations as their positioning is not suffering from the GNSS signal occlusion and estimated by aerial triangulation as well. The highly accurate positioned images can therefore provide precise locations of features. These features can be measured in the mobile laser scanning data as well.

The first task in the automatic correction pipeline is the matching of the point cloud images with aerial images, after a preprocessing step generates images from the point clouds. The conventional image matching techniques are tailor-made for specific applications, perform poorly and are unable to obtain substantial matches for our dataset, let alone the fulfilment of the near decimetre-level accuracy. Due to the unavailability of a suitable 2D matching technique which can also detect geometrically same feature across both datasets, we invest our efforts in the development of a new feature matching technique. The realization of feature matching technique starts from defining geometrically consistent features viable for matching.

Therefore, for feature detection, we have developed a variant of the Harris corner detector which adopts a threshold regardless of diverse image information. The developed technique detects clusters of keypoints on the vertices of road markings with pixel-level accuracy. For the feature descriptor, we use the LATCH binary descriptor, which is robust while matching the datasets from dissimilar sensors. Due to a large number of outliers, we also developed an outlier filtering technique which exploits the relative Euclidean-distances and relative angles between the corresponding topologies of feature

Summary

sets. In its entirety, the developed feature matching extracts the 2D correspondences with pixel-level accuracy. However, the pixel-level correspondences are not sufficient to improve the MLS dataset to a near decimetre accuracy level. Therefore, we map pixel-level correspondences to subpixel based on features detected from the Förstner-operator in the last phase of the feature matching.

Once the subpixel-level stereo correspondences are known, the remaining tasks are the multiview matching and the computation of the 3D tie points by 3D triangulation of the multiview matches. The 3D tie points determine the 3D-3D correspondences between the point cloud and aerial images, and are normally sufficient for the adjustment and the correction of the MLS dataset. For the evaluation of 3D tie point accuracy, it is necessary to develop a method which first performs an adjustment of the six degrees of freedom trajectory using 3D-3D correspondences and then evaluate the enhanced trajectory.

For the enhancement of the trajectory, changes of the six pose parameters over time are represented by B-spline functions and enhanced by updating the spline coefficients using three types of observations. The main observation for the trajectory enhancement is the 3D-3D correspondences between MLS point cloud and aerial images: it contains high local and global consistency. The high global consistency is necessary to achieve accuracy in the absolute coordinate system, while the local consistency is important to achieve the relative accuracy in the dataset. The second type of observation is based on data from the Inertial Measurement Unit (IMU). It provides only local consistency, but at a high frequency. The third type of observation is based on constraints applied to the pitch and yaw rotations of the vehicle, which reduces the degrees of freedom. To enhance the trajectory, these observations are added to a normal matrix and solved for the unknown changes to the coefficients. The computed changes to the coefficients are then added back to the coefficients of trajectory in an iterative process. The iterative process stops when the square sum of the residuals no longer improves. The procedure to select a trajectory segment for the experiments is based on several checks. First, we verify the high positioning accuracy of the start and end positions of the trajectory segment using checkpoints. Alternatively, if checkpoints are not available exactly at the start and end locations, we make sure that a sufficient amount of 3D tie points exist at those locations. The latter check also means that the trajectory's start and end positions are always located within the areas covered by more than one aerial image. Second, we verify that all other parts of the trajectory segment are as well covered by the aerial imagery. Third, depending on the evaluation criteria, we select the trajectory segment which has the availability of checkpoints after short intervals. The interval of the checkpoints availability should be shorter for the evaluation of the trajectory constructed only with the

IMU observations, while this interval can be larger for the evaluation of the trajectory enhanced by all observations.

When comparing with other available methods, the developed method is fully automatic while successful in obtaining a near decimetre-level accuracy. The quality of the trajectory enhancement is analysed under several conditions. Experiments are designed to test the influence of the number and quality of 3D-3D correspondences and try to determine the accuracy in the areas where checkpoints are not available. The experiment conducted on two datasets demonstrated that the enhanced trajectory achieved near decimetre-level accuracy. The RMSE values of the original Kalman filter results at the checkpoints were 0.26 m, 0.30 m and 0.47 m for the X-, Y-, and Z-coordinates in one of the two datasets and 1.10 m, 1.51 m, and 1.81 m in another dataset. After the trajectory adjustment these RMSE values were reduced to 0.09 m, 0.11 m, and 0.16 m for the first dataset and 0.12 m, 0.14 m and 0.18 m for the second dataset. In this project, the complete pipeline of the MLS dataset correction is implemented and as well delivered to the stakeholders.

Summary

Samenvatting

Mobiele laserscanning is een techniek om geo-informatie in te winnen voor grote gebieden met laserscanners die op een auto of een ander voertuig zijn gemonteerd. Het ingewonnen data wordt een mobiele laser scanning puntenwolk genoemd. Onder normale omstandigheden wordt een nauwkeurige positie bepaald door de integratie van Global Navigation Satellite Systems (GNSS) en Inertial Navigation Systems (INS). Vooral in stedelijke gebieden belemmeren hoge gebouwen echter een directe zichtliin naar navigatiesatellieten en leiden ze tot multipath-effecten. In deze situatie missen de afgeleide producten van mobiele laserscanpunt wolken de verwachte nauwkeurigheid en nemen ze een onbekende positioneringsfout over.

Dit probleem is onderzocht door veel onderzoekers, met als doel het verkleinen van de positiefouten met gebruik van externe datasets, zoals digitale kaarten, ortho-afbeeldingen of laser scangegevens vanuit de lucht. Deze datasets dienen als een betrouwbare bron voor de positionering en worden zodanig als referentiedatasets gebruikt.

Dit onderzoek heeft tot doel de positiebepaling van mobiele laserscanplatforms te verbeteren door multiview nadir luchtfoto's met hoge resolutie als referentiedataset te gebruiken. De luchtfoto's hebben zeer nauwkeurige positie, en objecten die in de foto's zichtbaar zijn, kunnen nauwkeurig worden ingemeten. Het idee is dat diezelfde objecten ook in de mobiele laserscandata gemeten kunnen worden en dat ze als het ware aan elkaar gematched kunnen worden.

De eerste taak in de automatische verbeterstap is het matchen van de puntenwolk met luchtfoto's. De standaard matchingstechnieken zijn vooral voor het matchen van luchtfoto's gemaakt en presenteren minder goed voor het matchen van luchtfoto's met mobiele laserscan data. Daarom hebben we een nieuwe "feature based" matchingtechniek ontwikkeld. De ontwikkeling van onze feature matching-techniek begint met het definiëren van geometrisch consistente features die geschikt zijn voor matching.

Daarom hebben we voor de featuredetectie een variant van de Harrishoekdetector ontwikkeld. De ontwikkelde techniek herkent clusters van punten op de hoeken van wegmarkeringen, zoals zebra's, met pixel-nauwkeurigheid. Om de eigenschappen voor die punten vast te leggen gebruiken we de LATCH binaire descriptor, die robuust voor zowel de mobiele laser scandata als de luchtfoto's. Vanwege een groot aantal data uitschieters hebben we ook een filtertechniek ontwikkeld die de relatieve Euclidische afstanden en relatieve hoeken tussen de overeenkomstige punten bekijkt. De matching krijgt op deze manier een nauwkeurigheid op pixelniveau, dat echter niet genoeg is om de MLS-data te verbeteren tot een nauwkeurigheid van bijna een decimeter. Daarom verbeteren we deze stap nog op basis van functies die zijn gedetecteerd door de Förstner-operator.

Zodra de matching op subpixelniveau bekend is, zijn de resterende taken de multiview-matching en de berekening van de 3D verbindingspunten door 3Dtriangulatie van de multiview-matches. De 3D-verbindingspunten bepalen de 3D-3D-overeenkomsten tussen de puntenwolk en luchtfoto's en zijn normaal gesproken voldoende voor de aanpassing en de correctie van de MLS-data. Voor de evaluatie van de 3D-verbindingspunten is een methode ontwikkeld die eerst een aanpassing van het traject uitvoert met behulp van de 3D-3Dcorrespondenties en vervolgens het verbeterde traject evalueert.

Voor de verbetering van het traject worden veranderingen van de zes trajectparameters in de loop van de tijd weergegeven door B-spline-functies en verbeterd door het bijwerken van de spline-coëfficiënten met behulp van drie soorten waarnemingen. De belangrijkste waarneming voor de trajectverbetering zijn de 3D-3D-overeenkomsten tussen MLS-puntenwolk en luchtfoto's: het bevat een hoge lokale en globale consistentie. De hoge globale consistentie is noodzakelijk om nauwkeurigheid in het absolute coördinatensysteem te bereiken, terwijl de lokale consistentie belangrijk is om de relatieve nauwkeurigheid in de data te bereiken. Het tweede type waarneming is gebaseerd op gegevens van de Inertial Measurement Unit (IMU). Het biedt alleen lokale consistentie, maar met een hoge frequentie. Het derde type waarneming is gebaseerd op voorwaardes aan de orientering van de auto, waardoor het aantal vrijheidsgraden worden verminderd. Om het traject te verbeteren, worden deze waarnemingen toegevoegd aan een normaal-matrix en opgelost voor de onbekende -te schatten- veranderingen in de coëfficiënten. De berekende veranderingen in de coëfficiënten worden vervolgens toegepast op de spline-coëfficiënten een iteratief proces.

De procedure om een trajectsegment voor de experimenten te selecteren, is gebaseerd op verschillende controles. Eerst verifiëren we de hoge positioneringsnauwkeurigheid van de begin- en eindposities van het trajectsegment met behulp van controlepunten. Indien controlepunten niet exact beschikbaar zijn op de start- en eindlocaties, zorgen we ervoor dat er voldoende 3D-verbindingspunten bestaan op die locaties. De laatste controle betekent ook dat de begin- en eindposities van het traject zich altijd binnen de gebieden bevinden die door meer dan één luchtfoto worden bedekt. Ten tweede controleren we of alle andere delen van het trajectsegment ook worden bedekt door de luchtfoto's. Ten derde selecteren we, afhankelijk van de evaluatiecriteria, het trajectsegment dat over controlepunten beschikt. In vergelijking met andere beschikbare methoden, is de ontwikkelde methode volledig automatisch en slaagt erin een nauwkeurigheid van bijna een decimeter te verkrijgen. De kwaliteit van de trajectverbetering wordt onder verschillende omstandigheden geanalyseerd. Experimenten zijn ontworpen om de invloed van het aantal en de kwaliteit van 3D-3D-verbindingspunten te testen en proberen de nauwkeurigheid te bepalen in de gebieden waar controlepunten niet beschikbaar zijn. Het experiment is uitgevoerd op twee datasets en toonde aan dat het verbeterde traject een nauwkeurigheid van bijna een decimeter bereikte. De RMSE-waarden van de oorspronkelijke Kalman-filterresultaten op de controlepunten waren 0,26 m, 0,30 m en 0,47 m voor de X-, Y- en Z-coördinaten in een van de twee datasets en 1,10 m, 1,51 m en 1,81 m in een andere dataset. Na de trajectaanpassing werden deze RMSE-waarden verlaagd tot 0,09 m, 0,11 m en 0,16 m voor de eerste dataset en 0,12 m, 0,14 m en 0,18 m voor de tweede dataset. In dit project is de volledige workflow van de MLS-data en traject verbetering geïmplementeerd en eveneens aan de gebruikerscommissie geleverd.