New Methods for 3D Reconstructions Using High Resolution Satellite Data

A thesis accepted by the Faculty of Aerospace Engineering and Geodesy of the University of Stuttgart in partial fulfilment of the requirements for the degree of Doctor of Engineering Sciences (Dr.-Ing.)

by

Ke Gong

born in Jingzhou, China

Main referee:	Prof. Dr.-Ing. habil. Prof. h.c. Dieter Fritsch
Co-referee:	Prof. Dr.-Ing. Konrad Schindler
Date of defense:	14.12.2020

Institute for Photogrammetry
University of Stuttgart
2021

This thesis was published online on:

http://www.dgk.badw.de/publikationen/reihe-c-dissertationen.html
and http://elib.uni-stuttgart.de

List of Acronyms

2.5D Two-and-half-dimensional. 7, 13–17, 19–21, 33, 49, 63, 107, 110, 111, 115, 119, 123–126

2D Two-dimensional. 23–25, 27, 39, 40, 49

3D Three-dimensional. 13–23, 26–31, 33, 38, 41, 45, 48, 49, 61, 70, 77, 81, 83–86, 88, 91, 93, 95, 98, 106, 107, 110, 115, 119, 123–126

ASP Ames Stereo Pipeline. 7, 8, 27, 107, 110–120

DLR Deutsches Zentrum für Luft- und Raumfahrt. 62, 64, 67, 85, 86, 88, 91, 124

DOM Digitales Oberflächen-Modell. 15, 16

DSM Digital Surface Model. 5, 13, 14, 17–21, 26–28, 33, 35, 37, 41, 45, 48, 49, 57, 58, 61–63, 75, 83–86, 88, 93–95, 104, 106, 107, 110, 115, 118, 119, 123, 124, 126

GCP Ground Control Point. 13, 15, 18, 20, 23–25, 38–41, 63–66, 68, 69, 83

GPU Graphics Processing Unit. 29, 107, 125

GSD Ground Sampling Distance. 13, 15, 17, 18, 22, 25, 45, 53, 54, 57, 61–63, 65, 69, 70, 75, 83, 85, 106, 107, 115, 118, 119

IARPA Intelligence Advanced Research Projects Activity. 27, 62–64, 70, 75, 85, 91, 106

IDW Inverse Distance Weighted. 48

ISPRS International Society for Photogrammetry and Remote Sensing. 18, 26, 61, 70, 75, 84

JAX Jacksonville Testsite. 63, 106, 107, 115, 118, 119

JHU/APL John Hopkins University Applied Physics Laboratory. 7, 18, 27, 63, 95, 97–100, 104, 106, 124

MGM More Global Matching. 27, 75, 107

MI Mutual Information. 26, 46

MVS Multi-veiw Stereo. 13–15, 17–22, 26, 28–31, 33, 40, 41, 45, 49, 61–64, 70, 85, 86, 88, 91, 95, 98, 100, 104, 106, 107, 115, 119, 123–126

NASA National Aeronautics and Space Administration. 27, 120

NCC Normalized Cross-Correlation. 27, 56

NMAD Normalized Median Absolute Deviation. 72, 74, 84, 88, 91, 115, 118, 119

OSGM Object-based Multi-image Semi-Global Matching. 28

PRP Project Reference Plane. 25, 26

PTE Projection-trajectory-based Epipolarity. 42, 43, 45

q68 68% quantile of the absolute residuals. 84, 88, 91, 104, 106, 115, 118–120

q95 95% quantile of the absolute residuals. 84, 88, 91, 104, 106

RFM Rational Function Model. 18, 22

RMSE Root-mean-squared-error. 64–70, 72, 74, 84, 88, 93, 94, 104, 106, 115, 118, 119, 124, 137

ROI Region of Interest. 77, 81–83, 86, 99, 100, 107, 110, 115, 119

RPC Rational Polynomial Coefficient. 13–16, 18–26, 33, 38–43, 45, 47–49, 52–56, 61, 63–70, 75, 85, 91, 93, 95, 106, 107, 123–125, 137

SGBM Semi-Global Block Matching. 26, 27, 107

SGM Semi-Global Matching. 18, 19, 26–29, 45, 46, 49, 85, 119

tSGM Tube-based Semi-Global Matching. 6, 7, 13, 15, 18–21, 27, 33, 45–47, 49, 75, 77, 81–88, 90–92, 95, 97–107, 110, 115, 118, 119, 124, 125

UCSD University of California San Diego Testsite. 63, 107, 115, 118, 119

UTM Universal Transverse Mercator. 23, 27, 33, 48, 53, 54, 75, 85, 93, 124

VHR Very High Resolution. 9, 17–19, 42, 61, 62, 64, 67, 70, 75, 85, 124

VRIP Volumetric Range Image Processing. 29, 30

WV WorldView series satellites. 28, 61–64, 68, 75, 77, 84–86, 88, 91, 106

List of Figures

3.1 Workflow of the pipeline 34
3.2 Example of the image has (a) normal brightness and contrast (b) low brightness (c) large contrast. 35
3.3 Example of the image has (a) small incidence angle (b) large incidence angle. 36
3.4 Stereo images collected in different season and their related point cloud.. 36
3.5 Stereo images collected in same season and their related point cloud. 37
3.6 (a) Point cloud generated from summer stereo image pair (b) point cloud generated from winter stereo image pair (c) the Lidar DSM. 37
3.7 Relative pointing error: (a) Red point is the corresponding point in the left image (b) The red point is the corresponding point in right image, the red line is the projected epipolar line and the arrow presents the distance from the epipolar line to the point. 39
3.8 (a) Epipolar geometry of the frame cameras (b) epipolar geometry of the satellite sensors 42
3.9 Epipolar image resampling strategy 44
3.10 Reprojection of pinhole camera 50
3.11 a. The relation of the variation of the ray $\mathbf{d}$ and the surface moving $\delta\mathbf{S}$; b. Relation of displacement within the surface and displacement in the image 51
3.12 Viewing ray of satellite data 55
3.13 The top of the building in: (a) the DSM mesh model (b) full resolution refined mesh model (c) coarse-to-fine refined mesh model (d) the Google Earth image. 57
3.14 The roof of the building in: (a) the DSM mesh model (b) full resolution refined mesh model (c) coarse-to-fine refined mesh model (d) the Google Earth image. 58
3.15 Roof refined with different triangle size: (a) 1 pixel (b) 2 pixels (c) 3 pixels (d) 4 pixels (e) 5 pixels (bottom-middle), and (f) the related area on the Google Earth . . 59
3.16 Tower building refined with different triangle size: (a) 1 pixel (b) 2 pixels (c) 3 pixels (d) 4 pixels (e) 5 pixels (bottom-middle), and (f) the related building on the Google Earth 60

4.1 WorldView-1 image of ISPRS VHR Satellite benchmark: a. La mola test site b. Terrassa test site c. Vacarisses test site. 62
4.2 WorldView-2 image of DLR's Munich dataset 62
4.3 WorldView-3 image of IARPA MVS benchmark: (a) test site 1, (b) test site 2, (c) test site 3. 63

4.4 WorldView-3 image of CORE3D benchmark: a. Jacksonville test site b. University of California San Diego test site. 64
4.5 Relative pointing error on the base stereo pair of San Fernando test site 1: (a) before correction (b) after correction 65
4.6 Relative pointing error on the example stereo pair of San Fernando test site 1: (a) before correction (b) after correction 66
4.7 Relative pointing error on the base stereo pair of San Fernando test site 2: (a) before correction (b) after correction 66
4.8 Relative pointing error on the example stereo pair of San Fernando test site 2: (a) before correction (b) after correction 67
4.9 Relative pointing error on the selected base stereo pair of Munich test site: (a) before correction (b) after correction 68
4.10 Relative pointing error of WorldView-2 data: (a) before RPC refinement (b) after RPC refinement 69
4.11 The anaglyph image of San Fernando test site 3 epipolar image pair 71
4.12 The sub-areas of the San Fernando test site 3 anaglyph image at: (a) the top-left corner (b) the top-right corner (c) the middle of the image (d) the bottom-left corner (e) the bottom-right corner 71
4.13 The vertical parallaxes of San Fernado epipolar image pair 72
4.14 The anaglyph image of Terrassa epipolar image pair 73
4.15 The sub-areas of the Terrassa test site anaglyph image at: (a) the top-left corner (b) the top-right corner (c) the middle of the image (d) the bottom-left corner (e) the bottom-right corner 73
4.16 The vertical parallaxes of Terrassa epipolar image pair 74
4.17 The point cloud of: (a) La Mola test site (b) Terrassa test site (c) Vacarisses test site 76
4.18 The generated DSM of: (a) La Mola test site (b) Terrassa test site (c) Vacarisses test site 77
4.19 La Mola test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud. 78
4.20 Terrassa test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud.. 79
4.21 Vacarisses test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud. 80
4.22 Mountainous area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison 81
4.23 Industrial area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison 82
4.24 Residential area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison 83
4.25 The fused point cloud of Munich test site 85
4.26 The fused DSM generated via our tSGM-based pipeline with WorldView-2 MVS images 86
4.27 Height Difference between (a) the spaceborne-based point cloud generated via our tSGM pipeline (b) the spaceborne-based point cloud generated via DLR's pipeline (c) the spaceborne-based point cloud generated via S2P pipeline and (d) the reference airborne-based point cloud. 87

4.28 Reconstruction details of Frauenkirche Munich: (a) point cloud generated by tSGM pipeline (b) point cloud generated by DLR's pipeline (c) point cloud generated by S2P pipeline (d) Reference point cloud (e) Height profile. 89
4.29 Reconstruction details of a sample building in Munich: (a) point cloud generated by tSGM pipeline (b) point cloud generated by DLR's pipeline (c) point cloud generated by S2P pipeline (d) Reference point cloud (e) Height profile. 90
4.30 Munich city hall area in (a) our tSGM generated point cloud (b) DLR's point cloud (c) S2P point cloud (d) reference point cloud (e) Google Earth 92
4.31 Relation between the number of point clouds and completeness 93
4.32 Relation between the number of point clouds and RMSE 94
4.33 The MVS reconstructed DSM of San Fernando (a) test site 1 (b) test site 2 (c) test site 3. 95
4.34 The fused point clouds of San Fernando (a) test site 1 (b) test site 2 (c) test site 3. . 96
4.35 Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 1. 97
4.36 Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 2. 98
4.37 Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 3. 99
4.38 Reconstruction details of the warehouse: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile. 101
4.39 Reconstruction details of building close to trees: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile. 102
4.40 Reconstruction details of high-rise building: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile. 103
4.41 Reconstruction details of intensive residential area: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile. 105
4.42 Example in JAX test site of (a) the DSM mesh (b) the input Poisson mesh (c) the refined mesh. 108
4.43 Example in UCSD test site of (a) the DSM mesh (b) the input Poisson mesh (c) the refined mesh. 109
4.44 Visualization of the Bank of America financial center generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model and (g) Google Maps snapshots. 111
4.45 Visualization of the Edward Ball building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots. 112

4.46 Visualization of the Florida Times Union building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots. 113
4.47 Visualization of the BB&T building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots. 114
4.48 Visualization of the railway station generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots. 116
4.49 Visualization of the bridge structure generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots. 117
4.50 Ground truth LiDAR DSM data of evaluated building areas of JAX test site (left) and UCSD test site (right). 118

List of Tables

1.1 Current mainstream VHR satellite sensors 19

3.1 Verification of local coordinate systems being close to Cartesian. 54
3.2 Verification of the influence of distance to surface on ray direction. 56
3.3 Verification of the influence of distance between H_1 and $H2$ on ray direction. 56

4.1 Relative pointing errors evaluation of the base stereo of San Fernando test site 1. . . 64
4.2 Relative pointing errors evaluation of the example stereo of San Fernando test site 1. 65
4.3 Object coordinates evaluation of the example stereo of San Fernando test site 1. . . . 65
4.4 Relative pointing errors evaluation of the base stereo of San Fernando test site 2. . . 67
4.5 Relative pointing errors evaluation of the example stereo of San Fernando test site 2. 67
4.6 Object coordinates evaluation of the example stereo of San Fernando test site 2. . . . 68
4.7 Evaluation of the relative pointing errors on the selected base stereo pair of Munich test site. 68
4.8 Relative pointing errors evaluation of Munich test site. 69
4.9 Object coordinates evaluation of the example stereo of Munich test site. 69
4.10 Evaluation of the vertical parallaxes. 72
4.11 Evaluation of the vertical parallaxes. 74
4.12 Height difference evaluation of the ISPRS WorldView-1 benchmark 84
4.13 Evaluation of the Munich test site. 91
4.14 Height difference evaluation of the IARPA MVS benchmark test site 1 104
4.15 Height difference evaluation of the IARPA MVS benchmark test site 2 104
4.16 Height difference evaluation of the IARPA MVS benchmark test site 3 106
4.17 Evaluation results of the JAX and UCSD test site for three MVS methods and corresponding refined surfaces. 115
4.18 Evaluation results of the JAX and UCSD test site for three MVS methods and corresponding refined surfaces. 118
4.19 Evaluation results of the JAX test site for the three MVS methods and corresponding refined surfaces. 119
4.20 Evaluation results of the UCSD test site for three MVS methods and corresponding refined surfaces. 120
4.21 Visualization of the building evaluation for the JAX test site. 121
4.22 Visualization of the building evaluation for the UCSD test site. 122

5.1 The relative pointing error evaluation for all stereo pairs in the Munich test site . . 137

5.2 The check points evaluation for all stereo pairs in the Munich test site 137
5.3 The relative pointing error evaluation for all stereo pairs in the San Fernando test site 1 137
5.4 The check points evaluation for all stereo pairs in the San Fernando test site 1 143
5.5 The relative pointing error evaluation for all stereo pairs in the San Fernando test site 2 149
5.6 The check points evaluation for all stereo pairs in the San Fernando test site 2 152

Contents

List of Acronyms **3**

List of Figures **8**

List of Tables **10**

Abstract **13**

Kurzfassung **15**

1 Introduction **17**
1.1 Motivation . . . 17
1.2 Objectives . . . 20
1.3 Main Contributions . . . 20
1.4 Outline . . . 21

2 Related Work **22**
2.1 Satellite RPC Model and Orientation . . . 22
2.2 Satellite Imagery Epoipolar Rectification . . . 24
2.3 Satellite 3D Reconstruction . . . 26
2.4 Mesh Refinement . . . 29

3 Methodology **33**
3.1 Image Pair Selection . . . 33
3.2 RPC Compensation . . . 38
3.2.1 RPC Model and Bias Compensation . . . 38
3.2.2 Relative Bias-compensating Algorithm . . . 39
3.3 Image Rectification . . . 41
3.3.1 Satellite Epipolar Geometry . . . 42
3.3.2 Piece-wise Epipolar Resampling Algorithm . . . 43
3.4 Point Cloud and DSM Generation . . . 45
3.4.1 tSGM Algorithm . . . 46
3.4.2 Triangulation and DSM Fusion . . . 47
3.5 Mesh Refinement . . . 49
3.5.1 Gradient for Pinhole Cameras . . . 49
3.5.2 Photometric Mesh Refinement for Satellite Imagery . . . 52

4 Experiments **61**
4.1 Description of Test Data . . . 61
4.2 RPC Compensation . . . 63
4.3 Image Rectification . . . 70
4.4 DSM and Point-cloud Generation . . . 75
4.4.1 Stereo Reconstruction for the ISPRS Benchmark . . . 75
4.4.2 3D Reconstruction for Munich Test Site . . . 85
4.4.3 3D Reconstruction for the San Fernando Test Site . . . 91
4.5 Mesh Refinement . . . 106
4.5.1 Reconstruction of 3D Structure . . . 107
4.5.2 Quantitative Evaluation of Multi-View Refinement . . . 115

5 Summary and Outlook **123**
5.1 Summary . . . 123
5.2 Limitations and Outlook . . . 125

Bibliography **127**

Appendix **137**

Acknowledgements **156**

Curriculum Vitae **157**

Abstract

Modern high resolution satellite sensors have been boosted to a new era in the last decades. Nowadays the optical sensors can collect multi-view stereo (MVS) panchromatic images with ground sampling distances (GSDs) of 30-50cm. In combination with the technologies used in the Photogrammetry and Computer Vision domain, dense and accurate Digital Surface Models (DSMs) can be generated from MVS satellite data. Because of the high re-visit frequency and large coverage, satellite imagery plays a big role in the application of global mapping, environmental monitoring, change detection and so on. Moreover, the oblique views of the high resolution satellite imagery enable the extraction of three-dimensional (3D) information of the surface, which has sparked a renewed interest in 3D reconstructions from satellite data.

The goal of this thesis is to develop a framework that generate point clouds and 2.5D DSMs and then reconstruct the 3D surface meshes from MVS satellite imagery. Because of the high data redundancy, the image selection strategy is discussed in our work, so that the process can be more accurate and efficient. While the satellite pushbroom sensor is different from the conventional pinhole camera, there are several problems that need to be solved. First, instead of the interior and exterior orientation parameters, the data vendors provide the Rational Polynomial Coefficients (RPCs) along with imagery, which could be inaccurate. We propose a relative RPC compensation method to refine the initial RPCs. Several tie points are first manually generated, and are then applied to eliminate the bias in one pair of the base stereo images. The base stereo pair is applied to define a virtual surface and the virtual ground control points (GCPs). The proposed relative compensation method accomplishes the RPC bundle block adjustment with the virtual GCPs to correct the RPCs by additional shift parameters and aligns the images to the identical virtual surface. The second problem is that the epipolar line of the satellite imagery is not straight and makes the image rectification challenging. Here, a modified piece-wise epipolar sampling method is proposed. We define an epipolar coordinate system first and then approximate the epipolar lines by several segments. The epipolar segments are resampled pixel by pixel and they are aligned to the same row in the epipolar coordinate system, so that the epipolar images are generated. The accuracies of the proposed relative orientation and epipolar resampling method are evaluated, which reaches sub-pixel level and fulfills the requirement for the good performance of MVS reconstruction. After these two problems are countered, the MVS satellite images are pair-wise densely matched by the tube-based Semi Global Matching (tSGM) algorithm, which has been proved to be accurate and effective in the airborne domain. The matched correspondences are triangulated via the RPC projection to generate the point clouds. Then the point clouds are fused and projected to regular-spaced grids to generate the DSMs. The experiments for DSM generation are conducted on seven different datasets. The height difference between the generated DSMs and reference ground truths are evaluated, which reveals that our pipeline generates accurate and robust DSMs. We also compare

our results to some other state-of-the-art pipelines to show the proposed framework is competitive. To breakthrough the mainstream 2.5D representations of the MVS satellite imagery reconstruction, we present a novel mesh refinement algorithm to recover the true 3D structures of the surface. The proposed method takes a coarse initial mesh as input and refines it by iteratively updating all vertex positions to maximise the photo-consistency between images. Photo-consistency is measured in image space, by transferring texture from one image to another via the surface. The equations to propagate changes in texture similarity are derived through the RPC projection to changes in surface shape, and devise a hierarchical scheme to optimise the surface with gradient descent. In experiments with two different datasets, we show that the refinement improves the initial DSMs generated by dense image matching. Moreover, we demonstrate that it is capable to reconstruct true 3D geometry, such as facade structures, if off-nadir views are available. The report closes with a summary and the discussion of the limitations and possible improvements in future work.

Kurzfassung

Moderne hochauflösende Satellitensensoren haben in den letzten Jahrzehnten eine neue Ära eingeleitet, in der die optischen Sensoren panchromatische Multi-View-Stereo (MVS) Bilder mit einer Ground Sample Distance (GSDs) von 30 bis 50 cm erfassen können. In Kombination mit den Technologien im Bereich Photogrammetrie und Computer Vision kann aus den MVS-Satellitendaten das dichte und genaue Digitale Oberflächen-Modell (DOM) generiert werden. Aufgrund der Häufigkeit von räumlich und zeitlich wiederholten Aufnahmen und der großen Abdeckung spielen Satellitenbilder heutzutage eine große Rolle bei der Anwendung von globaler Kartierung, Umgebungsüberwachung, Änderungserkennung und so weiter. Darüber hinaus ermöglichen die Schrägansichten der hochauflösenden Satellitenbilder die dreidimensionale (3D) Rekonstruktion der Oberfläche, was ein erneutes Interesse an der 3D-Rekonstruktion aus Satellitendaten geweckt hat.

Ziel der vorliegenden Arbeit ist es, eine methodische Arbeitsumgebung zu entwickeln, die Punktwolken und 2.5D-DOM erzeugt und anschließend 3D-Oberflächenmodells aus MVS-Satellitenbildern rekonstruiert. Aufgrund der hohen Datenredundanz wird die Bildauswahlstrategie diskutiert, damit der Prozess genauer und effizienter wird. Da sich der Satelliten-Pushbroom-Sensor von der herkömmlichen Lochkamera unterscheidet, müssen einige Probleme gelöst werden. Erstens stellen die Datenanbieter anstelle der inneren und äußeren Orientierungs-Parameter die Rationalen Polynom-Koeffizienten (RPC) zusammen mit den Bildern bereit, die häufig ungenau sind. Wir schlagen eine relative RPC-Kompensationsmethode vor, um die genauigkeit der RPCs zu verfeinern. Zuerst werden mehrere Verbindungspunkte manuell generiert und angewendet, um die Abweichung in einem Basis-Stereobildpaar zu beseitigen. Das Basis-Stereobildpaar wird angewendet, um eine virtuelle Oberfläche und die virtuellen Bodenkontrollpunkte (GCPs) zu definieren. Das vorgeschlagene relative Kompensationsverfahren wendet die RPC-Bündelblockanpassung mit den virtuellen GCPs an, um die RPCs durch zusätzliche Verschiebungsparameter zu korrigieren und kann die Bilder an der identischen virtuellen Oberfläche ausrichten. Das zweite Problem ist, dass die Epipolarlinie der Satellitenbilder nicht gerade ist und daher die Bildkorrektur schwierig macht. Hier wird ein modifiziertes stückweises epipolares Abtastverfahren vorgeschlagen. Wir definieren zuerst ein epipolares Koordinatensystem und approximieren dann die epipolaren Linien durch mehrere Segmente. Die epipolaren Segmente werden Pixel für Pixel neu abgetastet und im epipolaren Koordinatensystem auf dieselbe Zeile ausgerichtet, so dass epipolare Bilder erzeugt werden. Die Genauigkeit der vorgeschlagenen relativen Orientierung und des epipolaren Resampling-Verfahrens wird bewertet, welches Subpixel-Niveau erreicht und die Anforderung für eine gute Leistung der MVS-Rekonstruktion erfüllt. Nachdem diese beiden Probleme gelöst wurden, werden die MVS-Satellitenbilder paarweise dicht mit dem röhrenbasierten Semi-Global-Matching (tSGM)-Algorithmus angepasst, der sich im Luftbildbereich als genau und effektiv erwiesen hat. Die homologen Pixel werden über die RPC-Projektion trianguliert, um die Punktwolken zu erzeugen.

Die Punktwolken werden verschmolzen und auf ein Gitter mit regelmäßigem Abstand projiziert, um die DOMs zu erzeugen. Die Experimente zur DOM-Erzeugung werden an sieben verschiedenen Datensätzen durchgeführt. Der Höhenunterschied zwischen den generierten DOMs und den Referenzpunkten wird ausgewertet, was zeigt, dass unsere Vorgehensweise genaue und robuste DOMs generiert. Wir vergleichen unsere Ergebnisse auch mit einigen anderen State-of-the-art Verfahren, um zu zeigen, dass der vorgeschlagene Ansatz wettbewerbsfähig ist. Um die gängigen 2.5D-Darstellungen der MVS-Satellitenbildrekonstruktion zu durchbrechen, stellen wir in dieser Arbeit einen neuartigen Algorithmus zur Netzverfeinerung vor, mit dem die wahren 3D-Strukturen der Oberfläche wiederhergestellt werden können. Das vorgeschlagene Verfahren verwendet ein grobes Anfangsnetz als Eingabe und verfeinert es durch iteratives Aktualisieren aller Scheitelpunktpositionen, um die Fotokonsistenz zwischen Bildern zu maximieren. Die Fotokonsistenz wird im Bildraum gemessen, indem die Textur über die Oberfläche von einem Bild auf ein anderes übertragen wird. Die Gleichungen zur Ausbreitung von Änderungen der Texturähnlichkeit werden durch die RPC-Projektion auf Änderungen der Oberflächenform abgeleitet und entwickeln ein hierarchisches Schema zur Optimierung der Oberfläche mit Gradientenabstieg. In Experimenten mit zwei verschiedenen Datensätzen zeigen wir, dass die Verfeinerung die anfänglichen DOMs verbessert. Darüber hinaus zeigen wir, dass das Verfahren in der Lage ist, echte 3D-Geometrie wie Fassadenstrukturen zu rekonstruieren, wenn Off-Nadir-Ansichten verfügbar sind. Die Arbeit schließt mit einer Zusammenfassung und der Diskussion zur Einschränkung und möglichen Verbesserung zukünftiger Arbeiten.

Chapter 1

Introduction

1.1 Motivation

Three-dimensional (3D) reconstructions of surface models from airborne and space-borne imagery is a long standing topic in photogrammetry and computer vision. The imagery collected by airborne platforms has very high resolution. For instance, the latest DMC III camera from Leica Geosystems provides panchromatic images at 2.1cm Ground Sample Distance (GSD) and multi-spectral images at 6.7cm GSD, when the flight height is 500m. With well designed flight track, aerial images can be highly overlapped and highly redundant. Because of these features, the airborne imagery plays a big role in the surface reconstruction field. The state-of-the-art algorithms allow for country-scale reconstructions with an impressive level of detail and a high degree of robustness, as demonstrated for instance by the city models now included in virtual online globes. The 2.5D digital surface models (DSMs) generated from airborne nadir imagery is the most common product. More recent camera systems can also collect oblique views, which provide strong support to the 3D information (like facade elements) extraction and enable the reconstruction of true 3D structures. The reconstructed 3D surface geometry is typically represented as triangular mesh due to efficient storage, availability of neighborhood information and its good visualization capabilities.

Image acquisition via satellite sensors is a fast and efficient method. But compared to airborne imagery, traditional satellite sensors collect images with lower resolution, less redundancy and only nadir views. Since the successful launch of the first Very High Resolution (VHR) satellite IKONOS in September 1999, satellite sensors have made tremendous development and boosted to a new era over the last decade. In table 1.1, we list several current mainstream VHR satellite sensors in terms of spatial resolution of panchromatic images, temporal resolution and also swath width at nadir. The latest VHR satellite sensors, such as for instance WorldView-3, can even provide panchromatic images with down to 0.3m GSD, large swath width and also high redundancy. The sub-meter level resolution can reveal more details of the surface. These Earth observation satellites cover most regions of our planet, and they can collect large footprint data of any certain site. The high revisit frequency enables large data redundancy and makes the multiple view stereo (MVS) satellite imagery acquisition feasible. Note that the MVS satellite imagery is usually collected on different dates, which will be affected by the season changes of the terrain. Because of these benefits, the VHR satellite imagery becomes more and more valuable for global 3D mapping, urban planning, environmental environmental monitoring, change detection and so on. Therefore, the launch of new VHR satellites has lead to a renewed interest in detailed reconstruction from spaceborne imagery.

Compared to conventional airborne photogrammetry, VHR satellite data is able to provide the accurate 3D model for much larger area coverage. The high revisit frequency of the satellite sensor ensures that the MVS data is available in short time. The satellite data is free from the expertised flight route planning and it is a cheaper, stable and efficient approach to collect large urban scenes. Although the 3D models generated from airborne photogrammetry have higher precision, the latest satellite data with 30cm GSD can also provide many details. Moreover, with the development of satellite sensors, the precision of satellite data will be higher and higher in the future. The oblique views are available in MVS satellite imagery, which also enable the reconstruction of 3D information like building facades. Because of these features, VHR MVS satellite images have great potential in the area of accurate 3D reconstructions.

To enable research advancing the state-of-the-art in satellite images reconstructions, several well-organized benchmark datasets have been established, for instance the VHR benchmark dataset provided by the International Society for Photogrammetry and Remote Sensing (ISPRS) Working Group I/4 [ISPRS, 2010] and the publicly MVS benchmark of commercial satellite imagery released by the John Hopkins University Applied Physics Laboratory (JHU/APL) [Bosch et al., 2016]. With the new published datasets, plenty of researchers have explored the 3D reconstruction with VHR satellite imagery [d'Angelo and Reinartz, 2011, Wohlfeil et al., 2012, De Franchis et al., 2014, Qin, 2017, Facciolo et al., 2017]. Among these methods, the Semi-Global Matching (SGM) [Hirschmüller, 2008] method and its variants are the most popular algorithm to generate dense 3D point clouds, due to its good compromise between quality and computational costs. The tube-based Semi-Global Matching (tSGM) algorithm [Rothermel et al., 2012] has been proved to be effective and robust in the dense image matching of airborne imagery. Based on this method, we propose our MVS satellite imagery 3D reconstruction pipeline. We firstly apply tSGM to the 3D reconstruction of the space-borne imagery and then obtain accurate matching results. The pipeline starts from the MVS satellite image and their corresponding Rational Polynomial Coefficients (RPCs). The projection model of satellite sensors composed by RPCs is named as Rational Function Model (RFM), which is also known as RPCs model. The RPCs are the parameters that can easily describe the relation between image and object space, which is the dominant sensor model for satellite images. Usually, the RPCs require a further adjustment to improve its accuracy to sub-pixel level. Most methods conduct the RPC bundle block adjustment with some GCPs [Grodecki and Dial, 2003, Fraser and Hanley, 2003, Fraser et al., 2006]. Some works process the RPCs through a pair-wise relative orientation, but a further alignment of the point clouds would be required in the subsequent procedure [De Franchis et al., 2014]. To tackle this problem, we propose a relative additional bias-compensated method to refine the RPCs to sub-pixel accuracy without GCPs. The RPCs are aligned to a virtual surface during the compensation, so that no further alignment is needed. The tSGM algorithm needs the epipolar stereo pairs as inputs. Different from the traditional pinhole camera, the satellite sensors have changing altitude and perspective centers. It is hard to build the epipolar geometry and resample the epipolar images. Many references like [De Franchis et al., 2014, Ghuffar, 2016, Facciolo et al., 2017] generate the local epipolar straight lines and resample them tile-wise. In order to sidestep the discrepancy between different tiles' edges, we propose a modified piece-wise epipolar resampling strategy based on the work of [Oh, 2011]. The epipolar lines are approximated as a combination of segments and the epipolar image is resampled without tiling. With the epipolar images, we then employ the tSGM algorithm to match the images pairwise and triangulate the point cloud via forward intersection. At last, we project the point clouds into discrete regular size grids and fuse them to a final DSM as product.

Table 1.1: Current mainstream VHR satellite sensors

Satellite Sensor	Spatial Resolution (m)	Temporal Resolution (days)	Swath width (km)
Worldview-3/4	0.31	<1	13.1
Worldview-2	0.46	1.1	16.4
Worldview-1	0.46	1.7	17.6
Geoeye-1	0.46	2.8	15.2
Pléiades 1A/1B	0.5	1	20.0
SuperView-1	0.5	2	12.0
Kompsat-3A	0.55	1.4	12.0
QuickBird	0.65	1-3.5	16.8
Gaofen-2	0.8	5	45.0
TripleSat	0.8	1	23.4
IKONOS	0.82	3-5	11.3

As a common photogrammetric product which is widely applied in the surveying and mapping and Geographic Information System (GIS) area, DSM is also the predominant representation for satellite-based 3D reconstructions. But this 2.5D representation of the surface reconstruction is not satisfying. With the high resolution and high redundancy of the latest satellite imagery, the extraction of real 3D geometry like balconies and other facade structures, seems to be in reach. Most existing algorithms, however, only produce 2.5D height maps or surfaces [d'Angelo and Reinartz, 2011, Wohlfeil et al., 2012, Kuschk, 2013, Shean et al., 2016] and do not even attempt to recover 3D details. Besides the 2.5D scene representation, conventional reconstruction pipelines have further drawbacks. Since our MVS satellite imagery 3D reconstruction pipeline is based on the pair-wise (dense) stereo and subsequent fusion of stereo models, which does not fully exploit the multi-view redundancy. As we apply the tSGM algorithm for dense image matching, the price to pay are modeling errors such as fronto-parallel bias (caused by rectangular matching windows) and a preference for areas of constant disparity, the same as all the SGM-like algorithms [Roth and Mayer, 2019, Scharstein et al., 2017]. It is also well-documented, that methods that estimate sub-pixel disparities in discrete disparity space introduce further systematic errors [Shimizu and Okutomi, 2002, Szeliski and Scharstein, 2004, Gehrig and Franke, 2007]. The fusion of individual stereo models into a single, consistent height field is most often done with heuristic rules, which certainly improve robustness and accuracy, but are nevertheless sub-optimal. In particular, visibility and occlusions are often handled poorly, or not at all.

To sidestep the mentioned limitations, we extend our MVS satellite imagery reconstruction pipeline and propose a novel approach that reconstructs the surface by using a 3D mesh representation instead of the 2.5D DSM. Our method is a local optimisation starting from an initial mesh, i.e., it refines an existing surface model, for instance our conventional 2.5D stereo result. Following [Delaunoy et al., 2008, Vu et al., 2012], we assume that the coarse initial mesh is topologically correct and refine it by iteratively moving its vertices in the direction that most reduces the texture transfer error across all views. Technically, this is implemented as a variational energy minimisation, subject to a surface smoothness prior. Here, we formulate the corresponding energy function for the RPC model.

1.2 Objectives

The objectives of this thesis is to build and evaluate a new pipeline for MVS satellite images 3D reconstruction, that generate accurate surface models. The input of the pipeline is the satellite MVS imagery and the RPC files. The pipeline delivers the 2.5D DSMs and the true 3D mesh models or point clouds as outputs. In detail the following steps shall be investigated:

- **Image selection**: The satellite MVS imagery has very high redundancy. It is necessary to select the most useful stereo pair for our experiments, so that the computing efficiency and accuracy can be improved.
- **Orientation of satellite imagery**: The satellite MVS data usually provides RPCs to describe the projection relation between the image and the object space. Traditional image orientation methods with interior and exterior parameters are not suitable to satellite images. We need to solve the orientation problem of the MVS satellite images, sometimes even without any ground control points (GCPs).
- **Epipolar geometry of satellite imagery**: The satellite pushbroom sensor is different from the pinhole cameras. We need to approximate the epipolar geometry and then resample the epiplar images properly to achieve sub-pixel vertical parallaxes.
- **Dense image matching and DSM fusion**: The tSGM algorithm will be applied for the dense image matching. We need to adjust the parameters for the matching to obtain accurate 3D point clouds. The point clouds are then put into UTM grids to generate the DSMs. Since the matching process is done pair-wise, a DSM fusion process should be applied to generate the final DSM. The proper numbers of the DSMs involved in the fusion should also be investigated. And the quality of the generated DSM should be evaluated, as well as the processing time.
- **Mesh refinement**: The cost energy function of the refinement for satellite MVS imagery should be established and derived in detail. Based on a given initial surface, it should be proven that our method can refine the mesh and recover the 3D detail structures. The processing time, accuracy and robustness of the algorithm should be evaluated and discussed.

1.3 Main Contributions

The first contribution of this thesis is a relative RPC bias-compensated method. We first select some tie points from the satellite images. The RPCs of one stereo pair is refined with the tie points. The compensated RPCs of this stereo image pair are applied to generate a virtual surface. We then compensate the rest RPCs without GCPs and align them to the virtual surface plane. No further alignment of the generated point clouds is required in the subsequent process. We have conducted our method on different datasets and checked the discrepancies between the corresponding points, which show the accuracy is sub-pixel level and fulfill the requirement of the following matching process. The second contribution is that we propose a modified piece-wise epipolar resampling strategy. The epipolar line of the satellite images is approximated by multiple segments. We resample the whole image without tiling along the epipolar segments. The resampled epipolar stereo images are verified to have no vertical parallax. Next we employ the tSGM for the very first time to satellite data. The parameters of tSGM are adjusted to fit the satellite imagery. Accurate

3D point clouds and DSMs are generated by tSGM and the following forward intersection. In the case of MVS images, we conduct the fusion of the DSMs and discuss the proper number of the stereo pairs, which should get involved in the fusion. We present the pipeline to generate 2.5D DSMs from MVS satellite imagery. It is shown, that our pipeline can produce accurate and robust results. And last but not least, we establish a novel mesh refinement scheme for MVS satellite data. The mesh refinement starts from an initial surface and it is refined iteratively to reach the minimal cost energy by gradient decent. We formulate the energy function with the RPCs of the satellite images. The whole algorithm is implemented by our self-coded program, and it is tested on several benchmark datasets. It is demonstrated for the first time, that the reconstruction of full 3D surface structures in feasible, by incorporating satellite views with large off-nadir angles. Moreover, we show that the refinement also tends to improve the accuracy of the 2.5D elevation values.

1.4 Outline

This document will be separated in four main parts as follows. First we review the related work of our pipeline in the upcoming chapter 2. The basic concepts about the critical RPCs and the image orientation algorithms are introduced. The satellite epipolar geometry and resampling methods are then presented. Moreover, the state-of-the-art of the 3D reconstruction methods for satellite MVS imagery are reviewed. At last, we give an overview about the knowledge of standard mesh refinement pipelines. After going through the relevant work of our 3D reconstruction method, we explain conceptional ideas and implemented details of our 3D reconstruction method in chapter 3. We first show the workflow and then introduce the algorithms of every step. Specifically, we present the strategy of image selection, the proposed RPC relative bias-compensation method, the modified piece-wise epipolar geometry and resampling method, the overview of tSGM and also our application details. We also put emphasis on the explanation of the details of the proposed novel mesh refinement algorithm for MVS satellite imagery. Another main part of this document is the experiments and evaluations, which are demonstrated in chapter 4. In this chapter, we test our algorithms on different datasets and evaluate the results of several main steps. Furthermore, our 3D reconstruction products, such as DSMs and 3D mesh models, are compared to other state-of-the-art pipelines to identify the advantages and limitations of our algorithm. In the last chapter 5, we give a conclusion about our 3D reconstruction method and also have a discussion about the outlook for future work.

Chapter 2

Related Work

In the last two decades, optical satellite sensors have been boosted to a new era of satellite data applications. In this chapter, we first introduce the satellite imaging model and the geopositioning method in section 2.1. Section 2.2 will give an overview of the epipolar rectification procedure, and the section 2.3 introduces the related researches about the 3D reconstruction of high-resolution MVS imagery. At last, the related researches in the mesh refinement area are reviewed in 2.4.

2.1 Satellite RPC Model and Orientation

On 24th September 1999, the first commercial high-resolution Earth observation satellite IKONOS was launched, which made the collection of satellite imagery at 1m GSD publicly available. The overview of the IKONOS sensor has been presented in G. Dial and his colleagues [Dial et al., 2003]. This first high resolution satellite sensor inspired researchers to explore the satellite model that bonds the image coordinate space to the object coordinate space. As well-known, the high-resolution satellites apply pushbroom sensors, which collect a single image line at an instant of time. Unlike frame cameras that have unique exterior parameters for an entire image, the position of the perspective center and the attitude angles of the satellite pushbroom sensors are different for each scan line [Grodecki, 2001].

To avoid the expensive and time-consuming physical model of the optical satellite sensors, a pure mathematic model named as Rational Function model (RFM) or simultaneously Rational Polynomial Coefficients (RPC) [Hartley and Saxena, 1997] model is derived. The satellite data vendors also prefer to provide RPCs along with the imagery to avoid delivering further satellite information. The RPCs are composed of eighty coefficients. The RPC model of the satellite sensors has no physical meaning but only represents the relation between the image and object coordinates by a ratio of two third order polynomials. Each satellite image has its own RPCs document for imaging modelling. By applying the IKONOS datasets, [Grodecki, 2001] has verified that the RPC model can replace the rigorous physical sensor model and maintain the accuracy at the same time. The RPCs, which are provided along with the satellite imagery, can be inaccurate. The geo-positioning errors are mainly caused by the measurement of the attitude angles of the satellite sensor [Grodecki and Dial, 2003]. In image space, the geo-positioning errors introduce biases from pixel to tens of pixels. The RPCs can be refined by compensating the error with additional parameters. In 2002, [Dial and Grodecki, 2002] and [Fraser et al., 2002a] separately proposed a bias-compensation

method in the image space for the RPCs orientation. [Hanley et al., 2002] has verified that the drift effects are sub-pixel level if the image size is small. The simple shift bias compensation model is an effective model in most cases and the influence of the higher order parameters is negligible. [Fraser and Hanley, 2005] have verified that the orientation quality of the bias-compensated RPC model can reach sub-pixel level on both IKONOS and Quickbird data. They point out that the number and the location of the ground control points (GCPs) are critical for absolute orientation. A single GCP is sufficient for the shift compensation model. If the 2D affine model is applied for compensation, it requires at least three GCPs. In practical, four to six appropriately located GCPs are a better choice to the 2D affine compensation model. Grodecki and Dial [Grodecki and Dial, 2003] proposed the bias-compensated RPCs bundle block adjustment for stereo images. This solution compensates the biases with an additional affine model for image coordinates and it is widely applied for the RPCs orientation [d'Angelo and Kuschk, 2012, Ozcanli et al., 2015, Gong and Fritsch, 2016]. In their work, the 2D affine compensation model contains two shift parameters and four drift parameters for the horizontal and vertical coordinates. The compensation parameters are applied to absorb the effects of the ephemeris error, the satellite pitch attitude error, the interior orientation errors and the gyro drift during image scanning and so on. The RPC compensation method is straight-forward and can be conducted not only to compensate in image space but also in object space. But [Grodecki and Dial, 2003] pointed out that the compensation in the image space has more accurate results than the compensation in object space. [Hanley and Fraser, 2004] applied the shift compensation model to refine the quality of the RPCs. According to the compensation result, they also regenerate the RPCs with the corrections. The re-generated RPCs can be applied without the additional compensation parameters. Following the idea of [Grodecki and Dial, 2003] and [Hanley and Fraser, 2004], the 2D affine or shift compensation parameters in image space are applied to correct the RPCs in our work.

[Fraser et al., 2002b] pointed out that the 3D affine model can be considered as a special case of the RPC model. Instead of applying the RPC compensation method, [Fraser and Yamakawa, 2004] has given an insight of the 3D affine model for satellite sensors and tested it on IKONOS imagery. In their experiment, the object coordinates refer to the Universal Transverse Mercator (UTM) coordinate system. According to their tests, the 3D affine model is robust and practical for the orientation. [Noguchi et al., 2004] has verified the orientation performance of the bias-compensated RPC bundle block adjustment and also the 3D affine model on QuickBird imagery. They found out that the shift and drift terms warrant the RPC bundle block adjustment to produce high geopositioning accuracy. But the 3D affine model is less encouraging for the QuickBird imagery. The 3D affine model can not compensate the non-linear systematic image errors and produces poor orientation results.

All the methods mentioned above require a few number of ground control points (GCPs). The GCPs in a certain region are not always easily to access for the satellite data. In this situation, the relative orientation of the stereo image pairs is needed. Without the GCPs, the bias of the pixels in the imagery is hard to calculate. To solve this problem, [De Franchis et al., 2014] proposed the relative pointing error instead of the bias for compensation. A pair of stereo images are given, and two corresponding points exist on the images. As known, one point is related to one epipolar line. The epipolar line can be generated by the projection from the homologous image point. In the satellite case, the projection is achieved by the RPC model. If the RPCs are accurate enough, for point x in the base image, the corresponding point x' should be located on the the corresponding epipolar line EP in the warp image. If not, the distance from the corresponding point to the corresponding epipolar line is cited as the relative pointing error. They point out

that the relative point errors can be measured as simple translations if the image size is small (e.g. 1000×1000 pixels). They first pick several tie points and calculate the translations to compensate their relative pointing errors in small image tiles. The relative pointing error of the image is removed by applying the median of the translations. [Ghuffar, 2016] has verified that the relative pointing error compensation can achieve sub-pixel accuracy during the relative orientation. The proposed RPC correction algorithm in this work is also strongly related to the relative pointing error correction proposed by [De Franchis et al., 2014]. Instead of refine small tiles with shifts, we apply the 2D affine model to correct the relative pointing error for the whole image. We correct the relative pointing error for only one stereo pair to build a virtual surface. Then with the help the the virtual surface, the the bias-compensated RPCs bundle block adjustment [Grodecki and Dial, 2003] is employed for all the stereo pairs in our proposed algorithm.

[Qin, 2017] applied the relative orientation by using tie points for the bias compensation. If the height H in the object space and also the image coordinates s and l are known, the inverse RPC model can be applied to calculate the longitude and latitude. With the help of the inverse RPC model, he produces the artificial GCPs from the tie points by projecting the tie points to the mean height plane. The bias compensation is carried out with the artificial GCPs.

2.2 Satellite Imagery Epoipolar Rectification

Epipolar rectification warps an image pair such that the corresponding pixels share the same row index, which reduces the correspondences search range from 2D to 1D space. The rectified epipolar image pairs can be applied for dense image matching and improve the efficiency significantly. Unlike the pinhole cameras, the satellite image scene is generated from stitched 1D scan lines of the pushbroom sensors [Wolf and Dewitt, 2000]. The satellite pushbroom sensors are hard to establish the epipolar geometry because of the changing perspective center and attitude. Therefore, the rectification algorithms like [Fusiello et al., 2000, Loop and Zhang, 1999, Pollefeys et al., 1999] can not be applied for the satellite images. However, [Kim, 2000] have explained, that correspondences exist locally on a pair of epipolar lines across the images. Moreover, these epipolar lines are more like hyperbola curves than straight lines.

Some researchers have investigated the approximation of the satellite epipolar geometry and also the epipolar image resampling. Several critical characteristics of the satellite sensor can be summarized: the pushbroom sensor's scanner has very narrow angular field of view; the satellite image scene is acquired within a very short time so that the velocity and altitude is assumed as constant [Morgan et al., 2004b]. The constant velocity and altitude are not satisfied for a longer image strip. [Habib et al., 2005] has investigated the epipolar geometry of along-track and across-track satellite stereo images. They found that the image scenes with narrow angular field of view yield straight epipolar lines. Therefore, instead of the perspective projection, the satellite imagery can be assumed to comply with parallel projection. [Ono, 1999] proposed a method for the resampling of the satellite imagery based on 2D affine projection. The non-linear and linear forms of the parallel projection model are explained in his work. Ono also derived the transformation of the scene coordinates from the perspective to parallel projection. A similar parallel projection model based method is derived and applied to resample the high-resolution satellite epipolar images by [Morgan, 2004]. Sub-pixel level vertical parallaxes can be achieved on different satellite datasets with the parallel projection based model [Morgan et al., 2004a, Morgan et al., 2004b, Habib et al.,

2004, Morgan et al., 2006]. The parallel projection based method requires prior information like the scanner navigation data or GCPs.

As well-known, the RPCs are utilized for the projection between the object and image space. Instead of using the 2D affine projection, the RPCs projection can be applied to describe projection trajectory and establish the epipolar geometry. [Zhao et al., 2008] has proposed a projection trajectory based method to generate the epipolar lines via the RPCs. According to their method, the point on the left scene is projected to several elevation levels in object space and then projected into the right scene to obtain a series of image points on the right scene. These image points are applied to fit a straight line as the epipolar line. [Wang et al., 2010] applied the similar projection-trajectory based epipolar model to generate the epipolar pairs. They also proposed an simple epipolar arrangement method to resample the epipolar images. In the case of along-track stereo pairs, the horizontal axis of the epipolar image is parallel to the y-axis of the original image and the vertical axis indicates the indices of the epipolar lines. For a cross-track stereo image, the horizontal axis of the epipolar image is parallel to the x-axis of the original image and the vertical axis presents the epipolar indices. The epipolar image is resampled along the epipolar lines derived from the middle point of horizontal direction. This epipolar resampling method is practical but needs information to distinguish the along or cross track cases. To establish the epipolar geometry, we applied the RPCs and the projection-trajectory based epipolar model [Wang et al., 2010] in our work.

According to the local conjugacy and the possible straightness approximation of the satellite imagery's epipolar geometry, [Wang et al., 2011] defined a Project Reference Plane (PRP) in a local vertical coordinate system. The PRP is an average elevation plane in object space with certain elevation. They randomly select a point p on the base image and project it to two height levels which are upper and lower than the PRP. Then the points are projected to the matching image and obtain two image points p' and p''. p' and p'' are applied to project to the PRP. The intersections on the PRP are P' and P'', which will be used to approximate the direction of the epipolar straight line ED. Therefore the original satellite image stereo pair can be projected to the epipolar stereo pair that is parallel to the approximate epipolar line ED on the PRP. The projections are conducted via the RPCs model. The coverage of the epipolar images on the PRP is defined by the minimum rectangular region that contains both original images projection. The horizontal boundary of the coverage is parallel to the epipolar line direction of ED. The coverage area on the PRP is adjusted to grid units with the GSD of the original imagery. The relation between the planarity coordinates on the PRP and the pixel coordinates of the epipolar images can be described as a 2D affine transformation. The points on the original imagery are projected to the PRP and then transformed to the epipolar image points. In general, the PRP is applied as a bridge between the original and epipolar imagery.

Considering that the epipolar pair only exists for local areas, [Oh, 2011] proposed a piece-wise epipolar curve generation and epipolar resampling method for the entire image. In his method, the center point of the left image is selected to calculate the center epipolar line. The orthogonal line of the center epipolar line passing the center point can be generated. The points on the orthogonal with predefined interval (e.g. 1000 pixels) are selected as the start points. The start points on the left image are projected from the lowest height of the area and then projected to the right image to obtain the start points of the right image. The epipolar lines are expanded from the start points. The length of the expanded epipolar line is defined by a predefined height range (e.g. $0 - 1000$m), and the height range is equal to or larger than the actual terrain elevation range. The points on the

generated epipolar lines are re-arranged, so that the epipolar line pairs are assigned to a constant row. The curve-to-straight adjustment can be done by a high order polynomial transformation. The work of [Oh, 2011] inspired us, and we proposed a modified piece-wise epipolar resampling method.

Based on the work of [Wang et al., 2011] and [Oh, 2011], [Koh and Yang, 2016] proposed an unified piecewise epipolar resampling method in object space. The start points selected from the image space are projected to the PRP. The epipolar segments are derived on the PRP and the length is defined by the maximum and minimum height of the area. The fifth order polynomial function is applied to establish the relation between the epipolar curve on the PRP and the rows of the epipolar images.

[Zheng et al., 2015] proposed a minor solver for the inverse RPC mapping and then applied it to obtain the epipolar pairs. The inverse RPC function is converted to describe a polynomial equation system, which has up to nine solutions with a known elevation. The Gröbner basis method [Stewénius, 2005] has achieved great success in the field of computer vision, and it is applied to solve the polynomial equation system here. With the minor solver of the inverse RPC mapping, the points on the first image can be projected to several elevations in the object space. The epipolar lines in the second image are projected by the object points. With redoing the procedure from the second image, they find the epipolar pairs of the satellite imagery.

2.3 Satellite 3D Reconstruction

Generally, the MVS imagery reconstruction methods can be classified into two categories. The first category processes the single stereo pair separately and then fuses the reconstructed outputs, which can be noted as binocular method. The second category solves the multi-view triangulation problem simultaneously with all images, which is the so called true multi-view method. The true multi-view method is more rigorous but more complicated.

The majority of 3D reconstruction algorithms for optical satellite imagery employ the binocular method. The scheme of pairwise epipolar rectification and then dense image matching, followed by the fusion of depth maps, are widely applied in different satellite 3D reconstruction pipelines [d'Angelo and Reinartz, 2011, Wohlfeil et al., 2012, Shean et al., 2016, Rupnik et al., 2017]. Considering stability, computational efficiency and low memory consumption, many reconstruction pipelines employ some variants of Semi-Global Matching (SGM) methods for dense stereo matching. The SGM algorithm was first proposed by [Hirschmüller, 2008]. The SGM method computes the similarity of the potential corresponding pixels on the stereo images as the matching cost. The pixel-wise matching is achieved by employing the minimization of the energy function, which aggregats the cost along 8 or 16 paths. [d'Angelo and Reinartz, 2011] generate point clouds and DSMs on the ISPRS benchmark [Reinartz et al., 2010] with the classical SGM algorithm. Based on the SGM method, [Wohlfeil et al., 2012] established a full automatic pipeline to generate DSMs from high resolution satellite data. [d'Angelo, 2016] investigate the compensation of overcounting [Drory et al., 2014] in the context of SGM for satellite imagery and observe improved density but decreased precision. A modified version of SGM method is implemented in open source libraries [Bradski, 2000], which is called the Semi-Global Block Matching (SGBM) algorithm. Instead of searching along eight paths, SGBM only considers about five directions. SGBM calculate the cost function by Birchfield-Tomasi sub-pixel metric [Birchfield and Tomasi, 1998] instead of the mutual information (MI). The matching procedure focuses on blocks of the pixels but not on each individual pixels. [De Franchis et al., 2014] showed state-of-the-art performance on the stereo imagery of the Pleiades

data with this off-the-shelf SGM implemented in [Bradski, 2000]. [Di Rita et al., 2017] develop a fully automatic pipeline based on the SGBM algorithm and also acquire state-of-the-art results on Pleiades imagery. The SGBM algorithm is also applied in the stereo processing pipeline of National Aeronautics and Space Administration (NASA) [Moratto et al., 2010]. [Shean et al., 2016] present the reconstruction performance of NASA's open stereo reconstruction software Ames Stereo Pipeline (ASP) on WorldView-1/2 products. [Qin, 2017] show state-of-the-art performance on the IARPA satellite benchmark released by JHU/APL [Bosch et al., 2016], by employing the SGM methods. A free open source software MicMac is developed by [Rupnik et al., 2017], which is based on the SGM method for the DSM generation from satellite imagery. [Rothermel et al., 2012] proposed a hierarchical version of SGM. This modified tube-based SGM (tSGM) limits the disparity search range by applying the disparity priors obtained from the result of lower resolution pyramid levels. It reduces memory footprint and computation time, while also reducing matching ambiguities. The tSGM algorithm has achieved success for the 3D reconstruction from airborne photogrammetry [Haala, 2013]. In this thesis, the first time, the tSGM algorithm [Rothermel et al., 2012] is applied for satellite data. The More Global Matching (MGM) is implemented by [Facciolo et al., 2015], using a modified cost aggregation scheme that aims for globally more consistent disparities measure. Some researchers also try to solve the matching problem without applying the SGM-like methods. As the top-ranked competitor, [Facciolo et al., 2017] employ a pipeline based on MGM method and present state-of-the-art performance on the IARPA satellite benchmark [Bosch et al., 2016].

Instead of the SGM-like algorithms, some approaches start the matching of satellite images from the traditional area-based or feature-baed matching algorithms. [Capaldo et al., 2012] proposed a hierarchical matching method which combines the geometrical constrains and an area-based matching algorithm to improve the effectiveness and reliability. [Duan et al., 2016] employ the feature-based matching with an integrated similarity measure, which combines the distance vector, angle vector and the Normalized Cross-Correlation (NCC) of the points. [Wang and Frahm, 2017] construct dense correspondence maps from sparse feature matches via edge aware interpolation, so that the processing avoids costly energy minimisation altogether.

In order to obtain a consistent representation of the surface, the pair-wise matched results have to be fused eventually. Typically, this is realized by binning 3D points from multiple models into a regular and discretized 2D grid in the UTM coordinate system. Various filtering strategies are applied to derive a single elevation value per grid cell. One of the most popular strategy is the simple and robust median filtering method [Kuschk, 2013, d'Angelo and Kuschk, 2012, Wang and Frahm, 2017], which is also applied in our pipeline. [Facciolo et al., 2017] take the changing surface modes (e.g, vegetation) caused by different acquisition dates into account. They propose k-median clustering for each grid cell and favour observations from lower clusters. In the spirit of bilateral filtering, [Qin, 2017] proposed an adaptive depth fusion method, which considers the spatial consistency. According to his method, a polygon window containing cells, that have a certain degree of similarity, is applied to find the candidates. The median filtering is conducted on the elevations of the candidate cells, so that the noise in flat regions will be reduced. [Kuschk et al., 2017] cast the fusion of multiple stereo models as a convex energy minimization problem and solve it with a primal-dual algorithm, including an additional planarity prior in the form of a total variation (TV) or total generalized variation (TGV) regularization term and a L1 data fidelity term. [Rupnik et al., 2018] proposed a multi-directional dynamic programming approach based fusion method. To enhance the performance on non-texture area, the DSMs are first clustered to n-depth cells. The cells are then

smoothed with a recursive exponential filter. The multi-direction dynamic programming is applied to find the most probable depths of the cells by minimizing an energy function.

Some approaches circumvent the somewhat cumbersome pairwise processing and subsequent fusion and apply the true multi-view methods. [d'Angelo and Kuschk, 2012] directly estimate a DSM by assigning photometric similarity costs to a regular 3D cost structure in object space. The final elevation map is derived by semi global optimization. However, they find only limited gains compared to the more prevalent late fusion of binocular stereo models. Similarly, [Bethmann and Luhmann, 2014] proposed an object-based multi-image Semi-Global Matching (OSGM) method. The OSGM method discretizes the object space in to voxels and project the images on to these voxels. The matching cost calculation of OSGM is formulated with a dense voxel raster by applying the intensities of all the multi-view images. The matching correspondences are determined by the result of the semi-global minimization of the cost, which are the index-maps that indicate the 3D positions. [Ghuffar, 2016] implemented the pipeline with both SGM and OSGM method and then compared the performance. They found the accuracy of the two algorithms are close. [Wang et al., 2016] estimate an elevation grid and additionally fuse semantic information from multiple satellite images and corresponding semantic segmentation maps. To estimate surface elevations, Patch-Match Belief Propagation (PMBP) [Besse, 2013] is employed to maximize an energy function that encourages consistency of appearance and semantics across several images. Additionally, smoothness of semantics, height values and surface orientations are enforced in their method. [Pollard and Mundy, 2007, Pollard et al., 2010] propose a probabilistic voxel-based model to jointly reconstruct surface voxels and their corresponding colours. The prediction of the color model is based on the Gaussian mixture model. The geometric surface probability is initialized to a constant and the Gaussian mixture model is initialized with the color observed on the first image. Given a sequence of images and their camera models, the rays from the images are projected into the voxels. The surface probabilities and the color modes are updated according to the appearance of the rays until convergence. To our knowledge, this is the only published method in the satellite domain which is capable of extracting real 3D geometry. It does, however, not include any explicit surface prior, and [Ozcanli et al., 2015] found that both urban and rural reconstructions are less accurate than those from pairwise matching and late fusion on different satellite data sets.

As the one of the hottest research topic of last decade, deep learning approaches have also been introduced to the domain of 3D reconstruction from satellite imagery. To reduce the influence of the differences between different satellite MVS views, [Treible et al., 2018] represents one of the first attempts to generate DSM from WV-3 satellite imagery. Their work is built upon Pyramid Stereo Matching architecture [Chang and Chen, 2018]. In their experiments, multiple modules are applied for feature extraction. The feature maps are then aggregated with a correlation operation into a 4-dimensional cost volume. Each level represents a disparity between left and right feature maps. After regularization and smoothing, the disparity is regressed by taking the softmax of the cost. According to their results, the DSMs generated by deep learning approaches have sharper building edges than the DSMs generated by the SGM algorithm, but present less detailed information on the building roofs. [Chen et al., 2019] made a comparison between census feature based matching methods and fast Convolutional Neural Network [Girshick, 2015]. They generated DSMs from WV-1 satellite imagery via the two methods and the result's accuracy of two methods are in the same order of magnitude. However, DSMs generated by fast Convolutional Neural Networks show higher accuracy and completeness. [Zeng et al., 2018] apply deep neural network (DNN) to analyze the entire building structures. With the analyzed shape grammar rules, they reconstruct the CAD-

quality 3D models from satellite images. In their work, the residential buildings has sharp edges but low details. [Xu et al., 2020] focus on the building roofs and reconstruct 3D models from input point clouds with shape segmentation information generated by deep learning algorithm. PointNet [Qi et al., 2017] is applied as their segmentation model. They propose a hierarchical RANSAC to extract shapes from point cloud. The reconstructed 3D models of their algorithm has high completeness. But we note that only planar roof structures are concerned in their work, the facade 3D information are not considered.

2.4 Mesh Refinement

There are much less references in the satellite domain when comparing to the conventional pinhole camera model, presumably because of the limited availability of high-resolution imagery. Thus, we review the relevant work on the mesh refinement in the close range and airborne domains.

The typical approaches of the mesh refinement require a coarse mesh model to provide the initial geometry (e.g. [Pons et al., 2007], [Furukawa and Ponce, 2010], [Vu et al., 2012]). In order to gain the initial mesh model, the depth maps or point clouds with normals are first generated from MVS reconstruction. For the pinhole camera model, the depth map is usually generated via the searching of the maximal photo-consistency value between the image views [Esteban and Schmitt, 2004, Goesele et al., 2006, Campbell et al., 2008, Woodford et al., 2009]. Inspired by [Barnes et al., 2009], [Bleyer et al., 2011] propose the PatchMatch Stereo algorithm for 3D reconstruction. Their algorithm starts from random depth values, then refine the depth iteratively by a spatial propagation and a fast bisection search. Many researches implement the PatchMatch algorithm on graphics processing units (GPU) to accelerate the processing, and they show that this algorithm is effective and efficient to estimate the depth maps [Bailer et al., 2012, Zheng et al., 2014, Schönberger et al., 2016, Galliani et al., 2016]. [Furukawa and Ponce, 2010] propose a patch-based MVS algorithm for point cloud reconstruction. They detect feature points to generate the initial patches, then expand patches in nearby empty space, and at last apply filters to eliminate the erroneous patches. For MVS images collected from pinhole cameras, the SGM algorithm [Hirschmüller, 2008] is also widely applied for the point clouds or depth maps generation (e.g. [Rothermel et al., 2012, Mayer et al., 2012, Ahmadabadian et al., 2013, Rothermel, 2017]). As to satellite domain, the depth maps or point clouds can be generated by the methods introduced in section 2.3.

Next, the volumetric approaches followed by the marching-cube type algorithm or triangulation of the point cloud can be applied to bootstrap a topologically correct mesh representation. Poisson Surface Reconstruction [Kazhdan et al., 2006] is a popular and successful meshing algorithm, which considers the reconstruction as a spatial Poisson problem of the oriented points. [Kazhdan and Hoppe, 2013] extended the Poisson Surface Reconstruction by considering the points as soft constrains for interpolation, so that the oversmooth of the mesh is restrained. The signed distance is common applied to determine the mesh surface. [Curless and Levoy, 1996] proposed a function composed of an accumulated signed distance function and also an accumulated weighted function by applying all the range images for each voxel grid. The optimal zero iso-surface of the function is then extracted as the location of the mesh plane. This method is also called as Volumetric Range Image Processing (VRIP). Similarly, [Zach et al., 2007] calculated the truncated weighted signed distance field from range images. An energy function involving a total variation regularization term and a L1 data fidelity term is minimized, so that the zero iso-surface related to the distance field can be determined. [Zach, 2008] extended the former method by storing the signed distance values into

the histograms and improved the efficiency. [Fuhrmann and Goesele, 2011] implement a method close to VRIP, but only depth maps at compatible scales are involved into the cost calculation. By sharing the same basic idea, [Fuhrmann and Gösele, 2014] proposed a method based on an implicit function with floating scales. The implicit function is composed of basis functions, which are compactly supported by the positions, normals and scales of the sample surface. The zero iso-surface of the implicit function is extracted as the mesh surface without global operations. This method can handle the large noisy datasets and it is virtually parameter-free. [Ummenhofer and Brox, 2015] also introduced the scale information into the cost function of the signed distance, but they conducted the optimization globally on a balanced octree structure. Unlike the aforementioned algorithms, [Alliez et al., 2007] reconstruct the mesh surface directly from unoriented point clouds. The Principal Component Analysis (PCA) of the Voronoi diagram of the input point clouds is applied to estimate a tensor field that represent the most likely normal directions. They then computed an implicit function whose gradient matches the normal directions best, so that the iso-surface can be extracted as the mesh surface.

[Vogiatzis et al., 2005] considered the scene surface as the interface of the foreground and background and applied the graph-cuts on the voxel grid to optimize the surface. With an approximate surface as a hard constrain, the surface cost function is described with a weighted graph. The surface cost function is composed of a ballooning term and the photo-consistency measurement. To reduce the computing time and also the memory requirement, [Hernández et al., 2007] discretized the voxel grid with octree data structure. Moreover, instead of using the ballooning term, they applied the probabilistic evidence for the visibility and strengthen the performance in concave regions. [Sinha et al., 2007] apply the graph-cut on the tetrahedral grid, which is subdivided according to the photo-consistency values. Unlike the voxel based graph-cut algorithms, [Labatut et al., 2007] applied the graph-cut with the Delaunay triangulation to extract the surface model, so that no knowledge of the scene extent (silhouette) is required. First, a quasi-dense point cloud is generated with SIFT matching [Lowe, 1999]. Then a Delaunay triangulation is applied to decompose the space with tetrahedrons. The tetrahedrons are labelled as inside or outside of the scene. The triangular facets between inside and outside tetrahedrons are the mesh surface. The globally optimal label is achieved by computing the minimum s-t cut of the graph. Based on the work of [Labatut et al., 2007], [Labatut et al., 2009] improved the visibility term of the cost function by introducing a tolerance parameter and modifying the corresponding weight construction. The photo-consistency term of the cost function is replaced by a surface quality term, so that the formulation is free from any area-based smoothing. In order to handle the weakly-supported surfaces, [Jancosek and Pajdla, 2011] modified the graph weights from the formulation proposed by [Labatut et al., 2009] by taking the large free-space-support jump into consideration. A large free-space-support jump means the free-space-support of a tetrahedron close to the surface labelled as outside should be much larger than a tetrahedron near the surface labelled as inside. According to this assumption, the t-edge weights are changed as the number of the cameras related to the four points of the tetrahedron, and the free-space-support weight applies the number of the cameras occluded by the tetrahedron. In contrast to the algorithms with oriented point clouds, [Hornung and Kobbelt, 2006] applied the graph-cut algorithm to extract a mesh surface by using the unoriented point sets. For each voxel, an extended crust is generated via the morphological dilation operator to the neighborhood. The unsigned distance functions of the voxels are then computed via volumetric diffusion and represent the confidence. The graph is applied to the confidence weighted voxel grid and its minimum cut yields the faithful approximation of the surface. For the latest results of the MVS 3D reconstruc-

tion, we refer the readers to one of the active MVS benchmarks [Seitz et al., 2006, Schöps et al., 2017, Knapitsch et al., 2017].

In order to recover fine details and improve precision, the vertex positions of the initial variational meshes can be further refined [Pons et al., 2007, Delaunoy et al., 2008, Vu et al., 2012]. The common framework of the mesh refinement employs all the MVS image pairs to re-evaluate the photo-consistency score or reprojection error over the initial surface. The photo-consistency score is usually defined by an energy function. The vertex positions are optimized to achieve the minimum energy or maximum similarity. The adjusted movement of the vertex can be presented as the gradient, and the gradient decent is usually applied for the optimization [Furukawa et al., 2015]. [Yezzi and Soatto, 2003] proposed a variational formulation of the continuous (smooth) surfaces for stereoscopic segmentation and reconstruction. A cost function presenting the re-projection errors is applied to minimize variations, which is composed of data fidelity term, smoothness term and a geometric prior. They set up the gradient flow equation to find the surface and radiance that minimize the cost function. [Gargallo et al., 2007] apply the same variational formulation as [Yezzi and Soatto, 2003] to MVS case and minimize the variations via gradient descent. Their work provides the computation method of the exact gradient of the reprojection error, which also takes the visibility into consideration. The two former researches require Lambertian scenes, which is not common in real lives. To reconstruct non-Lambertian objects, [Soatto et al., 2003] introduce a constrained radiance tensor to the energy cost function. Based on the work of [Yezzi and Soatto, 2003] and [Gargallo et al., 2007], [Delaunoy et al., 2008] proposed the rigorous formulation of the gradient of the reprojection error for discrete meshes (non-smooth surface) to circumvent the limitation of a continuous surface. Their computation takes the visibility change into account and requires the input mesh surface close to the geometry of the true surface. They also propose a modified Lambertian model to improve the robustness. [Pons et al., 2007] propose a global image-based matching score for the MVS mesh refinement and also the scene flow estimation. By employing the estimated surface, an input image view is projected to predict other image views. The global matching score combines the image similarity measurement of the input view and its predicted view and a user-defined mean curvature based regularization term. The regularization term is applied to smooth the regions with poor evidence. [Vu et al., 2012] propose a MVS pipeline to reconstruct high detailed models for large scale scenes with uncontrolled imaging conditions, which is strongly related to our work. Their triangular mesh-based variational refinement is inspired from the work of [Pons et al., 2007]. A thin-plate term is applied in their work, which considers the principal curvatures of the surface to penalize strong bending. More details will be reviewed in section 3.5.2. [Tyle_ek and Šára, 2010] apply not only the photo-consistency measurement but also the contour matching results to establish the energy function and minimized variations via gradient descent. When the 3D reconstruction is done via patch-based MVS algorithm, [Furukawa and Ponce, 2010] refine the mesh model based on the patches. The depth and orientation information of the surface are estimated via the patch optimization, and they are used to build the energy function. A PatchMatch based mesh refinement is implemented by [Heise et al., 2015]. They integrated the current confidence in the depth estimation and minimize the photo-consistency with a local plane approximation. Because the optimization of the energy function is an iterative global process, the computation is quite expensive. [Li et al., 2016] propose to apply adaptive resolution control to accelerate the process. The regions of the surface are divided into significant and insignificant classes, and they are meshed in adaptive resolution. The refinement is only conducted on the significant regions, so that the process is more efficient and maintains the detail quality. To suppress noises and refine details of

less texture regions, an alternate minimisation of the reprojection error and mesh denoising is given by [Li et al., 2015]. An inter-image similarity measure is applied for reprojection error minimization and a content aware prior is applied to image denoising. The semantic segmentation can provide priors to the regularization and improve the accuracy of the reconstruction. A guideline for the mesh refinement by semantic information is pursued in [Blaha et al., 2017, Romanoni et al., 2017]: semantic consistency across views is enforced and smoothness priors for each class are adapted individually. To the best of our knowledge, all refinement algorithms were formulated for pinhole camera models and cannot be directly applied to satellite data.

Chapter 3

Methodology

In this work, we propose a pipeline to generate the 2.5D elevation maps and also 3D mesh model from multi-view stereo satellite imagery. Comparing to other state-of-the-art satellite MVS 3D reconstruction pipelines [Facciolo et al., 2017, Qin, 2017, Rupnik et al., 2017, Rupnik et al., 2018], the pipeline proposed in this work focus more on ture 3D structures and introduce all the MVS pairs simultaneously to improve the results. Our workflow is shown in Figure 3.1. The pipeline contains several core procedures. First we select a number of suitable image pairs from a large redundancy of the image combination manually. The selected image pairs are processed by a following binocular matching method to generate the 3D point clouds and the DSM. As preparation of the dense image matching, the RPCs are refined via our relative bias-compensated algorithm. We then apply a modified piece-wise epipolar resampling method for the image rectification. The epipolar image pairs are densely matched by the tSGM method individually for each stereo pair. The point clouds of every image pair are triangulated by forward intersection. We then project the point clouds into the regular-size discrete UTM grids and fused them to produce the final DSM. The DSM generating pipeline is also introduced in our previous work [Gong and Fritsch, 2018] and [Gong and Fritsch, 2019]. To recover the 3D structures of the surface, our MVS refinement algorithm requires the input as an initial mesh model with homogeneous triangle distribution. The DSMs are converted to mesh representations and the facade is re-meshed by Poisson Reconstruction. By applying a hierarchical approach, the input Poisson mesh is refined to the true 3D mesh model. The first-hand results of our mesh refinement experiments are also presented in [Rothermel et al., 2020]. Our tSGM-based 3D reconstruction pipeline is semi-automatic and is implemented in our self-programmed C++ modules. In the following sections, we will present further details about our algorithms and implementations.

3.1 Image Pair Selection

As known, the MVS satellite images are usually collected at different dates. Thus, illumination, objects on the ground and seasonal features of the vegetation might vary from image to image. A suitable image selection procedure is needed to avoid the influence of the illumination situation, geometric configuration and season change on subsequent processing. The MVS satellite dataset can provide hundreds or even thousands of stereo pairs to our experiments. The processing of all the MVS satellite stereo pairs is computationally expensive and not meaningful. The image selection strategy will improve the processing efficiency.

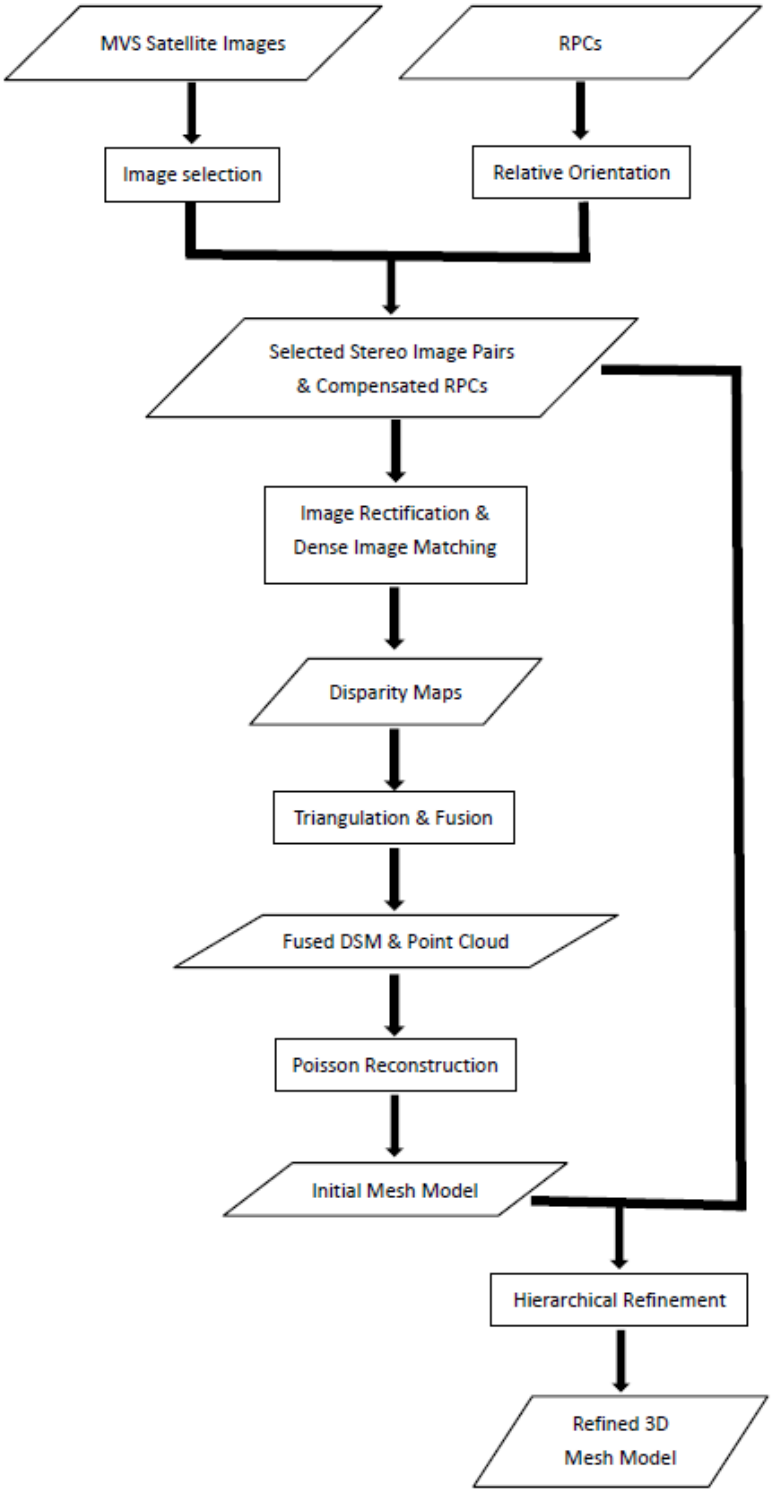


Figure 3.1: Workflow of the pipeline

[d'Angelo et al., 2014] suggested that the intersection angle of the image pairs' views is the biggest factor that impacts performance. They selected the image pairs having intersection angles between 15 and 25 degrees. In order to learn the factors of image selection, [Facciolo et al., 2017] sorted all possible image pairs by the completeness of their computed DSMs, and they built a Pearson's correlation matrix for different factors. According to their observation, the temporal proximity, maximum incidence angle and the intersection angle are three main factors that affect more on the dense image matching side. They suggest the selected images should be acquired on nearby dates with incidence angles less than 40 degrees, and the intersection angle should be between 5 and 45 degrees. [Qin, 2017] also agreed that the intersection angle plays a big role in the quality of the generated DSMs. He pointed out that when the intersection angle of the image pair is smaller than 8 degrees or larger than 40 degrees, the generated DSM performs poor. The image pairs with intersection angles from 10 to 30 degrees are chosen in his work.

Combining the recommendation of [d'Angelo et al., 2014, Facciolo et al., 2017, Qin, 2017] and our own experimental experience, we eliminate the low-quality satellite images from three aspects: brightness, contrast and the incidence angle of the image view. The brightness and contrast of some collected images are much lower than the other images (as Figure 3.2b shows). The contrast of some satellite images could be larger than the regular images (as Figure 3.2c displays). According to our experience, this happens more frequently in winter. The images, which have low brightness and too large contrast, perform poor in the similarity measurement. These images have to be eliminated from the experiments. We also check the influence of the incidence angle for the image. Those images that have large incidence angles (e.g. Figure 3.3b) should be excluded. Because the spatial resolution of the satellite image becomes lower when the view's incidence angle is larger. The detail information of the surface is missing a lot in the images with large incidence angle. In our image selection strategy, we only use the satellite images whose incidence angle is less than 35 degrees. The images having incidence angles larger than 35 degrees are turned out to be less useful in our experiments.

(a) *(b)* *(c)*

Figure 3.2: Example of the image has (a) normal brightness and contrast (b) low brightness (c) large contrast.

When the low-quality images are abandoned, we select the proper stereo pairs from the remaining images. According to former researchers' work [d'Angelo et al., 2014, Qin, 2017, Facciolo et al., 2017], the intersection angle and the collected dates of the stereo images are the two biggest factors that affect a dense image matching. Small intersection angles lead to a high image similarity, but poor

(a) *(b)*

Figure 3.3: Example of the image has (a) small incidence angle (b) large incidence angle.

depth precision. On the other hand, the stereo pairs have high depth precision but low similarity when the intersection angles is too large. Therefore, we select the stereo image pairs with neither too large nor too small intersection angles. According to our test experience, the stereo pairs having intersection angles between 5 degrees and 35 degrees are applied in our pipeline.

In most cases, the images with close collecting date should be our prior choice. Because there will be little changes between the scene of the images. But according to our experiments, there can be two exceptions: 1. The image collecting dates of the stereo images are relatively close, but there are season changes. 2. The interval between the image collecting dates is large, but the surface features are in the same season. These exceptions mainly occur in vegetation areas, because the season change makes tremendous differences. If the data is collected in the same season, the vegetation has minor differences and the objects in the views might keep consistency in neighbouring years.

(a) *(b)* *(c)*

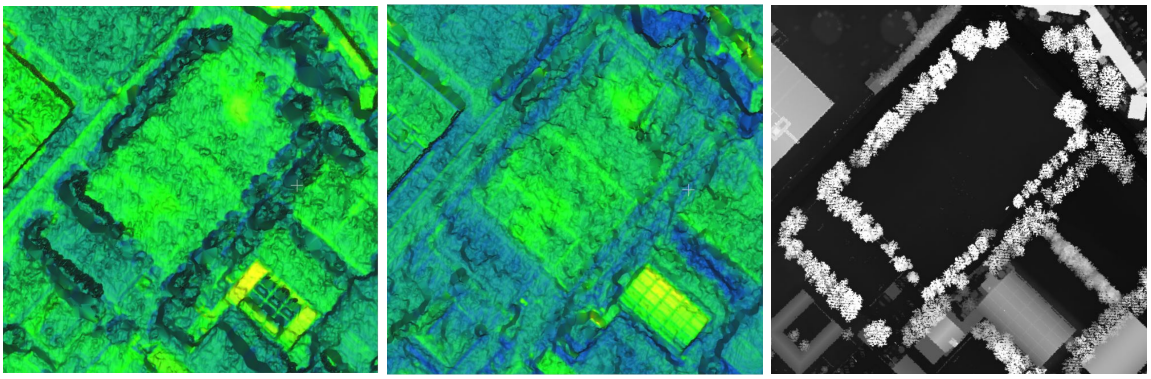
Figure 3.4: Stereo images collected in different season and their related point cloud..

Figure 3.4 and Figure 3.5 demonstrate these two exceptions. Figure 3.4a and Figure 3.4b present two images collected at close dates. We can find that, in the vegetation area highlighted by the red

(a) *(b)* *(c)*

Figure 3.5: Stereo images collected in same season and their related point cloud.

circle, the new leaves have grown in Figure 3.4b. Although these two images are collected in close dates, they have significant differences caused by season change. As Figure 3.4c shows, the point cloud generated from this stereo pair lost a lot of information in this area. Figure 3.5a and 3.5b are the images collected in different years but same season. For the same area shown in Figure 3.4, there are minor season changes in these two images. The related point cloud generated by same season stereo pair, which is displayed in Figure 3.5c, is more complete than the result shown in Figure 3.4c. Taking these two exceptions into account, our image pair selection considers not only the collecting dates but also the collecting seasons.



(a) *(b)* *(c)*

Figure 3.6: (a) Point cloud generated from summer stereo image pair (b) point cloud generated from winter stereo image pair (c) the Lidar DSM.

The season changes have strong effect on the vegetation area. If there is no significant object changes in the earliest and latest collected images, we sort the images into two groups: as winter and summer instead of in a simple chronological order. Figure 3.6a exhibits the point cloud generated from images collected in summer and Figure 3.6b presents the point cloud generated from the stereo image collected in winter. We also show the Lidar DSM of the same area in Figure 3.6c. The images of the summer group can rebuild the vegetation in the point cloud. The images of the winter group

produces complete point cloud but the vegetation could be missing. Because the summer group images usually have better illumination situations and reconstruct more complete vegetation, we recommend to put more focus on the summer group images. Anyway, the vegetation area affects the quality of the 3D reconstruction. If the vegetation is masked out, the whole reconstruction would be more accurate and robust. Both winter and summer groups are applied in our work. In each group, we ignore the year of data collection and order the images only by month. The image pairs with close collecting dates in the same season group are selected as the inputs of our 3D reconstruction pipeline.

In conclusion, we will eliminate the images that have low brightness, too high contrast or incidence angle of the view larger than 35 degree. We divide the qualified images into summer and winter groups and order them according to the collecting dates without considering the year. In each group, we select the image pairs collected in close dates and exclude the pairs that have intersection angles smaller than 5 degrees or larger than 35 degrees. The selected image pairs and the related RPC files are applied as the inputs of our 3D reconstruction pipeline.

3.2 RPC Compensation

To refine the RPCs, we propose a relative bias-compensating algorithm, which does not need the support of GCPs. The RPC compensation method introduced here is based on our former researches [Gong and Fritsch, 2017]. In section 3.2.1, we quickly go through the important concepts of the RPC model of satellite sensors. In section 3.2.2, the details of our relative bias-compensating method is introduced.

3.2.1 RPC Model and Bias Compensation

As known, many satellite vendors would rather keep the interior elements and exterior elements confidential. Instead, the Rational Polynomial Coefficients (RPCs) files are provided along with the imagery to the data users. The RPCs files contain eighty coefficients and also the offset and scale parameters for the object and image coordinates. With these coefficients, a ratio of two third order polynomials is established to present the relation between the image and object space, which is the RPC model of satellite imagery. The RPC model has been proved that this pure mathematic model can replace the rigorous sensor model [Grodecki, 2001, Fraser et al., 2002b, Grodecki and Dial, 2003]. A comprehensive form of the RPC model can be presented as:

$$\begin{bmatrix} s \\ l \end{bmatrix} = \begin{bmatrix} s_{scale} \dfrac{\mathbf{Num}_s^T \mathbf{P}(B_n, L_n, H_n)}{\mathbf{Den}_s^T \mathbf{u}(B_n, L_n, H_n)} + s_{offset} + \Delta_s \\ l_{scale} \dfrac{\mathbf{Num}_l^T \mathbf{P}(B_n, L_n, H_n)}{\mathbf{Den}_l^T \mathbf{u}(B_n, L_n, H_n)} + l_{offset} + \Delta_l \end{bmatrix}. \tag{3.1}$$

Here, s and l are the image's horizontal and vertical coordinates. s_{offset}, l_{offset}, s_{scale} and l_{scale} are the offset and scale parameters of the image coordinates, which we can get from the RPC files. As to $\mathbf{Num}_s$, $\mathbf{Den}_s$, $\mathbf{Num}_l$ and $\mathbf{Den}_l$, each of them contains 20 coefficients of the RPCs. $\mathbf{P}$ presents the third order polynomial of the normalized geographic coordinates B_n, L_n and H_n. We have $B_n = (B - B_{offset})/B_{scale}$, $L_n = (L - L_{offset})/L_{scale}$, $H_n = (H - H_{offset})/H_{scale}$. B, L and H are the longitude, latitude and height. The offset parameters of object coordinates are B_{offset}, L_{offset} and H_{offset}. B_{scale}, L_{scale} and H_{scale} are the scale parameters related to the geographic

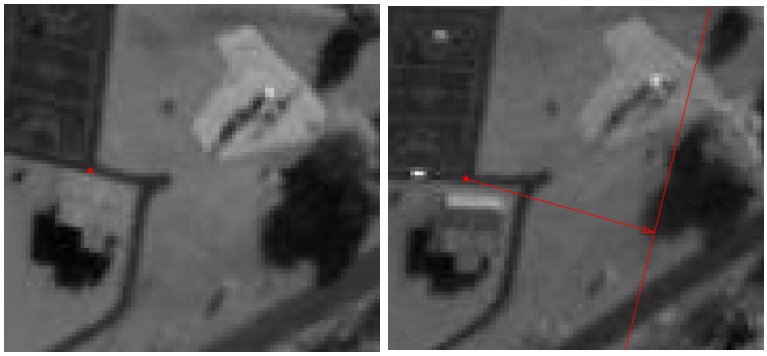
coordinates. When we project points from the object space to the image with the RPCs provided by the data vendors, it may have biases between the calculated image coordinates and the true locations. Δ_s and Δ_l are the bias of the image coordinates. The biases can vary from sub-pixel to tens of pixels for different data. An effective and wide-used method to correct the errors of the RPCs is compensating the bias in image space with some additional parameters. The bias can be approximated as polynomials of the image coordinates, like:

$$\begin{bmatrix} \Delta_s \\ \Delta_l \end{bmatrix} = \begin{bmatrix} a_0 + a_s \times s + a_l \times l + a_{sl} \times s \times l + a_{l2} \times l^2 + a_{s2} \times s^2 + ... \\ b_0 + b_s \times s + b_l \times l + b_{sl} \times s \times l + b_{l2} \times l^2 + b_{s2} \times s^2 + ... \end{bmatrix}. \tag{3.2}$$

where a_0, a_l, a_s, a_{l2}, a_{s2} ... and b_0, b_l, b_s, b_{l2}, b_{s2} ... are the bias-compensation parameters for the image coordinates. The shift parameters a_0 and b_0 are applied to absorb the offsets in the line direction and sample direction caused by the ephemeris error and the satellite pitch attitude error. The drift parameters for line direction a_l and b_l can absorb the effects of gyro drift during image scanning. The drift parameter for sample direction a_s and b_s can reduce the errors of radial ephemeris error and interior orientation errors. a_{l2}, a_{s2}, b_{l2}, b_{s2} and the higher order parameters correct the effects of other small systematic errors [Grodecki and Dial, 2003]. Usually the shift model is enough to compensate the RPCs, because the effects of the higher order parameters are negligible when the image coverage is less than 50km × 50km [Dial and Grodecki, 2002, Fraser et al., 2002b]. To compensate the RPCs in image space, GCPs are required. The shift bias compensation model needs at least one GCP and the 2D affine bias compensation model needs four to six well distributed GCPs. The compensation parameters for RPCs refinement can then be solved by the least squares method with tie points on multiple images and the GCPs.

3.2.2 Relative Bias-compensating Algorithm

The RPC bias compensation requires a proper number of GCPs, but the ground control information is not always available. To circumvent the requirement of the GCPs, we propose a relative bias-compensating algorithm. Our proposed algorithm has two different strategies: for single stereo cases and multiple stereo cases.



(a) *(b)*

Figure 3.7: Relative pointing error: (a) Red point is the corresponding point in the left image (b) The red point is the corresponding point in right image, the red line is the projected epipolar line and the arrow presents the distance from the epipolar line to the point.

For only one stereo image, some tie points on both images are selected manually. We apply the remote sensing software Envi to obtain reliable tie points from the stereo image. According to the work of [De Franchis et al., 2014], the inaccurate RPCs lead to a relative pointing error in image space. Via the RPC model, an image point on the right image is back-projected to the object space and then projected on the left image to get its corresponding point and epipolar line. If the RPCs are accurate, the corresponding point should be located on the corresponding epipolar line. The relative point error is defined as the distance between the corresponding point and the corresponding epipolar line, which is presented in Figure 3.7. [De Franchis et al., 2014] crop the images into small tiles and correct the relative pointing error by simple shifts in image space. To circumvent the inconsistent between different tiles, in our proposed algorithm, the relative pointing errors are corrected for the whole stereo images by a 2D affine model instead of the shift model:

$$\mathbf{D}_p = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \begin{bmatrix} \mathbf{I} \\ \mathbf{S} \\ \mathbf{L} \end{bmatrix}. \tag{3.3}$$

where a_1, a_2, a_3, b_1, b_2 and b_3 are the correction parameters. $\mathbf{I}$ is the unity matrix, $\mathbf{S}$ and $\mathbf{L}$ are the matrices composed by the sample coordinates and line coordinates of the corresponding points. $\mathbf{D}_p$ presents the relative pointing errors of the corresponding points. With the selected tie points, we have the coordinates of the corresponding points and we can calculate their distances to the corresponding epipilar lines. The correction parameters are then resolved by least squares parameter estimation.

For the case of MVS imagery, we select the tie points on all the images. One stereo pair is then selected from the dataset. We followed the fore-mentioned method to correct the relative pointing error. With the correction parameters and the RPCs, we calculate the object coordinates of the tie points via a forward intersection. A part of the tie points are selected as the virtual GCPs and the rests are applied as the check points. The virtual GCPs and check points are applied in the subsequent processing. [Grodecki and Dial, 2003] proposed the bias-compensating RPC bundle block adjustment, which is a popular solution to refine the RPCs with GCPs. They apply an additional affine model to compensate the bias. In our algorithm, we apply the virtual GCPs and the tie points for the bias-compensating RPC bundle block adjustment on all the input images. Because the image size is barely larger than 50km, we abandon the affine model but employ the simple shift model to compensate the bias caused by the inaccurate RPCs. According to 3.1, we can have an observation equation $\mathbf{F}$ as:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_s \\ \mathbf{F}_l \end{bmatrix} = \begin{bmatrix} -s + s_{scale} \frac{\mathbf{Num}_s^T \mathbf{P}(B_n, L_n, H_n)}{\mathbf{Den}_s^T \mathbf{u}(B_n, L_n, H_n)} + s_{offset} + s_{shift} + \epsilon_s \\ -l + l_{scale} \frac{\mathbf{Num}_l^T \mathbf{P}(B_n, L_n, H_n)}{\mathbf{Den}_l^T \mathbf{u}(B_n, L_n, H_n)} + l_{offset} + l_{shift} + \epsilon_l \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{3.4}$$

here $\mathbf{F}_s$ and $\mathbf{F}_l$ are the observation equations for sample coordinates and line coordinates. s_{shift} and l_{shift} are the shift parameters for compensation. ϵ_s and ϵ_l are random errors. The Taylor expansion is applied for the observation equation.

$$\mathbf{F} = \mathbf{F}_0 + d\mathbf{F} + \epsilon = 0 \tag{3.5}$$

where $\mathbf{F}_0$ is the misclosures for the image coordinates. Taking our input data into 3.4 and we can get $\mathbf{F}_0$. The unknowns $\mathbf{x}$ of our observation equation are the shift compensation parameters $\mathbf{x}_C$ and the object coordinates $\mathbf{x}_G$. $d\mathbf{F}$ can be presented as

$$dF = \begin{bmatrix} \frac{\partial \mathbf{F}_s}{\partial \mathbf{x}} \\ \frac{\partial \mathbf{F}_l}{\partial \mathbf{x}} \end{bmatrix} dx = \begin{bmatrix} \frac{\partial \mathbf{F}_s}{\partial \mathbf{x}_C} & \frac{\partial \mathbf{F}_s}{\partial \mathbf{x}_G} \\ \frac{\partial \mathbf{F}_l}{\partial \mathbf{x}_C} & \frac{\partial \mathbf{F}_l}{\partial \mathbf{x}_G} \end{bmatrix} \begin{bmatrix} d\mathbf{x}_C \\ d\mathbf{x}_G \end{bmatrix} = \begin{bmatrix} \mathbf{A}_C & \mathbf{A}_G \end{bmatrix} \begin{bmatrix} d\mathbf{x}_C \\ d\mathbf{x}_G \end{bmatrix}. \tag{3.6}$$

$d\mathbf{x}$ is the correction for the approximate unknowns, which are composed of the correction for shift compensation parameters $d\mathbf{x}_C$ and objects coordinates $d\mathbf{x}_G$. Combining 3.5 and 3.6, we change the observation equation to

$$\begin{bmatrix} \mathbf{A}_C & \mathbf{A}_G \\ \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} d\mathbf{x}_C \\ d\mathbf{x}_G \end{bmatrix} + \epsilon = \begin{bmatrix} -\mathbf{F}_0 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \tag{3.7}$$

or we can write the matrices in short form as:

$$\mathbf{A}d\mathbf{x} + \epsilon = \mathbf{W} \tag{3.8}$$

The least squares solution is applied to solve the unknowns of the equation:

$$d\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{W} \tag{3.9}$$

Because the observation equation is not linear, the bundle block adjustment calculates the approximate value of the shift compensation parameters and the object coordinates iteratively until it reaches convergence. The iteration will stop, when the object coordinates correction is less than 10^{-4}m or the iteration times are over 10,000. With the result of the bundle block adjustment, the biases of the RPCs are compensated by the shift parameters. Moreover, the MVS images are aligned to the surface where our virtual GCPs are located on. Therefore, no further registration is needed for the point clouds and DSMs in the subsequent processing of our pipeline. Note that there is a 3D translation from the surface defined by the virtual GCPs to the true surface in the real world.

3.3 Image Rectification

As a critical pre-procedure of the dense image matching step, the image rectification produces epipolar stereo image pairs. Every corresponding pixels in the epipolar image pairs are located on the same line, so that the search range of the matching is reduced to one dimension. Based on the projection trajectory epipolar model, we propose a piece-wise epipolar resampling strategy for satellite imagery. The proposed image rectification method here is based on our work [Gong and Fritsch, 2017]. In this section 3.3.1, the main concepts of the satellite epipolar geometry are reviewed. Then we introduce the details of our epipolar resampling algorithm in section 3.3.2.

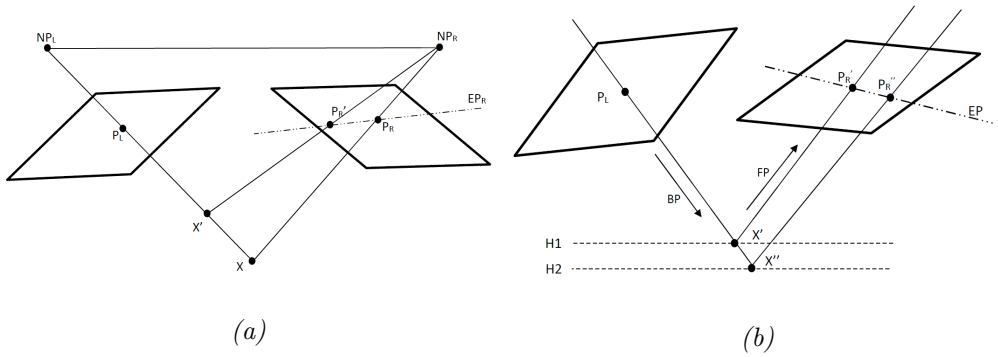


Figure 3.8: (a) Epipolar geometry of the frame cameras (b) epipolar geometry of the satellite sensors

3.3.1 Satellite Epipolar Geometry

In the case of the traditional frame camera, all images are perspective images and have unique perspective centers. The epipolar line is the intersection of the image plane and the epipolar plane. As an example in Figure 3.8a, NP_L and NP_R are the perspective centers for left and right images. P_L is an image point on the left image. X and X' are two object points on the ray when we back-project P_L to the object space. Project X and X' to the right image, and the intersections are P_R and P'_R. The Epipolar line in the right image is EP_R. P_R, P'_R and the corresponding points of P_L in the right image are located on EP_R.

Unlike the frame cameras, the VHR satellite imagery is hard to build its epipolar geometry, because the linear push broom satellite sensors have different perspective centers and attitude in each scanning line. Most satellite sensors have a very narrow field of view. The satellite images are collected in a short period, so that the attitude is assumed as the same for the scene collected and the scanner is assumed that the moving velocity is constant. The projection rays from object to image space can be assumed as parallel [Morgan et al., 2006]. Therefore, no exact perspective center in the satellite imagery and the principal distance is treated as infinity.[Kim, 2000] has verified that the epipolar lines of the satellite sensors exist locally and they are not straight lines but more like hyperbola curves. In small areas, the epipolar lines of satellite imagery can be modelled as straight lines or segments. In order to build the local epipolar geometry of VHR satellite images, the RPC projection-trajectory-based epipolarity (PTE) model proposed by [Wang et al., 2010] is a widely used method. Figure 3.8b presents the basic concept of this model. The projection relation of the satellite imagery is defined by the RPCs. Let the RPC transform from object space to image space [latitude, longtitude, height] $\rightarrow$ [line, sample] as the forward projection (FP) and the transform from image space to object space [line, sample, height] $\rightarrow$ [latitude, longitude] as the back projection (BP). We choose the left image as the base image and the right image as the slave image. P_L is the point in the left image. We back-project it to two different height levels H1, H2 via the RPC model. Points X' and X'' are the intersections in object space. Then we forward-project X' and X'' to the right image and acquire the image points P'_R and P''_R. P'_R and P''_R are applied to approximate the epipolar line EP. The corresponding points of P_L should also be located on EP. We get the corresponding epipolar line in the left image, when redoing the same procedure from one point in the right image (e.g. point P'_R). Note that the selected height levels should be in the true

height range of the covering area. The reason for this constraint is that the RPC model is invalid when the height is too far away from the true surface.

3.3.2 Piece-wise Epipolar Resampling Algorithm

According to the PTE model [Wang et al., 2010], we build the epipolar geometry of the satellite images. Another key point is to approximate the epipolar line and resample the epipolar images. Considering the characteristics of satellite epipolar geometry, [De Franchis et al., 2014] divided the image into small tiles and approximate the epipolar line as straight line in each tile. We prefer to resample the whole image without tiling to circumvent the errors between the borders of each tile. Inspired by the work of [Oh, 2011], we propose a modified piecewise epipolar resampling strategy. Figure 3.9 depicts the details of our strategy. On the left side, the figure shows the procedures for the master image. The procedures for the slave image are demonstrated on the right side. There are three main steps in our epipolar resampling strategy:

In step one, we establish the coordinate system of the epipolar image. As presented in Figure 3.9a, the centre point of the left image C_L is used to calculate the direction of the initial epipolar line in the left image EP_{iniL} with the RPC model. The orthogonal line to the initial epipolar line is OR_{iniL}. Two lines EP'_{iniL} and EP''_{iniL} parallel to the EP_{iniL} and two lines OR'_{iniL} and OR''_{iniL} parallel to OR_{iniL} are applied to define the boundary of the left epipolar image. EP'_{iniL}, EP''_{iniL}, OR'_{iniL} and OR''_{iniL} pass the four corners of the original image. The line OR'_{iniL} is selected as the y-axis Y_{epiL}, and the line EP'_{iniL} is selected as the x-axis X_{epiL} of the left epipolar image coordinate system. The intersected point of the x-axis and y-axis is the origin point O_{epiL} of the left epipolar image coordinate system. Symmetrically, the initial epipolar curve EP_{iniR} and its orthogonal line OR_{iniR} are derived from the center point of the right image C_R. The boundary of the right image is also defined by the parallel lines across the corners of the original image. The bottom boundary line is set as the x-axis X_{epiR} and the left boundary line is chosen as the y-axis Y_{epiR}. Their intersection is the origin point O_{epiR}.

Step two is the approximation of satellite epipolar lines. According to Oh's method, the points on the line perpendicular to the epipolar curve of the left image's centre point are the start points for the epipolar line generation. These start points are set along the perpendicular line with a predefined interval like 1,000 pixels. The epipolar lines are derived piece-wisely from the start points. Instead of this strategy, the points along the y-axis (the bottom boundary line parallel to the orthogonal line of the initial epipolar line) of the left image are selected as the start points. For example, we select a point S_{1L} on the y-axis of the left epipolar image Y_{epiL} as a start point in Figure 3.9b. Starting from the start point, we derive the segment along the direction parallel to the initial epipolar line EP_{iniL}. Once the extended segment reaches the boundary of the original image, we mark the intersection as the first re-start point P_{1L}. Then a new direction of the epipolar segment is calculated via the RPC based PTE method from P_{1L}. The epipolar segment will keep growing until it reaches a proper length. We apply the height range given by the RPCs to define the length of each segment. The end point of the segment P'_{1L} is the next re-start point to generate the following segment. We repeat the epipolar segments generation until the epipolar segment is out of the boundary of the original image. In the case of the right image, we start from the start point S_{1R} on the y-axis of the right epipolar image. The epipolar segments are derived according to the direction of the initial epipolar line of the right image EP_{iniR}. When we find the re-start points of epipolar segments (P_{1L}, $P'_{1L}\ldots$) in the left image, the conjugate points on the right image

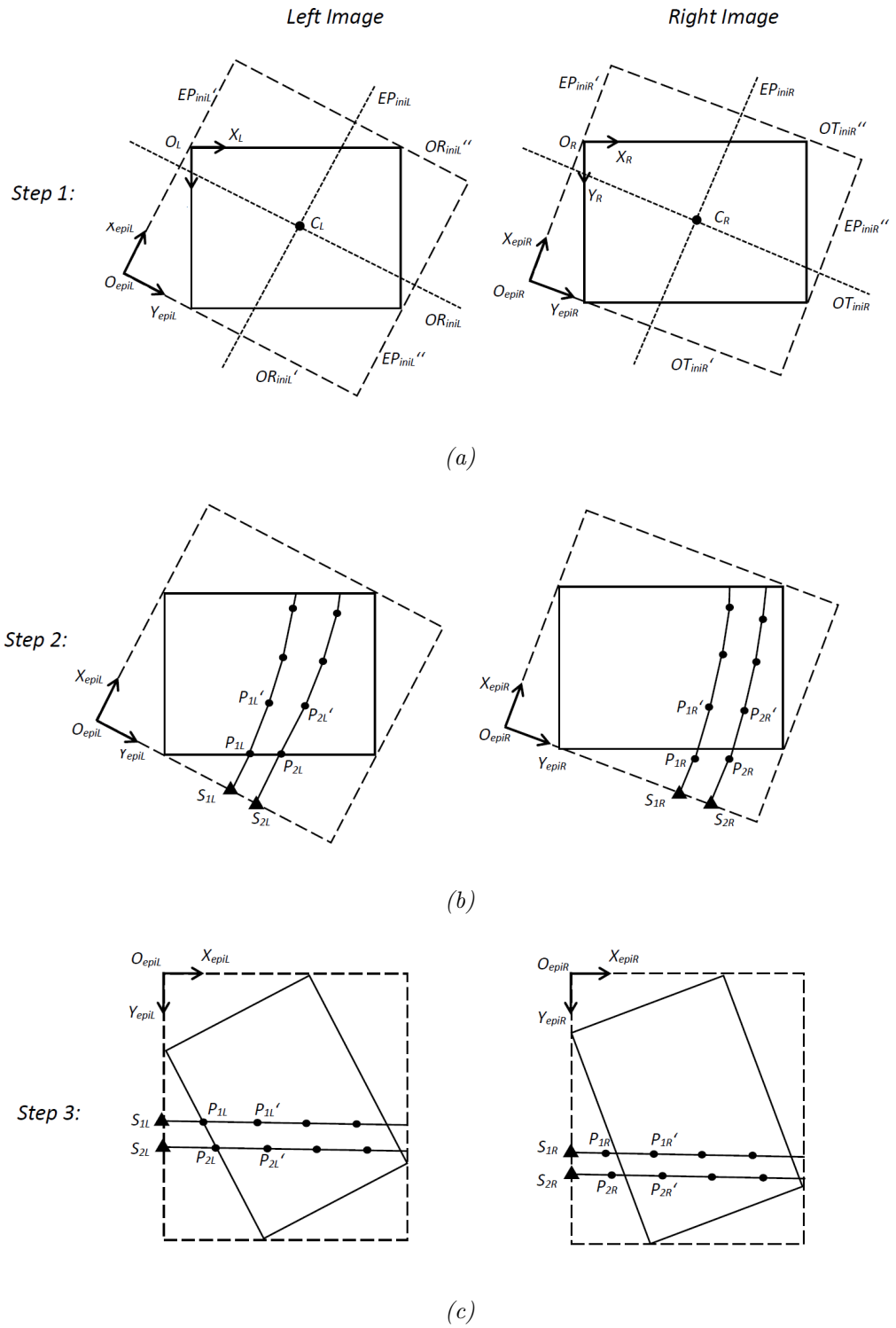


Figure 3.9: Epipolar image resampling strategy

are also set as the re-start points (P_{1R}, $P'_{1R}\dots$). The conjugate points are calculated by the RPC PTE model with a pre-defined height level. This height level is selected as the mean height of the area in our experiments. It only affects the x-parallax and would not affect the y-parallax. When the epipolar segments are completely derived from one start point S_{1L}, we move along the y-axis Y_{epiL} for one GSD and get the next start point S_{2L}. Undertaking the same procedure for S_{2L}, we approximate the next epipolar line with segments. The processing stops when the start point is out of the epipolar image boundary.

The last step is resampling the epipolar stereo images (Figure 3.9c). The former steps have built an epipolar geometry of satellite imagery and approximate every epipolar line pairs by several segments. The epipolar segment pairs generated in Figure 3.9b are aligned to the same row along the direction of the x-axis of the epipolar image coordinate system. Then we resample the rows of the epipolar image from top to bottom. For each row, we resample it from left to right. The resample distance equals to the GSD of the original image. The bi-linear interpolation is applied for the image resampling procedure.

During our rectification procedure, several critical parameters, that present the transfer relation between the original pixel coordinates and the epipolar pixel coordinates, need to be stored. As mentioned above, the epipolar curve is approximated by several epipolar segments. Because the first segment ends when it intersects the border of the original image, the length of the first segment is len_{seg0}. The lengths of the first epipolar segment for every epipolar curves are stored in a matrix $\mathbf{Len_{seg0}}$. The size of $\mathbf{Len_{seg0}}$ is $epi_{num} \times 1$. epi_{num} is the number of the rows in the epipolar image (the number of the epipolar curves). Except the first epipolar segment, the rest segments will be derived till they reach the pre-defined length l_{seg}. The formula of each epipolar segment can be simply presented as $l = bs + a$. s and l denote the original horizontal and vertical image coordinates. b and a are the linear parameters of the epipolar segment. For each epipolar segment, the original horizontal image coordinates of the start or re-start points s_{start} are marked. We also record the parameters b and a. They are stored in three $epi_{num} \times seg_{num}$ matrices $\mathbf{S_{start}}$, $\mathbf{B}$ and $\mathbf{A}$. seg_{num} is the maximum number of the epipolar segments. The pre-defined epipolar segment length len_{seg}, the first segment matrix $\mathbf{Len_{seg0}}$, the start/re-strat horizontal coordinate matrix $\mathbf{S_{start}}$, the epipolar segment linear parameter matrix $\mathbf{B}$ and $\mathbf{A}$ will be applied in the subsequent triangulation procedure.

3.4 Point Cloud and DSM Generation

In the former steps, we employ the image rectification to force the corresponding pixels sharing the same row index. The epipolar stereo pairs will be matched pixel-wise separately. As a core algorithm of the software SURE, the tube-based SGM (tSGM) algorithm [Rothermel et al., 2012] has been proved to have impressive performance on several benchmark datasets for airborne MVS imagery [Haala, 2013]. We firstly introduce the tSGM algorithm into the application for dense matching of satellite MVS stereo imagery. The implemented library libTsgm for tSGM algorithm is authorized by nFrames GmbH to be used in our work. A brief introduction about tSGM algorithm is presented in section 3.4.1. When the correspondences are acquired from the dense image matching, it is necessary to transfer the corresponding points from the epipolar images to the original images. This is, because the RPCs are only valid for the original images. With the corresponding points on the original images, we calculate the 3D point cloud via forward intersection and fuse them to the final DSM. The methods related to the triangulation and fusion are presented in section 3.4.2.

3.4.1 tSGM Algorithm

Let I_b to be the base image and I_m be the matching image. Since we already have the epipolar images as the inputs of dense image matching, for a pixel on the base image $\mathbf{p}_b(x, y)$, its corresponding pixel on the matching image will be $\mathbf{p}_m(x+d, y)$. Here d presents the disparity of the corresponding pixel pair. The tSGM algorithm is based on the SGM method proposed by [Hirschmüller, 2008]. As known, the key idea of the SGM algorithm is minimizing the global cost function to estimate the disparities. The disparity map $\mathbf{D}(\mathbf{p}_b)$ stores the estimated disparities of each corresponding pixel on the base image. The general formula of the global cost E is

$$E(\mathbf{D}) = \sum_{\mathbf{p}_b}(C(\mathbf{p}_b, \mathbf{D}(\mathbf{p}_b)+\sum_{\mathbf{p}_N} P_1 T[|\mathbf{D}(\mathbf{p}_b)-\mathbf{D}(\mathbf{p}_N)| = 1]+\sum_{\mathbf{p}_N} P_2 T[|\mathbf{D}(\mathbf{p}_b)-\mathbf{D}(\mathbf{p}_N)| > 1]). \quad (3.10)$$

The first part of the equation is the overall pixel-wise matching cost of the whole image. The second part are the penalties for the pixels that have different disparities to their neighbour pixels. $\mathbf{p}_N$ are the neighbour pixels of pixel $\mathbf{p}_b$ in the base image. $C(\mathbf{p}_b, \mathbf{D}(\mathbf{p}_b)$ represents the pixel-wise matching cost. T is an operator that equals to one if the subsequent condition is true. Otherwise, T equals to zero. $P1$ is the penalty parameter for the small disparity differences to the neighbour pixels and $P2$ is the penalty parameters for the large differences. The SGM method estimates the MI as the matching cost in a hierarchical way. The result of the higher pyramid level (low resolution) is used to refine the matching cost of the lower pyramid level (high resolution). The matching cost along i image paths are accumulated sequentially (typically 8 paths). For pixel $\mathbf{p}$, the cost accumulated along a path direction $\mathbf{r}$ at disparity d refers to $L_{\mathbf{r}}(\mathbf{p}_b, d)$:

$$\begin{aligned} L_{\mathbf{r}}(\mathbf{p}_b, d) =& C(\mathbf{p}_b, d) + min(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d), \\ & L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d-1) + P_1, \\ & L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d+1) + P_1, \\ & min_i(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, i) + P_2)) \\ & - min_k(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, k)). \end{aligned} \quad (3.11)$$

The calculation of the cost on one direction starts from $C(\mathbf{p}_b, d)$, which is the matching cost with disparity d along path r. $\mathbf{p}_b - \mathbf{r}$ represent the previous pixel along the path r. $\min_k(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, k))$ is the lowest path cost of the previous pixel from the whole term. It is a constant value for a pixel p and it guarantees that the cost would not increase permanently. According to 3.11, the upper border of the cost $L_{\mathbf{r}}(\mathbf{p}_b, d)$ would be $C_{max} + P_2$. The sum of all directions is

$$S(\mathbf{p}_b, d) = \sum_{\mathbf{r}} L_{\mathbf{r}}(\mathbf{p}_b, d). \quad (3.12)$$

$S(\mathbf{p}_b, d)$ contains both the matching cost and the potential disparity set of each pixel. The final disparity d of pixel $\mathbf{p}_b$ corresponding to the minimum cost $\min_d(S(\mathbf{p}_b, d))$ is selected and stored in $\mathbf{D}$.

Different from the classic SGM algorithm, tSGM algorithm applies the 9×7 Census cost to calculate the matching cost instead of the MI, because of higher robustness. In order to accelerate the processing speed and reduce the memory requirement, tSGM applies a dynamic disparity search range. The integer range of the potential disparity is set as $[d_{min}, d_{max}]$. The tSGM algorithm will

adapt the disparity range hierarchically. Starting from pyramid level l, the estimated disparity map of pixel $\mathbf{p}_b(x, y)$ in this level is $\mathbf{D}^l_{\mathbf{p}_b}$. For the first pyramid level, no prior knowledge can be used, so the initial disparity range is defined as $d_{min} = -x$ and $d_{max} = n_c - x$. Here n_c represents the number of the image columns. To re-define the disparity search range, successfully matched pixels and unsuccessfully matched pixels are taken into consideration separately. For the matched pixels, d_{min} and d_{max} are determined in a small 7×7 window, and they are stored into two images $\mathbf{R}^l_{min}$ and $\mathbf{R}^l_{max}$. As to the unsuccessfully matched pixels, d_{min} and d_{max} are searched in a large searching window like 31×31. In the searching window, if at least three successful matched pixels are found, the disparity $d_{\mathbf{p}_b}$ is set as the median value of all disparities contained in the window. If there are less than three matched pixels in the search window, $d_{\mathbf{p}_b}$ is set as the mean disparity of the whole disparity image. d_{min} and d_{min} are signed to half of the initial range and also stored in $\mathbf{R}^l_{min}$ and $\mathbf{R}^l_{max}$. The $\mathbf{D}^l_{\mathbf{p}_b}$, $\mathbf{R}^l_{min}$ and $\mathbf{R}^l_{max}$ are upscaled to the next pyramid level $l-1$, and the disparity search range of pixel $\mathbf{p}_b$ in this level is $[2 \times (x + d_{\mathbf{p}_b} - d_{min}), 2 \times (x + d_{\mathbf{p}_b} + d_{max})]$. Iteratively, the disparity range of the higher pyramid level is introduced as the prior information to reduce the range of the lower level until $l = 0$. Because the cost strings of neighboring pixels may be only overlapped slightly, tSGM algorithm introduce the bottom or top elements of the cost strings $d_{max}(\mathbf{p}_b - \mathbf{r})$ and $d_{min}(\mathbf{p}_b - \mathbf{r})$. The enhanced formula of 3.11 is

$$
\begin{aligned}
&\textbf{if: } d > d_{max}(\mathbf{p}_b - \mathbf{r}) \\
&\quad L_{\mathbf{r}}(\mathbf{p}_b, d) = C(\mathbf{p}_b, d) + L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d_{max}(\mathbf{p}_b - \mathbf{r})) + P_2 - \min_k(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, k)) \\
&\textbf{elseif: } d < d_{min}(\mathbf{p}_b - \mathbf{r}) \\
&\quad L_{\mathbf{r}}(\mathbf{p}_b, d) = C(\mathbf{p}_b, d) + L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d_{min}(\mathbf{p}_b - \mathbf{r})) + P_2 - \min_k(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, k)) \\
&\textbf{else:} \\
&\quad L_{\mathbf{r}}(\mathbf{p}_b, d) = C(\mathbf{p}_b, d) + min(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d), L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d-1) + P_1, \\
&\quad L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, d+1) + P_1, \min_i(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, i) + P_2)) - \min_k(L_{\mathbf{r}}(\mathbf{p}_b - \mathbf{r}, k)).
\end{aligned}
\tag{3.13}
$$

Moreover, the penalty parameter P_2 in the tSGM algorithm is based on the canny edge image instead of the base image. If edges are detected, a lower smoothing parameter $P_2 = P_{21}$ is utilized. Otherwise, a higher smoothing parameter $P_2 = P_{21} + P_{22}$ is employed. The tSGM algorithm greatly reduces computing time and optimizes memory efficiency.

3.4.2 Triangulation and DSM Fusion

The epipolar stereo images are applied as the input to the tSGM algorithm. The product of the dense image matching process is the disparity map. The corresponding pixel $\mathbf{p}_m(x + d, y)$ in the matching image can be easily calculated from the pixel $\mathbf{p}_b(x, y)$ in the base image and disparity map D. Note that the corresponding pixels $\mathbf{p}_b$ and $\mathbf{p}_m$ are located on the epipolar image pair but not the original stereo image pair. While the RPCs are related to the original images, we need to transfer the coordinates back from the epipolar image coordinate system to the original image coordinate system. As described in section 3.3.2, the start points of each epipolar segments, the linear parameters of the epipolar segments and the length of each epipolar segments have been recorded in the image rectification procedure. According to these parameters and our epipolar resampling strategy, the corresponding pixels on the original images can be derived. Pixel $p_{epi}(x_{epi}, y_{epi})$ locates on the ith

epipolar segments of the y_{epi}th epipolar curve in the epipolar image, the start point of the segment is $\mathbf{S_{start}}(y_{epi}, i)$. The linear parameters of the ith epipolar segment are $\mathbf{B}(y_{epi}, i)$ and $\mathbf{A}(y_{epi}, i)$. The length of the first segment of the y_{epi}th epipolar curve is $\mathbf{Len}_{seg0}(y_{epi})$. Our predefined epipolar segment length is len_{seg}. The transfer equations from the epipolar image space to the original image space are derived as:

$$\begin{cases} s = \mathbf{S_{start}}(i) + \sqrt{\dfrac{1+\mathbf{B}(i)^2}{\mathbf{B}(i)^2}}(x_{epi} - \mathbf{Len}_{seg0}(y_{epi}) - len_{seg} \times (i-1)) \\ l = \mathbf{A}(i) \times s + \mathbf{B}(i) \end{cases} \tag{3.14}$$

where s and l are the horizontal coordinate and vertical coordinates of the corresponding pixel in the original image space.

Following 3.14, all the corresponding epipolar image points are transferred to the corresponding original image points. With these correspondences, the RPC model is employed to project the image points to the object space and the object coordinates are calculated via the forward intersection. We apply a similar observation equation like 3.4 and 3.5 for the forward intersection. The observations are the image coordinates s and l of the corresponding points and our shift compensation parameters. The unknowns are the object coordinates X of the 3D point cloud. F_0 is the misclosures for the image coordinates of the corresponding pixels. Unlike 3.6, here dF is derived as

$$d\mathbf{F} = \begin{bmatrix} \frac{\partial \mathbf{F}_s}{\partial x} \\ \frac{\partial \mathbf{F}_l}{\partial x} \end{bmatrix} d\mathbf{X} = \mathbf{A} d\mathbf{X} \tag{3.15}$$

where dX is the correction for the object coordinate. Therefore, the equation of the forward intersection is

$$\begin{bmatrix} \mathbf{A} \\ 1 \end{bmatrix} d\mathbf{X} + \epsilon = \begin{bmatrix} -\mathbf{F}_0 \\ 0 \end{bmatrix}. \tag{3.16}$$

The RPC model is non-linear, so that we solve the object coordinates by least squares estimation iteratively until they reach convergence. We set 0.001m as the threshold value for the unknown correction. The calculated point clouds are in a geographic coordinate system, which is not suitable to visualization. Hence, the object coordinates of the point cloud are projected to some Cartesian coordinate system, for example, the UTM coordinate system.

The point clouds of each stereo pair are computed separately. As explained in section 3.2.2, the point clouds have already been aligned to the same virtual surface with the refined RPCs, so no additional registration procedure is needed. In order to generate the final DSM, the point clouds are first projected into a regularly spaced and discretized grid in the UTM coordinate system. In our implementation, a simple median filter is applied for the DSM fusion. The median value of the height of each grid cell is computed as the final height value of the fused DSM. The Inverse Distance Weighted (IDW) interpolation method is applied if no points are projected to the cell. The search radius of the interpolation is set as 30 pixels. The power parameter of the IDW method is three in our implementation.

3.5 Mesh Refinement

Our 3D reconstruction pipeline generates the point clouds and the DSMs based on the binocular dense image matching algorithm and the following fusion procedure. Like other SGM-like algorithms, the result of the tSGM algorithm also suffers from the fronto-parallel bias and the estimation of the sub-pixel disparities [Rothermel et al., 2012]. Our fusion algorithm applies the simple median filter, which might be sub-optimal. Moreover, the fused DSMs are only 2.5D representations of the surface and can not recover the 3D structures. We propose a novel mesh refinement algorithm and extend our pipeline to reconstruct true 3D models from the satellite MVS imagery. The algorithm is inspired by the work of [Delaunoy et al., 2008] and [Vu et al., 2012], and it handles the occlusion and visibility well. Cooperating with the colleagues from ETH Zurich, the mesh refinement algorithm is designed and implemented with C++ codes. The algorithms described in this section is based on our latest research [Rothermel et al., 2020].

The basic idea of the photometric mesh refinement is to position the vertices of a mesh so that the pixel intensities of a set of images projected via the mesh to surrounding views resemble the actual, observed intensities. The refinement starts from a topological correct but low-detail surface mesh. For each vertex a gradient can be computed defining the direction of vertex translation, which increases the similarity of real and projected views, or in other words, decreases the reprojection error. We minimize the energy composing of the reprojection error of all image views and the surface smoothness by using the variational optimization, so that the position of vertices are refined. In section 3.5.1, the computation of the gradients for a traditional frame camera is reviewed. In order to adapt the energy function to the satellite MVS images, we apply the RPC model. We show the adaption details for the satellite sensor model in section 3.5.2. At last, section 3.5.2 also shows some implementation details.

3.5.1 Gradient for Pinhole Cameras

The final aim of the mesh refinement algorithm is to minimize the photo-consistency energy of the mesh model, so that the mesh surface is closer to the actual surface. In order to understand the adaptions necessary for satellite-imagery-based mesh refinement, in this section we review the computation of the photo-consistency energy for pinhole models.

As shown in Figure 3.10, I_i and I_j are two images taken from the camera position i and j. $\mathcal{S}$ denotes the (infinite) set of admissible 2D surface manifolds in $\mathbb{R}^3$. Let $\Pi_j, \Pi_i : \mathbb{R}^3 \to \mathbb{R}^2$ be the projection mapping object coordinates to image coordinates of I_j and I_i respectively. $\Pi_{i,\mathcal{S}}^{-1}, \Pi_{j,\mathcal{S}}^{-1} : \mathbb{R}^2 \to \mathbb{R}^3$ denote the re-projection from image I_i and I_j to the surface $\mathcal{S}$. We can reproject image I_j to the mesh surface and then project to the space of image I_i. The transfer function is given by $I_j \circ \Pi j \circ \Pi_{i,\mathcal{S}}^{-1}$. Therefore, the photo-consistency energy is formulated as

$$E(\mathcal{S}) = \sum_i \sum_{j \neq i} E_{ij}(\mathcal{S}). \tag{3.17}$$

In addition the pair-wise measure of the photo-consistency energy $E_{ij}(\mathcal{S})$ of projected image $I_j : \Omega_j \to \mathbb{R}^1$ to image $I_i : \Omega_j \to \mathbb{R}^1$ via the surface $\mathcal{S}$ is presented as

$$E_{ij}(\mathcal{S}) = \int_{\Omega_i \cap \Omega_j} M(I_i, I_j \circ \Pi_j \circ \Pi_{i,\mathcal{S}}^{-1})(\mathbf{x}_i) d\mathbf{x}_i. \tag{3.18}$$

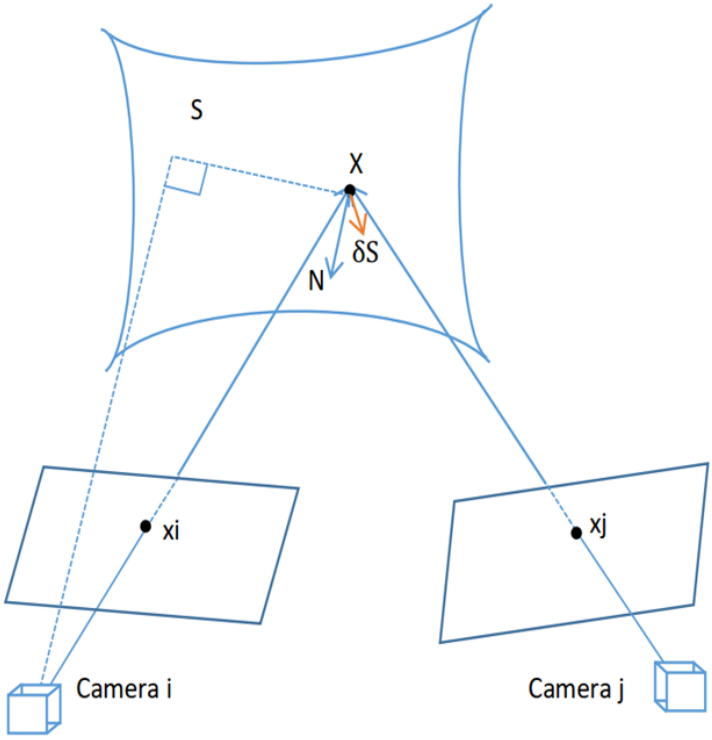


Figure 3.10: Reprojection of pinhole camera

where M is a function measuring photo consistency. In our algorithm, we seek to minimize the negative zero-normalised cross-correlation $M(\mathbf{a}, \mathbf{b}) = -ZNCC(\mathbf{a}, \mathbf{b})$. $\Omega_i \cap \Omega_j$ denotes the overlapping surface area on $\mathcal{S}$ of the observation of image I_i and I_j. We compute the variation of the pair-wise energy E_{ij} with respect to an infinitesimal vector displacement $\delta\mathcal{S}$ of the surface $\mathcal{S}$. According to the chain rule of function composition, the variation is presented as:

$$\left.\frac{\partial E_{ij}(\mathcal{S}+\epsilon\delta\mathcal{S})}{\partial\epsilon}\right|_{\epsilon} = \int_{\Omega_i\cap\Omega_j} \partial_2 M(\mathbf{x}_i) DI_j(\mathbf{x}_j) D\Pi_j(\mathbf{X}) \left.\frac{\partial \Pi^{-1}_{i,\mathcal{S}+\epsilon\delta\mathcal{S}}(\mathbf{x}_i)}{\partial\epsilon}\right|_{\epsilon=0} d\mathbf{x}_i. \tag{3.19}$$

Thereby $\mathbf{x}_i$ and $\mathbf{x}_j$ are the pixels' coordinates in image I_i and I_j. $\mathbf{X}$ is the projected object coordinates on the mesh surface $\mathcal{S}$. Here $\partial_2 M(\mathbf{x}_i)$ denotes the derivation of the similarity measure with respect to the second argument I_j. DI_j denotes the gradient image of I_j and $D\Pi_j$ denotes the partial derivatives of object-to-image space mapping with respect to an object point on the surface. Let vector $\mathbf{d}$ be the be the ray from the projection center of view i passing through the pixel coordinates $\mathbf{x}_i$ to the mesh surface. The term

$$\left.\frac{\partial \Pi^{-1}_{i,\mathcal{S}+\epsilon\delta\mathcal{S}}(\mathbf{x}_i)}{\partial\epsilon}\right|_{\epsilon=0} \tag{3.20}$$

represents the change along $\mathbf{d}$ when moving the surface by $\delta\mathcal{S}$.The relation between δS and the change of the ray light vector is displayed in Figure 3.11a.

In Figure 3.11a, $\mathbf{n}$ is the outward surface normal vector at the intersection point of image ray and the surface $\mathbf{X}$. We have α as the angle between $\mathbf{d}$ and $\mathbf{n}$ and β as the angle between $\delta\mathcal{S}$ and

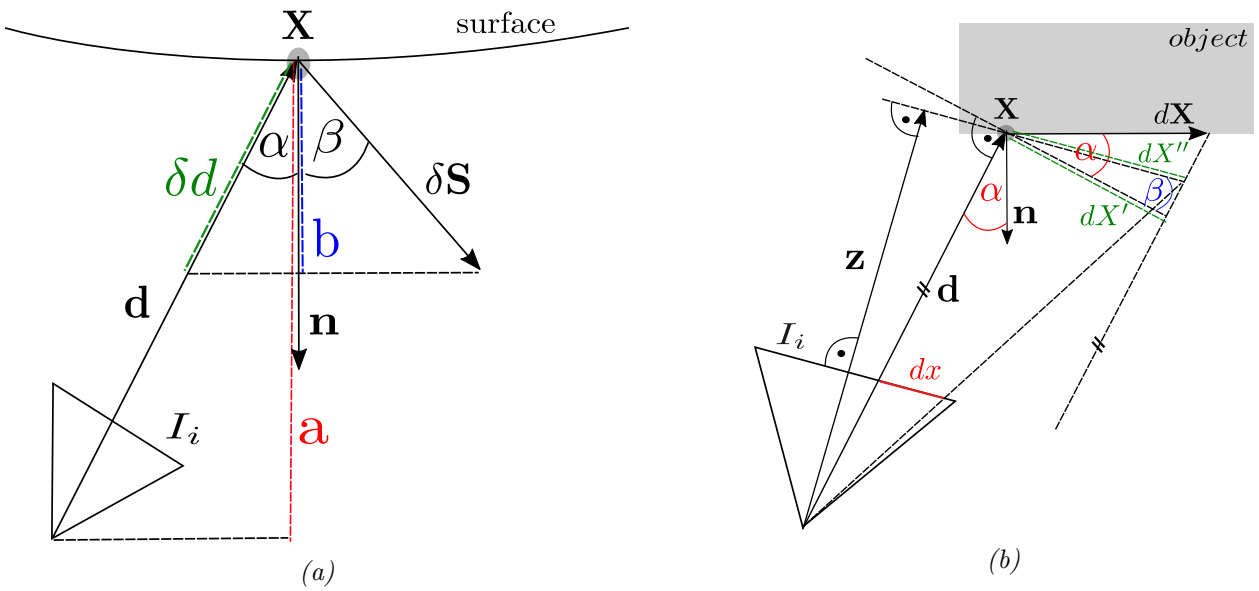


Figure 3.11: a. The relation of the variation of the ray $\mathbf{d}$ and the surface moving $\delta\mathbf{S}$; b. Relation of displacement within the surface and displacement in the image

$\mathbf{n}$. With $d = |\mathbf{d}|$ trigonometry yields $-\mathbf{n}^T\mathbf{d} = |\mathbf{d}|\cos\alpha := a$ and $\mathbf{n}^T\delta\mathcal{S} = |\delta\mathcal{S}|\cos\beta := b$. According to the interception theorem, we obtain $\delta d = |\mathbf{d}|b/a$. Therefore, the variation of the ray can be rewritten as

$$\left.\frac{\partial\Pi^{-1}_{i,\mathcal{S}+\epsilon\delta\mathcal{S}}(\mathbf{x}_i)}{\partial\epsilon}\right|_{\epsilon=0} = \delta d\frac{\mathbf{d}}{|\mathbf{d}|} = \frac{\mathbf{n}^T\delta S(\mathbf{X})}{\mathbf{n}^T\mathbf{d}}\mathbf{d}. \tag{3.21}$$

Figure 3.11b presents the relation of the displacement within the surface and the displacement in the image. Here, dx is the image displacement and dX is the object displacement. dX' is the displacement along the direction perpendicular to the changed ray, and dX'' is the displacement along the direction parallel to the image plane. z represents the z-component of a surface point in the camera coordinate system of view i. With $\cos\alpha = dX'/dX = -\mathbf{n}^T\mathbf{d}/d$ and $\cos\beta = dX'/dX'' = z/d$ follows $dX'' = -dX\mathbf{n}^T\mathbf{d}/z$. The interception theorem yields the change of image space coordinates caused by an infinitesimal displacement of object coordinates as

$$d\mathbf{x}_i = -\frac{\mathbf{n}^T\mathbf{d}d\mathbf{X}}{z^2}. \tag{3.22}$$

By substitution of 3.21 in 3.19 and using 3.22 to change the integration domain from image space to the surface domain we obtain

$$\left.\frac{\partial E_{ij}(\mathcal{S}+\epsilon\delta\mathcal{S})}{\partial\epsilon}\right|_{\epsilon=0} = -\int_{\Omega_i\cap\Omega_j}\partial_2 M(\mathbf{x}_i)DI_j(\mathbf{x}_j)D\Pi_j(\mathbf{X})\frac{\mathbf{d}}{z^2}\mathbf{n}^T\delta\mathcal{S}(\mathbf{X})d\mathbf{X}. \tag{3.23}$$

From the work of [Delaunoy and Prados, 2011] and [Solem and Overgaard, 2005], we know that

for the variation of the energy δE attached to the surface $\mathcal{S}$ and a variation of the surface $\delta\mathcal{S}$ the gradient vector field ∇E fulfills

$$\delta E = \left.\frac{\partial E_{ij}(\mathcal{S}+\epsilon\delta\mathcal{S})}{\partial\epsilon}\right|_{\epsilon=0} = -\int_{\Omega_i\cap\Omega_j} \nabla E_{ij}(S)(\mathbf{X})\delta\mathcal{S}(\mathbf{X})d\mathbf{X}, \tag{3.24}$$

Consequently, by comparison of 3.24 and 3.23 the gradient of the matching function is given by

$$\nabla E_{ij}(S)(\mathbf{X}) = -\delta_{\Omega_i\cap\Omega_j}\left[\partial_2 M(\mathbf{x}_i)DI_j(\mathbf{x}_j)D\Pi_j(\mathbf{X})\frac{\mathbf{d}}{z^2}\right]\mathbf{n}. \tag{3.25}$$

Thereby $\delta_{\Omega_i\cap\Omega_j}$ is the Kronecker Symbol. It accounts for the visibility, evaluating to one if the surface is visible in both views I_i, I_j and to zero otherwise. Note that this continuous formulation can be directly used to compute gradients of discrete surface. In practice a gradient for every vertex is computed by weighted integration within its one ring neighborhood. In the case of triangular meshes, the barycentric coordinates corresponding to the vertices $\phi(X)$ are employed. The discrete gradient of the energy from the continuous one can be presented as

$$\frac{dE(S)}{d\mathbf{X}} = \int_{\Omega_i\cap\Omega_j}\phi(X)\nabla E(S)(\mathbf{X})d\mathbf{X} = \phi(X)\sum_i\sum_{j\neq i}\int_{\Omega_i\cap\Omega_j}\nabla E_{ij}(S)(\mathbf{X})d\mathbf{X}. \tag{3.26}$$

Introduce 3.25 to 3.26, the final discrete gradient is

$$\begin{aligned}\frac{dE(S)}{d\mathbf{X}} &= -\phi(X)\sum_i\sum_{j\neq i}\int_{\Omega_i\cap\Omega_j}\delta_{\Omega_i\cap\Omega_j}\left[\partial_2 M(\mathbf{x}_i)DI_j(\mathbf{x}_j)D\Pi_j(\mathbf{X})\frac{\mathbf{d}}{z^2}\right]\mathbf{n}d\mathbf{X}\\ &= -\phi(X)\sum_i\sum_{j\neq i}\int_{\Omega_i\cap\Omega_j}\delta_{\Omega_i\cap\Omega_j}\left[\partial_2 M(\mathbf{x}_i)DI_j(\mathbf{x}_j)D\Pi_j(\mathbf{X})\frac{\mathbf{d}}{z^2}\right]\mathbf{n}(-\frac{z^2}{\mathbf{n}^T\mathbf{d}})d\mathbf{x_i}\\ &= \phi(X)\sum_i\sum_{j\neq i}\int_{\Omega_i\cap\Omega_j}\delta_{\Omega_i\cap\Omega_j}\left[\partial_2 M(\mathbf{x}_i)DI_j(\mathbf{x}_j)D\Pi_j(\mathbf{X})\mathbf{d}\right]\frac{\mathbf{n}}{\mathbf{n}^T\mathbf{d}}d\mathbf{x_i}.\end{aligned} \tag{3.27}$$

The gradient descent flow is given by $-\frac{dE(S)}{d\mathbf{X}}$. According to 3.27, the gradient of the vertex is computed by summing the weighted contribution of all the pixels, which are located in the projected triangles containing this vertex in all image pairs.

3.5.2 Photometric Mesh Refinement for Satellite Imagery

Since the camera model of the satellite sensors is different from the pinhole camera, the mesh refinement algorithm need to be adapted for the RPC model. The partial derivatives $D\Pi_j$ are modified with the relation between the object and image space built by RPC projection. Furthermore, the RPC model for satellites relates Cartesian image coordinates to object coordinates in a polar coordinate system. This is in contrast to the pinhole model where object coordinates are defined in a Cartesian coordinate system. Finally, the gradient computation for the pinhole model depends on entities not explicitly modeled for satellite imagery, namely projection center and depth. We also show how to circumvent this problem in this section.

As aforementioned, the RPC model can be presented as 3.1. For $\mathbf{Num}_s$, $\mathbf{Den}_s$, $\mathbf{Num}_l$ and $\mathbf{Den}_l$, each of them holds 20 coefficients. The third order polynomial of the normalized geographic

coordinates is $\mathbf{P}(B_n, L_n, H_n)$, which are calculated from the offset and scale parameters provided by the RPC files. s_{offset}, l_{offset}, s_{scale} and l_{scale} are the offset and scale parameters of the image coordinates. The offset and scale parameters of the object coordinates are $[B_{offset}, L_{offset}, H_{offset}]$ and $[B_{scale}, L_{scale}, H_{scale}]$. The RPCs are already refined by our RPC compensation algorithm. Then the 2×3 Jacobian matrix $D\Pi(\mathbf{x})$ w.r.t geographic coordinates is given by

$$\begin{aligned} D\Pi_j &= \begin{bmatrix} \frac{\partial s}{\partial B_n}, \frac{\partial s}{\partial L_n}, \frac{\partial s}{\partial H_n} \\ \frac{\partial l}{\partial B_n}, \frac{\partial l}{\partial L_n}, \frac{\partial l}{\partial H_n} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\mathbf{Num}_s^T(\mathbf{Den}_s^T\mathbf{P}) - \mathbf{Den}_s^T(\mathbf{Num}_s^T\mathbf{P})}{(\mathbf{Den}_s^T\mathbf{P})^2} \left[\frac{\partial \mathbf{P}}{\partial B_n}, \frac{\partial \mathbf{P}}{\partial L_n}, \frac{\partial \mathbf{P}}{\partial H_n}\right] \begin{bmatrix} \frac{1}{L_{scale}}, 0, 0 \\ 0, \frac{1}{B_{scale}}, 0 \\ 0, 0, \frac{1}{H_{scale}} \end{bmatrix} s_{scale} \\ \frac{\mathbf{Num}_l^T(\mathbf{Den}_l^T\mathbf{P}) - \mathbf{Den}_l^T(\mathbf{Num}_l^T\mathbf{P})}{(\mathbf{Den}_l^T\mathbf{P})^2} \left[\frac{\partial \mathbf{P}}{\partial B_n}, \frac{\partial \mathbf{P}}{\partial L_n}, \frac{\partial \mathbf{P}}{\partial H_n}\right] \begin{bmatrix} \frac{1}{L_{scale}}, 0, 0 \\ 0, \frac{1}{B_{scale}}, 0 \\ 0, 0, \frac{1}{H_{scale}} \end{bmatrix} l_{scale} \end{bmatrix}. \end{aligned} \tag{3.28}$$

Where

$$\left[\frac{\partial \mathbf{P}}{\partial B_n}, \frac{\partial \mathbf{P}}{\partial L_n}, \frac{\partial \mathbf{P}}{\partial H_n}\right] = \begin{bmatrix} 0,0,1,0,L_n,0,H_n,0,2B_n,0,L_nH_N,0,2L_nB_n,0,L_n^2,3B_n^2,H_n^2,0,2B_nH_n,0 \\ 0,1,0,0,B_n,H_n,0,2L_n,0,0,B_nH_n,3L_n^2,B_n^2,H_n^2,2L_nB_n,0,0,2L_nH_n,0,0 \\ 0,0,0,1,0,L_n,B_n,0,0,2H_n,B_nL_n,0,0,2L_nH_n,0,0,2B_nH_n,L_n^2,B_n^2,3H_n^2 \end{bmatrix}. \tag{3.29}$$

The RPC model relates image coordinates in a Cartesian coordinate system (COS) and geographic coordinates in a polar COS. However, the whole derivation of the discrete gradient in the former section requires a Cartesian coordinate system. Usually, the geographic coordinates are transferred to the Cartesian COS by map projection such as UTM projection. For mesh refinement, the non-linear map projection procedure has unnecessarily high-cost for computation. In order to circumvent this problem, one possibility is to transform the geographic coordinates $[B; L; H]$ into a local "Quasi-Cartesian" coordinate system. To achieve the "Quasi-Cartesian" COS, the latitude B and the longitude L are scaled to the metric unit of the height component H. We set the center point of the reconstruction area as $\mathbf{x}_{geo} = [B_c, L_c, H_c]^T$. Its correspondence in the UTM coordinate system is $\mathbf{x}_{utm} = [X_c, Y_c, H_c]^T$. We move the next point by one unit in the UTM system, which is presented like $[X_c + 1.0, Y_c + 1.0, H_c]^T$. This point is corresponding to $[B_{c+1}, L_{c+1}, H_c]^T$ in the geographic coordinate system. Then we can transform the geographic coordinates to our own defined local quasi-Cartesian COS like

$$\mathbf{x}_{loc} = f(B, L, H) = \left[\frac{B}{B_c - B_{c+1}}, \frac{L}{L_c - L_{c+1}}, H\right]^T - \mathbf{x}_{geo}. \tag{3.30}$$

This transformation mimics a Cartesian coordinate system only locally. However, the approximation is valid for a large enough area, because the exploiting parallelism in computation refinement algorithms are applied to tiles of limited extend. We apply the WorldView-3 imagery at 30cm GSD as an example to test our local coordinate system. The quality of the approximation (non-orthogonality and scale anisotropy of the coordinate axes) is given in table 3.1. In order to verify the degree of orthogonality, two orthogonal unit vectors $\mathbf{x} = \mathbf{p}_1 - \mathbf{p}_0$ and $\mathbf{y} = \mathbf{p}_2 - \mathbf{p}_0$ are defined in the UTM coordinates. The points $\mathbf{p}_0$, $\mathbf{p}_1$ and $\mathbf{p}_2$ are located in a horizontal plane at the average

terrain height of the reconstruction area. $\mathbf{p}_0$ is the origin of our quasi-Cartesian COS. $\mathbf{p}_1$ is the point moving along the UTM's x-axis with certain length from the origin. $\mathbf{p}_2$ represents the point moving along the UTM's y-axis. Here we apply the same unit of length to move for both axes. We transform the vectors $\mathbf{x}$ and $\mathbf{y}$ in UTM coordinates to the geographic coordinates and then transfer them to our local Cartesian COS. The unit vectors of local COS are $\mathbf{x}'$ and $\mathbf{y}'$. In table 3.1, we show the vector's length in UTM coordinate system in the first column. The second and third column display the length of the vector of the x and y axis in local Cartesian COS. At last we show the orthogonality check in the last column. As table 3.1 presented, the scale difference is less than 7mm in x- and y-direction when the vector length is as large as 2,000m. The scale difference between the x- and z-axis is 3.8cm, which is ca. 0.13 GSD. The angle between the vectors $\mathbf{x}'$ and $\mathbf{y}'$ is about 0.005° and thus can be assumed as quasi-orthogonal. Generally, we find that the effect of our quasi-Cartesian COS is marginal if the vector length is less than 2,000m. Therefore the local quasi-Cartesian COS is feasible for the small areas. In our pipeline, we will transform the meshes to a quasi Cartesian COS before the refinement. Using the chain rule, the Jacobian of the mapping from object to quasi Cartesian coordinates then renders as

$$D\Pi(X,Y,H) = \begin{bmatrix} \frac{1}{B_c - B_{c+1}} & \frac{1}{L_c - L_{c+1}} & 1 \\ \frac{1}{B_c - B_{c+1}} & \frac{1}{L_c - L_{c+1}} & 1 \end{bmatrix} \odot D\Pi(B,L,H). \tag{3.31}$$

Thereby $\odot$ denotes element-wise multiplication.

scale s	length $\lvert\mathbf{x}'\rvert_2$ [m]	length $\lvert\mathbf{y}'\rvert_2$ [m]	angle $\mathbf{x}'$, $\mathbf{y}'$ [deg]
100	100.002	100.002	90.000
200	200.003	200.003	90.001
500	500.009	500.008	90.001
1000	1000.018	1000.016	90.003
2000	2000.038	2000.031	90.005
5000	5000.110	5000.067	90.014

Table 3.1: Verification of local coordinate systems being close to Cartesian.

According to 3.27, a key variable to the discrete gradient computation is the vector of the ray that connects the projection center and a surface point. For satellite imagery, the projection center and the altitude is different for each scan line. Fortunately, we find that 3.27 is actually independent of the absolute length of $\mathbf{d}$, which means the vector can be re-scaled to an arbitrary length. Therefore, $\mathbf{d}$ is re-written as $\mathbf{d} = d\mathbf{n}_d$, with $\mathbf{n}_d$ the unit vector in its direction and length $d = |\mathbf{d}|$. Equation 3.27 can therefore be rewritten as

$$\frac{dE(S)}{d\mathbf{X}} = \phi(X) \sum_i \sum_{j \neq i} \int_{\Omega_i \cap \Omega_j} \delta_{\Omega_i \cap \Omega_j} \left[\partial_2 M(\mathbf{x}_i) DI_j(\mathbf{x}_j) D\Pi_j(\mathbf{X}) \mathbf{n}_d\right] \frac{\mathbf{n}}{\mathbf{n}^T \mathbf{n}_d} d\mathbf{x_i}. \tag{3.32}$$

The direction of the viewing ray $\mathbf{n}_d$ is necessary to the discrete gradient calculation. Because the RPC model is a pure mathematic model to approximate the relation between image and object space without actual physical meanings, the viewing rays of the satellite imagery are not approximated as rigorous straight lines which might prevents efficient ray casting. Usually, we can set two different elevation levels H_1 and H_2 to define the direction of the viewing ray (as Figure 3.12). In Figure 3.12, P is the perspective center, and x_i is the point in the image $\mathbf{I}$. The ray starts from P via x_i

intersects on H_1 and H_2 at points X_1 and X_2 by the RPCs. The connection of X_1 and X_2 presents the direction of the viewing ray.

To verify the performance of the ray direction approximated by the RPCs, we organize two experiments. For the first experiment, the influence of the distance from X_1 to the actual surface is verified. The distance between X_1 and the surface is varied from 1m to 1,000m. The elevation difference of X_1 and X_2 is set as 1m for every approximation. Because the RPCs are invalid if the object points are too far away from the surface, the elevation distances to the surface which are larger than 1,000m are not taken into consideration. The off-nadir angles of the viewing ray and their differences to $H_1 = 1m$ are displayed in Table 3.2. As Table 3.2 presents, the direction difference between the distance at $H_1 = 0m$ and $H_1 = 1000m$ is smaller than 0.006 °. That clarifies that the distance from H_1 to the surface has minor effect on the viewing rays direction. The ray direction is more stable when H_1 is closer to the actual surface. In the second verification, we set X_1 exactly onto the actual surface. The elevation difference of X_1 and X_2 varies from 1m to 1,000m. The difference larger than 1,000m is ignored, because the RPCs are invalid if the elevation is too large. Then, we compute the off-nadir angles of the viewing ray and their differences. The results are shown in Table 3.3. According to Table 3.3, the direction differences between the ray generated with height difference $h = 0m$ and $h = 1000m$ are smaller than 0.004°. The test has verified, that the ray direction is well approximated when the height difference of X_1 and X_2 is small. Therefore, we define a horizontal plane located at $H_1 = H_{mean}$, where H_{mean} refers to the average terrain height of the test area. The height difference between H_1 and H_2 applies 1m. Then we approximate the ray direction by H_1, H_2 and RPCs. In this way, we can assume that the curvature of the RPC approximated viewing rays is low enough to represent them by straight rays. Note that all object coordinates are mapped in our pre-defined quasi Cartesian COS.

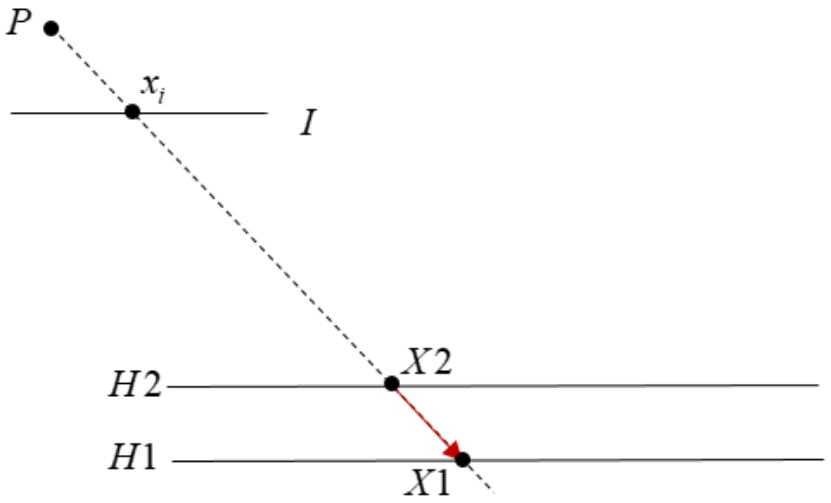


Figure 3.12: Viewing ray of satellite data

The computation of the derivative of the similarity measure $\partial_2 M$ also requires mapping the image I_j into the view i via the surface $\mathcal{S}$, which involves ray casting. As explained above, the ray direction is approximated by RPCs. Another critical requirement of ray tracing is the perspective center of the image. As known, no unique perspective center exists in satellite imagery. Here

H_1[m]	1	2	5	10	20	50	100	300	500	1000
ϕ [°]	30.131	30.131	30.131	30.131	30.131	30.131	30.130	30.129	30.128	30.126
$\Delta\phi$ [°]	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.002	0.003	0.005

Table 3.2: Verification of the influence of distance to surface on ray direction.

ΔH[m]	1	5	10	50	100	300	500	1000
ϕ [°]	30.131	30.131	30.131	30.131	30.131	30.130	30.129	30.128
$\Delta\phi$ [°]	0.000	0.000	0.000	0.000	0.000	0.001	0.002	0.003

Table 3.3: Verification of the influence of distance between H_1 and $H2$ on ray direction.

we define a virtual perspective center for each image pixel. We apply the maximum height H_{max} provided by the RPC files to establish a horizontal plane in our local quasi-Cartesian COS. For each image pixel, the intersection point projected by the RPCs on this plane is defined as its virtual perspective center. The RPC approximated viewing ray directions and the related virtual perspective centers are stored in matrices for the implementation of the ray casting.

The overall proceeding for the mesh refinement pipeline is iterative. For one iteration, we proceed as follows: First, the input meshes are transformed to quasi-Cartesian coordinate system. In the stereo case, we have images I_i and I_j. Then the image intensities of I_j^v are projected onto the mesh surface and back into I_i via the RPC projection. The similarity (NCC in our implementation) between the projected image and original image as well as its derivative are densely computed. For each pixel in image I_i, the ray direction $\mathbf{n}_d$ and the virtual perspective center are derived. Per-vertex discrete gradients are calculated by integration of 3.32 over all faces in a one ring neighborhood. The remaining components needed for 3.32 are obtained by following the adaption, which we explained before. For each vertex of the input mesh, the resulting discrete gradients are summed over all stereo models. Then the gradients are scaled with a step size to obtain a field of vertex displacements. In addition, the displacement vectors corresponding to the thin-plate energy [Kobbelt et al., 1998] are added to the photometric gradients for regularizing smoothness of the surface [Vu et al., 2012]. The final displacement field is applied to the mesh vertices to shape an updated surface, which then serves as input to the next iteration. Formally, the overall energy we minimize

$$E_{all}(\mathcal{S}) = \alpha E(\mathcal{S}) + \beta \int_{\mathcal{S}} (\kappa_1 + \kappa_2) d\mathcal{S}, \tag{3.33}$$

where κ1,κ2 denote the principal curvatures. The weight α balances the photometric term and the smoothness. Homogenization of smoothness and photometric energies is implemented by an additional parameter $\beta = \frac{1}{gsd^2}$. This accounts for different scales across datasets and mesh resolutions. In all experiments we run 20 iterations of gradient descent, after which the energy barely decreases any more. More advanced stopping criteria are of course possible, which can be implemented in the future's work. The weight factor and the step size for gradient decent were derived by a grid search.

We observed convergence problems for input meshes with vertices located too far from the correct surface. [Li et al., 2016] point out the coarse-to-fine scheme can improve the convergence. We design a comparison experiments to verify if hierarchical processing scheme is needed. We select an example area from the WorldView-3 imagery covering Downtown of Jacksonville, Florida, USA. The input mesh is processed seperately with full resolution scheme and hierarchical scheme, and all rest conditions are set as the same. For the full resolution scheme, the full resolution mesh is refined

via our algorithms for 20 iterations. To produce low resolution mesh model for hierarchical scheme, the full resolution mesh is converted to a cloud of oriented points and extract a low-resolution mesh via Poisson reconstruction [Kazhdan and Hoppe, 2013]. The minimal voxel resolution (respectively, octree leaf dimensions) is applied as 2^l GSD. The low resolution is refined using downscaled versions of the original images (factor 2^l) for 20 iterations. Then the mesh is densified by splitting each triangle face into four smaller ones, and refinement is repeated with image scale 2^{l-1}, and so forth until the full resolution is reached. Note that densification factor is the same as the increase in the number of pixels from one pyramid level to the next, hence the average number of pixels per triangle remains the same.

The results are demonstrated in Figure 3.13 and Figure 3.14. The Google Earth snapshots are applied as the reference (in Figure 3.13d and Figure 3.14d). As we can see from Figure 3.13a, the mesh model generated from the DSM has gross errors beside the edge of the building and also some strip structures on the facade. The full resolution scheme can not fully remove the gross errors and strip structures (Figure 3.13b). According to Figure 3.13c, we find that the hierarchical scheme can eliminate these errors. Another example is shown in Figure 3.14. As shown in Figure 3.14a, the DSM based mesh also has some gross errors on the roof of the building and strip structures on the facade plane. Moreover, one of the building's corner is poorly reconstructed because of the shadows. It is obvious, that the gross error and wrong strip structures of the facade exist in the result of the full resolution scheme (Figure 3.14b). The information of the buildings right-bottom corner is still missing. In contrast, the hierarchical scheme reconstructs the right bottom corner more completely and gets rid of the gross errors and the wrong facade strip structures (Figure 3.14c). Therefore, we implement our mesh refinement pipeline in a hierarchical processing scheme because of the improved performance comparing to the full resolution scheme.

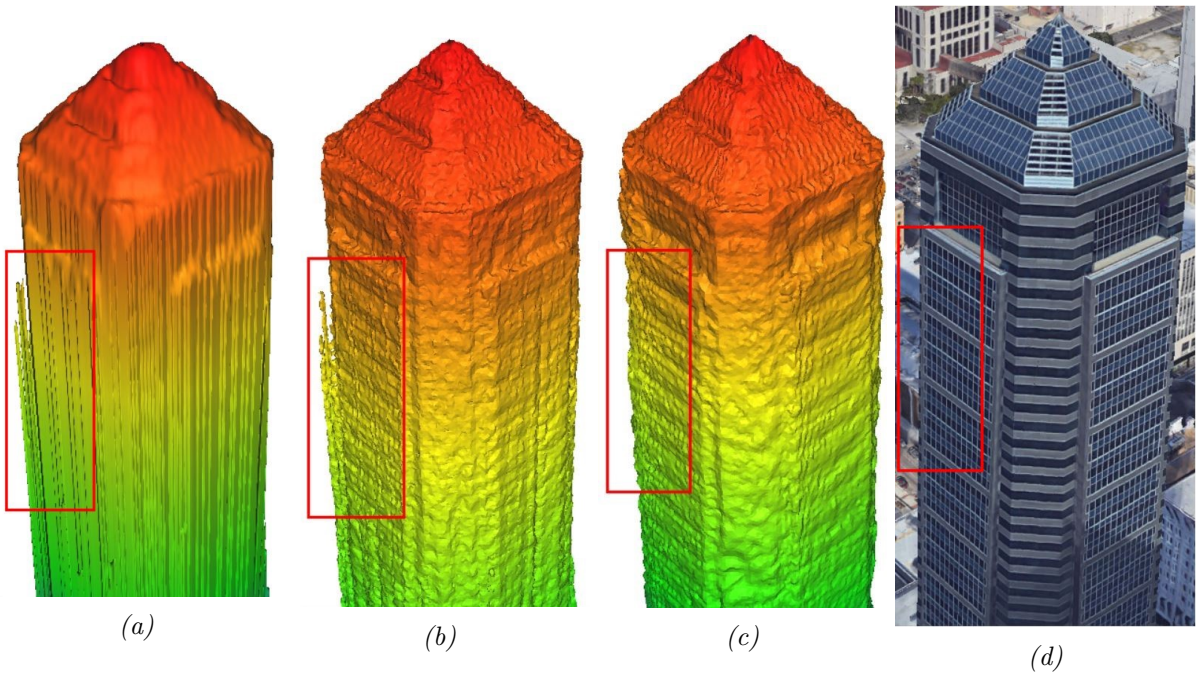


Figure 3.13: The top of the building in: (a) the DSM mesh model (b) full resolution refined mesh model (c) coarse-to-fine refined mesh model (d) the Google Earth image.

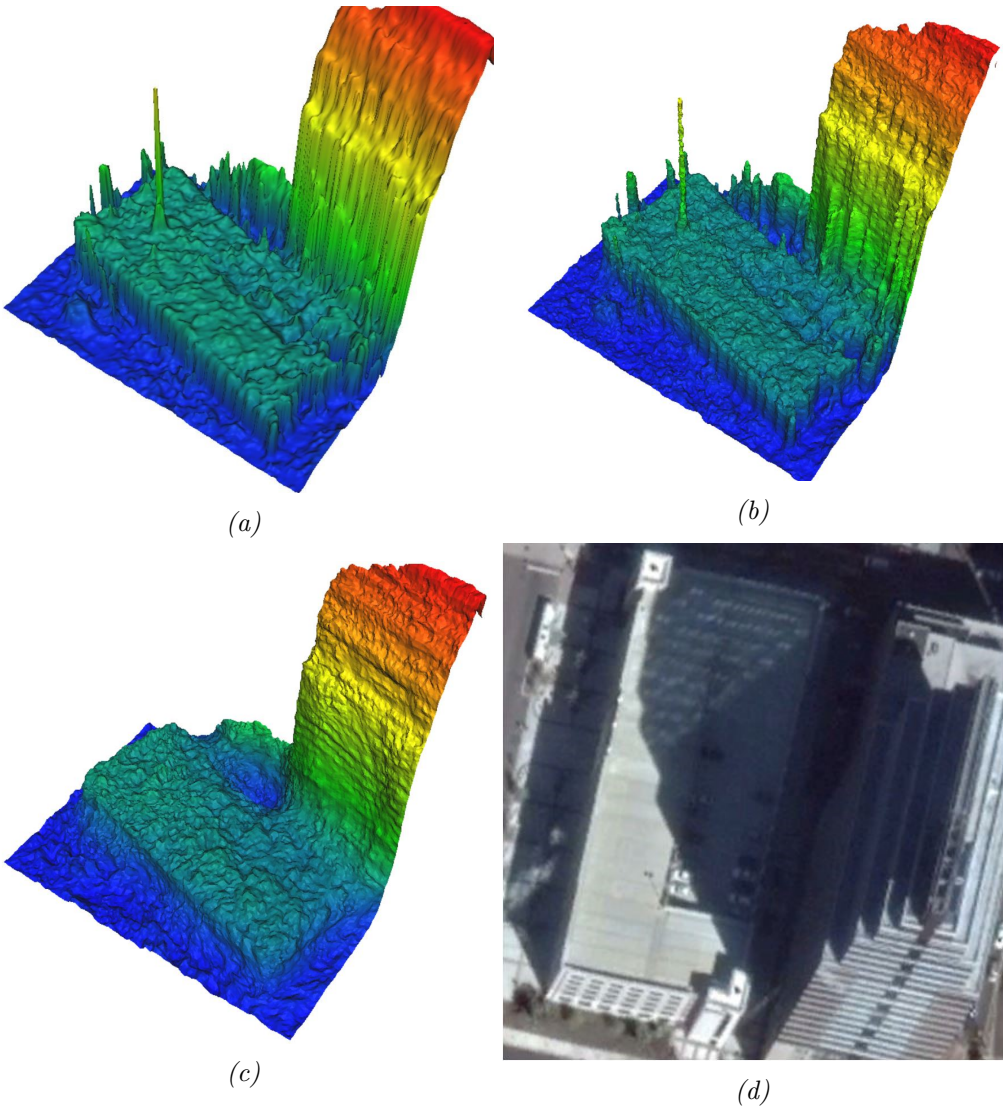


Figure 3.14: The roof of the building in: (a) the DSM mesh model (b) full resolution refined mesh model (c) coarse-to-fine refined mesh model (d) the Google Earth image.

The size of the mesh triangles affects directly the refinement performance. To investigate the influence of the mesh triangle size, the meshes with triangle size from 1 to 5 pixels are reconstructed on the Jacksonville test site. Here, two examples are shown in Figure 3.15 and 3.16. 3.15 displays a parking lot and its neighbouring building, which are refined with different mesh triangle sizes. The rectangles surround some detail structures on the roof. When the triangle size is larger than 3 pixels, the mesh is smooth but many detail structures are blurred (e.g. 3.15c, 3.15d, 3.15e). On the other hand, when the triangle is too small, such as 1 pixel, the reconstructed mesh model is too noisy especially considering the flat plane (3.15a). 3.16 demonstrates the Bank of America tower meshed with different triangle sizes. We find when the triangle size is larger than 3 pixels, the facade get blurred and loses some information. According to our tests, we recommend to refine the mesh with triangle size between 1 pixel and 3 pixels, so that the refined result has structures with

more explicit details and less noises. Therefore, the triangle size of the refined mesh model in each pyramid level is ca. 2 pixels in our processing.

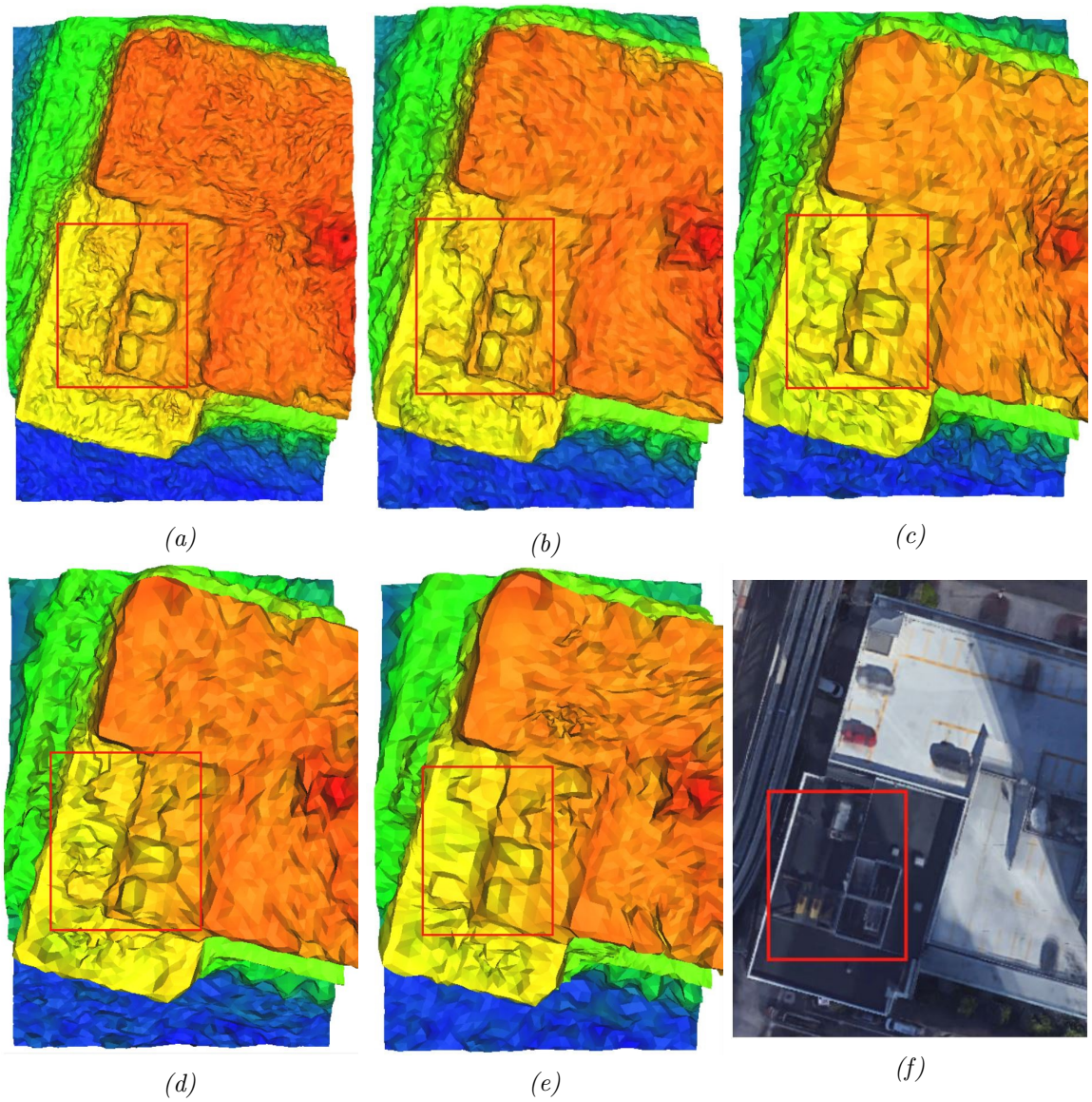


Figure 3.15: Roof refined with different triangle size: (a) 1 pixel (b) 2 pixels (c) 3 pixels (d) 4 pixels (e) 5 pixels (bottom-middle), and (f) the related area on the Google Earth

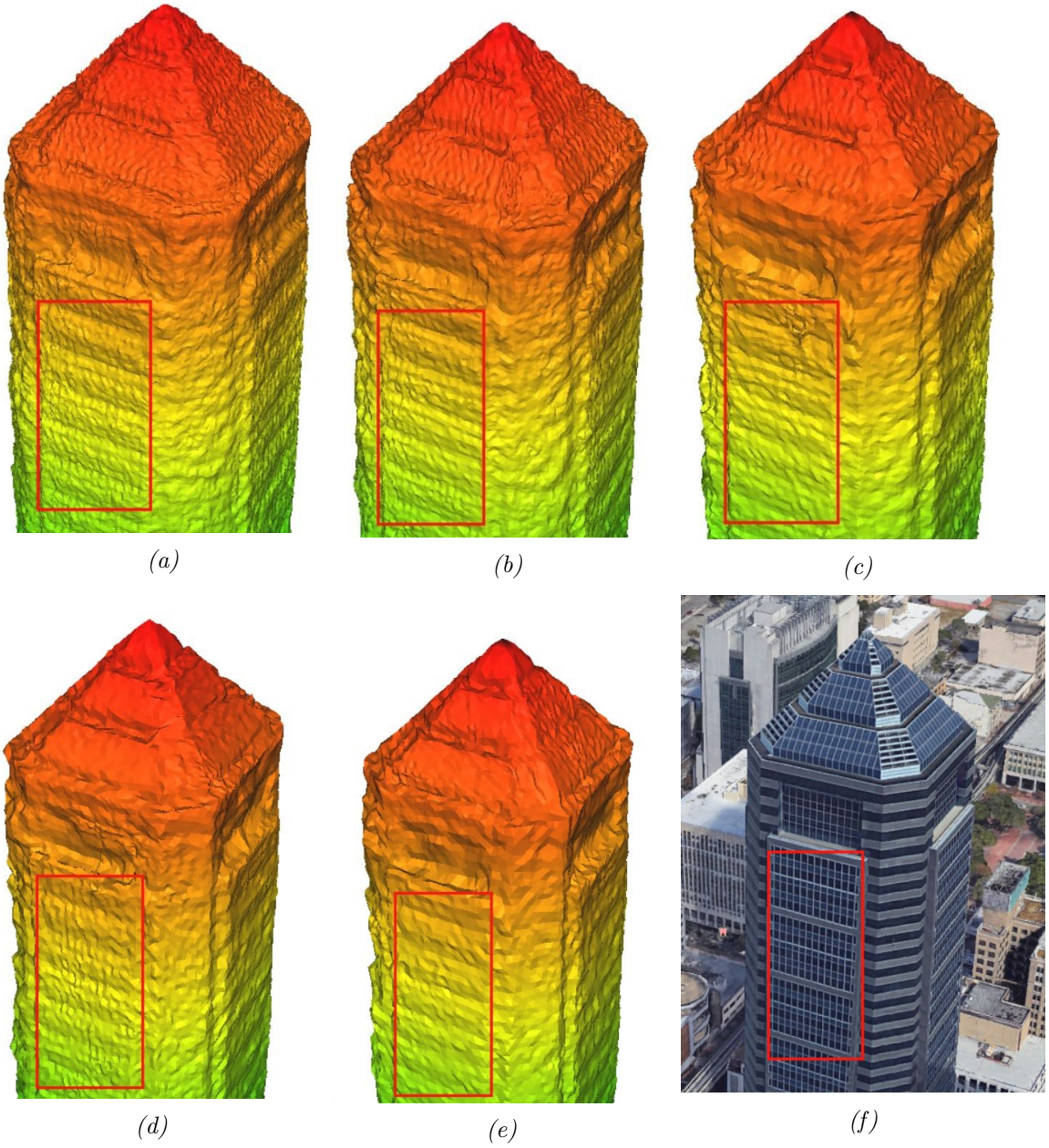


Figure 3.16: Tower building refined with different triangle size: (a) 1 pixel (b) 2 pixels (c) 3 pixels (d) 4 pixels (e) 5 pixels (bottom-middle), and (f) the related building on the Google Earth

Chapter 4

Experiments

In this chapter we present the experiments for our proposed 3Dreconstruction pipeline on different datasets. The datasets applied by our work are presented in section 4.1. The MVS images are first selected according to our image selection strategy. The selected images and the related RPCs are the inputs of the pipeline. The RPCs are refined by the proposed relative bias-compensating method. We test and evaluate the relative orientation with RPCs, and the results are introduced in section 4.2. The RPCs and the compensation parameters are then utilized to generate the epipolar images of satellite data. The experiments and the evaluation results of the proposed modified piece-wise epipolar resampling strategy are presented in section 4.3. With the epipolar images produced by the image rectification procedure, we densely match all the images pair-wisely and generate the point clouds and DSMs. The experiments are conducted on some benchmark data and the results are compared to other state-of-the-art 3Dreconstruction pipelines. The results and evaluations are shown in section 4.4. At last, we generate an initial mesh based on the DSM product and refine it to a true 3D mesh model via our novel mesh refinement method. The comparison between the mesh model based on DSM and the model after refinement is conducted. We show our 3D reconstruction performance and also compare it to some state-of-the-art pipelines in section 4.5.

4.1 Description of Test Data

Several VHR satellite datasets are applied for the testing and evaluation of our 3D reconstruction pipeline. Here we introduce every test dataset in details:

1. ISPRS VHR satellite benchmark [d'Angelo and Reinartz, 2011]: This VHR satellite dataset is established by the ISPRS Working Group I/4 on "Geometric and Radiometric Modelling of Optical Spaceborne Sensors". The benchmark data provides one stereo pair of Worldview-1 (WV-1) panchromatic covering areas in Catalonia, Spain. The GSD is at 0.5m. The data are acquired on the 29th August, 2008. For the stereo pair, the intersection angle of the view is ca. 35°. There are three different test sites in the benchmark data: La mola, Terrassa and Vacarisses (see Figure 4.1). Each test site covers an area of about 5,000×5,000m. The regions of the test sites contain different terrain types, for example rural areas, urban areas, forest areas and mountainous areas. The satellite data are delivered in level 1B (the images are radiometrically corrected but not projected to a plane by map projection) and along with the corresponding RPC files. The benchmark data also

provides the LiDAR (Light Detection and Ranging) point clouds at 1m GSD as the reference data for evaluations.

(a) *(b)* *(c)*

Figure 4.1: WorldView-1 image of ISPRS VHR Satellite benchmark: a. La mola test site b. Terrassa test site c. Vacarisses test site.

2. DLR's VHR satellite dataset: Four WorldView-2 (WV-2) panchromatic images are provided by Deutsche Zentrum für Luft- und Raumfahrt (DLR) Oberpfaffenhofen. The imagery is collected on July 12, 2010, and delivered at Level 1B. The imagery covers the downtown area of Munich, Germany. The GSD of the images is at 0.5m. The size of the applied test site is about 2,000×2,000m (Figure 4.2). A DSM generated from multi-stereo airborne images by software SURE is employed as reference data. The airborne images are collected by DMC II camera. The GSD of the reference DSM is at 0.1m. The reference DSM is acquired from the European Spatial Data Research Organisation (EuroSDR) benchmark [Haala, 2013].

Figure 4.2: WorldView-2 image of DLR's Munich dataset

3. IARPA Multi-View Stereo benchmark [Bosch et al., 2016]: The public MVS benchmark of commercial satellite imagery is released by the John Hopkins University Applied Physics Laboratory

(JHU/APL), USA. The benchmark contains 50 WorldView-3 (WV-3) panchromatic and multi-spectral images. The MVS images are collected from November of 2014 to January of 2016. Three different test sites among the benchmark data are applied in our pipeline. All the test sites are close to San Fernando, Argentina, with GSD of the nadir images at ca. 0.3m. Test site 1 is ca. 1,000×1,000m. The sizes of test site 2 and 3 are about 700×700m. The benchmark data provides a LiDAR point cloud collected on June 2016 as ground truth, with nominal point spacing of 20cm. DSMs at 30cm GSD are produced from the LiDAR point cloud, in order to make equally-spaced comparisons with the results generated from WV-3 panchromatic imagery ([Bosch et al., 2016]). The three test sites are shown in Figure 4.3.

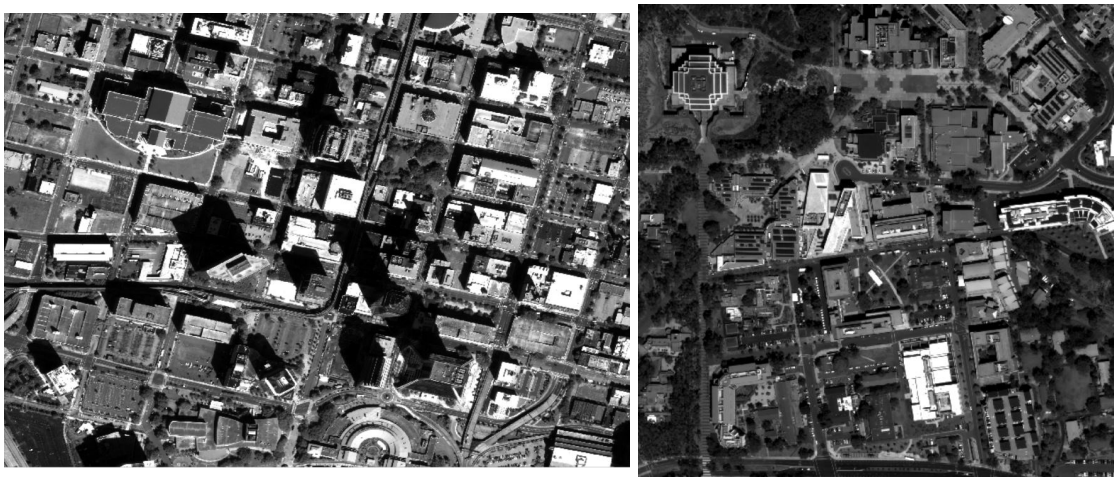
(a) *(b)* *(c)*

Figure 4.3: WorldView-3 image of IARPA MVS benchmark: (a) test site 1, (b) test site 2, (c) test site 3.

4. IARPA CORE3D Public Data [Brown et al., 2018]: This publicly available benchmark, which provides multi-view collections of 16bit panchromatic WV-3 images with 0.3m GSD (at nadir, the actual GSD in off-nadir views can be up to a factor of ≈1.5 lower). Two different test sites were applied in our tests: Downtown of Jacksonville (JAX), FL, USA and University of California San Diego (UCSD), CA, USA. The JAX test site covers an urban area with size of about 750m×750m. It is covered by 26 images, which are collected between October of 2014 and February of 2016. The UCSD test site consists of 35 images covering an area of ca. 600m×600m. The MVS imagery is collected between October of 2015 and August of 2017. For both test sites, 2.5D LiDAR DSMs with 0.5m GSD are provided as the ground truth. The test site overview is displayed in 4.4.

4.2 RPC Compensation

As described in section 3.2.2 and in order to apply our RPC relative-compensated method, we generate a number of tie points from the stereo images manually, with software Envi. We then select one stereo pair as base stereo pair to generate the virtual surface. The relative pointing error is calculated via the RPC projection. An additional affine model for the image coordinates is evaluated to correct the relative pointing error. The tie points are projected to object space to calculate the virtual GCPs by using the RPCs and the related correction parameters. A part of the tie points are applied as the virtual GCPs. The rest of the tie points are used as the check points in the next steps of the orientation. For all the images, we then conduct the relative bias-compensating



(a) *(b)*

Figure 4.4: WorldView-3 image of CORE3D benchmark: a. Jacksonville test site b. University of California San Diego test site.

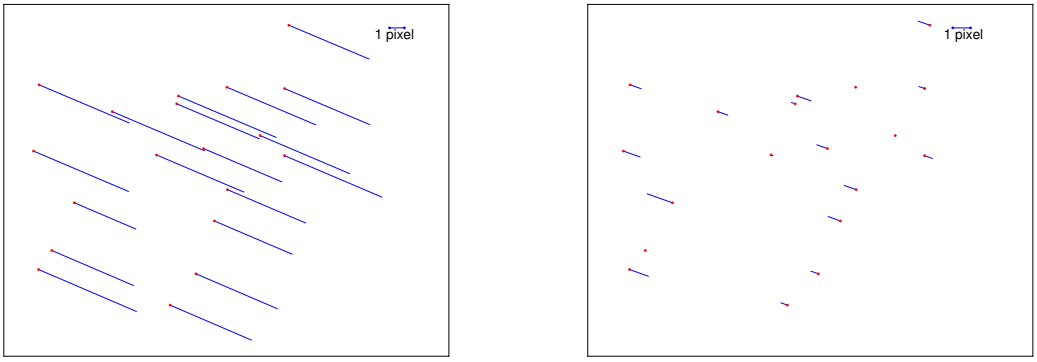
	RMSE [pixels]	Mean [pixels]
Before Refinement	6.543	6.511
After Refinement	0.654	0.555

Table 4.1: Relative pointing errors evaluation of the base stereo of San Fernando test site 1.

bundle block adjustment with the virtual GCPs and the check points. Additional shift parameters are applied to compensate the bias caused by the RPCs. All the RPCs are aligned to the surface where the virtual GCPs are located on. In the procedure of our relative RPC compensation, no actual GCPs are required. In order to check the performance of our relative RPC compensation algorithm, the San Fernado test site 1 and test site 2 of the IARPA MVS benchmark and the Munich test site from DLR's VHR satellite dataset are applied in our experiments.

For the San Fernado test site 1 of IARPA WV-3 MVS benchmark, nineteen corresponding points are generated manually. For the MVS satellite images, we name each image by simple number index. We select Nr. 6 and Nr. 19 image from all the stereo pairs as the base stereo pair. The relative pointing error correction is conducted on the base stereo pair. The affine parameters are applied to reduce the errors. The relative pointing error measured before and after the correction is presented in Figure 4.5. On the top-right corner, the segment presents the scale of one pixel. The red points represent the location of the corresponding points and the blue lines indicate the bias to the corresponding epipolar lines (pointing error). According to Figure 4.5a, all tie points have relative pointing errors larger than one pixel. With the correction, the relative pointing error is decreased to sub-pixel level (Figure 4.5b). We compute the mean error and the root-mean-squared-error (RMSE) of the relative pointing errors for further quantitative evaluation. The result is presented in Table 4.1. As we can observe from the table, the RMSE and the mean error of the relative pointing error are decreased from 6.5 pixels to sub-pixel level via our correction.

With the corrected RPCs, the object coordinates of the tie points are calculated via forward



(a) *(b)*

Figure 4.5: Relative pointing error on the base stereo pair of San Fernando test site 1: (a) before correction (b) after correction

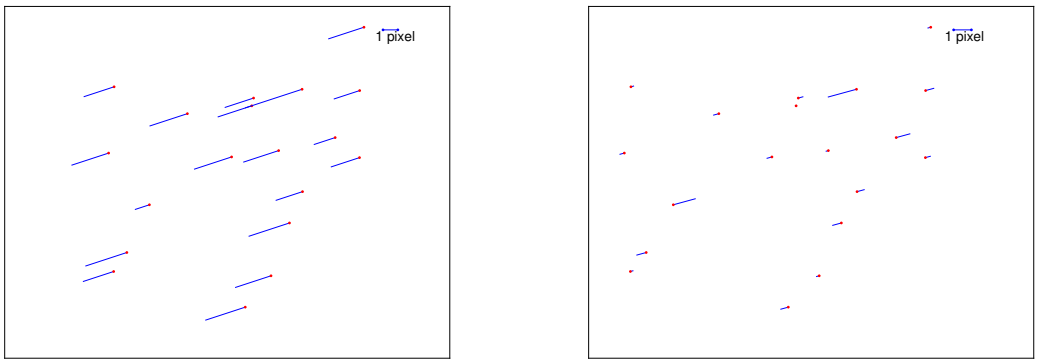
	RMSE [pixels]	Mean [pixels]
Before Refinement	2.420	2.366
After Refinement	0.510	0.414

Table 4.2: Relative pointing errors evaluation of the example stereo of San Fernando test site 1.

intersection. Nine of them are then selected as virtual GCPs and ten points are applied as the check points. We apply our relative RPC compensation method for all the images. Because there are hundreds of different combinations of the stereo pairs in San Fernando test site 1 to be chosen, we only select one example stereo here to show the performance of our relative RPC compensation. Here we present the compensation result of image Nr. 1 and image Nr. 2 as example. The results of all the stereo pairs are presented in the Appendix. The relative pointing errors before and after the compensation are depicted in Figure 4.6. According to the pictures, it is obvious that the relative pointing error of this example stereo pair is reduced to sub-pixel level after our relative RPC compensation. We compute the RMSE and mean error of the relative pointing error. Table 4.2 presents the evaluation result. Before the relative RPCs compensation, the RMSE and mean error of the bias caused by RPC are ca. 2.4 pixels. Through our RPC compensation, the RMSE is reduced to 0.5 pixels and the mean error is reduced to 0.4 pixels. For the check points on the example stereo pair, we calculate the related object coordinates and evaluate the accuracy. The RMSEs of the longitude, latitude and elevation are displayed in Table 4.3. The planar coordinates of the object points have a RMSE less than 1GSD (0.5m), and the height's RMSE is 0.8m.

	Longitude [m]	Latitude [m]	Height [m]
RMSE	0.468	0.065	0.802

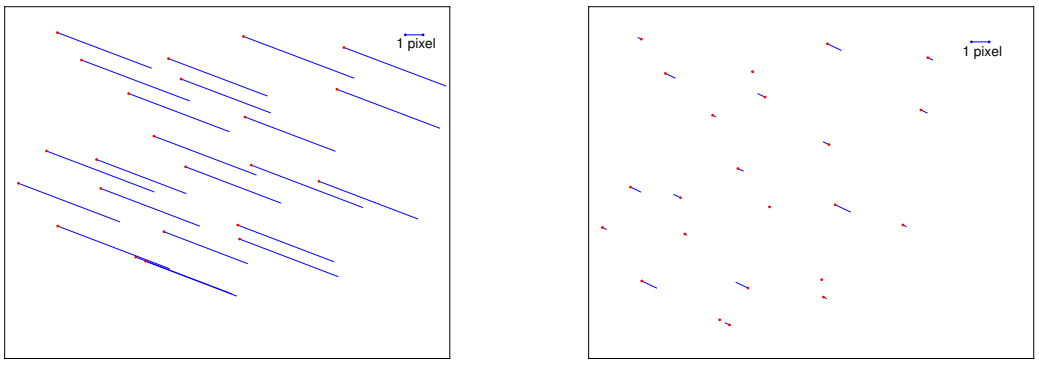
Table 4.3: Object coordinates evaluation of the example stereo of San Fernando test site 1.



(a) *(b)*

Figure 4.6: Relative pointing error on the example stereo pair of San Fernando test site 1: (a) before correction (b) after correction

As to the San Fernando test site 2, in total, 23 tie points are generated manually with Envi. Similar to test site 1, we select Nr. 1 image and Nr. 6 image as the base stereo pair for virtual surface generation. The relative pointing errors of the tie points before and after correction are depicted in Figure 4.7. The figures show that the correction procedure reduces the relative pointing errors from several pixels to sub-pixel level. In table 4.1, the RMSE and mean error of the relative pointing errors on the base stereo are shown. The RMSE and mean error of the bias before correction are as large as 6.1 pixels. Both RMSE and mean error are decreased to less than 0.5 pixels after the correction.



(a) *(b)*

Figure 4.7: Relative pointing error on the base stereo pair of San Fernando test site 2: (a) before correction (b) after correction

We generate the object coordinates of the tie points with the corrected RPCs. Ten of the tie points are applied as virtual GCPs and the rests are used as the check points. With the virtual GCPs

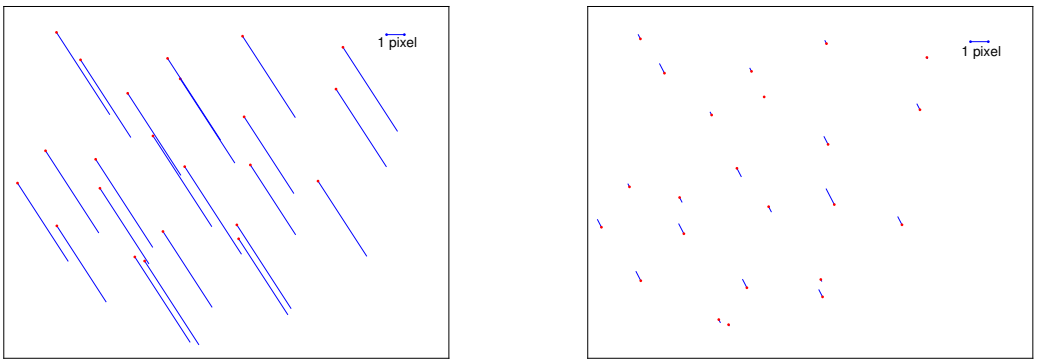
	RMSE [pixels]	Mean [pixels]
Before Refinement	6.170	6.154
After Refinement	0.451	0.367

Table 4.4: Relative pointing errors evaluation of the base stereo of San Fernando test site 2.

	RMSE [pixels]	Mean [pixels]
Before Refinement	6.863	6.851
After Refinement	0.463	0.405

Table 4.5: Relative pointing errors evaluation of the example stereo of San Fernando test site 2.

and the check points, we refine the RPCs by the relative bias-compensating algorithm. Similar to the San Fernando test site 1, an example stereo pair composed of image 1 and 26 is applied to present the evaluation of the compensation results here. The evaluations for all stereo pairs are demonstrated in the Appendix. We measure the relative pointing errors of all the tie points before and after our relative RPC compensation. Figure 4.8 reveals the pointing errors of the example stereo pair image. The relative pointing error is significantly decreased after the RPC compensation. We compute the RMSE and the mean error of the relative pointing error and present them in Table 4.5. After refinement, the RMSE and the mean error of the relative pointing error are improved by 6 pixels and reach sub-pixel level. We calculate the object coordinates of the check points with the compensated RPCs. The accuracy of the check points is then evaluated, which is presented in Table 4.6. The RMSE of the longitude is 0.26m and the latitude is ca. 0.1m. The RMSE of the height is close to half meter. The accuracy of the calculated object coordinates of the check points is at sub-meter level.



(a) *(b)*

Figure 4.8: Relative pointing error on the example stereo pair of San Fernando test site 2: (a) before correction (b) after correction

At last, we test the relative RPC compensation on the Munich test site from DLR's VHR satellite

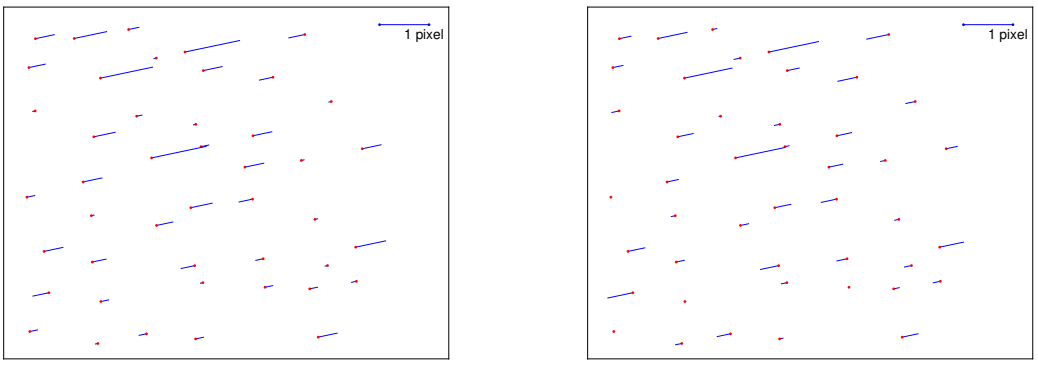
	Longitude [m]	Latitude [m]	Height [m]
RMSE	0.263	0.086	0.480

Table 4.6: Object coordinates evaluation of the example stereo of San Fernando test site 2.

	RMSE [pixels]	mean [pixels]
Before Refinement	0.339	0.271
After Refinement	0.303	0.240

Table 4.7: Evaluation of the relative pointing errors on the selected base stereo pair of Munich test site.

dataset. On the four WV-2 images, 45 tie points are generated by Envi manually. From all stereo pair combinations, one stereo pair is selected randomly to generate the virtual GCPs. Here we select image 1 and 2 as the base stereo pair. The relative pointing error of the base stereo pair is corrected by utilizing the image coordinates of the tie points. We measure the relative pointing error of the tie points on the base stereo pair and compare the results before and after the correction. The pointing error comparison of the base stereo image pair is demonstrated in Figure 4.9. The scaled vector maps are depicted like the other two test sites. The segment on the left top presents the scale of 1 pixel. The red points are the image points and the blue lines present the bias. As shown in 4.9a, we find that the pointing errors before the correction is already at sub-pixel level. After the refinement, the pointing errors of refined RPCs is at a close level. The improvement is hard to observe by visual check. We calculate the RMSE and the mean value of the relative pointing errors. The comparison of the results before and after correction is presented in Table 4.7. According to the table, the improvement of the mean error and RMSE are improved slightly after the relative pointing correction, which is less than 0.1 pixels.



(a) *(b)*

Figure 4.9: Relative pointing error on the selected base stereo pair of Munich test site: (a) before correction (b) after correction

The object coordinates of the tie points are calculated with the corrected RPCs. Fifteen of them

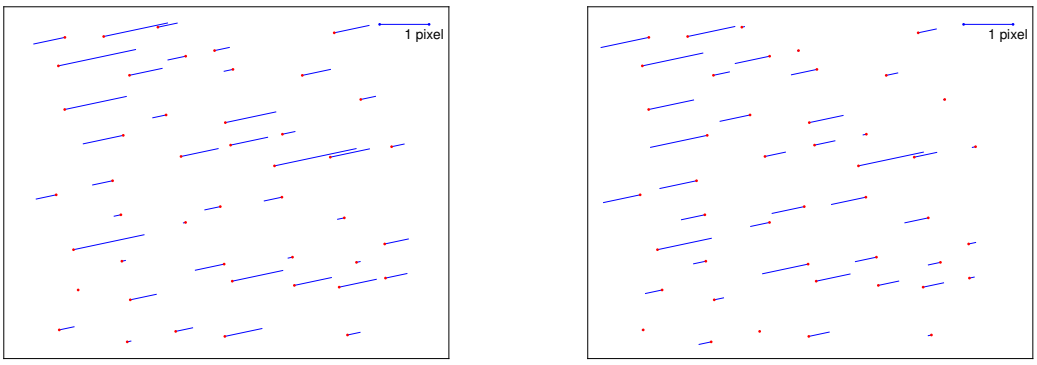
	RMSE [pixels]	Mean [pixels]
Before Refinement	0.629	0.517
After Refinement	0.563	0.463

Table 4.8: Relative pointing errors evaluation of Munich test site.

	Longitude [m]	Latitude [m]	Height [m]
RMSE	0.287	0.109	0.701

Table 4.9: Object coordinates evaluation of the example stereo of Munich test site.

are selected to generate the virtual GCPs and the remaining 30 points are set as the check points. We then conduct the relative RPC compensation for all the image pairs with the virtual GCPs and the check points. In Figure 4.10, we show the relative pointing errors of all the corresponding points on an example stereo pair. The selected example image pair is the 1st and the 3rd image of the dataset. The evaluation results of all the stereo pairs are presented in the Appendix. As shown in Figure 4.10a, most relative pointing errors are at sub-pixel level. Through our relative RPC compensation, the relative pointing error is slightly reduced in Figure 4.10b. The comparison for the mean error and RMSE of the relative pointing error are presented in Table 4.8. The RMSE and the mean error are decreased by 0.06pixels. The object coordinates of the check points are calculated with the compensated RPCs. The evaluation of the object coordinates of the check points is shown in Table 4.9. The RMSEs of the longitude and latitude are less than 1 GSD. The height's RMSE is at sub-meter level.



(a) *(b)*

Figure 4.10: Relative pointing error of WorldView-2 data: (a) before RPC refinement (b) after RPC refinement

According to the experiments on these three different test sites, the large systematic residuals of the relative pointing error can be significantly decreased from pixel level to sub-pixel level. If the relative pointing error of the RPCs is already at sub-pixel level, the result of our relative RPC

compensation maintains the same level accuracy and provides slight improvements. The object coordinates of the check points have less than 1GSD accuracy for planar coordinates and sub-meter accuracy for heights. In general, the experiments have proved that the proposed relative RPC compensation algorithm can reduce the relative pointing error effectively, so that the compensated RPCs can be applied in the subsequent procedures of our 3D reconstruction pipeline.

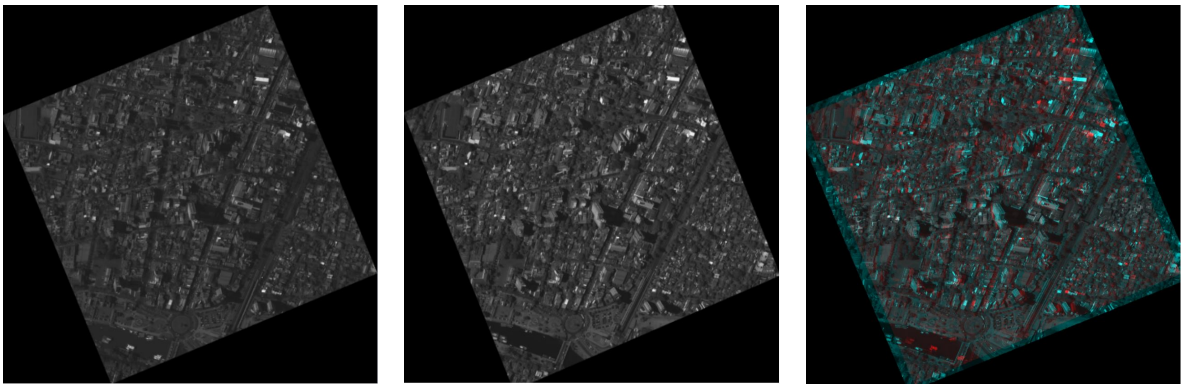
4.3 Image Rectification

Following the workflow displayed in Figure 3.1, the image rectification starts right after the RPCs are compensated. Through the image rectification procedure, the stereo images are rectified to the epipolar images so that the corresponding points locate on the same line. We apply our modified piece-wise epipolar resampling method described in section 3.3.2 to generate the satellite epipolar images. First, the epipolar image coordinate system is defined by the center point of the image. Along the y-axis of the epipolar image coordinate system, we find the start points. We then derive the segments for epipolar line approximation. Once all the epipolar segments are generated, we adjust them to the same row in the epipolar image coordinate system. The parameters of the epipolar segments are recorded for the use of the later point cloud calculation. The epipolar images are pixel-wise resampled along each epipolar segment with the bi-linear interpolation method. The sample distance of the epipolar images is equal to the pixel size of the base image.

In this section, we present the performance of the image rectification procedure on the San Fernando test site 3 from IARPA Multi-View Stereo benchmark and the Terrassa test site from ISPRS VHR satellite benchmark. We show some example epipolar stereo pairs selected from the MVS dataset, and we also highlight some detail areas. In order to numerically validate the accuracy of the generated epipolar images, we apply the software Envi to extract some corresponding image points from the epipolar images. The generated corresponding points are evenly distributed in the whole overlapping area. We measure the absolute differences of the y-coordinates of the corresponding points on the epipolar image, which present the vertical parallaxes (or y-parallaxes).

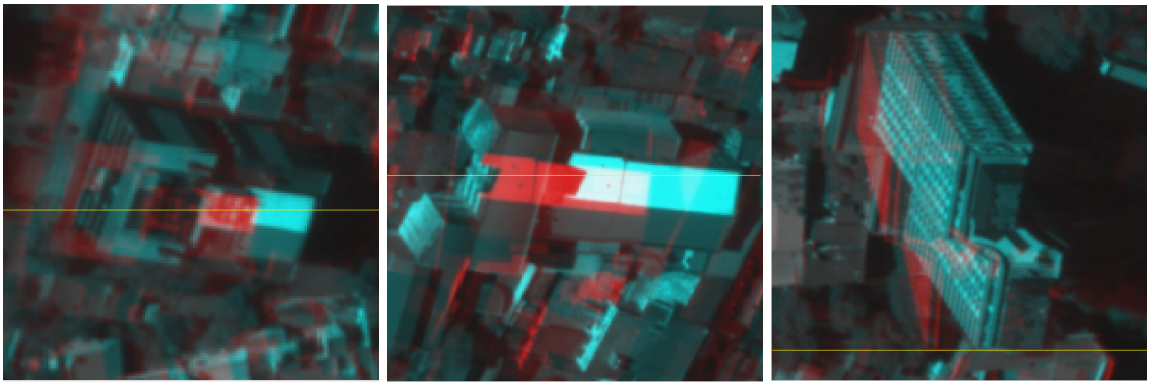
The first example stereo pair is selected from San Fernando test site 3. The epipolar stereo pair is displayed in Figure 4.11a and 4.11b. For better visualization, the epipolar images are overlaid to generate the stereo anaglyph, which is shown in Figure 4.11c. The y-parallax can be checked as the difference of the row coordinates between the left view (red) and right view (cyan). For this stereo pair, the image size is ca. 2,500 $\times$ 2,500 pixels. In order to investigate the accuracy of our epipolar resampling over the entire image, several sub-areas on the stereo anaglyph are extracted to present more details. Five sub-areas located at the middle and the four corners of the image are selected. The anaglyph of the sub-areas are shown in Figure 4.12. The yellow straight lines in the sub-images are applied to observe whether the corresponding points in left scene and right scene are located on the same row. According to Figure 4.12, no matter the sub-area is located in the middle of the image or at the corners, the corresponding points on the epipolar images are located in the same row.

With the help of Envi, we generate five hundreds corresponding image points from the selected example epipolar stereo pair in San Fernando test site 3. The absolute differences of the vertical coordinates between the corresponding points on the left and right epipolar image are measured. Using the measured vertical parallaxes, we draw the line graph in Figure 4.13. According to the figure, the vertical parallaxes of all the corresponding points are less than 0.4 pixels. The mean value and the RMSE of the vertical parallaxes are evaluated. To evaluate the robustness of our

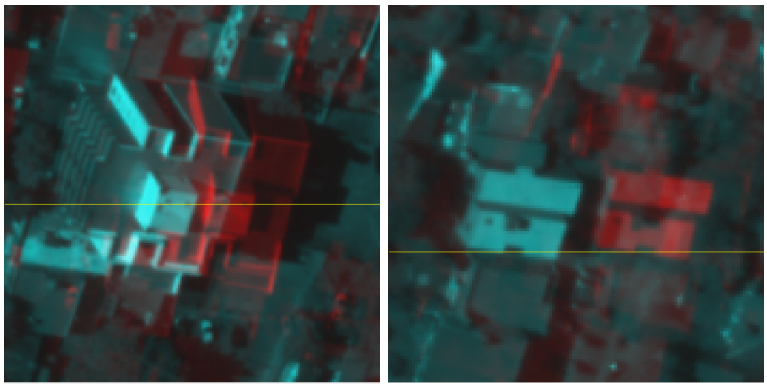


(a) *(b)* *(c)*

Figure 4.11: The anaglyph image of San Fernando test site 3 epipolar image pair



(a) *(b)* *(c)*



(d) *(e)*

Figure 4.12: The sub-areas of the San Fernando test site 3 anaglyph image at: (a) the top-left corner (b) the top-right corner (c) the middle of the image (d) the bottom-left corner (e) the bottom-right corner

	mean [pixels]	RMSE [pixels]	NMAD [pixels]
San Fernado 3	0.156	0.179	0.104

Table 4.10: Evaluation of the vertical parallaxes.

epipolar resampling, the Normalized Median Absolute Deviation (NMAD) of the vertical parallax is calculated. The results of the two test sites are presented in Table 4.10. As the table shows, the mean error and the RMSE of the vertical parallax is less than 0.2 pixels, and the NMAD is 0.1 pixels. For this small range stereo pair, our epipolar resampling method is accurate and robust.

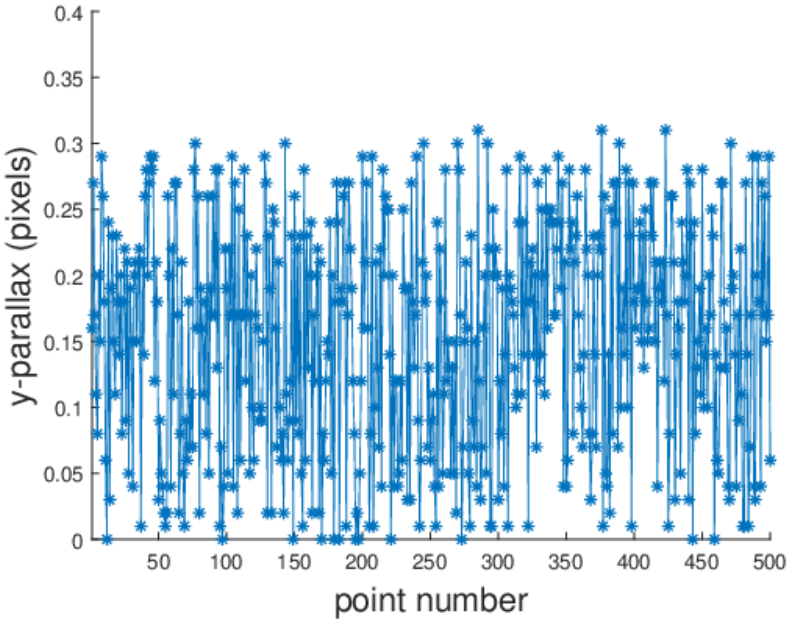
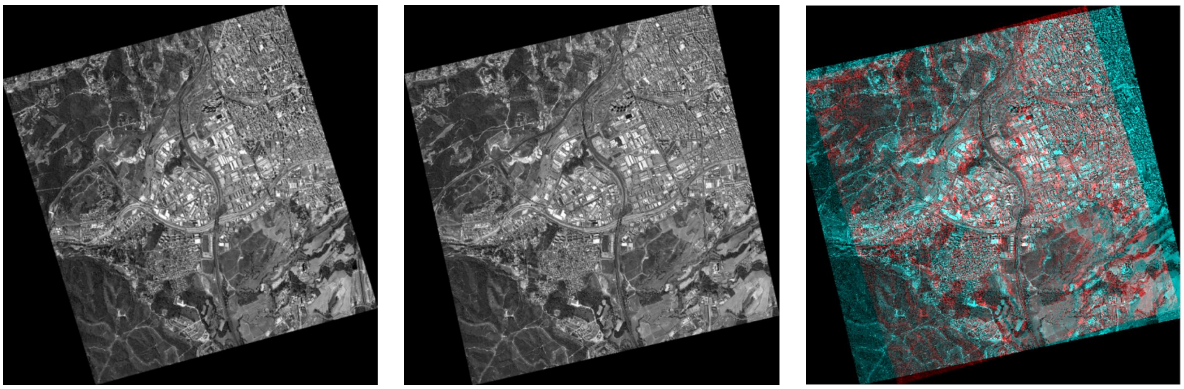


Figure 4.13: The vertical parallaxes of San Fernado epipolar image pair

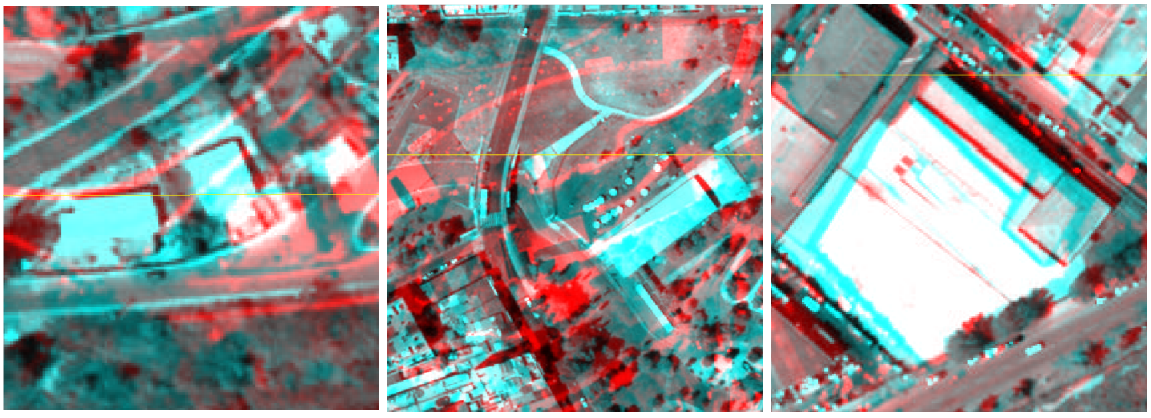
The second example is from Terrassa test site. The stereo images are processed by the modified piece-wise resampling method. The generated epipolar stereo images are shown in Figure 4.14. Similar to the San Fernando test site 3, the overlapped stereo anaglyph is also generated for the epipolar images, which is shown in 4.14c. Different from the San Fernando test site 3, the image size of Terrassa test site is over 10,000 pixels, which is much larger than the former dataset. It is important to check if the piece-wise epipolar resampling works correctly over the whole image scene in this large coverage stereo pair. We choose five sub-areas located in the middle and at the four corners of the image to show more details. The stereo anaglyphs of the sub-areas are shown in Figure 4.15. The yellow lines in the sub-images are some example rows to verify if the corresponding points are located in the same line. As Figure 4.15 displays, not only the corresponding points in the middle area of the image, but also the corresponding points at the corners are located in the same line .

For the Terrassa test site, a thousand corresponding image points are generated from the epipolar stereo pairs. The absolute differences of the vertical coordinates of the corresponding points are

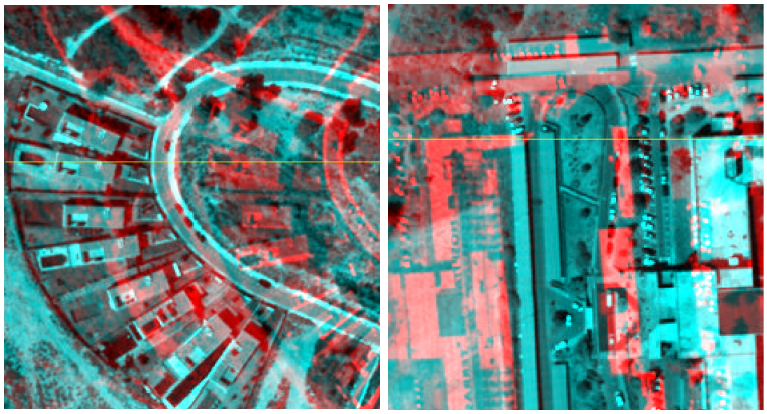


(a) *(b)* *(c)*

Figure 4.14: The anaglyph image of Terrassa epipolar image pair



(a) *(b)* *(c)*



(d) *(e)*

Figure 4.15: The sub-areas of the Terrassa test site anaglyph image at: (a) the top-left corner (b) the top-right corner (c) the middle of the image (d) the bottom-left corner (e) the bottom-right corner

	mean [pixels]	RMSE [pixels]	NMAD [pixels]
Terrassa	0.568	0.615	0.193

Table 4.11: Evaluation of the vertical parallaxes.

calculated to present the vertical parallaxes. The line graph of the vertical parallaxes is depicted in Figure 4.16. In Figure 4.16, we find that most corresponding points have the vertical parallaxes less than 1 pixel. But there are few vertical parallaxes of the corresponding points larger than 1 pixel. We compute the mean value and the RMSE for the vertical parallaxes. We also calculate the NMAD of the vertical parallax for the robustness evaluation. The results of the Terrassa test sites are presented in Table 4.11. Table 4.11 reveals that the mean value of the vertical parallaxes is about half pixel and the RMSE of the vertical parallaxes is close to 0.6 pixels. The proposed epipolar resampling algorithm has worse performance for this large range stereo images, but the accuracy is still at sub-pixel level. The NMAD of the vertical parallaxes is 0.2 pixels, which means our method is robust.

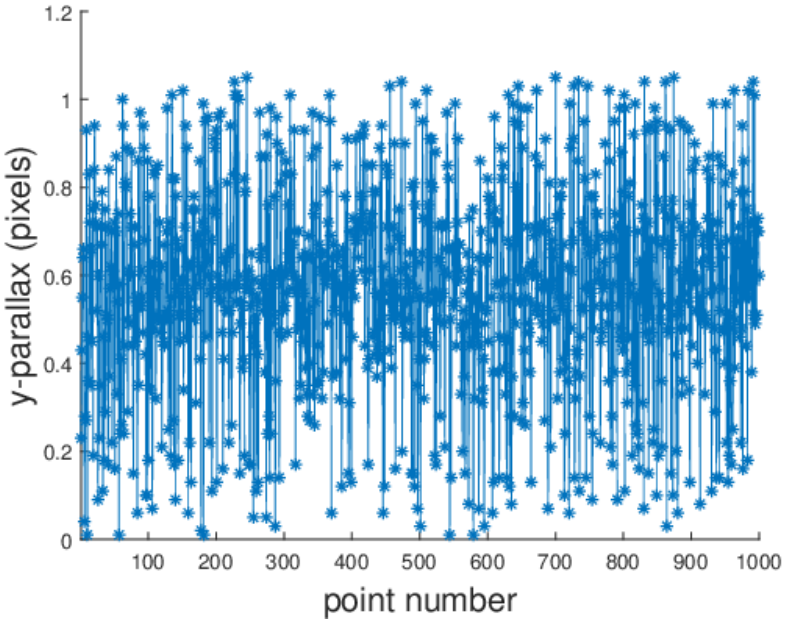


Figure 4.16: The vertical parallaxes of Terrassa epipolar image pair

The experimental results have proved, that our modified piece-wise epipolar resampling method can generate robust epipolar images with sub-pixel level vertical parallaxes, no matter the size of the input image is large or small.

4.4 DSM and Point-cloud Generation

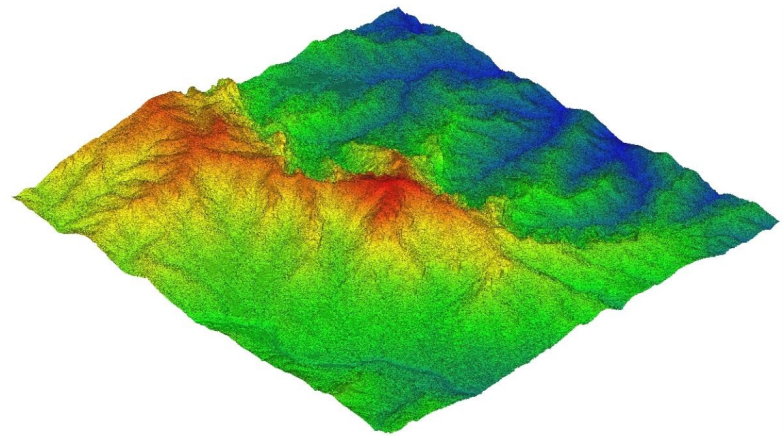
According to our workflow, the dense image matching is pairwisely conducted when the epipolar stereo images are generated. The tSGM algorithm is applied for the dense matching in our pipeline. The disparity maps, which contains the corresponding information of every pixels in the base map, are the products of the pixel-wise matching procedure. With the help of the epipolar segments parameters, the corresponding pixels in the original stereo images are derived and projected to the object space with our compensated RPCs. The point clouds are generated via the forward intersection. The object points calculated directly from the RPCs are in the geometric coordinate system. For better visualization, the point clouds are usually transferred to some Cartesian coordinate systems by map projection, for example the UTM coordinate system. To produce the DSMs, the point clouds are converted into a discretized and regular space grid. In the case of multi-stereo, the DSMs are fused. The simple and robust median filter is applied to give the median height values to grids of the final DSM. Regarding to the processing time, for example in the 2km by 2km Munich test site, our proposed pipeline takes 2.8min to generate the DSM from one pair of high resolution satellite images.

In this section, we show the binocular reconstructed point clouds and the fused DSMs of different test sites. The pipelines provided by the benchmark organizers and one of the top open-source 3D reconstruction pipeline – S2P [Facciolo et al., 2017] are also applied to generate the fused point clouds and DSMs for comparison. The S2P pipeline divides the whole satellite imagery into small tiles, then it corrects the relative pointing error by shift parameters and approximates the epipolar lines by straight lines in each tile. Their dense image matching module is based on MGM algorithm [Facciolo et al., 2015]. For all pipelines, we apply the same selected stereo images and the corresponding RPC files as input data. The Lidar point clouds and DSMs provided along with the benchmark data are applied as the ground truth. Both quantitative and qualitative evaluations are presented on each test sites. The performance of different pipelines will also be compared and discussed. In section 4.4.1, it exhibits the performance of our pipeline for only one stereo pair, which is provided by the ISPRS VHR satellite benchmark. The multi-stereo reconstruction results of the Munich test site are demonstrated in section 4.4.2. At last we reconstruct and take experiments and analysis on the IARPA Multi-View Stereo benchmark with results shown in section 4.4.3.

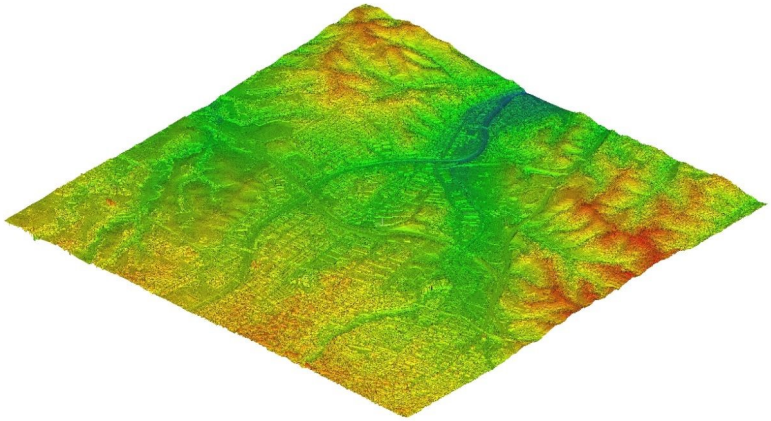
4.4.1 Stereo Reconstruction for the ISPRS Benchmark

As we introduced in section 4.1, the ISPRS VHR satellite benchmark data provides one stereo pair WV-1 image for La Mola, Terrassa and Vacarisses test sites. For these three test sites, we reduce the relative pointing error by following along the correction method for one stereo. The epipolar images then are generated and densely matched. With the correspondences, we triangulate the point cloud for each test site. The point clouds are transferred to the UTM coordinate system and the GSD is 0.5m. The oblique views of the point clouds of the three test sites are presented in 4.17, which is color-coded according to the elevations. Because of the lack of image redundancy, we generate the DSMs at 1m GSD from the point clouds. The DSMs of the three test sites are shown in Figure 4.18.

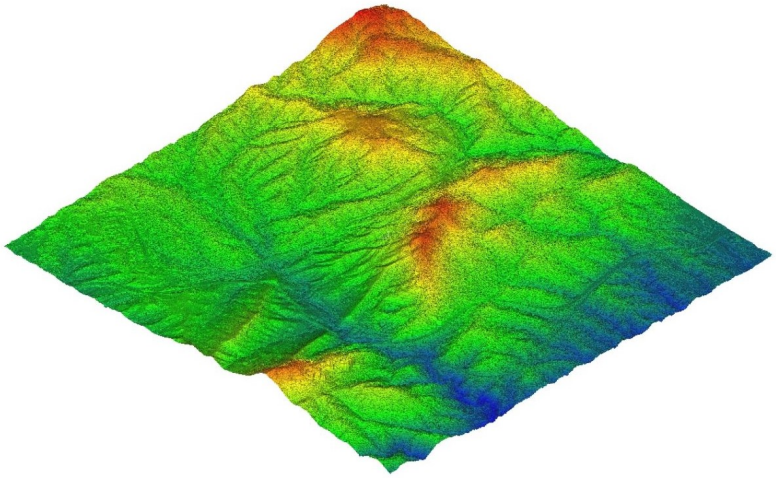
With the same RPCs and stereo images, the point clouds and DSMs of these three test sites are also generated via S2P pipeline. We compare the point clouds generated from WV-1 stereo images to the reference LiDAR point clouds. Figure 4.19, 4.20 and 4.21 demonstrate the comparison results.



(a)

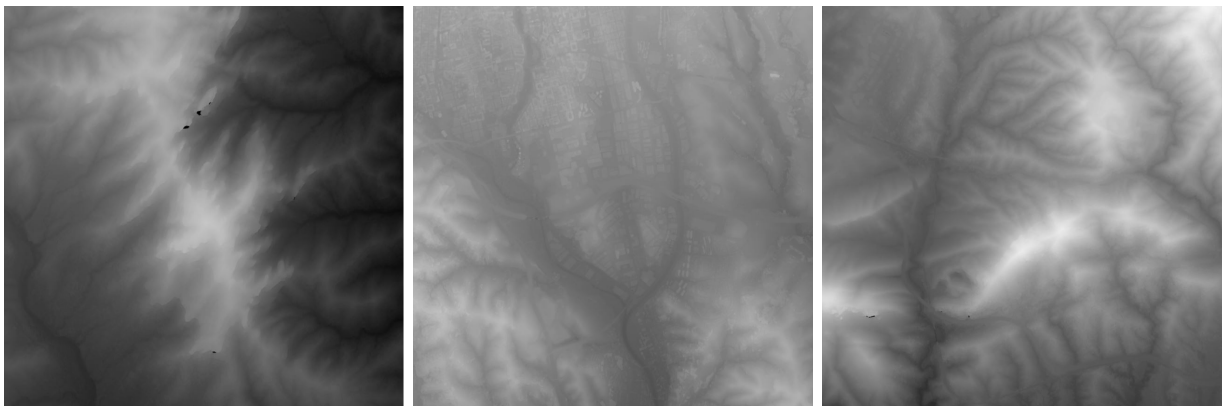


(b)



(c)

Figure 4.17: The point cloud of: (a) La Mola test site (b) Terrassa test site (c) Vacarisses test site



(a) *(b)* *(c)*

Figure 4.18: The generated DSM of: (a) La Mola test site (b) Terrassa test site (c) Vacarisses test site

The shaded LiDAR point clouds are presented in Figure 4.19c, 4.20c and 4.21c. With the color which coded by the absolute height difference to the reference LiDAR point clouds, the height difference maps of the point clouds generated via our tSGM 3D reconstruction pipeline are shown in Figure 4.19b, 4.20b and 4.21b. The height different maps of S2P's point clouds are demonstrated in Figure 4.19a, 4.20a and 4.21a. At the right side of the height different maps, the scale bar and the distribution of the height differences are presented. The stereo images are tile-wise processed in S2P pipeline. As shown in Figure 4.19c, La mola test site is a mountainous area. According to Figure 4.19a, the reconstructions of this mountainous region are failed in some tiles. The height differences of most successfully reconstructed areas are less than 3m in S2P's result. Figure 4.19b shows that our reconstruction has height differences less than 3m in most areas. The point cloud generated from our tSGM pipeline is more complete. The middle of the tSGM point cloud's height difference map has a hole because of missing points in the reference data. Large errors are mainly caused by the steeps in the area. Terrassa test site contains more urban areas in the scene. Most height differences between our reconstructed point cloud and the LiDAR point cloud are less than 2.5m in Terrassa test site. The S2P point cloud also has the height differences less than 2.5m in most areas of Terrassa test site. But two tiles of S2P point clouds are failed to rebuild the terrain. In the urban area, the shadows of the buildings lead to errors. For both our tSGM and S2P pipeline, the height differences are below 2.5m for most areas in Vacarisses test site. In the North-east of the point cloud, an area has significant changes, which lead to a large height difference to the reference data. S2P pipeline has problems with the reconstruction of some southern tiles and lost a lot of information. For most areas in these three test sites, the surfaces generated from the WV-1 stereo pair via two 3D reconstruction pipelines have height differences less than 3m to the ground truth. Somehow, not all of the tiles are reconstructed successfully via the S2P pipeline in the three large test sites. The problems mainly occurs in mountainous and forest regions.

To peak a sight of the details of our 3D reconstruction's performance, we select three Region of Interests (ROIs). The first ROI is a mountainous forest area. The reference Lidar point cloud and the point clouds generated by tSGM and S2P pipelines are presented in Figure 4.22a, 4.22b and 4.22c. All the point clouds are color-coded according to the height values. Compared to the point clouds generated from satellite stereo images, the LiDAR point cloud reconstructs more vegetation

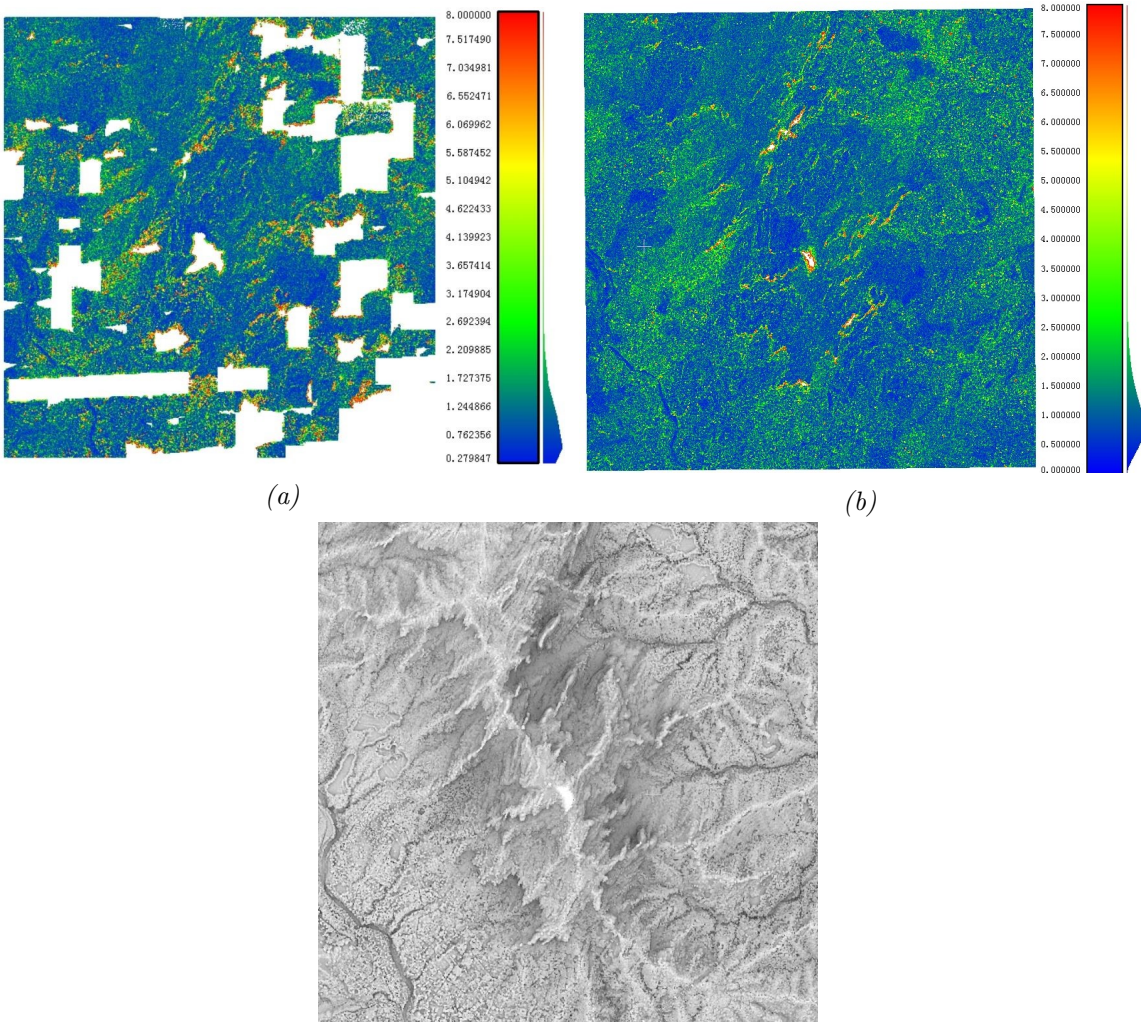
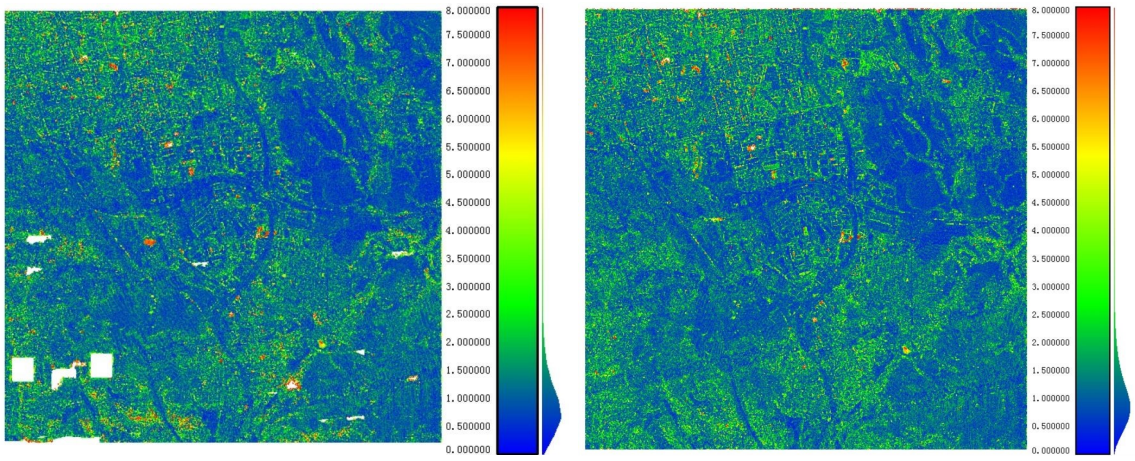


Figure 4.19: La Mola test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud.



(a) *(b)*

(c)

Figure 4.20: Terrassa test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud..

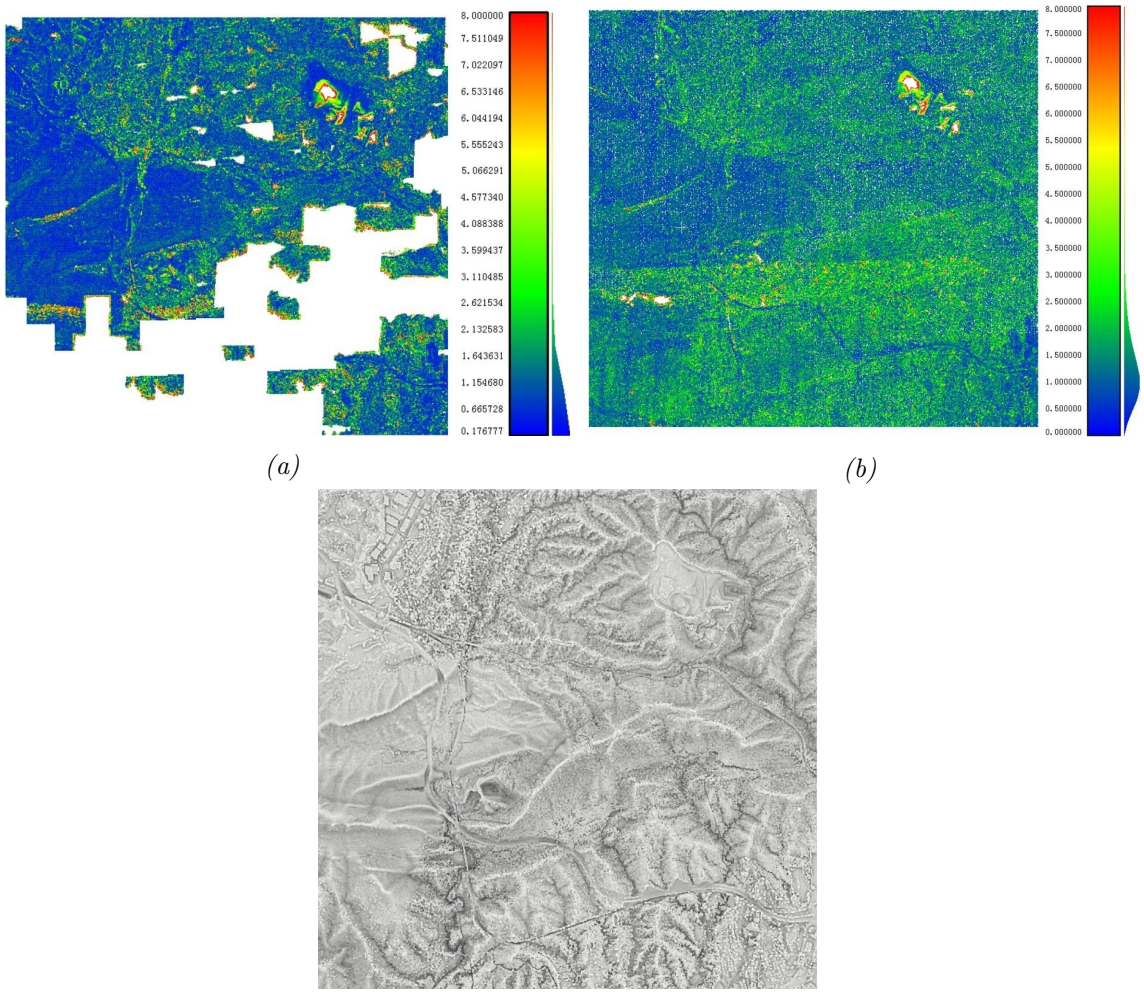


Figure 4.21: Vacarisses test site: (a) Height difference map of S2P pipeline (b) Height difference map of our pipeline (c) shaded LiDAR point cloud.

on the mountain. More outliers exist in S2P point cloud than tSGM point cloud. The tSGM point cloud is smoother than other point clouds. To do further investigation, we select the identical profile segments on all point clouds, which is presented as the red lines in the point cloud scenes. The comparison of the profiles extracted from different point clouds are depicted in Figure 4.22d. In the profile sketch, the x-axis presents the distance along the profile segment and the y-axis presents the height. The red line in the sketches represents the LiDAR point cloud, the blue line represents the tSGM point cloud and the green line represents the S2P point cloud. According to Figure 4.22d, the slope of the mountain is accurately reconstructed in the point cloud generated from both 3D reconstruction pipelines. For the point clouds generated from satellite stereo images, the saddle parts between the two peaks has larger errors because of the shadow. In both 3D reconstruction pipelines, the forests on the mountain are hard to reconstruct because of less texture. The tSGM point cloud is closer to the reference point cloud on the lowest peak, whereas the S2P point cloud has significant error.

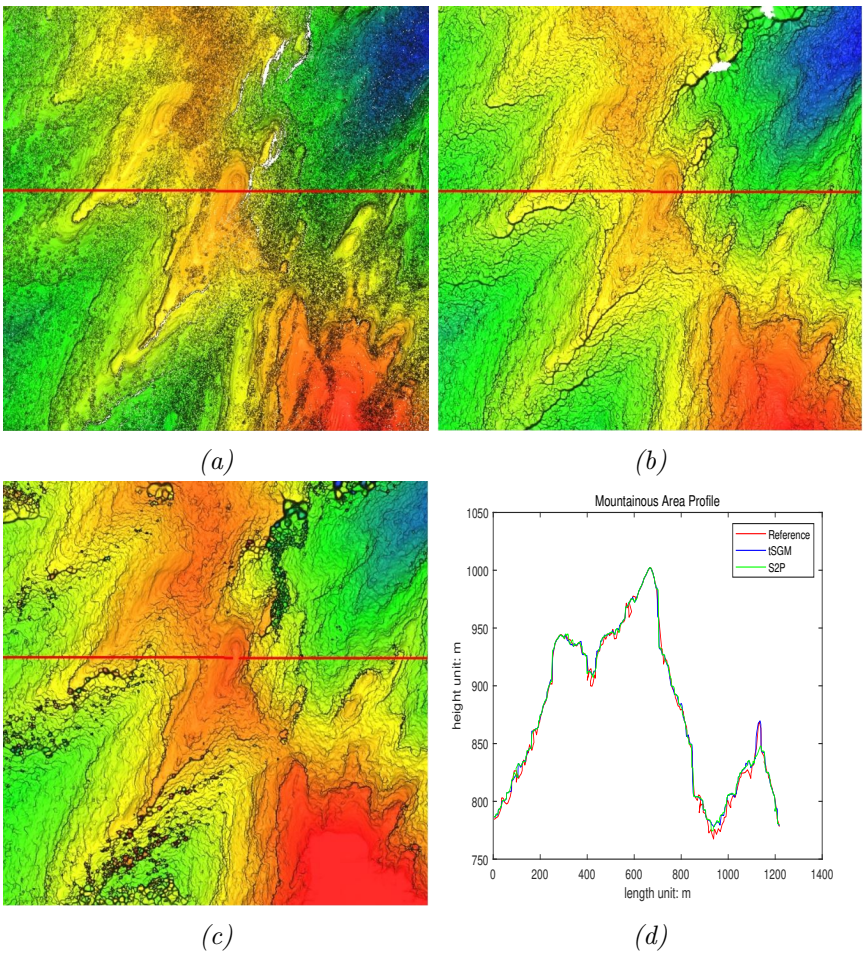


Figure 4.22: Mountainous area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison

The second ROI is an industrial area. As same as the first mountainous ROI, the Lidar point cloud, point clouds generated from tSGM and S2P pipeline are color-coded by the elevations and are

presented in Figure 4.23. In Figure 4.23b and 4.23c, the shape of the buildings are well reconstructed, but the edges of the buildings are not as sharp as the edges in the LiDAR point cloud. The identical segment across the industrial building is selected to extract the profiles from all the point clouds, which is shown as the red straight lines in the scenes of the point clouds. The comparison of the profiles is shown in Figure 4.23d. In the profile sketch, the red line presents the Lidar point cloud. The tSGM point cloud presented by blue line and the the S2P point cloud presented by green line fit the reference point cloud well at the right part of the profile, but both point clouds have noisy and inaccurate edges at the left part. The shadow cast by the building lead the inaccurate edges. For this building, S2P pipeline handles the shadow slightly better than our proposed pipeline.

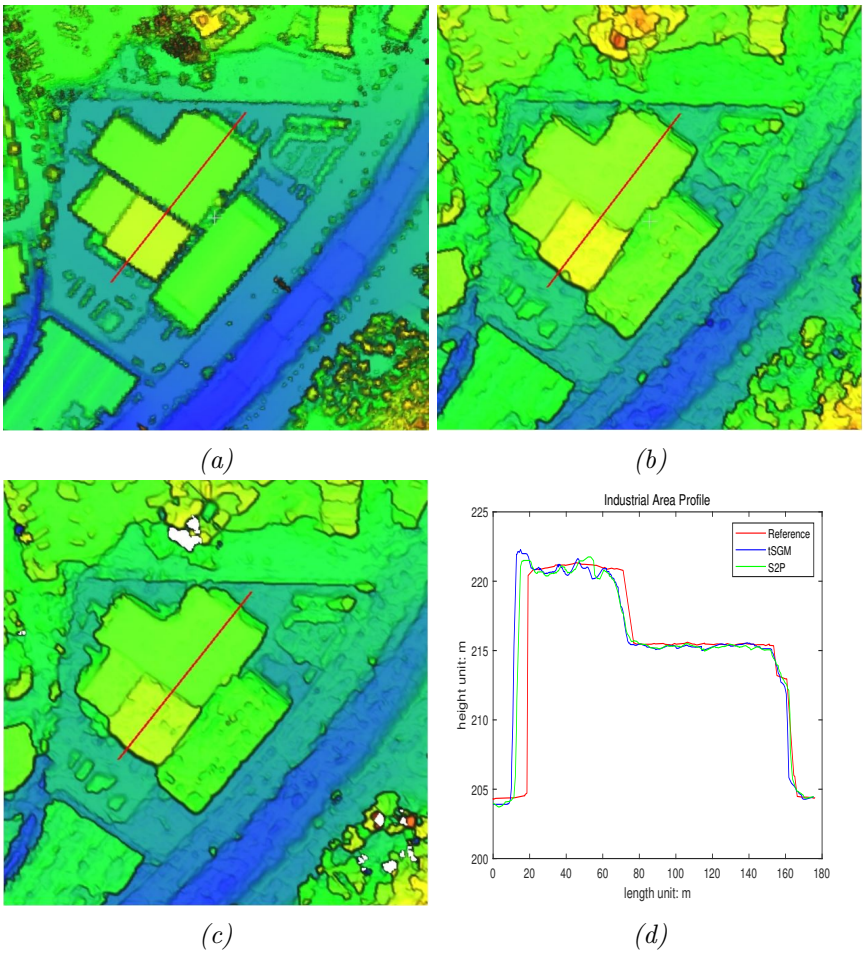


Figure 4.23: Industrial area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison

Figure 4.24 displays the third ROI, which is an intensive residential area. We color-code the Lidar point cloud, tSGM point cloud and S2P point cloud and select the identical segment for profile extraction (red lines) in Figure 4.24a, 4.24b and 4.24c. The point cloud generated from satellite data meets problems when the buildings stand close to each other. In 4.24b and 4.24c, the roads between the building blocks are not completely reconstructed. Figure 4.24d depicts every profiles. The profiles extracted from tSGM and S2P point cloud and the reference profile are fitted in most

parts. The outliers, which mainly occur in the area between two close standing buildings, are caused by the shadows.

According to the height difference maps and the extracted profiles of the ROIs, the point cloud generated by our proposed pipeline reconstructs the main features on the surface accurately, and it is competitive to the point clouds generated by S2P pipeline. Both point clouds generated with the satellite stereo images are suffering from the steep, vegetation and shadow. We also notice that only one stereo pair is applied to generate the point clouds and DSM in the three test sites. If more stereo pairs are involved into the procedure, the noise and erros can be improved.

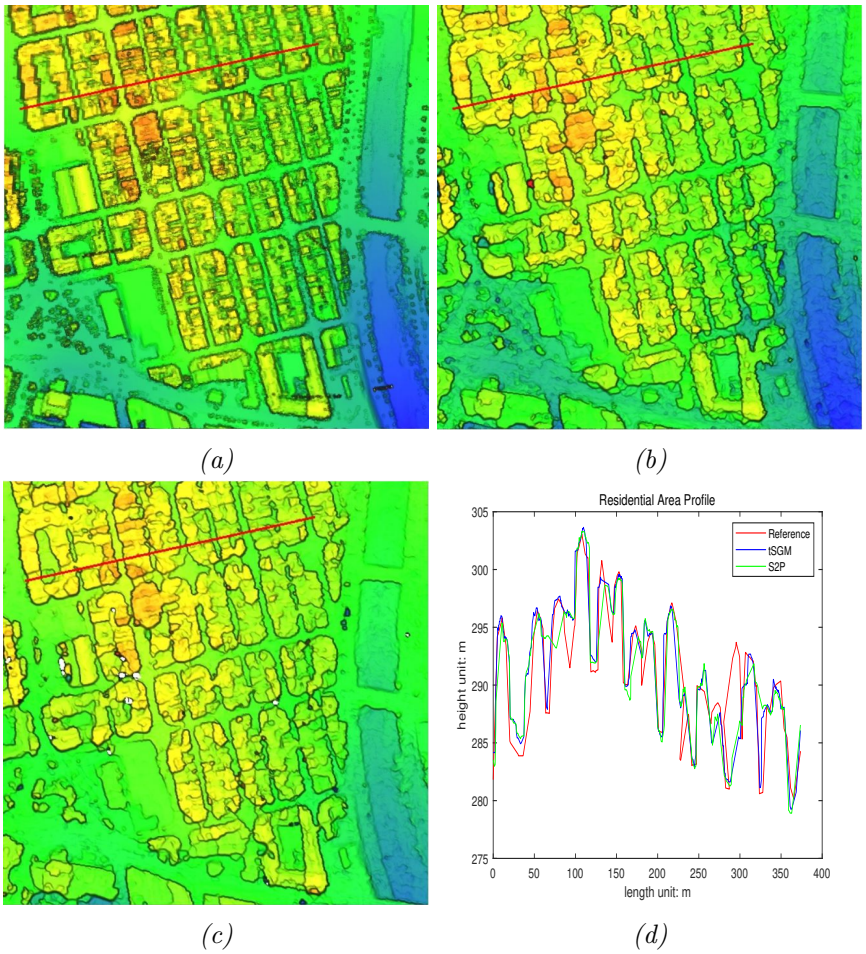


Figure 4.24: Residential area: (a) LiDAR point cloud (b) point cloud generated from our tSGM pipeline (c) point cloud generated from S2P pipeline (d) Profiles comparison

In order to conduct the quantitative evaluation, we follow the evaluation method proposed by the benchmark paper [d'Angelo and Reinartz, 2011]. The reference LiDAR point clouds and the DSMs generated from tSGM and S2P pipelines are at 1m GSD. Both tSGM and S2P DSMs are aligned to the Lidar point cloud surface before the evaluation, because no GCPs are used in the orientation procedure. The 3D translation is applied to minimize the median error of the height differences between the reference point clouds and the generated DSMs. For the full scene of the overlapping area, we measure the absolute height difference between the reference point clouds and the DSMs

	La mola			Terrassa			Vacarisses		
	tSGM	S2P	ISPRS	tSGM	S2P	ISPRS	tSGM	S2P	ISPRS
RMSE [m]	**2.98**	15.61	4.71	**2.61**	6.43	2.90	**2.78**	13.30	3.72
NMAD [m]	**0.84**	1.27	1.28	**0.58**	0.59	0.64	**0.85**	0.90	1.08
q68 [m]	**1.41**	2.42	1.79	0.97	1.03	**0.75**	**1.40**	1.62	1.41
q95 [m]	**6.55**	62.62	9.26	**5.29**	8.41	5.87	**6.17**	34.16	7.80

Table 4.12: Height difference evaluation of the ISPRS WorldView-1 benchmark

generated from WV-1 stereo images. The RMSE of the height difference is calculated to show the accuracy. The NMAD is applied to estimate the robust of the RMSE. The 68% quantile and 95% quantile of the absolute residuals are also taken into account to show the quality of the generated DSM, which are denoted by q68 and q95. The value of q68 and q95 are calculated such, that 68% and 95% of the residuals are $\leq$ q68 and q95, corresponding to $\pm 1{\cdot}\sigma$ and $\pm 3{\cdot}\sigma$ of a Gaussian error distribution. The results of the evaluation on tSGM and S2P point clouds are presented in Table 4.12. In Table 4.12, we also refer the evaluation result generated by ISPRS Working Group 4 of Commission I on "Geometric and Radiometric Modelling of Optical Spaceborne Sensors" [d'Angelo and Reinartz, 2011]. Note that their height difference distribution has a mean error close to zero, so that the standard deviation is identical with the RMSE.

For the La mola test site, the RMSE of the height difference in our 3D reconstruction pipeline is 3m, which is significantly lower than the 4.7m RMSE in the benchmark's pipeline. The RMSE of S2P's result is over 15m, because there are a lot of failure matched tiles in this test site. The NMAD of the height difference is 0.8m, the q68 is 1.4m and the q95 is 6.5m in our tSGM pipeline, which are the best of all three pipelines. The NMAD of S2P and ISPRS pipelines are very close. The q68 of S2P DSM is 1m larger than tSGM pipeline and 0.6m larger than the ISPRS pipeline. The ISPRS pipeline's result has q95 larger than 9m. The q95 of S2P pipeline is over 60m, which indicates the large outliers and errors in their reconstruction of this region.

As to the Terrassa test site, all the pipelines have better performance. Because there are more city areas and less mountainous areas in this test site. The proposed tSGM pipeline has the RMSE as 2.6m, which is less than the 2.9m RMSE of the ISPRS benchmark's pipeline and the 6.4m RMSE of S2P pipeline. The q95 of our pipeline is 0.6m lower than the ISPRS benchmark's pipeline and 3m lower than the S2P pipeline. Our tSGM pipeline's NMAD is very close to S2P pipeline and it is slightly better than the benchmark's pipeline. The q68 of ISPRS benchmark's pipeline is the best, which is 0.2m lower than the results of tSGM and S2P pipeline.

In the Vacarisses test site, the RMSE of the height difference of our pipeline is 1m lower than the benchmark's pipeline and 10m lower than S2P pipeline. Compared with the result of ISPRS benchmark's result, both DSMs generated from tSGM and S2P have 0.2m lower NMAD. The q68 of two pipelines is at an extremely close level. The q95 of tSGM DSM is 6.2m, which is 1.6m less than the DSMgenerated by the benchmark's pipeline. The q95 of S2P pipeline is more than 34m, because of the failed tiles.

According to the experimental results, our satellite 3D reconstruction pipeline can generate robust and accurate point clouds and DSMs. The tile-wise processing of S2P pipeline meets some problems in the reconstruction of the mountainous areas and misses a lot of terrain information.

The tSGM-based pipeline proposed in this work has advantages to the benchmark's pipeline and S2P pipelines in these three large-range test sites, especially in the mountainous areas.

4.4.2 3D Reconstruction for Munich Test Site

In the last section, we present the 3D reconstruction performance of the proposed tSGM pipeline on the datasets with one stereo pair. If more stereo images are involved in the reconstruction, the result may be more robust and accurate. To verify the quality of the MVS 3D reconstruction, here we present the experiments on the DLR's VHR satellite dataset for Munich. The tests for MVS case in the test sites of IARPA Multi-View Stereo benchmark are introduced in the following section.

There are four images in the DLR's VHR satellite dataset, which covers the downtown of Munich. By selecting different base images for the dense image matching, there can be as maximum as twelve possible stereo pair combinations for this dataset. In our experiments, all twelve stereo pairs are applied. The RPCs of these WV-2 images are refined by our relative compensation method. The images are then transformed into epipolar images via the image rectification procedure. The stereo pairs are densely matched separately and triangulated to the point clouds. We convert the point clouds from a geographic coordinate system to the UTM coordinate system. The point clouds are fused and the final DSM is generated. We show an oblique view of the fused point cloud in Figure 4.25. The point cloud is color-coded by the height values. In Figure 4.26, the fused DSM generated from our tSGM based pipeline at 0.5 m GSDis demonstrated.

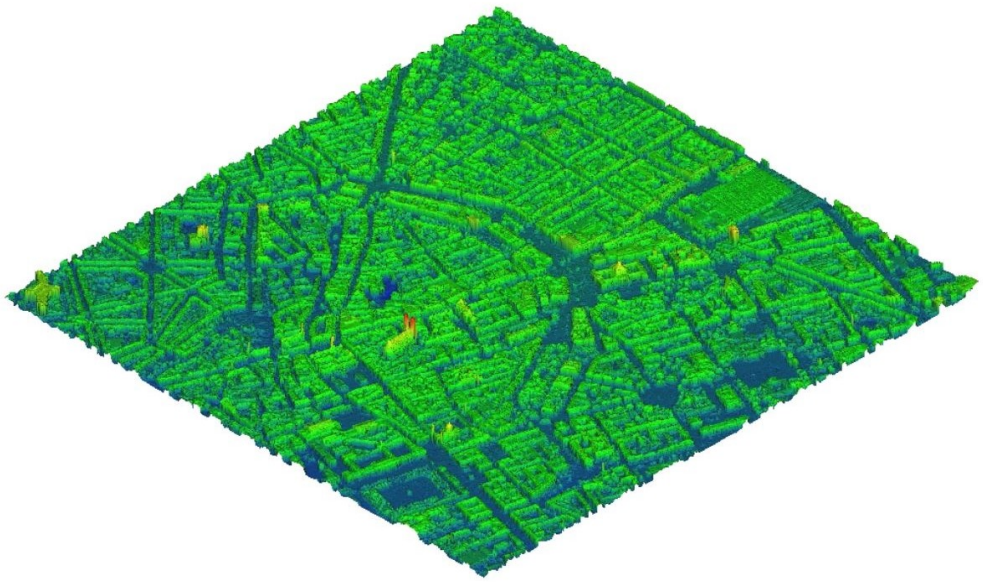


Figure 4.25: The fused point cloud of Munich test site

For the Munich test site, we can only access the DSM generated from the aerial MVS images at 0.1m GSD as a reference data. From DLR, we acquire a DSM at 0.5m GSD, which is generated via their SGM-based 3D reconstruction pipeline with the same data in 2016. The DSM generated by DLR's pipeline and the reference airborne DSM are also converted into point clouds. Applying the same stereo pairs and RPCs, we process the data via S2P pipeline and also generate the fused point cloud and DSM at 0.5m GSD. Because the covering region of the DSM generated from the airborne

Figure 4.26: The fused DSM generated via our tSGM-based pipeline with WorldView-2 MVS images

data has differences to the WV-2 MVS images. We only extract the overlapping areas between the airborne and spaceborne data for comparison. The DSMs generated with the satellite imagery are aligned to the surface of the reference airborne DSM with 3D translations. The shifting is done iteratively to minimize the median error of the height difference between the reference DSM and the spaceborne DSMs.

For the overlapping areas, we compute the height differences from three spaceborne point clouds to the reference airborne point cloud and show the results in Figure 4.27. The shaded reference DSM is displayed in Figure 4.27d. The point clouds generated from satellite MVS images are color-coded by the absolute height difference values. The scale bar is displayed at the right side of the point cloud. According to the pictures, the results of all pipelines have height differences less than 2m in most areas. The outliers are introduced by the object changing or vegetation. In this downtown area, the errors mainly come from the shadows cast by the buildings. We find that DLR's pipeline handles the shadow better than our tSGM and S2P pipelines and generates sharper edges.

Some ROIs are selected to show the reconstruction details. The first ROI is the area around the Frauenkirche in Munich center. For this subarea, the point clouds generated by three pipelines and the reference point cloud are shown in Figure 4.28. As shown in Figure 4.28a and 4.28b, the point clouds generated from the satellite MVS images have reconstructed the main features of the surface, but the edge of the buildings are relatively noisy if they are compared to the point cloud shown in Figure 4.28d. As to the point cloud of S2P pipeline, more outliers exist on the roof ridge and the tower of the church is failed to rebuild. It is visible that the point clouds generated from satellite data are suffering from the shadows and the trees close to the buildings. The identical profile line

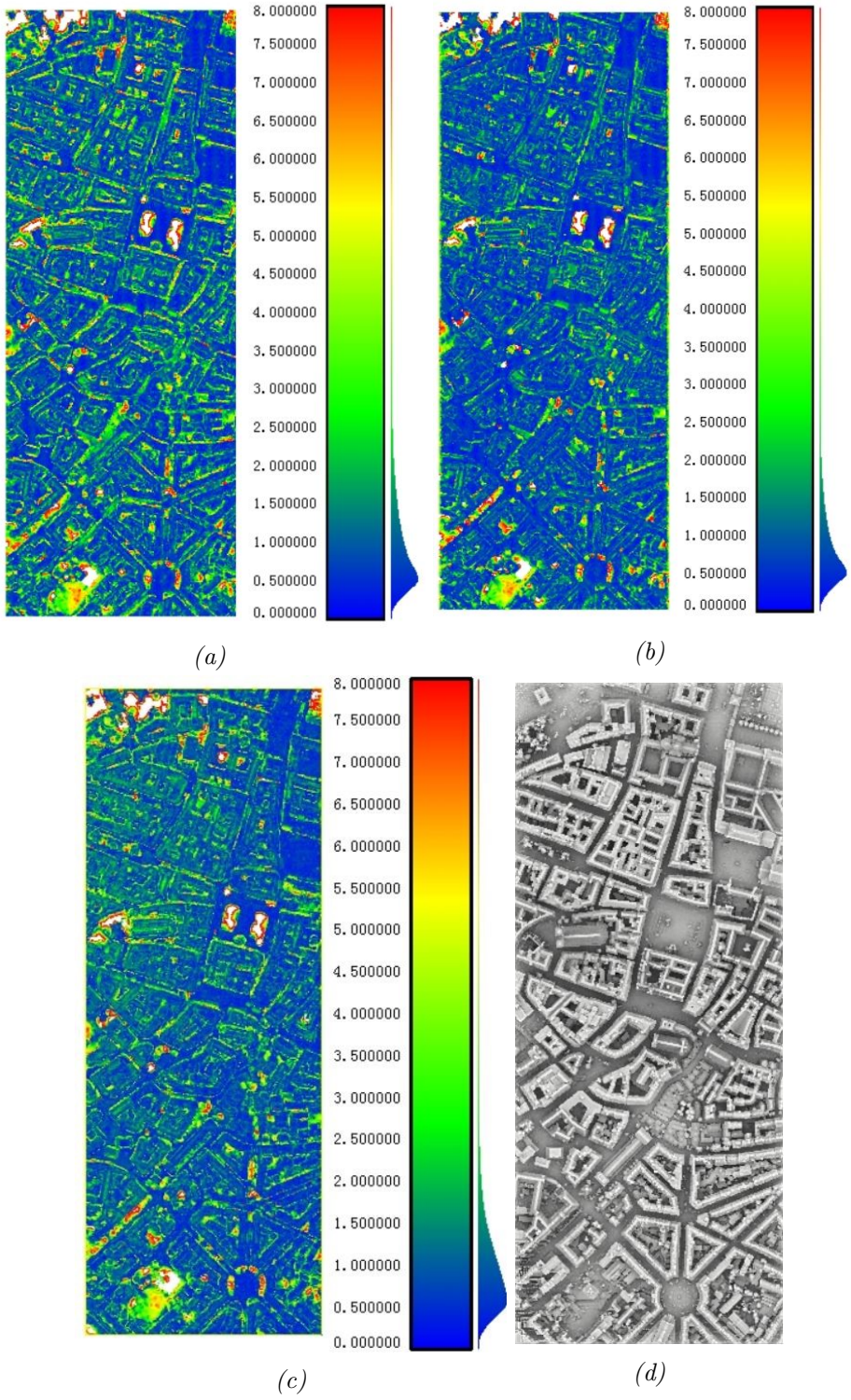
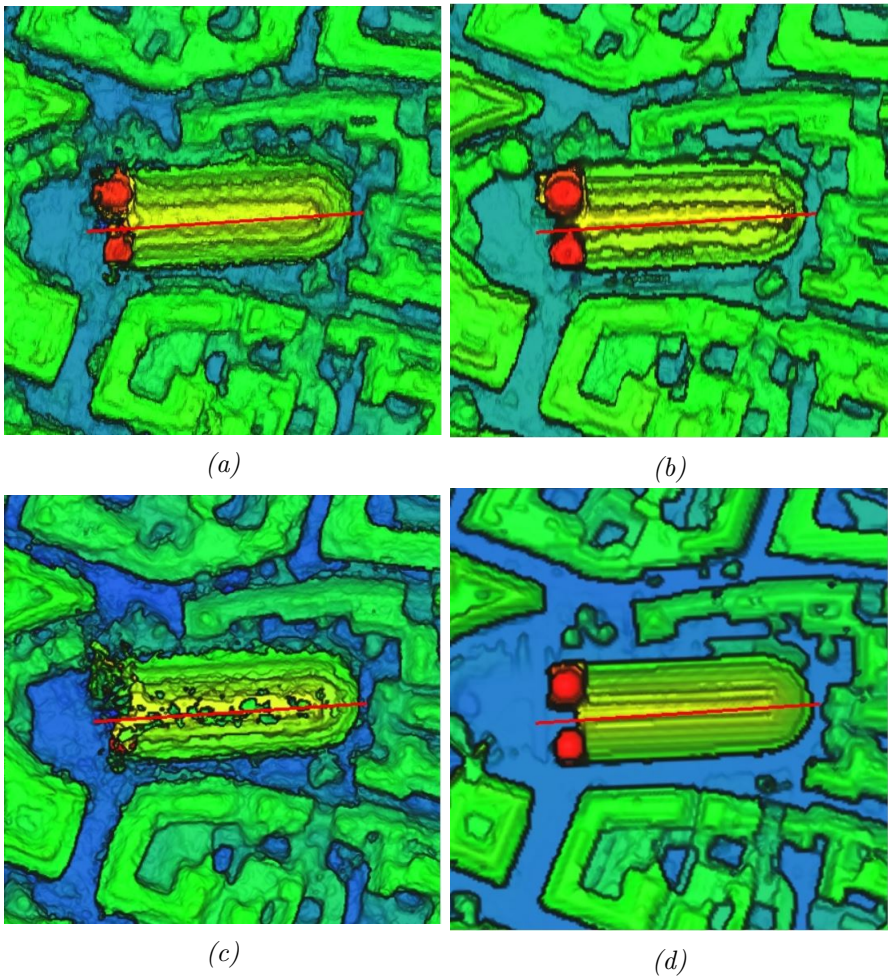


Figure 4.27: Height Difference between (a) the spaceborne-based point cloud generated via our tSGM pipeline (b) the spaceborne-based point cloud generated via DLR's pipeline (c) the spaceborne-based point cloud generated via S2P pipeline and (d) the reference airborne-based point cloud.

is selected for all the point clouds, which are presented as the red straight lines in 4.28a, 4.28b, 4.28c and 4.28d. The extracted height profiles are drawn in Figure 4.28e. In this figure, the red line depicts the height profile of the reference point cloud, the green line depicts the height profile of the DLR point cloud, the blue line depicts the height profile of the tSGM-based point cloud and the magenta line depicts the height profile of S2P point cloud. The x-axis represents the distance along the profile segment and the y-axis represents the height. We find, in the middle parts and the right parts along the profile, the point clouds generated from tSGM and DLR pipelines fit the reference point cloud well. At the tail of the church, the point cloud generated via DLR's pipeline has wrong high-rise parts and our generated point cloud has an inaccurate edge. The height profile of S2P pipeline has significant distances to the reference data because of the ourliers on the ridge. At the head of the profile, all point clouds generated with satellite data have tremendous errors like tens of meters. The shadow between the twin-tower of the church leads to the errors during the image matching.

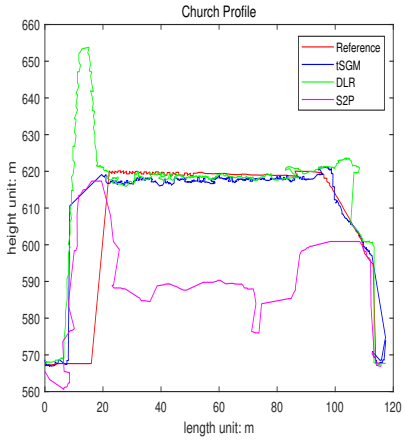
We select a building in Munich city center as an example and present the point clouds in Figure 4.29. The main structure of the buildings is reconstructed clearly by all the point clouds generated with WV-2 MVS images. The boundary between the buildings and streets have clearer results when there is no shadow. The reconstructed streets generated with satellite data meet troubles when the buildings is too close to trees or other buildings. The point cloud generated by DLR has better performance to control the effect of the shadows. As same as the former example, an identical profile segment is selected for all four point clouds. The height profiles are depicted in Figure 4.29e. Here the red line represents the reference point cloud, the blue line represents our tSGM point cloud, the green line represents DLR's point cloud and the magenta line represents the S2P point cloud. The x-axis of the sketch is the distance along the profile segment and the y-axis is the height. In most parts of the profile segment, the point clouds generated by satellite data are well fitted to the surface of the reference point cloud and the height differences are at sub-meter level. The profiles, which vary between 30 and 50m along the axis, have large height difference like several meters. Moreover, at the end of the profile segment, the point clouds generated from satellite MVS images have significant errors to the reference surface. The shadow cast by the building leads to these errors.

In oder to verify the improvement of MVS reconstruction, we also select one stereo pair and then generate a DSM from them. To analyze the reconstruction result quantitatively, we calculate the absolute height difference from all the DSMs to the reference airborne photogrammetric DSM. The RMSE is calculated to show the accuracy and the NMAD, the q68 and q95 are applied for the results' robustness estimation. Table 4.13 displays the statistic evaluation results. As the table shows, if MVS images are applied, the RMSE is decreased by 1m, the q68 are improved by 0.3m, the NMAD is decreased by 0.1m, and the q95 is reduced by 1m. Obviously, the result of MVS reconstruction is more accurate and robust than the result generated by only one stereo pair. Among the three MVS pipeline, the tSGM DSM has the best RMSE, which is slightly lower than DLR DSM and half meters lower than S2P DSM. The NMAD of DLR DSM is under 1m, whereas the NMAD of tSGM and s2P DSM is larger than 1m. DLR DSM has the lowest q68 as 1.5m, which is 0.2m less than S2P DSM and 0.3m less than tSGM DSM. The q95 of our DSM is 0.8m less than DLR's DSM and 1.6m less than S2P DSM. We can conclude that, in this Munich MVS test site, our tSGM-based 3D reconstruction pipeline achieves the same level of quality as DLR's reconstruction pipeline, and performs better than S2P pipeline. In general, our proposed pipeline is competitive to the state-of-the-art 3D reconstruction pipelines.



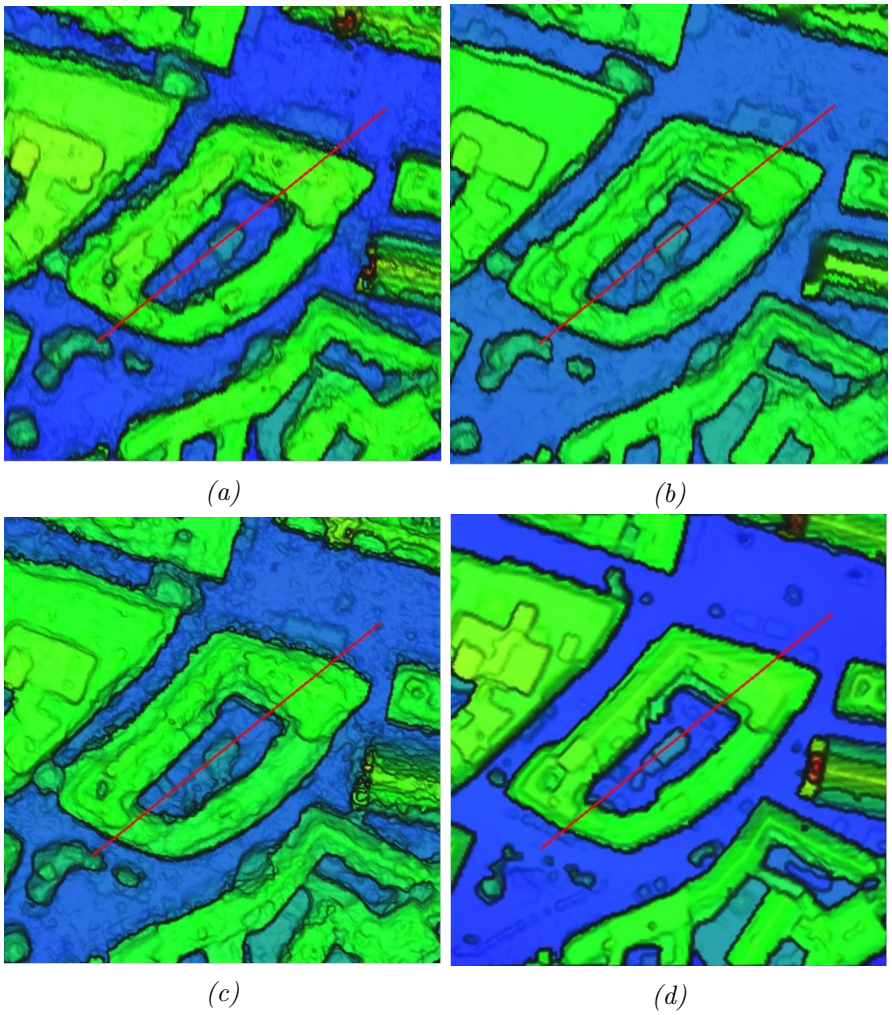
(a) *(b)*

(c) *(d)*



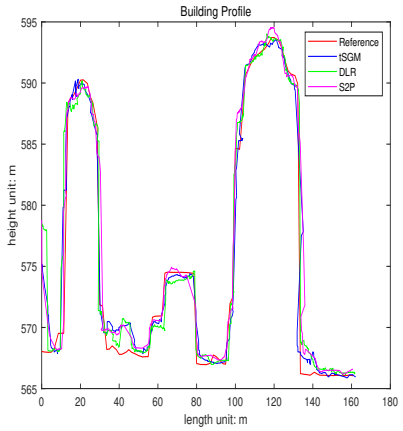
(e)

Figure 4.28: Reconstruction details of Frauenkirche Munich: (a) point cloud generated by tSGM pipeline (b) point cloud generated by DLR's pipeline (c) point cloud generated by S2P pipeline (d) Reference point cloud (e) Height profile.



(a) (b)

(c) (d)



(e)

Figure 4.29: Reconstruction details of a sample building in Munich: (a) point cloud generated by tSGM pipeline (b) point cloud generated by DLR's pipeline (c) point cloud generated by S2P pipeline (d) Reference point cloud (e) Height profile.

	RMSE [m]	NMAD [m]	q68 [m]	q95 [m]
tSGM One Stereo	5.678	1.277	2.116	13.768
tSGM MVS	**5.141**	1.116	1.841	**12.691**
DLR MVS	5.180	**0.909**	**1.492**	13.458
S2P MVS	5.682	1.018	1.705	14.302

Table 4.13: Evaluation of the Munich test site.

From the former analysis, we have already realized that the result generated from DLR's pipeline has better NMAD and q68. We present the area close to the Munich city hall in Figure 4.30 for further investigation. Here we present the point clouds generated by three satellite 3D reconstruction pipelines in Figure 4.30a, 4.30b and 4.30c. The reference point cloud is shown in Figure 4.30d and the snapshot of the corresponding area on Google Maps is displayed in Figure 4.30e. In Figure 4.30e, we can find two towers on the new city hall, one tower on the old city hall, one tower on the Heilig-Geist-Kirche and one tower on the St. Peter's Church. All towers are surrounded by red rectangles. We also mark the related areas on four point clouds. It is clear that all five towers are reconstructed on the reference point cloud generated from airborne MVS imagery. In Figure 4.30a, the small tower on the new Munich city hall is missing and the rest four towers are reconstructed. Only one tower on the old Munich city hall can be recognized in Figure 4.30b. The other four towers are not constructed in the point cloud generated from DLR's pipeline. According to Figure 4.30c, the towers of two churches and one tower on the new city hall are rebuilt, but the rest two towers are not reconstructed. As this example shows, DLR's pipeline does have sharper building edges but it seems over-smoothed and sacrifices some important structures on the surface. On the other hand, tSGM-based pipeline and S2P pipeline preserve more features and has better completeness. This can also explain why our tSGM pipeline has lower q95 and higher q68 than DLR's pipeline.

4.4.3 3D Reconstruction for the San Fernando Test Site

As described in 4.1, the IARPA MVS benchmark data provides 50 WV-3 images covering the area close to San Fernando. The large data redundancy brings thousands of different image pair combinations, so that we can not simply apply all the stereo pairs like in the Munich test site. To decrease the computing volume and make our processing more efficient, our image selection strategy is applied for this benchmark. Following this strategy, we keep the images having an incidence angle less than 35 degrees and good illuminating situations. We sort the images into summer and winter group, and order the images by dates. The stereo image pairs collected in close dates (for example collecting interval less than 30 days) are selected. Among the selected image pairs, the images that have intersection view angles less than 5 degrees or larger than 35 degrees are eliminated. We apply the image selection for all three test sites here. At last, 748 stereo pairs are selected as input data in test site 1. In test site 2 and 3, 394 and 484 stereo pairs are the winners. For each test site, we manually select the tie points for all the images and correct the RPCs via our relative RPC compensation method. The epipolar stereo images are generated by our modified piece-wise epipolar resampling strategy. The stereo images are separately matched by the tSGM algorithm. Using the disparity maps generated by the matching module, the point clouds are triangulated. These point clouds have already been aligned to the virtual surface defined by our selected tie

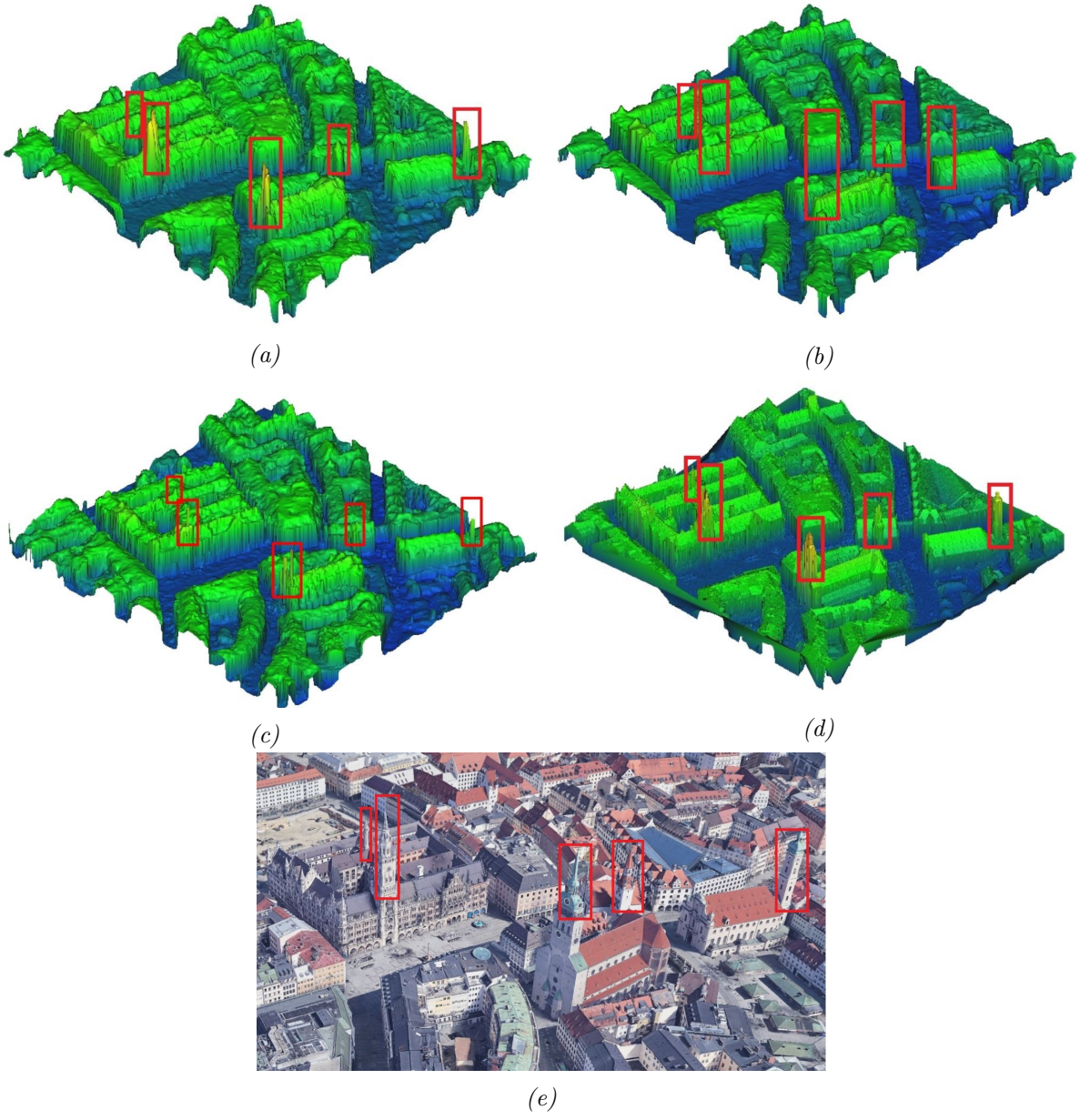


Figure 4.30: Munich city hall area in (a) our tSGM generated point cloud (b) DLR's point cloud (c) S2P point cloud (d) reference point cloud (e) Google Earth

points and corrected RPCs. With hundreds of point clouds, we fuse them to the final point cloud and DSM in the UTM coordinate system.

But more involved stereo pairs do not always bring better fusion results. To investigate the proper number of the input point clouds for the fusion step, the point clouds are first ranked according to their quality. According to the benchmark paper [Bosch et al., 2016], the completeness can be applied to represent the quality of the reconstructed point clouds. In their work, the completeness is referred to the percentage of the points that have height differences to the ground truth of less than 1m. For every point clouds, we calculate their height difference to the reference DSM. The point clouds are first moved from our defined virtual surface to the true surface of the reference DSM by 3D translation shifts. Iteratively, the shifts are adjusted until the median error of the height differences is minimized. In this way, we can minimize the bias between the point cloud and the ground truth, which is caused by our relative orientation procedure. When the height differences to the ground truth are measured, the completeness of every point clouds is computed. The point clouds are then ranked descending by their completeness. For testing, we apply different numbers of the top-ranked point clouds for the DSM fusion. For each fused DSM, the completeness and the RMSE of the height differences are computed to evaluate the accuracy of the fused DSMs.

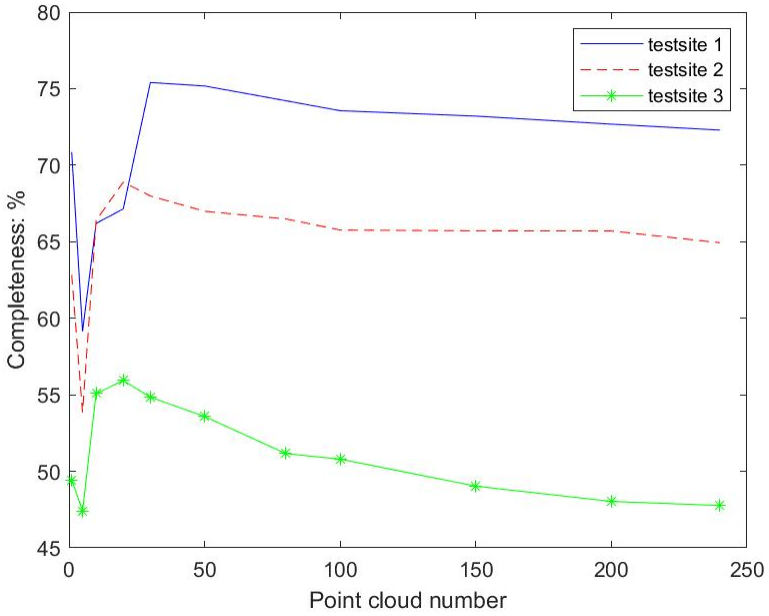


Figure 4.31: Relation between the number of point clouds and completeness

Figure 4.31 demonstrates the completeness as a function of the number of the input point clouds for the three test sites. In Figure 4.31, the blue solid line represents the result of test site 1, the red dash line represents the result of test site 2 and the green dot line represents the result of test site 3. According to the figure, we note that the completeness is decreased when the number of point clouds used for fusion is too low. Then along with the increasing number of the point clouds used, the completeness keeps increasing until it reaches the peak. For test site 1, the completeness has the highest value of 75% when the number of the input point clouds is about 30. For test site 2, the completeness reaches a peak of 68% when about 25 point clouds are applied. Finally, for test

site 3, the best completeness is 56% with a corresponding point cloud count of 25. Over the peak, the completeness becomes lower if more point clouds are fed to the fusion step. We also notice that the completeness of test site 1 is the highest of the three areas, and that test site 3 has the lowest completeness. Moreover, the completeness is decreased more significantly when too many point clouds are applied in the DSM fusion of test site 3. The reason is that test site 1 has more flat field areas, but there are more residential areas in test site 2 and 3. In particular, several high-rise buildings exist in test site 3, which lead to larger shadow areas in the image scenes. The high-rise buildings reconstructed from different stereo pairs may have tremendous height differences in the boundary areas of the buildings. The dense residential areas and the high-rise buildings lead to more loss of the completeness.

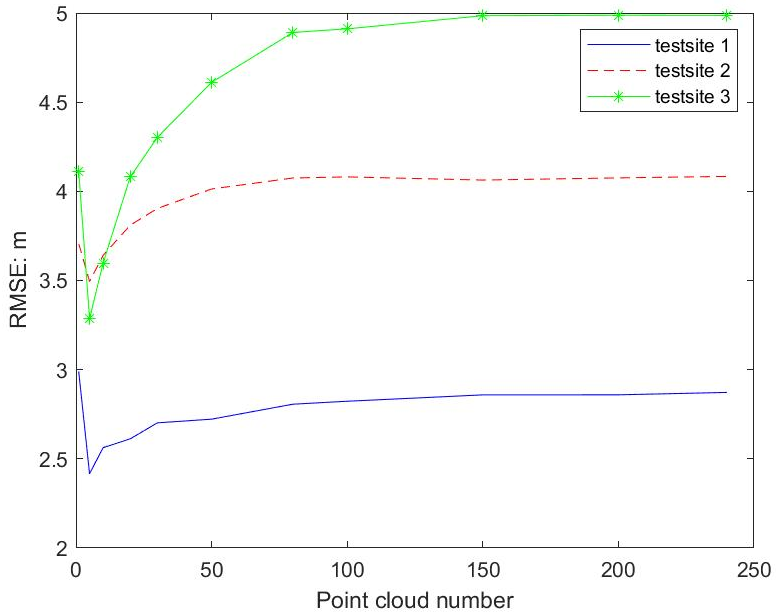
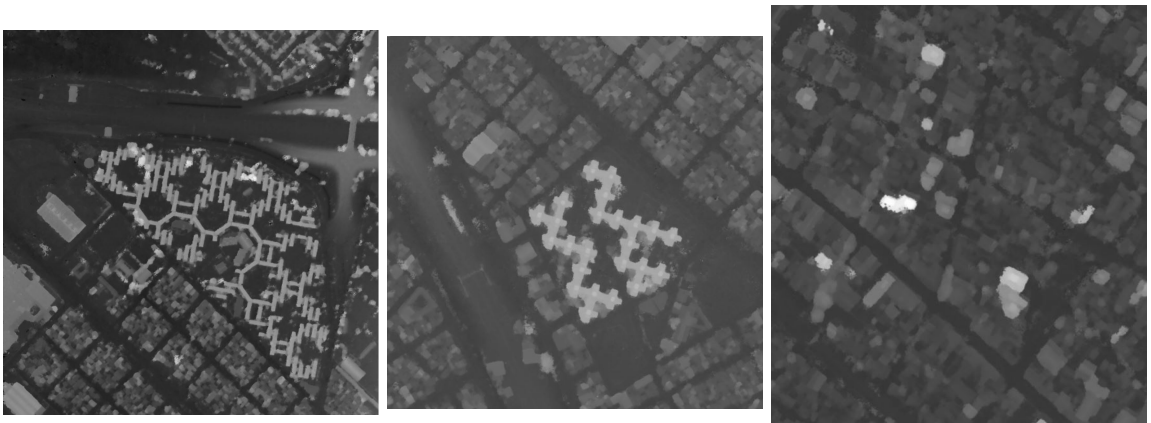


Figure 4.32: Relation between the number of point clouds and RMSE

In Figure 4.32, the line graph presents the RMSE as a function of the number of the involved point clouds in the three test sites. Like Figure 4.31, the blue line represents the RMSEs of test site 1, the red line represents test site 2 and the green line represents test site 3. By observing the relations presented in the figure, the RMSE is decreasing when the number of fused point clouds is small at the beginning. Then the RMSE increases if more point clouds are applied. The accuracy is reduced because more errors are introduced by some lower quality image pairs. Test site 1 has the best RMSE, then test site 2, and test site 3 has the largest height differences to the ground truth. As explained before, more dense residential areas and high-rise buildings exist in test site 3, which reduces the accuracy of the reconstruction.

Considering both the completeness and the RMSE, we select the number of point clouds which can provide the best completeness while having a relatively small RMSE. Therefore, the number of the applied point clouds for the DSM fusion is 30 for test site 1. In our case, the proper number for test site 2 is 20 point clouds. We select 20 point clouds as the proper number to generate the final fused point cloud and DSM for test site 3. By checking the selected point clouds for the final fusion

step, we find that most of these point clouds are generated from stereo images collected within close date and the intersection angles of most stereo pairs are between 10 and 30 degrees. This proved that the image selection strategy is effective, but the intersection angles of the stereo pairs can be limited from 10 to 30 degree for this benchmark data. The final fused DSMs of the three test sites, which are generated by the selected number of the point clouds, are displayed in Figure 4.33. The oblique views of the fused point clouds are displayed in Figure 4.34. The color of the point clouds is coded by the elevations.

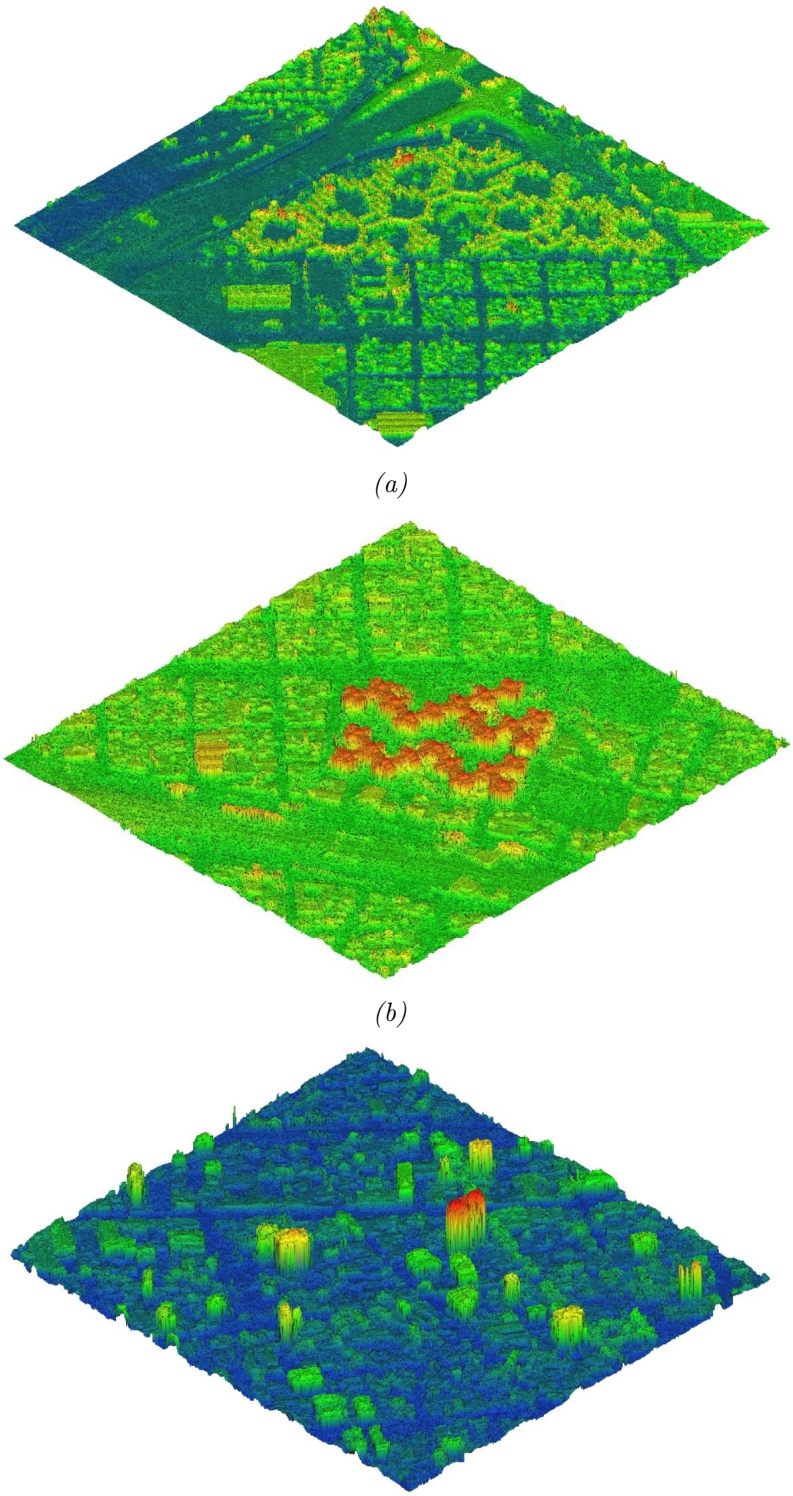


(a) *(b)* *(c)*

Figure 4.33: The MVS reconstructed DSM of San Fernando (a) test site 1 (b) test site 2 (c) test site 3.

Along with the benchmark data, JHU/APL also provides their 3D reconstruction algorithms for comparison. The readers who are interested to their algorithms can view the details from [Bosch et al., 2016]. We generate the fused point clouds and DSMs with their algorithms with our selected input data in all three test sites. The S2P pipeline is also applied to generate fused point clouds and DSMs with our selected stereo pairs and the related RPCs in these test sites. To verify the accuracy of the 3D reconstruction, we calculate the height differences from point clouds generated from satellite MVS images to the reference Lidar point cloud. All point clouds are first aligned to the ground truth surface first with 3D translation shifts, which can minimize the median value of height differences. The result of the height differences to the reference data from the three pipeline's point clouds are shown in Figure 4.35, 4.36 and 4.37. The shaded point clouds of the reference LiDAR data are displayed in Figure 4.35d, 4.36d and 4.37d. Figure 4.35a, 4.36a and 4.37a present the point clouds generated via our proposed pipeline. Figure 4.35b, 4.36b and 4.37b present the point clouds generated via the pipeline of JHU/APL. The height difference map of S2P point clouds are displayed in Figure 4.35c, 4.36c and 4.37c. The point clouds are color-coded according to the value of the height differences.

In test site 1, all pipeline's reconstructed point clouds have height differences less than 1m in most areas (the blue parts). The boundaries of the building and vegetation lead to higher differences. Comparing to JHU/APL's pipeline,our tSGM pipeline and S2P pipeline have sharper and more accurate edges of the building. The point cloud generated by S2P pipeline has more points with height difference less than 0.5m but also more points with height difference more than 5m than the point cloud of tSGM pipeline.



(a)

(b)

(c)

Figure 4.34: The fused point clouds of San Fernando (a) test site 1 (b) test site 2 (c) test site 3.

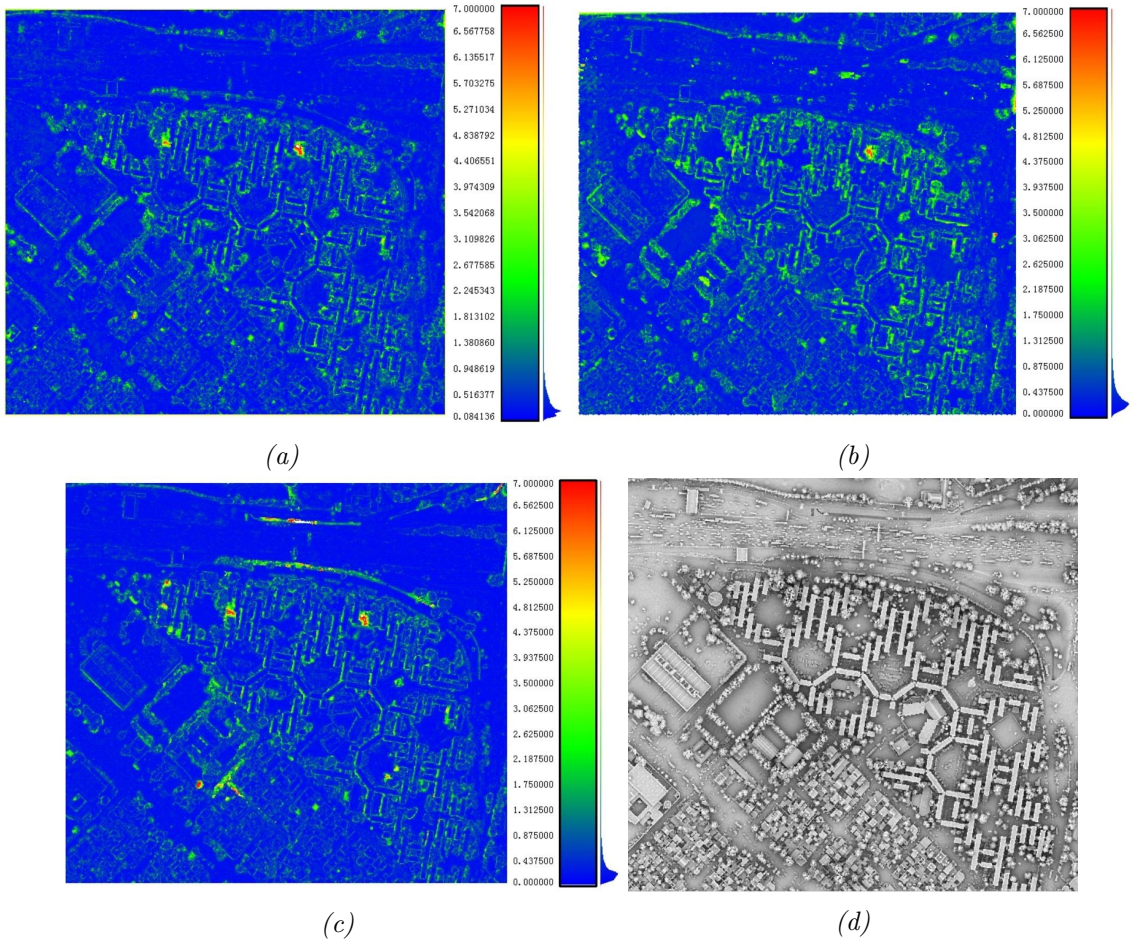


Figure 4.35: Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 1.

The point clouds generated from all three pipelines suffer a lot from the trees planting close to the buildings in test site 2. But for all reconstruction results, the major areas have height differences below 1m. According to Figure 4.36a, 4.36b and 4.36c, the point cloud generated by JHU/APL's pipeline is stronger affected by the shadows and vegetation than our pipeline. The proposed tSGM reconstructs the highway in the left-bottom corner best.

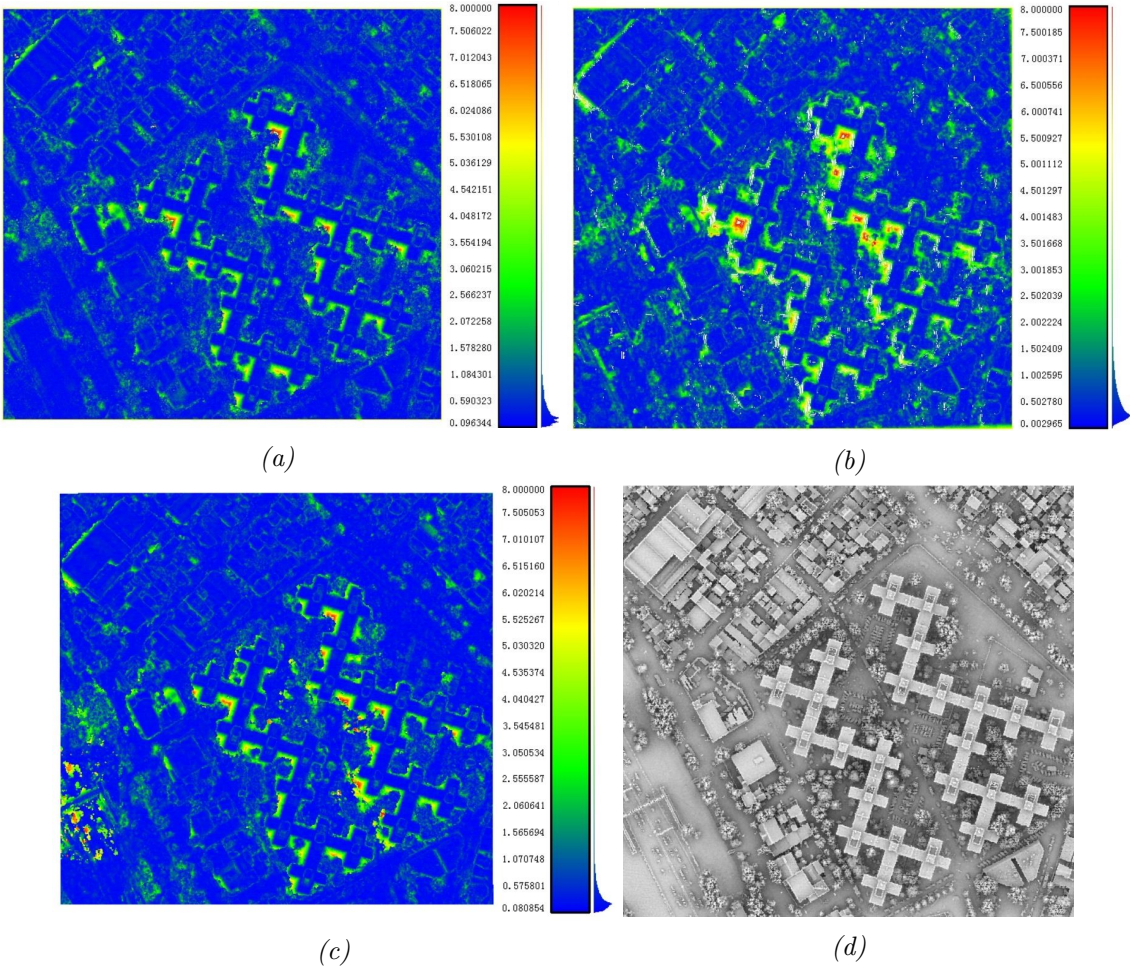


Figure 4.36: Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 2.

In the middle of test site 3, the high-rise building is a challenge for the 3D reconstruction with satellite MVS imagery. Because the building cast large range shadows and cause difficulties to rebuild sharp and clear edges of it. In the point cloud reconstructed via our tSGM pipeline, the high-rise building is reconstructed but there are significant errors in the building edges. The red parts in the middle of Figure 4.37b and 4.37c means the errors are larger than 8m. In the point cloud generated from JHU/APL's pipeline and S2P pipeline, the area contains the high-rise building is not correctly reconstructed. In most areas of this test site, all point clouds generated with satellite

data have less than 2m height differences to the ground truth. The tSGM point cloud has better performance on the reconstruction of the shadow areas and the vegetation.

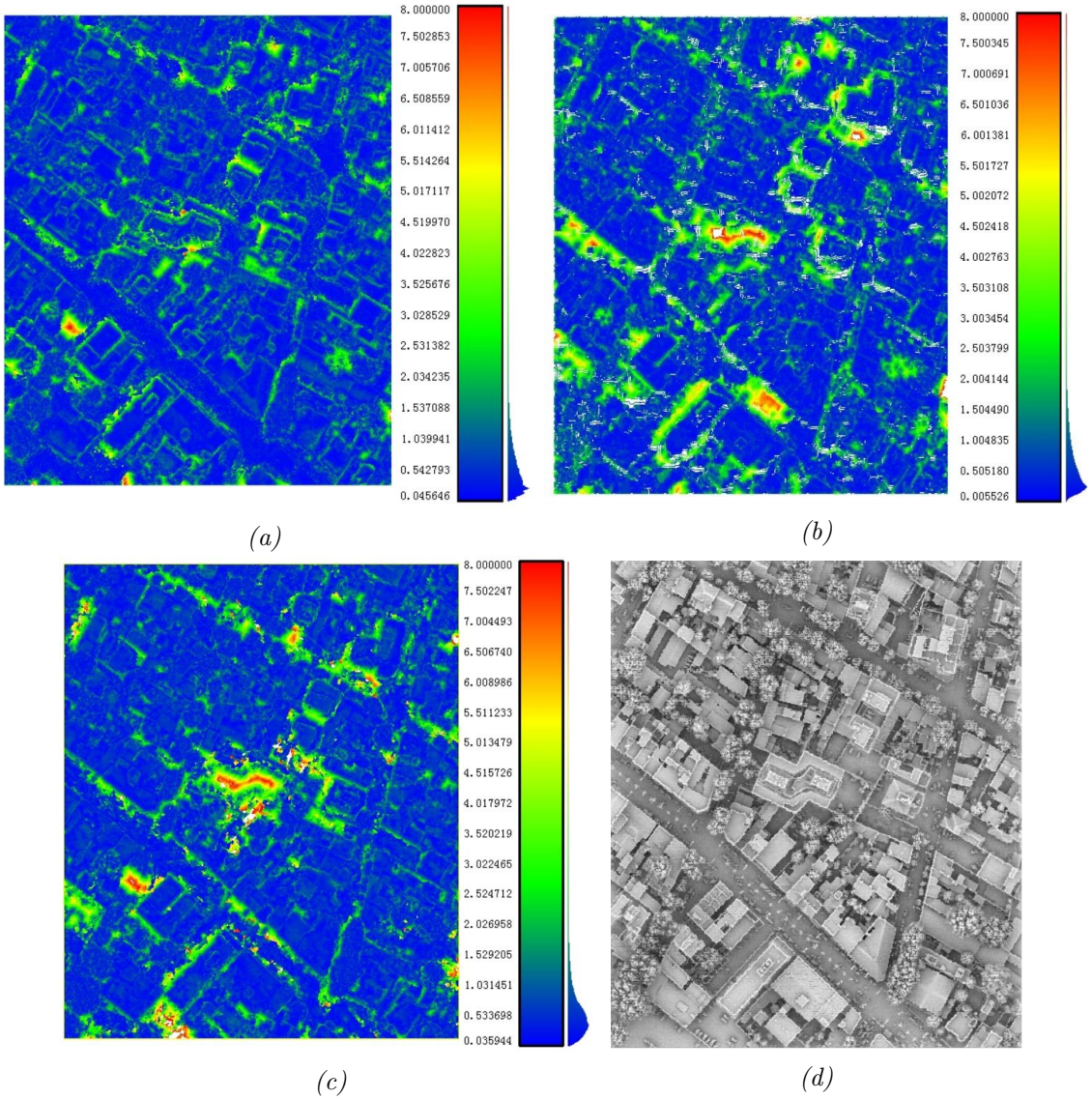


Figure 4.37: Height difference from (a) the point cloud generated via tSGM pipeline (b) the point cloud generated via JHU/APL's pipeline (c) the point cloud generated via S2P pipeline to (d) reference Lidar point cloud in San Fernando test site 3.

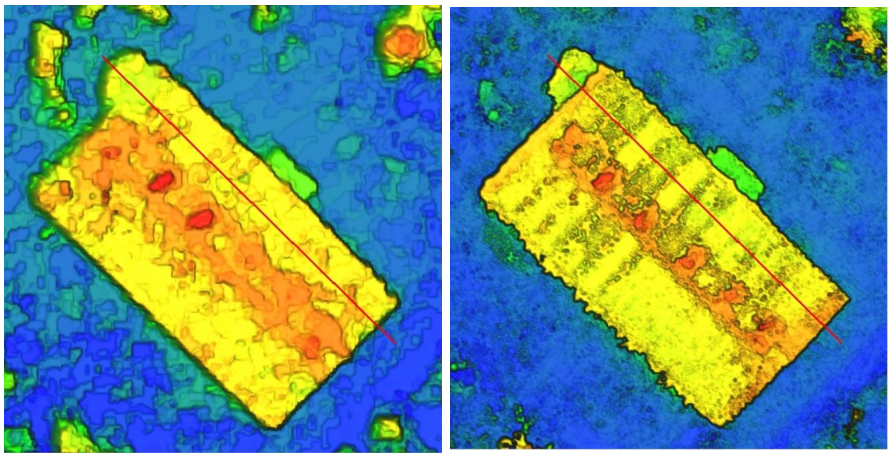
To show the details of the reconstructed point clouds and to conduct some further analysis, several subareas are extracted from the three test sites as ROIs. The first ROI is an isolated large warehouse and it is demonstrated in Figure 4.38. The point cloud generated by JHU/APL's pipeline is shown in Figure 4.38a, the point cloud generated by our proposed pipeline is shown in Figure 4.38b and the S2P point cloud is shown in Figure 4.38c. The reference LiDAR point cloud is displayed in Figure 4.38d. All point clouds are color-coded by the elevation. The building's edge in 4.38c is straighter than the edges in another two pipeline's results. The result reconstructed by

tSGM and S2P pipelines have more details on the roof of the warehouse. The S2P point cloud is smoother than the other point clouds. We cut an identical vertical profile for these four point clouds. The profile segment is shown as the red lines in every point cloud scenes. The profiles are depicted in Figure 4.38e. The x-axis means the distance along the profile segment and the y-axis means the height. The red line represents the profile line of the reference point cloud, the blue line represents the profile line of the tSGM point cloud, the green line represents the JHU/APL point cloud's profile and the magenta lines show the profiles of the S2P point cloud. On the roof of the building, All the reconstructed result generated from satellite MVS are very noisy. The S2P and tSGM pipeline generate point clouds with sharper and more accurate building edges.

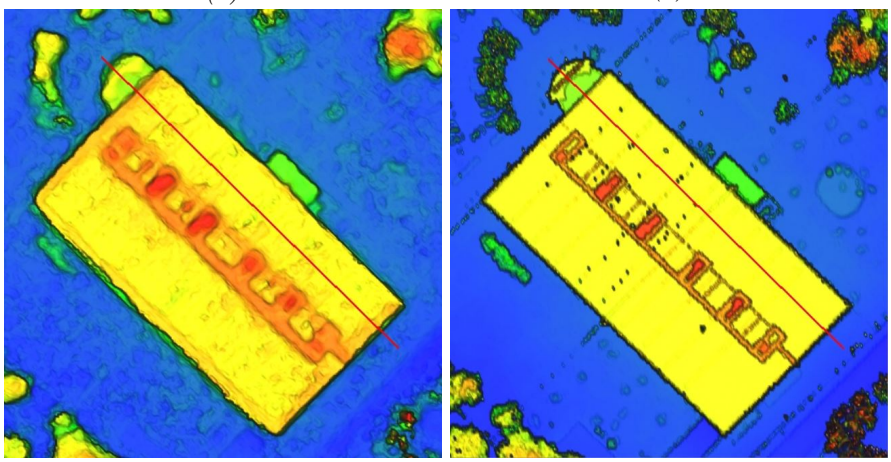
In Figure 4.39, we show an area which have some connected buildings surrounded by trees. The reference data is shown in Figure 4.39d. The point clouds generated from JHU/APL's pipeline, tSGM pipeline and S2P pipeline are separately presented in Figure 4.39a, 4.39b and 4.39c. The point clouds are color-coded by the heights. According to the figures, the upper and right boundaries of the buildings are reconstructed more clearly in the three point clouds generated from the MVS satellite data, because they are less affected by the shadows and trees. We can observe that the boundaries of buildings are hard to distinguish if the trees are planted too close. Again, a profile segment is selected for the point clouds. The profile sketch is shown in Figure 4.39e. The x-axis represents the distance along the profile segment and the y-axis represents the height. The profile of the reference point cloud is presented by the red line, the profile of the tSGM point cloud is presented by the blue line, the profile of JHU/APL point cloud is presented by the green line and the S2P point cloud is presented as the magenta line. As we can see in the beginning of the profile lines, the upper edge of the building in tSGM point cloud fits the reference data best, and the JHU/APL's point cloud has significant planar errors for the edge. As to the bottom edge of the building, All point clouds generated with satellite data have tremendous errors because of the influence of the shadows and the trees.

The next ROI contains a high-rise building, which is displayed in Figure 4.40. The point clouds generated by JHU/APL, tSGM and S2P pipelines are shown in Figure 4.40a, 4.40b and 4.40c. The LiDAR point cloud is shown in Figure 4.40d. All point clouds are colored according to the heights. The high-rise building is not completely reconstructed in Figure 4.40a and 4.40c. In tSGM result, the building is reconstructed but with noisy edges. The shadows lead to the information loss in the southern part of the building. A profile on the roof of the high-rise building is selected to draw the sketch shown in Figure 4.40e. The red, blue, green and magenta lines represent the profiles cut from LiDAR, tSGM, JHU/APL's and S2P point clouds. The x-axis and y-axis represent the distance along the profile and the height. Comparing the red and blue lines, our point cloud has rebuilt the basic shape of the roof. The green and magenta profile lines also show the incomplete reconstruction in JHU/APL's and S2P point clouds.

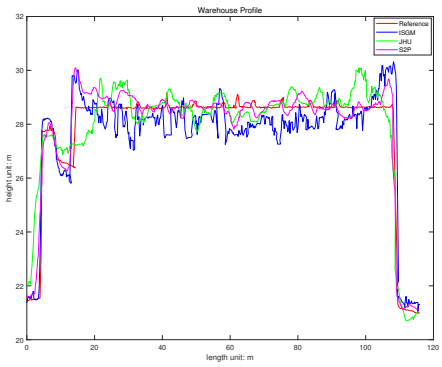
The last ROI selected by us is an area which contains many low-rise but intensive residential buildings. Figure 4.41a to 4.41d display the point clouds color-coded by heights of this sub-area. The fused point clouds of all three pipelines have poor performance because the buildings are too close, so that the shadows of the buildings are often casting on the nearby buildings. The reconstructed buildings are connected to each other and have totally blurred boundaries. The profile segment is selected and drawn as the red lines on the point clouds. The sketches of the profiles are displayed in Figure 4.41e. The red lines represent the contour of the LiDAR point cloud. It is observed that the buildings are separately reconstructed. The blue line, green line and magenta line present the profile of the point clouds generated by tSGM pipeline, JHU/APL's pipeline and S2P pipeline separately.



(a) *(b)*



(c) *(d)*



(e)

Figure 4.38: Reconstruction details of the warehouse: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile.

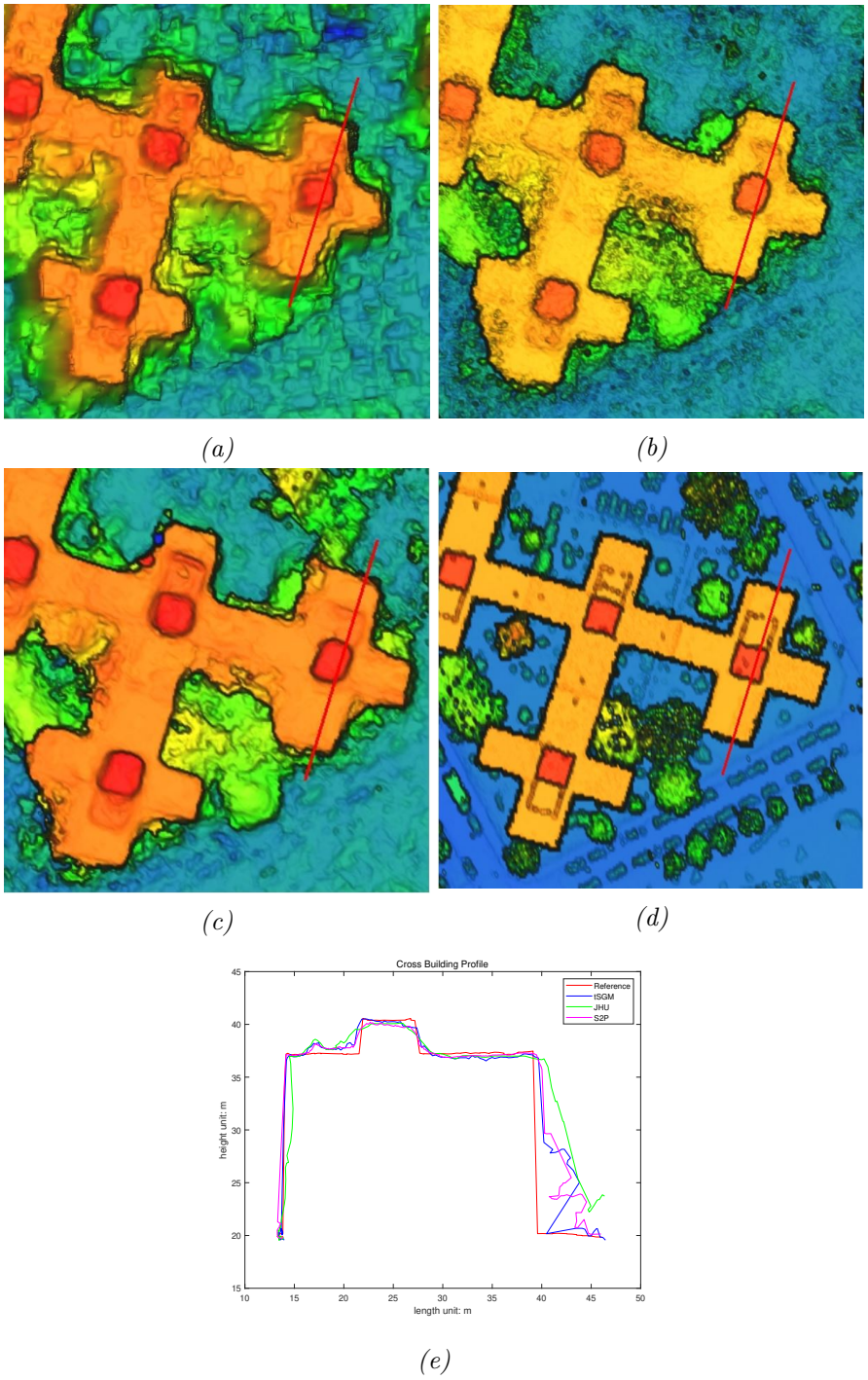


Figure 4.39: Reconstruction details of building close to trees: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile.

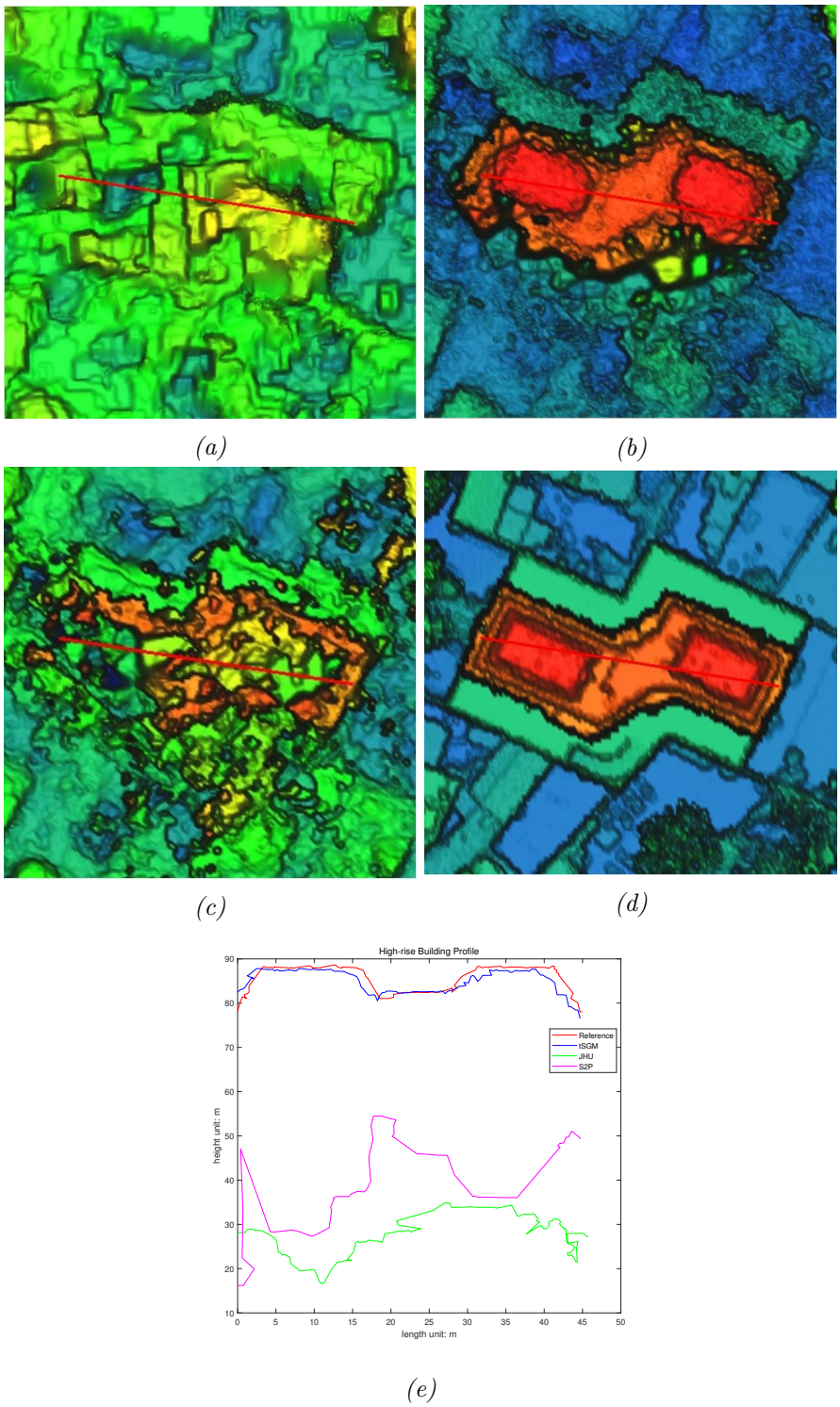


Figure 4.40: Reconstruction details of high-rise building: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile.

	Test site 1		
	tSGM	JHU/APL	S2P
Completeness (<1m) [%]	75.399	72.357	**77.038**
RMSE [m]	**2.702**	2.712	2.709
median [m]	0.320	0.322	**0.218**
q68 [m]	0.890	0.692	**0.536**
q95 [m]	**3.380**	6.762	6.806

Table 4.14: Height difference evaluation of the IARPA MVS benchmark test site 1

	Test site 2		
	tSGM	JHU/APL	S2P
Completeness (<1m) [%]	**68.886**	59.797	67.794
RMSE [m]	**3.810**	3.970	4.310
median [m]	**0.390**	0.592	**0.390**
q68 [m]	1.390	1.382	**1.010**
q95 [m]	**5.210**	10.327	11.520

Table 4.15: Height difference evaluation of the IARPA MVS benchmark test site 2

It is clear that these intensive standing buildings are very nosiy and are connected with each other in the point clouds generated from the MVS satellite images.

To evaluate the quality of our fused DSMs quantitatively, a comparison is made between the fused DSMs generated from three pipelines and the reference LiDAR DSM for all three test sites. The RMSE of the height difference and completeness of the results are computed to check the accuracy. We keep using the percentage of the points with less than 1m absolute height differences as the completeness. Moreover, we computed the median error, q68 andq95 to evaluate the robustness of the fused DSMs.

The statistic evaluation results of test site 1 are illustrated in Table 4.14. The RMSEs of the height differences of all fused DSMs are at the same level as 2.7m. The median errors of S2P DSM is 0.22m, which is 0.1m lower than the other two pipeline's results. The S2P pipeline's fused DSM has a completeness over 77%, which is 2% higher than tSGM DSM and 5% higher than JHU/APL DSM. The q95 of tSGM result is the best and it is 3m less than JHU/APL's and S2P DSM. The q68 of S2P fused DSM is only 0.5m, that is 0.35m better than tSGM dsm and 0.15m better than JHU/APL's DSM.

For test site 2, the evaluation results are shown in Table 4.15. The RMSE of our tSGM DSM is the lowest among three pipeline's results. The median error of tSGM result is as same as S2P DSM, and they are 0.2m lower than JHU/APL's DSM. The q95 of tSGM DSM has significant advantage to the JHU/APL's and S2P fused DSM. The DSM generated from our proposed pipeline has q95 as 5m, while q95s the other two pipeline's DSM are larger than 10m. The q68 of S2P DSM is the lowest like 1m. The other two pipelines have close q68 values as 1.4m. The completeness of tSGM DSM is 68.9%, which is the best of three DSMs. The completeness of S2P DSM is 1% lower and JHU/APL's DSM falls 9% behind.

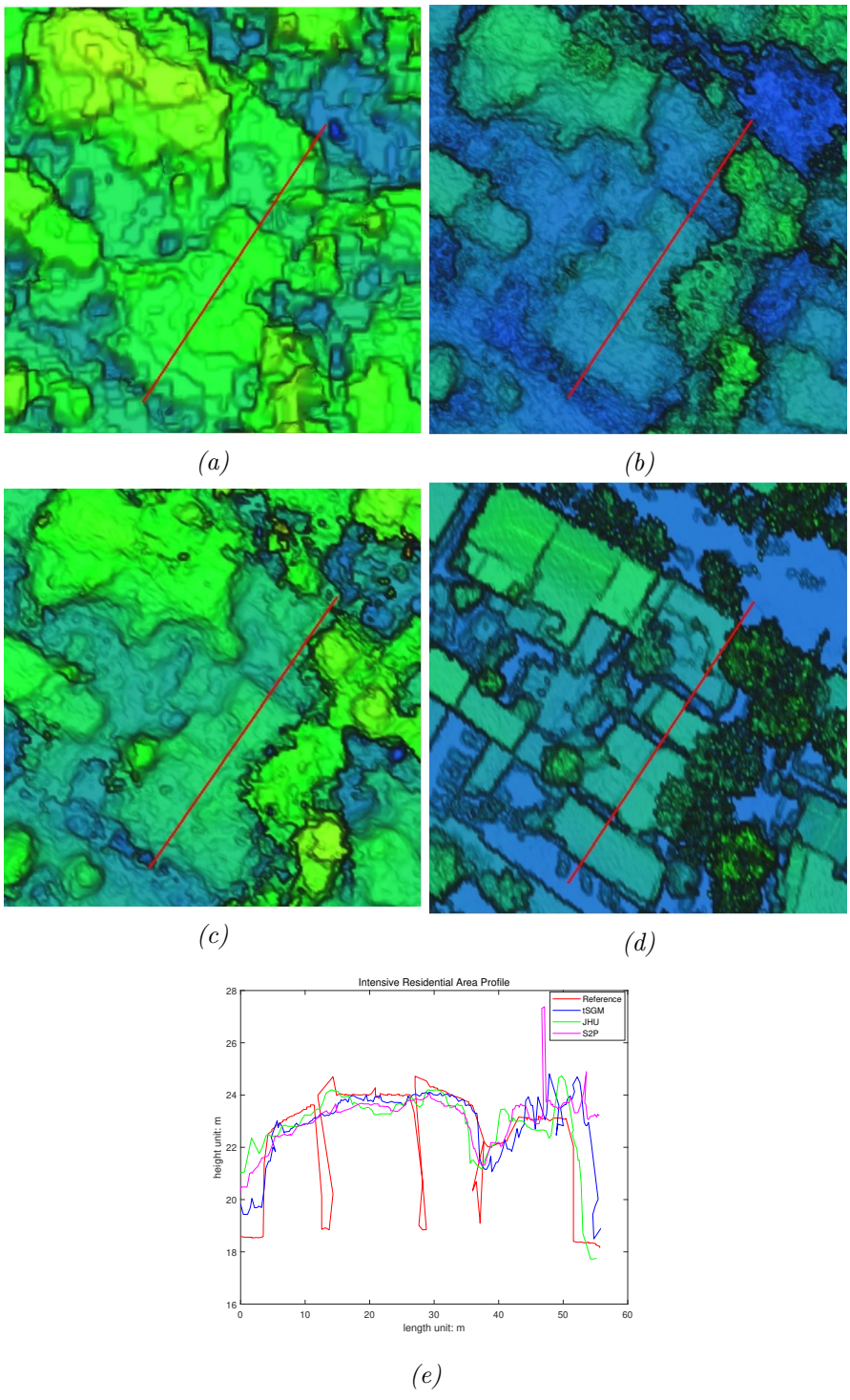


Figure 4.41: Reconstruction details of intensive residential area: (a) point cloud generated by JHU pipeline (b) point cloud generated by tSGM pipeline (c) point cloud generated by S2P pipeline (d) Reference LiDAR point cloud (e) Height profile.

	Test site 3		
	tSGM	JHU/APL	S2P
Completeness (<1m) [%]	**55.931**	49.978	52.210
RMSE [m]	**4.081**	7.209	6,546
median [m]	**0.728**	0.915	0.920
q68 [m]	2.165	2.387	**2.016**
q95 [m]	**5.265**	10.946	12.110

Table 4.16: Height difference evaluation of the IARPA MVS benchmark test site 3

Table 4.16 demonstrates the evaluation results of test site 3. The tSGM fused DSM win the most comparison with the other two pipeline's DSM, because they are failed to rebuild the high-rise building in the reconstructed result. The completeness of our result is 56%, which is 6% higher than JHU/APL's result and 4% higher than S2P result. The RMSE of our result is 4m, and the RMSE of JHU/APL's and S2P results are more than 6m. The median error of the absolute height difference is 0.7m in our fused DSM and it is 0.2m lower than JHU/APL's and S2P DSM. The q95 of our fused DSM is less than half of the q95 of JHU/APL's and S2P DSM, which is 5.2m. The S2P pipeline generates the DSM with lowest q68. The q68 of tSGM DSM is 0.1m higher and The q68 of JHU/APL DSM is 0.4m higher

Generally, the fused point clouds generated by the pipeline proposed in this work can reconstruct the terrain surface with some details correctly. Test site 1 has more isolated standing buildings and less vegetation, so the reconstructed result has the best performance. Test site 2 and 3 have worse performance, because the vegetation and shadows cause troubles during the reconstruction and decrease the quality. According to our experiments, the 3D reconstruction pipeline for satellite MVS imagery has worse performance on the high-rise buildings and intensive residential areas. Our proposed pipeline is more accurate than the JHU/APL's 3D reconstruction pipeline provided by the benchmark, and is competitive to the S2P pipeline. According to the evaluated median errors and q68 values, the S2P pipeline is more robust than our tSGM pipeline. But as the q95 values shown, our pipeline constrains the outliers better. The proposed tSGM pipeline has the lowest RMSE and generate more accurate point clouds and DSMs.

4.5 Mesh Refinement

When processing the satellite MVS images through our pipeline, we obtain the compensated RPCs at sub-pixel accuracy which fulfills the requirement of the MVS refinement for good performance. We conduct the dense image matching for each input image pairs and then fuse them to the final point clouds and DSMs. With the DSM, the initial mesh for the proposed mesh refinement algorithm is generated by Poisson reconstruction. We then can apply our refinement algorithm to generate more accurate mesh models with true 3D structures.

The proposed mesh refinement algorithm is tested on the publicly available IARPA CORE3D benchmark [Brown et al., 2018]. As introduced in Section 4.1, the benchmark data provides 16 bits panchromatic WV-3 multi-view images with 0.3m GSD at nadir. The actual GSD in off-nadir views can be up to a factor of ≈1.5 lower. We have done experiments on JAX test site and

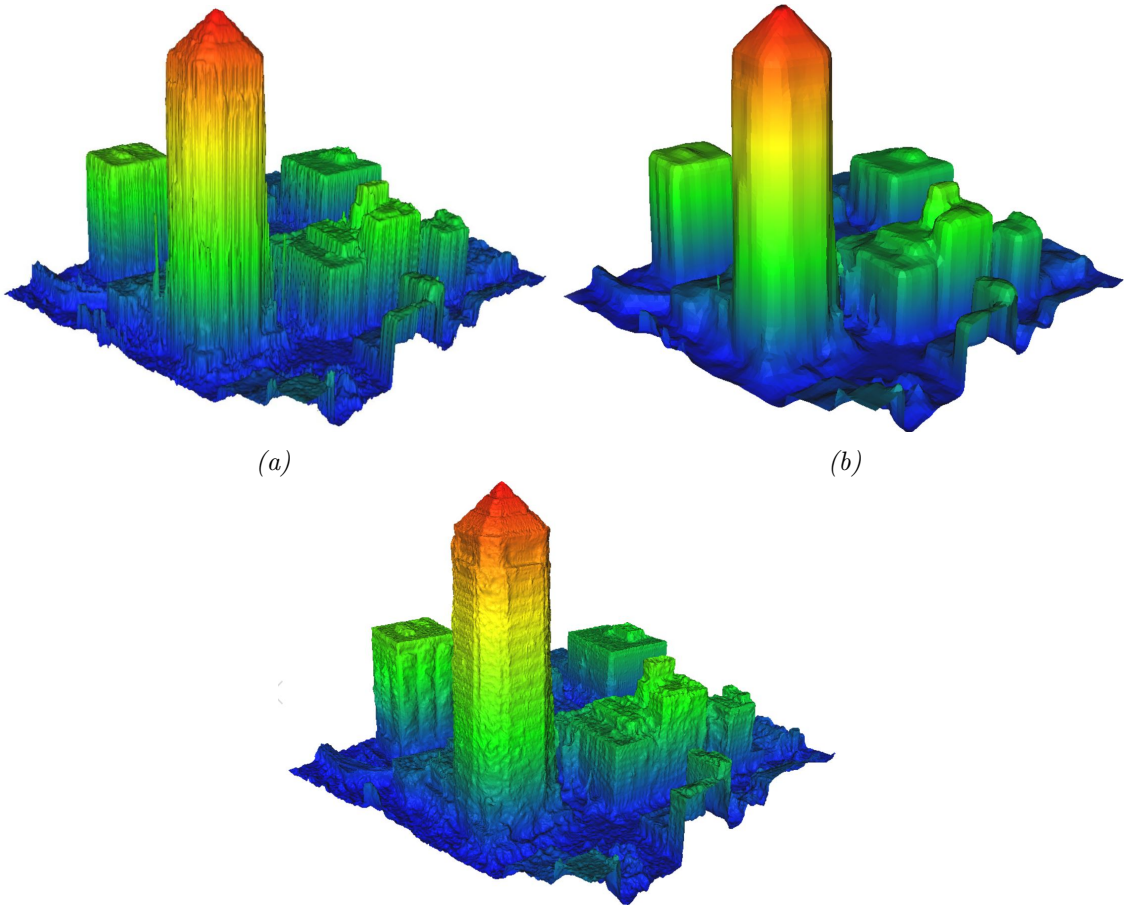
UCSD test site. According to our image selection strategy, we manually remove images with poor illumination conditions. Our image selecting strategy for the DSM generation select the image pairs with intersection angles of the viewing directions between 5° and 35°. Because the proposed algorithm depends on the image similarity, we select the image pairs with shorter baselines as the inputs for the mesh refinement. Considering that the refinement algorithm is very computational expensive, we also prefer to select as less images as possible to improve the efficiency. Therefore, according to our experimental experience, we select 80 suitable image pairs for JAX and 86 for UCSD, with intersection angles of the viewing directions between 5° and 13°. We also give the pre-set parameters to the refinement procedure for both test sites. The parameter α which controls the contribution of the unary term was set to $3.5 \cdot 10^4$ for UCSD and $4 \cdot 10^4$ for JAX respectively. The parameter β, which steers the smoothness was set to 0.05 for both datasets. As we have demonstrated in Section 3.5.2, the average size of the projected triangles is 2 pixels. The input meshes are then refined using the coarse-to-fine scheme starting at 1/8 resolution and stopping at the full resolution images. With the current, unoptimised implementation the runtime of one iteration for the full-resolution refinement is 61mins for JAX and 39mins for the smaller UCSD. The most time consuming part is ray casting, which consumes ≈70% of the computation time. We note that the ray casting is suitable for GPU implementation, as is the computation of $\partial M(\mathbf{x}_i)$, which furthermore could be reused and updated only periodically after several iterations. Together with stricter mechanisms for stereo-pair selection and masking of regions outside the stereo overlap, a >10× speed-up is almost certainly possible. For the two test sites, the examples of the mesh triangulated on the DSM, the input Poisson mesh and also the mesh refined by the proposed algorithm are displayed in Figure 4.42 and 4.43. The mesh models are color-coded by the height.

To test the sensitivity of our method w.r.t. the initialisation, except the DSM produced from our current tSGM based pipeline, we also generate the DSMs with other two state-of-the-art satellite MVS systems: the S2P pipeline [Facciolo et al., 2017] based on MGM [Facciolo et al., 2015] and the ASP pipeline [Moratto et al., 2010] based on SGBM [Bradski, 2000]. In our experiments, all three DSMs serve as baselines to compare with. For a fair comparison we employ our selected stereo images and the corresponding compensated RPCs in all MVS systems and in the actual mesh refinement. For the refinement on three pipelines' DSMs, same weights and mesh triangle sizes are applied. A 2.5D LiDAR DSM with 0.5m GSD is provided and serves as ground truth for both test sites. The lack of full-3D ground truth makes it impossible to quantitatively evaluate 3D elements, such as indentations on facades, in our reconstructed mesh (see Figure 4.42c). Nevertheless, since the aim of our approach is its ability to reconstruct true 3D geometry, we first present a qualitative evaluation of such 3D areas in the section 4.5.1, while the quantitative evaluation based on 2.5D elevation maps is discussed in section 4.5.2.

4.5.1 Reconstruction of 3D Structure

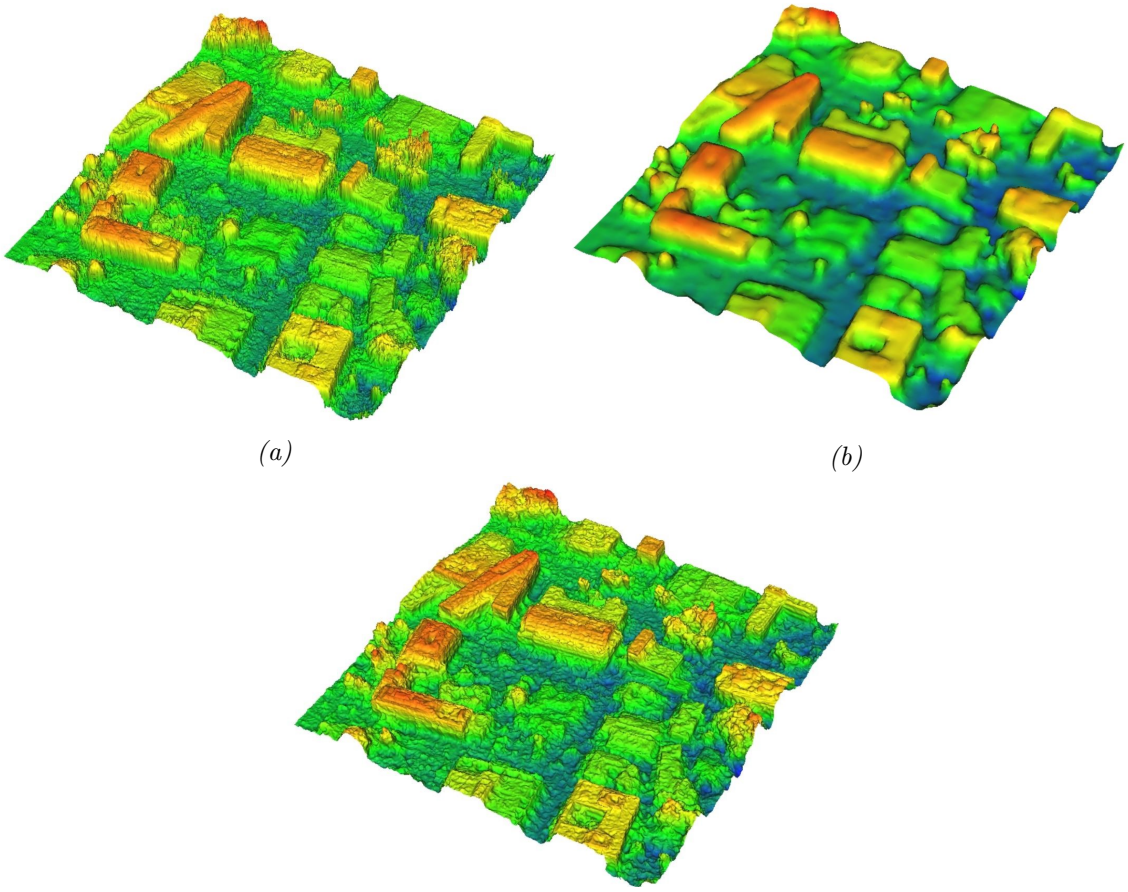
In this section we qualitatively assess the reconstructions obtained with the proposed 3D mesh refinement algorithm. In particular, we illustrate its ability to recover 3D shape details that are not representable in a 2.5D height field, and its superior treatment of sharp discontinuities on man-made structures. Here we select six ROIs from the test sites and show the reconstruction details.

The first ROI is the Bank of America financial center, the reconstructed mesh models and the corresponding area in Google Maps are demonstrated in Figure 4.44. The first row of the figure shows the 2.5D models generated from tSGM pipeline (4.44a), S2P pipeline (4.44b) and ASP pipeline



(a) *(b)* *(c)*

Figure 4.42: Example in JAX test site of (a) the DSM mesh (b) the input Poisson mesh (c) the refined mesh.



(a) *(b)* *(c)*

Figure 4.43: Example in UCSD test site of (a) the DSM mesh (b) the input Poisson mesh (c) the refined mesh.

(4.44c), while the second row shows the 3D models refined from these three pipelines' results (4.44d, 4.44e, 4.44f). At last we show Google Maps snapshots as a reference. Comparing Figure 4.44a and 4.44d, the roof geometry (little step shapes) are improved. The step shapes of the roof are very blur in the 2.5D model of S2P pipeline (4.44b), and the refinement algorithm makes the step shaped roof sharper. Although the roof reconstructed by ASP pipeline is worse (4.44c), the step roof is rebuilt after our refinement. This kind of step shapes are inherently difficult for 2.5D approaches, where these elements are represented only by very few pixels of the elevation map. Facades of the building displayed in Figure 4.44d and 4.44e are clearly smoother than in the 2.5D version displayed in 4.44a, 4.44b. For 2.5D models, the facade geometry is defined by roof and ground elevations. Errors in such elevation estimates are propagated over the whole facade. 3D refinement in facade regions is supported by image similarity in oblique views, leading to smoother surfaces without raster artifacts, and without destroying high-frequency crease edges. Thus, the refined models have visibly crisper crease edges. Moreover, the facades feature indentations and vertical edges that are, by construction in the refined 3D models, not representable in a 2.5D heightfield. Unfortunately, the 2.5D model generated by ASP pipelines lost too much information for this high-rise building, and we can not fully refine the model. Anyway, we still rebuild a part of the facades in Figure 4.44f.

The Edward Ball building in Jacksonville is selected as the second ROI. We present the reconstructed models and the reference Google Maps snapshot in Figure 4.45. The first row presents the 2.5D model based on the DSMs generated by our tSGM pipeline (4.45a), S2P pipeline (4.45b) and ASP (4.45c) pipeline. The refined version of tSGM pipeline (4.45d), S2P pipeline (4.45e) and ASP pipeline(4.45f) are presented in the second row. For all three pipelines, the roof substructures are crisper and more geometric detail is extracted on facades after the refinement. It is also observed that the facades are bumpy in places, presumably due to repetitive structures, specular materials and insufficient evidence in the image set due to the uneven distribution of viewing directions.

The third ROI is the area containing the Florida Times Union building, which is shown in Figure 4.46. Figure 4.46a and 4.46d display the 2.5D model based on tSGM pipeline and its refined 3D model. The 2.5D model and the refined 3D model of S2P pipeline are displayed in Figure 4.46b and 4.46e. Figure 4.46c shows the 2.5D model generated from ASP pipeline and Figure 4.46f shows its refined 3D model. In the refined 3D models, the roof structures are crisper. For all three pipelines, the facade of the left lower building are clearer and smoother. But the Florida Times Union building on the right has bumpy facades, because of the less texture and specular glass walls. The most left part of the area is very noisy in S2P's 2.5D model (4.46b) because of the shadows. This part has too large differences to the true surface, so that the proposed refinement algorithm can not correctly recover the structure. For the same area, the noise in our pipeline's result (4.46a) are eliminated after the refinement (see Figure 4.46d).

The BB&T building is selected as the forth ROI. We present the models and reference Google Maps snapshot in Figure 4.47. The first column of the figure displays the 2.5D model and the refined model of tSGM pipeline. The 2.5D model generated by S2P pipeline and also the corresponding 3D model are displayed in Figure 4.47b and 4.47e. The last column demonstrates the model based on the DSM generated by ASP pipeline and its refined 3D model. In the refined 3D models of the three pipelines, the roof substructures are crisper and the entrance of the building is reconstructed clearer. It is challenging to rebuild the facade, because the dark glass surface impair the reconstruction and leads to the bumpy structures.

Figure 4.48a-4.48g demonstrate the fifth ROI, which is an area close to a railway station. In all refined models, more detail structures on the roof of the building are clearer reconstructed. For

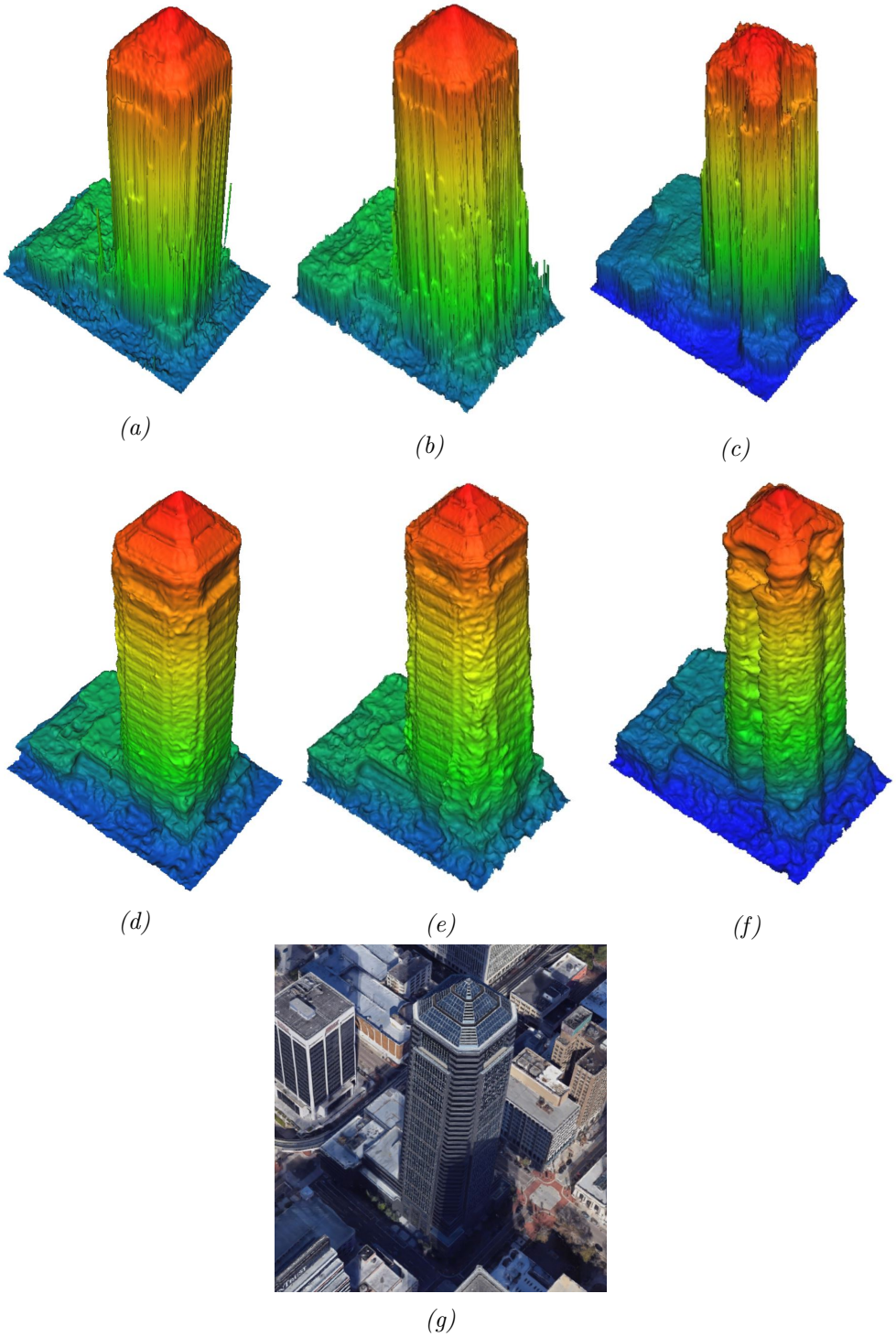


Figure 4.44: Visualization of the Bank of America financial center generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model and (g) Google Maps snapshots.

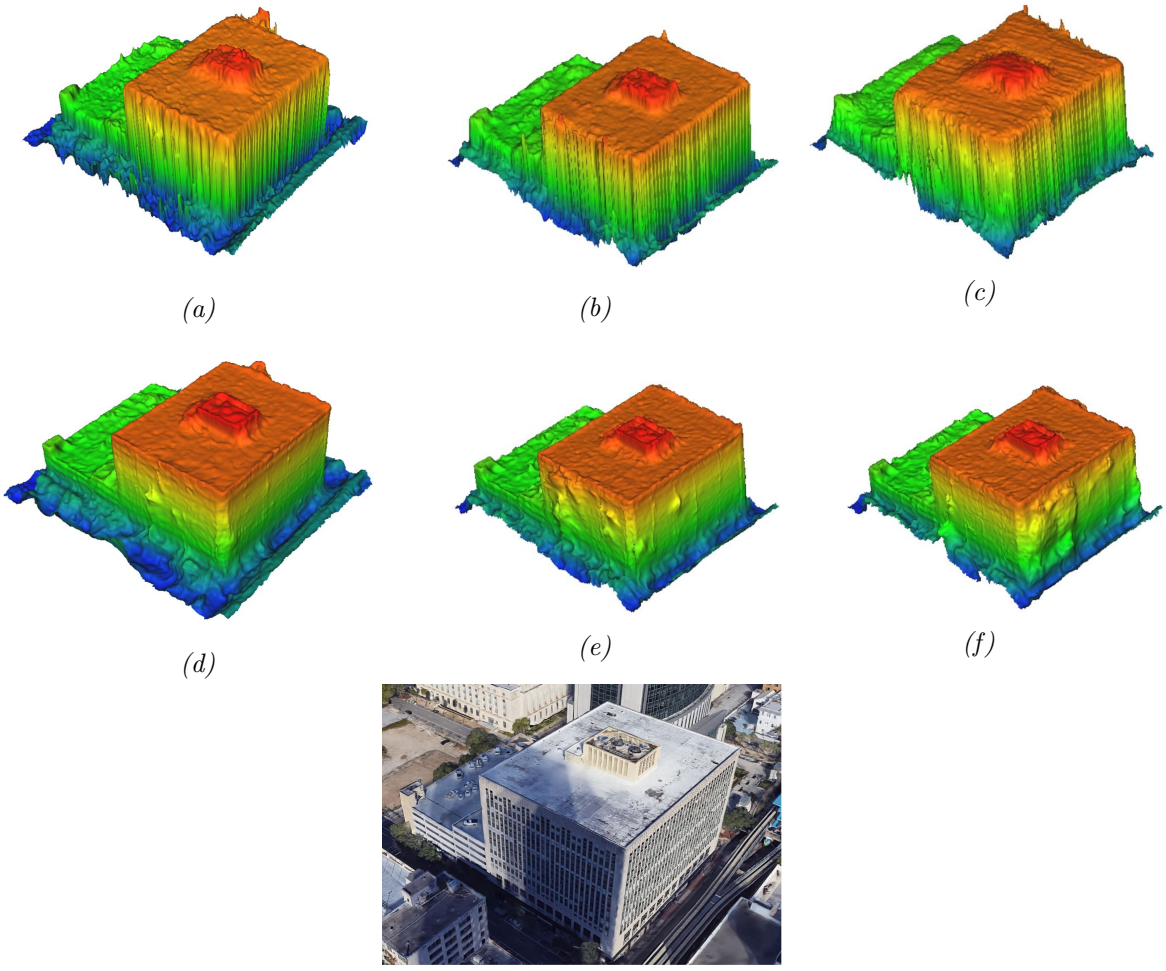


Figure 4.45: Visualization of the Edward Ball building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots.

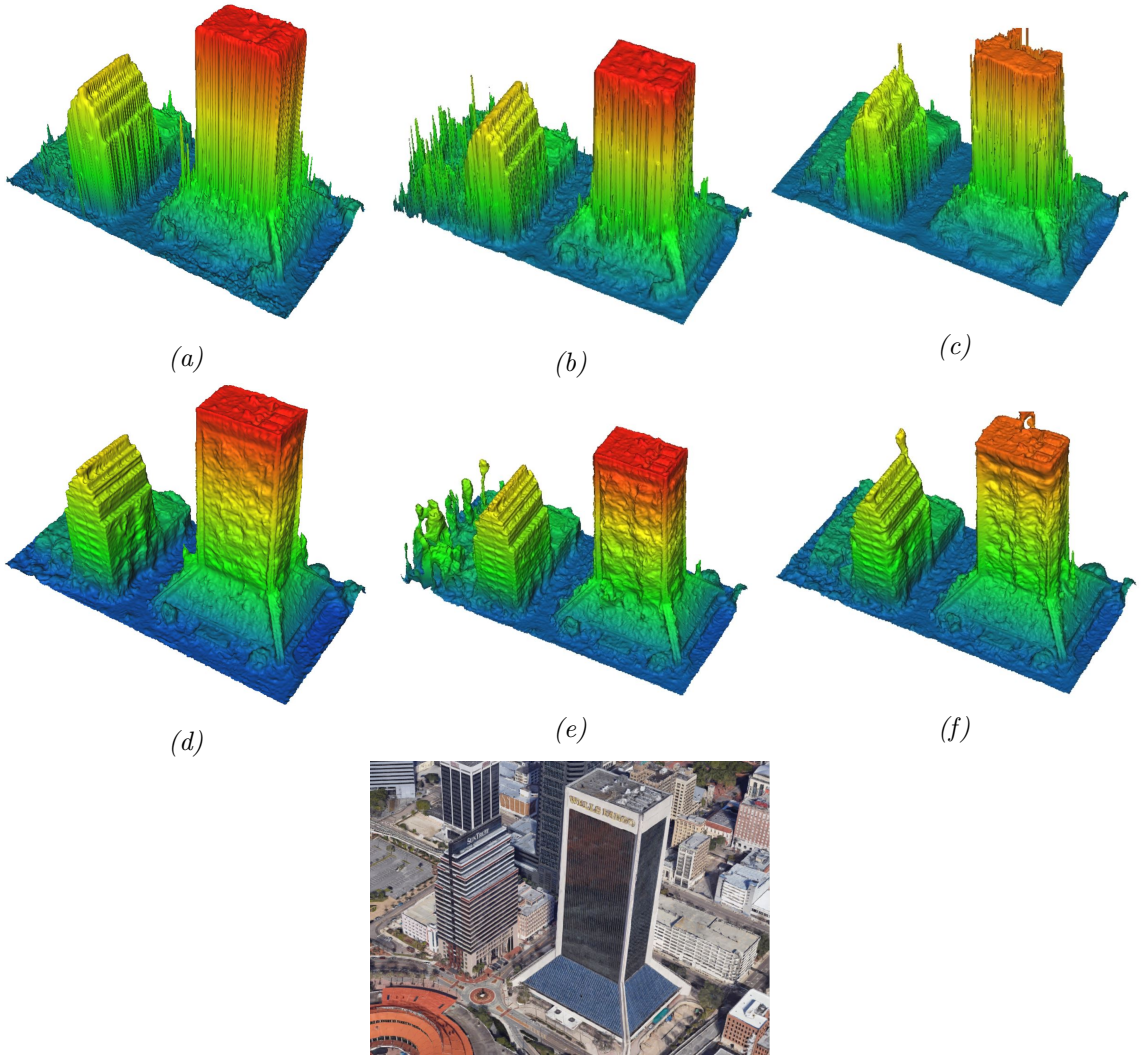


Figure 4.46: Visualization of the Florida Times Union building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots.

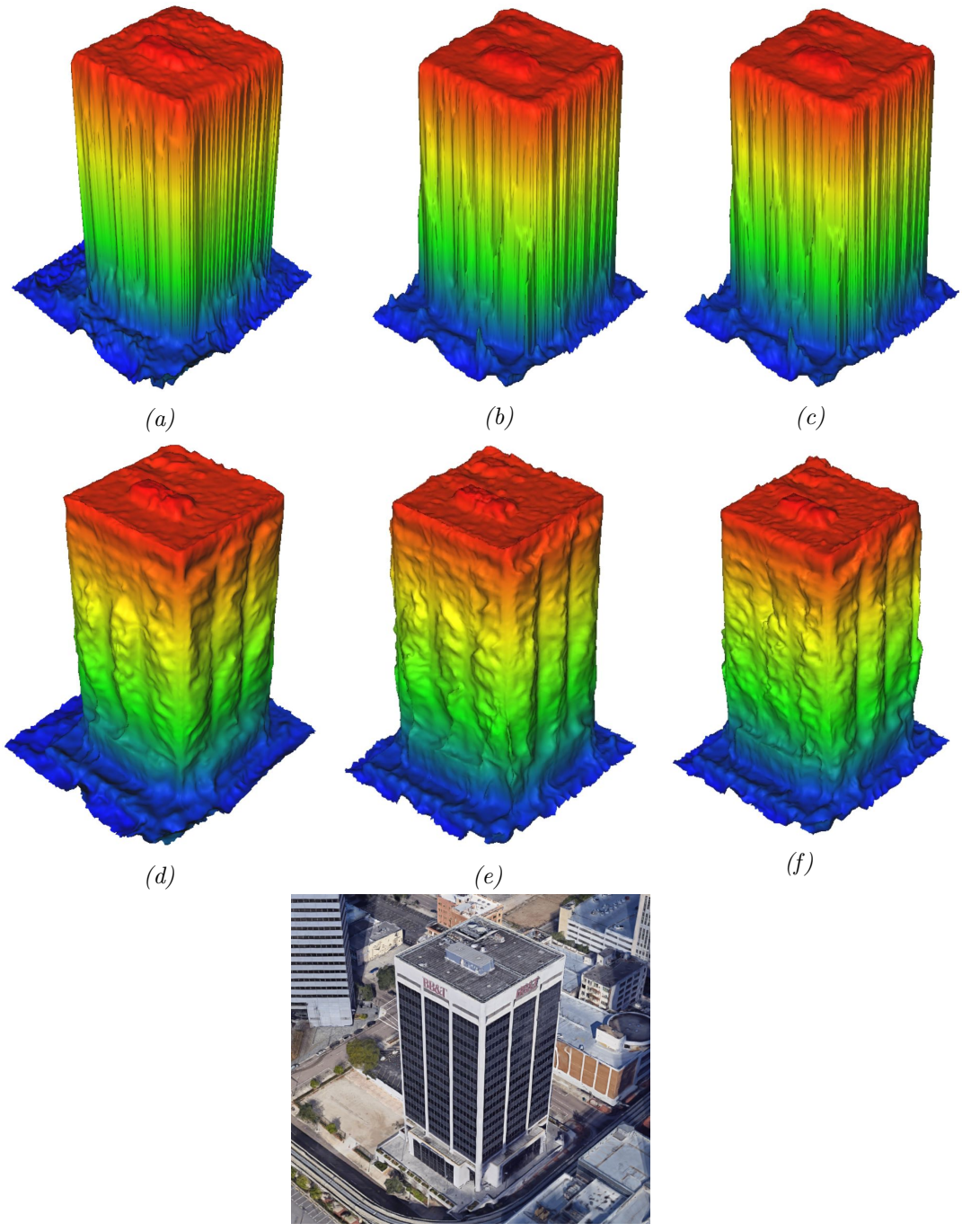


Figure 4.47: Visualization of the BB&T building generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots.

		tSGM	tSGM ref	S2P	SP2 ref	ASP	ASP ref
JAX	Compl. [%]	71.82	**76.66**	78.89	**81.67**	76.62	**80.95**
	RMSE [m]	1.33	**1.00**	1.02	**0.97**	1.06	**0.94**
	NMAD [m]	1.31	**0.77**	0.79	**0.64**	0.98	**0.65**
	q68 [m]	2.43	**1.67**	1.52	**1.28**	1.77	**1.27**
UCSD	Compl. [%]	78.46	**79.56**	80.01	**80.02**	71.39	**77.13**
	RMSE [m]	1.21	**1.18**	**1.15**	1.17	1.31	**1.19**
	NMAD [m]	1.14	**1.09**	**1.02**	1.06	1.59	**1.17**
	q68 [m]	1.93	**1.85**	**1.74**	1.79	2.65	**2.02**

Table 4.17: Evaluation results of the JAX and UCSD test site for three MVS methods and corresponding refined surfaces.

the three test pipelines, the tracks of the railway are reconstructed with less outliers and noise after the refinement, even some parts of the tracks are missing in Figure 4.44c. Comparing to the DSM based models, the ridge of the station is crisper in the refined models. In our refined models, the reconstructed vegetation is also less noisy than the models generated from the conventional methods.

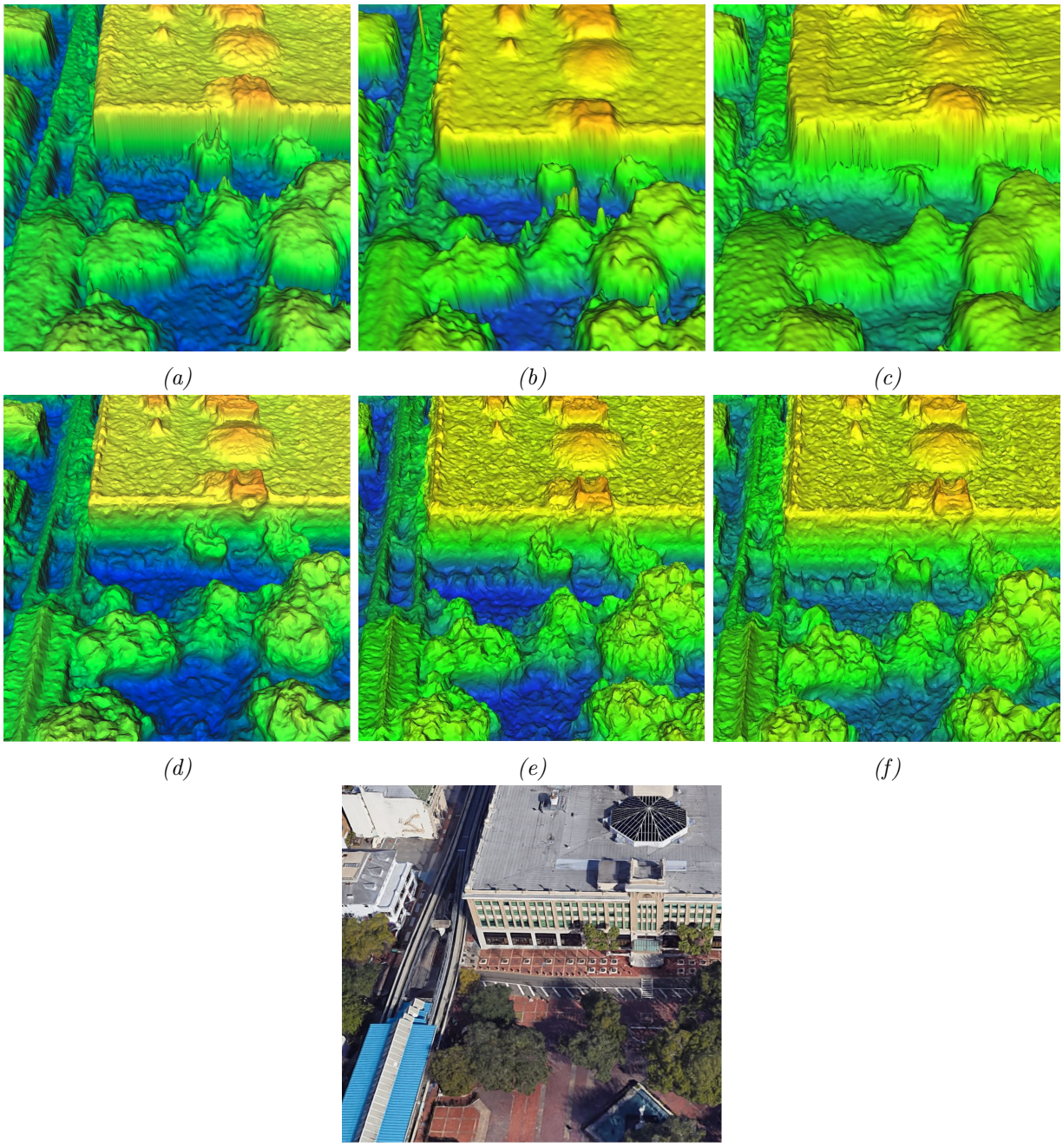
A bridge between the two buildings is selected as our last ROI, which is depicted in Figures 4.49a-4.49g. Compared with the models directly generated by three pipelines, the refined models offer crisper structures on the roof, less outliers and appear less noisy. The same holds true for reconstructed vegetation. The sunken structure of the bridge is reconstructed in the refined models. The street under the bridge is not reconstructed correctly in 4.49d, 4.49e and 4.49f – while one can see the attempt to "carve out" the empty space under the bridge, the refinement is limited by the faulty topology of the initial surface, which cannot be changed by the algorithm.

4.5.2 Quantitative Evaluation of Multi-View Refinement

The LiDAR ground truth is provided in the form of a 2.5D gridded DSM. Consequently, the refined 3D meshes must be converted back to 2.5D elevation maps. To accomplish this process, we align the mesh to the LiDAR DSM [Bosch et al., 2016], then cast a vertical ray through the centre of each grid cell, and extract the highest intersection point with the reconstructed mesh.

Table 4.17 displays error statistics for the MVS results and the corresponding refinement results, for both JAX and UCSD sites. The comparisons to ground truth were carried out with the test suite provided by [Bosch et al., 2016], where we add additional, robust error metrics: namely, a truncated RMSE, computed from only those residuals that are <3m (ca. 10GSD), and the corresponding completeness (percentage of residuals below that threshold). Furthermore, we list the NMAD and the 68% percentile of absolute residuals (q68) to show the robustness of the results.

According to Table 4.17, for all three MVS satellite systems, the DSMs are significantly improved after the refinement after the mesh refinement in JAX test sites. For tSGM pipeline, the completeness raises 5%, the RMSE reduced by 1GSD, the NMAD is improved by 2GSD and the q68 is 0.8m reduced after the refinement. As to the S2P pipeline, the completeness is raised by 3%, the NMAD decreases 0.1m and the q68 value reduces 0.2m via our mesh refinement. The RMSE of the S2P pipeline improves slightly and stays at very close level after the processing. For the ASP pipeline, the mesh refinement improves the completeness by 5%, decreases the RMSE by 0.1m, reduce the NMAD by 0.3m and decrease the q68 value by 0.5m. In UCSD test site, the ASP pipeline

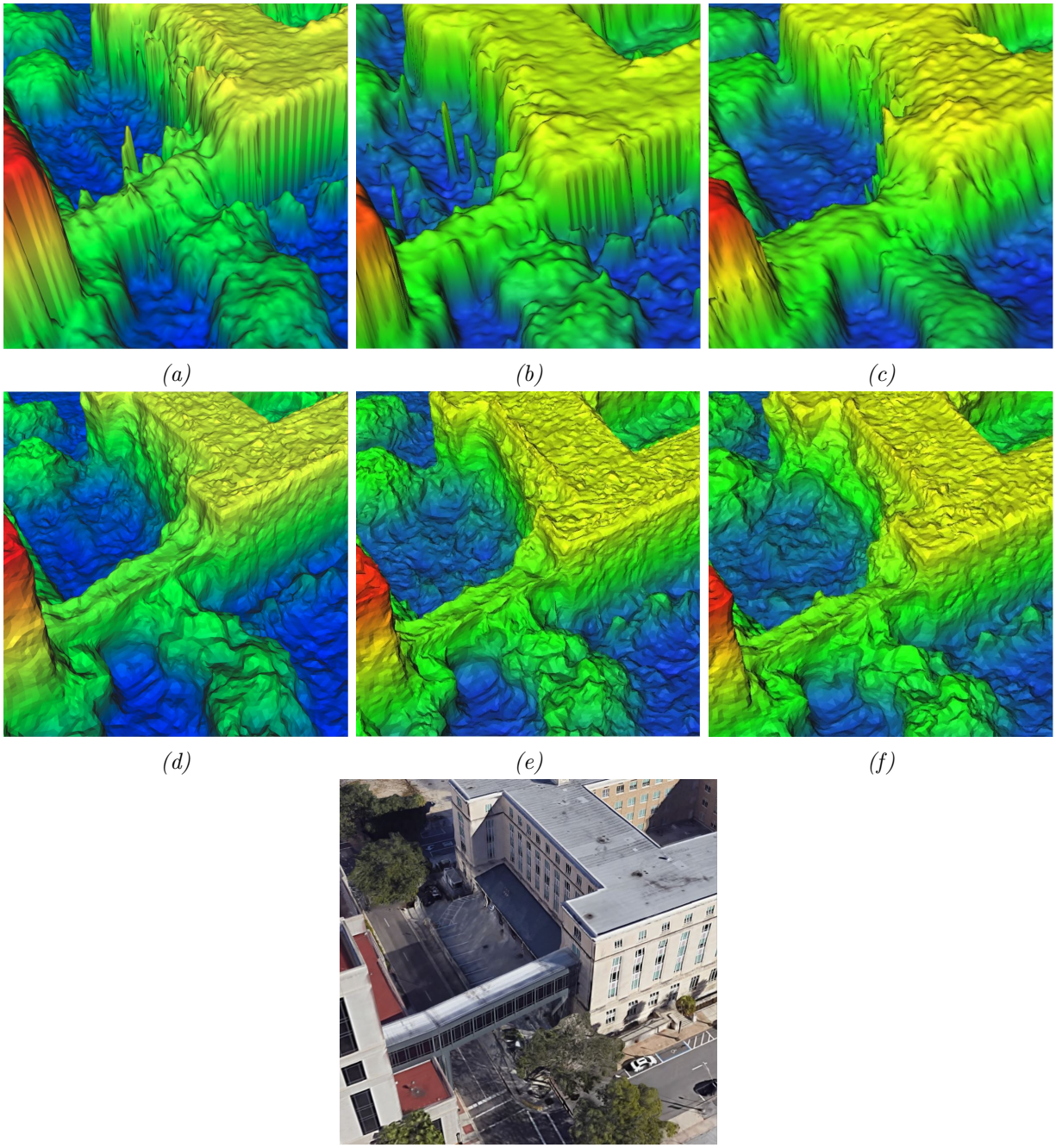


(a) *(b)* *(c)*

(d) *(e)* *(f)*

(g)

Figure 4.48: Visualization of the railway station generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots.



(a) *(b)* *(c)*

(d) *(e)* *(f)*

(g)

Figure 4.49: Visualization of the bridge structure generated by (a) tSGM 2.5D model (b) S2P 2.5D model (c) ASP 2.5D model, the refined version of (d) tSGM 3D model (e) S2P 3D model (f) ASP 3D model. and (f) Google Maps snapshots.

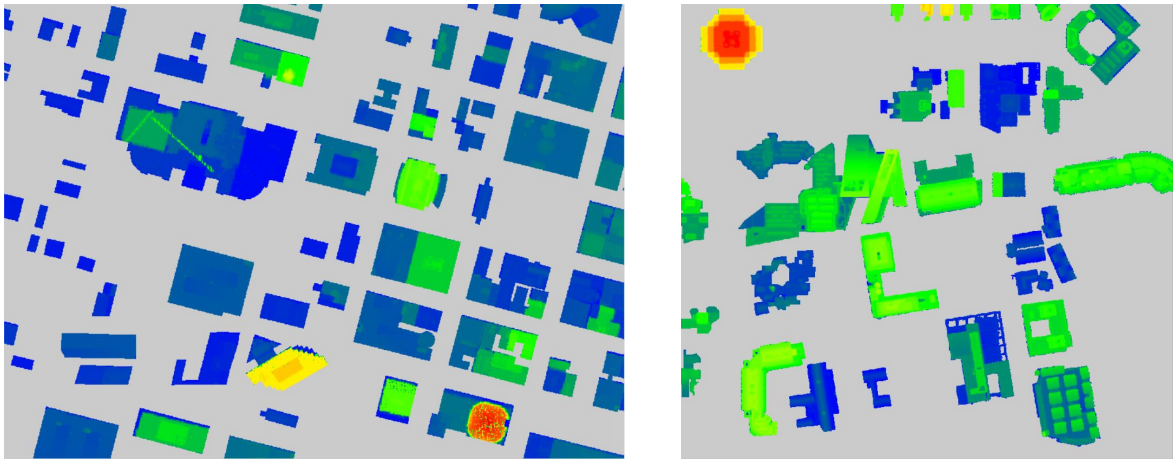


Figure 4.50: Ground truth LiDAR DSM data of evaluated building areas of JAX test site (left) and UCSD test site (right).

		tSGM	tSGM ref	S2P	SP2 ref	ASP	ASP ref
JAX	Compl. [%]	88.46	**88.77**	87.68	**88.01**	84.22	**87.14**
	RMSE [m]	0.86	**0.80**	0.89	**0.79**	1.0	**0.83**
	NMAD [m]	0.51	**0.39**	0.48	**0.37**	0.74	**0.46**
	q68 [m]	0.88	**0.73**	0.92	**0.73**	1.25	**0.80**
UCSD	Compl. [%]	95.69	**96.09**	**96.72**	96.71	93.23	**96.44**
	RMSE [m]	0.97	**0.86**	0.85	0.85	1.16	**0.86**
	NMAD [m]	0.58	**0.47**	0.45	0.45	0.79	**0.47**
	q68 [m]	0.95	**0.77**	**0.74**	0.75	1.28	**0.77**

Table 4.18: Evaluation results of the JAX and UCSD test site for three MVS methods and corresponding refined surfaces.

is improved significantly after the refinement: the completeness is increased by 6%, the RMSE is improved by 0.1m, the NMAD is improved by 0.4m and the q68 value is improved by 0.6m. The tSGM pipeline has the improvement like 1% higher completeness. For RMSE, NMAD and q68 values, the DSM generated from refined mesh models are slightly improved, which is in centimeter level. After the refinement, the refined result of S2P pipeline has slightly higher completeness, while the RMSE, NMAD and q68 value are slightly worse (less than 0.2GSD). Generally, the accuracy and robustness of the DSM derived from the refined mesh stays at the same level as the DSM generated by the S2P pipeline.

We also conduct an evaluation only on selected building roofs (see Figure 4.50) to rule out error sources like seasonal changes, extreme height discontinuities, temporal changes and moving objects. [Brown et al., 2018] provide a building mask for the Jacksonville test site. Since for the UCSD site the building mask is not publicly available, we manually created one. The evaluation results of the building roofs are demonstrated in Table 4.18.

As shown in table 4.18, our mesh refinement has significant improvement on the ASP pipeline for both two test sites. In both test sites, ASP pipeline has 3% improved completeness, 0.2m decreased RMSE, 1GSD decreased NMAD and 0.5m decreased q68. As to our tSGM pipeline, the completeness and RMSE improved slightly, the NMAD and q68 value decreased by 0.15m in JAX

		tSGM	tSGM ref	S2P	SP2 ref	ASP	ASP ref
ROI 1	Compl. [%]	92.69	**93.35**	**93.82**	92.83	89.76	**90.57**
	RMSE [m]	0.80	**0.79**	**0.67**	0.77	0.85	**0.76**
	NMAD [m]	0.31	**0.24**	**0.22**	0.23	0.46	**0.25**
	q68 [m]	0.70	**0.64**	**0.51**	0.56	0.85	**0.61**
ROI 2	Compl. [%]	94.69	**94.71**	**95.06**	94.54	89.71	**93.49**
	RMSE [m]	0.77	**0.75**	**0.69**	0.74	1.07	**0.75**
	NMAD [m]	0.37	**0.28**	**0.27**	0.30	0.67	**0.30**
	q68 [m]	0.71	**0.64**	**0.60**	0.62	1.22	**0.65**
ROI 3	Compl. [%]	79.31	**80.54**	**83.79**	80.86	63.26	**80.37**
	RMSE [m]	1.26	**1.18**	1.15	**1.13**	1.56	**1.16**
	NMAD [m]	1.18	**1.00**	0.88	**0.87**	2.15	**0.92**
	q68 [m]	1.93	**1.75**	**1.52**	1.68	3.32	**1.78**
ROI 4	Compl. [%]	93.68	**95.52**	94.63	**94.86**	88.81	**92.64**
	RMSE [m]	0.89	**0.76**	0.82	**0.74**	1.05	**0.75**
	NMAD [m]	0.46	**0.36**	0.36	**0.35**	0.67	**0.36**
	q68 [m]	0.84	**0.66**	0.68	**0.63**	1.20	**0.68**

Table 4.19: Evaluation results of the JAX test site for the three MVS methods and corresponding refined surfaces.

test site. In the UCSD test site, through our refinement, the tSGM pipeline has the 0.4% improved completeness, the 0.1m improved RMSE and NMAD and the 0.2m improved q68 value. For S2P pipeline, the completeness increases 0.4%, the RMSE decreases 0.1m, the NMAD decreases 0.1m and q68 value decrease 0.2m in JAX test site. While in UCSD test site, there are no significant quantitative differences to S2P after the refinement: the completeness is 0.01% lower, the q68 is 0.01m higher and the RMSE and the NMAD stay the same. It appears that in terms of average 2.5D roof accuracy the S2P method is already close to the achievable limit (NMAD ca. 1.5 GSD), so that the further improvement is hard to reach. In general, the majority of the residuals are <1.5 GSD and the accuracy of the DSMs on roof surfaces is in the sub-meter range, which we find very encouraging.

We note that the error metrics do not fully reflect the visual quality of the reconstructions. The mesh refinement does bring out additional 3D structure and suppress noise (see Sec. 4.5.1). Note that the mesh refinement exhibits little sensitivity to the initialisation, its results are quantitatively very similar in all cases, independent of which stereo method was used to generate the input surface.

To characterise the reconstruction in more detail and gain further insights into its behaviour, we examine four building roofs for each test site in detail. As we have explained, we exclude building edges, since aliasing at large height jumps makes a meaningful evaluation impossible.

Figures 4.21 and 4.22 display exemplary surface reconstructions of the different tested methods for both ROIs. Tables 4.19 and 4.20 list the corresponding quantitative results. Again, mesh refinement generally improves the accuracy over SGM and ASP, at both sites. With S2P we get mixed results. Its accuracy is already very high (for several buildings at, or below GSD), so the results after refinement are quantitatively almost the same. Still, visual inspections make it clear that the quantitative metrics do not fully characterise the model quality: The refined surfaces are crisper and more correct on complicated roof structures. The numbers do not reflect this, because after refinement the surface tends to be slightly noisier on flat, fronto-parallel areas that play to the strength of the constant-height prior built into most MVS and 2.5D fusion methods. Overall,

		tSGM	tSGM ref	S2P	SP2 ref	ASP	ASP ref
ROI 1	Compl. [%]	**99.70**	99.20	98.50	**99.0**	94.40	**99.04**
	RMSE [m]	0.97	**0.87**	**0.86**	0.88	1.21	**0.86**
	NMAD [m]	0.61	**0.50**	**0.48**	0.52	0.90	**0.51**
	q68 [m]	0.90	**0.74**	**0.74**	0.75	1.34	**0.73**
ROI 2	Compl. [%]	92.46	**92.22**	**94.17**	92.33	86.15	**91.80**
	RMSE [m]	1.12	**0.97**	**0.96**	0.97	1.24	**0.98**
	NMAD [m]	0.68	**0.52**	0.48	**0.52**	0.90	**0.54**
	q68 [m]	1.22	**0.93**	0.92	0.92	1.67	**0.96**
ROI 3	Compl. [%]	96.00	**98.67**	97.66	**98.33**	85.21	**98.31**
	RMSE [m]	1.10	**1.07**	**1.09**	1.10	1.65	**1.12**
	NMAD [m]	0.69	**0.59**	0.67	**0.61**	1.60	**0.69**
	q68 [m]	1.09	**1.06**	1.10	1.10	2.27	**1.17**
ROI 4	Compl. [%]	99.14	**99.44**	99.53	**99.52**	96.44	**99.31**
	RMSE [m]	0.95	0.95	**0.88**	0.94	1.35	**0.96**
	NMAD [m]	0.57	**0.46**	0.46	**0.45**	0.70	**0.43**
	q68 [m]	0.92	**0.85**	**0.83**	0.84	1.48	**0.87**

Table 4.20: Evaluation results of the UCSD test site for three MVS methods and corresponding refined surfaces.

we observe crisper reconstructions of roof details after refinement. These details are recovered even when starting from the roughest initial surface (generated with NASA's ASP), where they are completely missing. The meshes after refinement are also visually comparable, independent of the initialisation, indicating favourable convergence properties of the hierarchical optimisation. We note that the q68 metric is comparable with the numbers published in [Brown et al., 2018].

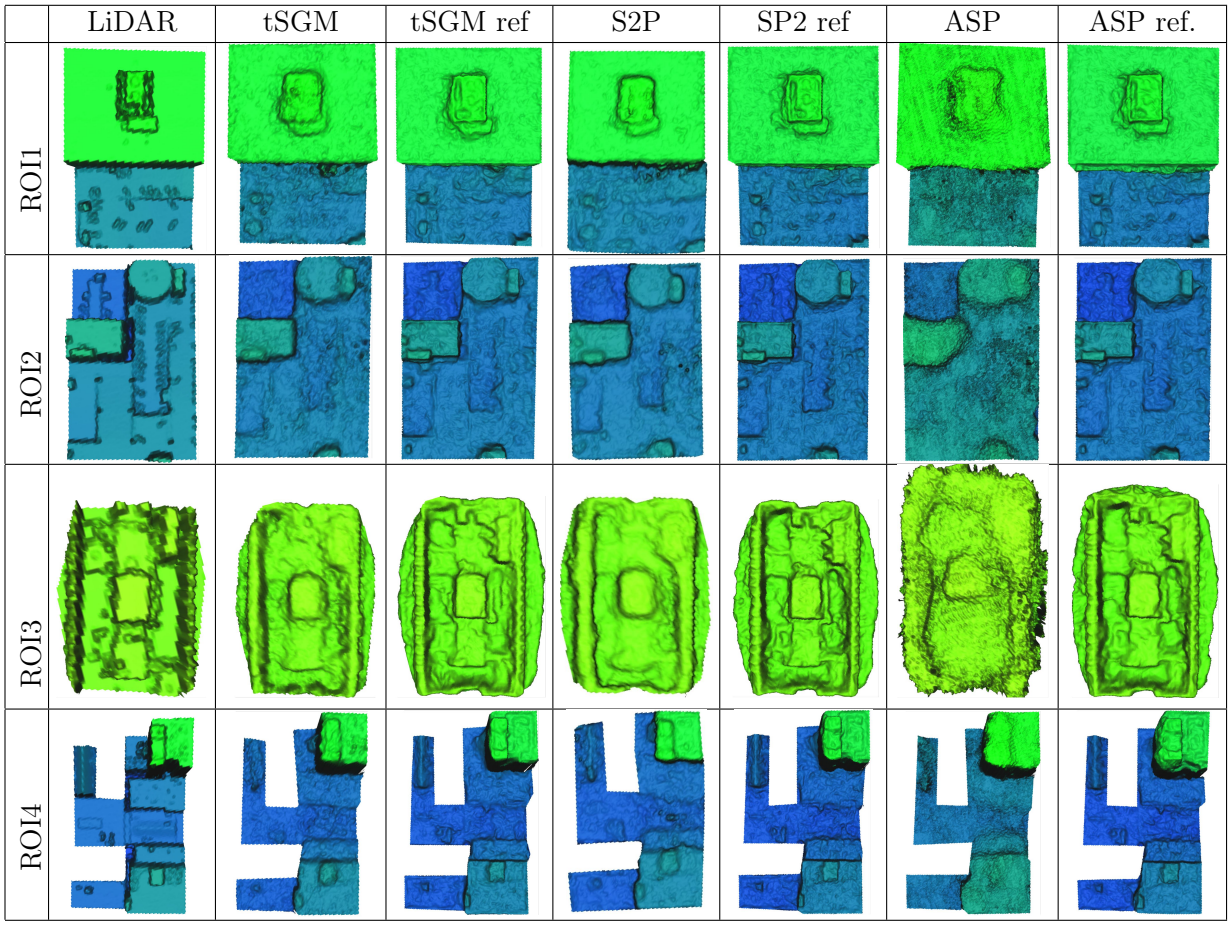


Table 4.21: Visualization of the building evaluation for the JAX test site.

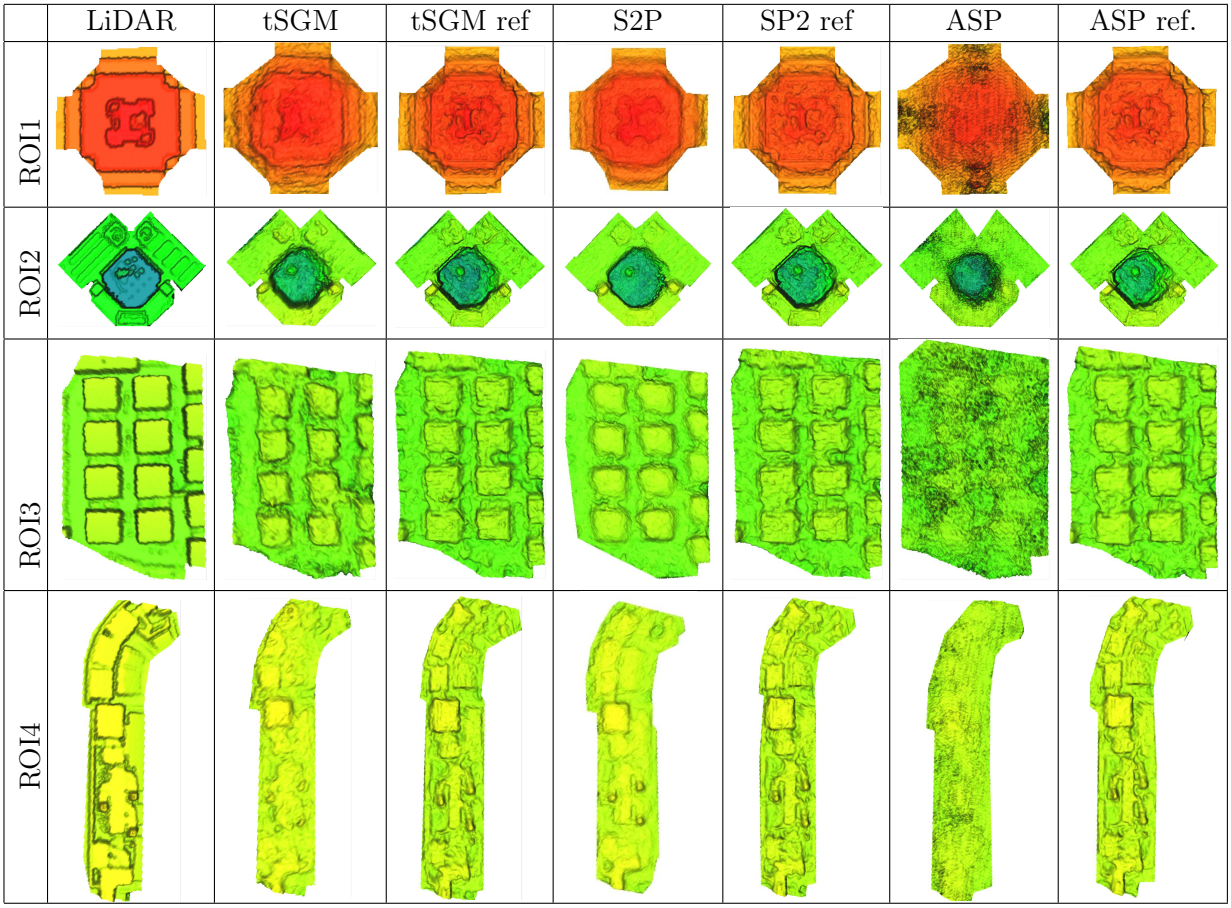


Table 4.22: Visualization of the building evaluation for the UCSD test site.

Chapter 5

Summary and Outlook

In this work, we have presented a new 3D reconstruction pipeline for MVS satellite images, which produces 3D point clouds, 2.5D DSMs and also 3D mesh models. We will summarize the proposed algorithms and procedures in section 5.1. Then some discussion about the limitations and future work are presented in section 5.2.

5.1 Summary

Image selection: Taking the season changes, illumination situation and the incidence angles of the view into account, we propose a strategy of image selection. According to the experiments on MVS satellite imagery taken from different dates, the image views with large incidence angles are less useful to process. The overexposure images or dim illuminated images are harmful to the image matching. The stereo pairs which have too large or too small intersection angles are eliminated, so that the dense image matching performs better. Most state-of-the-art MVS pipelines order the images chronologically and then select stereo pairs with close collecting dates. We find the season changes is critical to reduce the outliers and first sort the images according to the seasons. The stereo pairs collected in the same season are applied in our pipeline. Our image selection strategy improve the reconstruction effectively.

RPC compensation: In our pipeline, we implement a relative RPC bias-compensated method to refine the accuracy of the RPCs. Our method requires no ground control information. Based on multiple tie points, we generate a virtual surface and conduct the image orientation on this surface. All RPCs are aligned to the virtual surface and no further registration is need in the subsequent processing. Experiments are taken on WorldView-2 and WorldView-3 datasets. It has been verified that our proposed relative orientation method decreases the relative pointing error to sub-pixel level.

Image rectification: We propose a modified piece-wise epipolar resampling method to approximate the epipolar curve of satellite imagery by the combination of segments. Many satellite MVS pipelines rectify the images tile-wise. But our proposed algorithm resamples the stereo pairs along the epipolar segments over the entire image without tiling processing. Our image rectification method has been tested on a small coverage WorldView-3 test site and a large coverage WorldView-2 test site. For both test sites, the vertical parallaxes of the obtained corresponding epipolar images are at sub-pixel level. We also show that, no matter the corresponding points are located in the middle or in the corner of the image, the vertical parallax is close to zero.

Dense image matching and forward intersection: We introduce the tSGM algorithm to process the satellite imagery and then generate the dense point cloud by forward intersection. The experiments are carried on different VHR satellite datasets. It has been proved that the tSGM algorithm is a reliable and robust method to generate dense disparity maps for satellite data. The generated 3D point cloud is accurate.

DSM fusion: We project the point clouds into the regular sized discrete grids in the UTM coordinate system. To generate accurate DSMs from MVS satellite imagery, a fusion step is conducted in our pipeline. Unlike the state-of-the-art pipelines like S2P [Facciolo et al., 2017] and ASP [Moratto et al., 2010], the point clouds are already aligned to the identical virtual surface and do not need the additional registration before the fusion step in our proposed method. The simple median fusion is applied and generates robust and accurate final DSM. An investigation about the influence of the number of the involved stereo pairs for DSM fusion is done in our work. We apply those point clouds which lead to the best completeness and maintain a relatively low RMSE. The errors are increased if too many point clouds are involved, because some of them are of low quality. The structures on the surface are well restored in our fused DSM. The DSMs generated from our proposed pipeline is accurate and robust

Mesh refinement: To recover the true 3D structures of the surface mesh model from satellite images, we have proposed a novel mesh refinement algorithm which minimizes the photometric transfer error between multiple pairs of images. The RPC model is applied to specify the satellite sensor poses. In experiments on high-resolution MVS images from WorldView-3, we have shown that, the refinement extracts additional 3D surface details and reproduces crisper edges. When the oblique satellite views are introduced, the method produces clearer facade structures. The refinement procedure also decreases the residual errors of the elevation of the recovered surface in most cases.

We have compared the point clouds and 2.5D DSM generated from our proposed tSGM based pipeline to the results generated by other state-of-the-art satellite 3D reconstruction pipelines like DLR's pipeline, JHU/APL's pipeline and S2P pipeline. In large range mountainous areas, the S2P pipeline has some problems during their tile-wise processing and our proposed algorithm has better performance. The DSMs' height accuracy of shadow and vegetation area of our tSGM based pipeline is higher than the JHU/APL pipeline, close to S2P pipeline and lower than the DLR's pipeline. However, our proposed pipeline preserve more detail structures, while DLR's pipeline is slightly over-smoothed and lost some details. Our tSGM based pipeline constrains the outliers better than other pipelines. But the evaluation results like median errors also show that our pipeline is slightly less robust than the top-ranked S2P pipeline. In general, considering about the completeness, the accuracy and the robustness, it turns out that our pipeline is competitive to the state-of-the-art pipelines in the satellite MVS reconstruction field. Moreover, the state-of-the-art algorithms only reconstruct 2.5D models. Through the mesh refinement procedure, the proposed satellite MVS 3D reconstruction pipeline generates not only the DSMs but also true 3D surface models. It has been verified in our experiments, that no matter what kind of initial mesh is applied, our mesh refinement algorithm can improve the accuracy of the model and recover more details and 3D structures. The 3D models generated by the proposed pipeline could be advantageous for downstream building modelling, surface structuring and analysis based on geometric features, and of course visual display from off-nadir viewpoints.

5.2 Limitations and Outlook

With a number of selected stereo image pairs, our 3D reconstruction pipeline produces accurate and robust 2.5D elevation maps and true 3D mesh models. However, there are some limitations existing in our pipeline. Here we highlight these problems and also have a discussion about the directions to tackle them.

Fully automatic pipeline: Our 3D reconstruction pipeline is designed as a modular but semi-automatic pipeline. We need to select the qualified stereo pairs manually. The image orientation module also requires some given tie points generated from other tools like Envi and VisualSFM. An end-to-end automatic pipeline would be more convenient in the applications by minimizing the user interaction. An image selection module can be designed, so that the image incidence angles are read and the image seasons are tagged according to image collecting dates. Then the pipeline can select the stereo pairs according to our strategy automatically. The SIFT and RANSAC algorithms are a proper methods to be implemented in our pipeline, so that the tie points selection and RPC compensation can be fully automatic.

Parameter adaption: The tSGM algorithm has robust parameters. As to our 3D mesh refinement algorithm, the parameters are set according to our experimental experience and are varied from dataset to dataset. To get rid of the heuristic parameterization, the 3D mesh refinement algorithm should be tested on more test sites. The proper common preset parameters should be learned, and parameters can be adapted automatically if needed.

Processing speed: Our proposed algorithms are all implemented in C++ lab codes and marginally optimized. There are still many spaces to improve on our programming, for example the cache locality and the number of constructing acceleration structures for the ray casting to the surface. There are also several parts of our 3D reconstruction algorithm suitable to be processed on GPU, like the epipolar image resampling, the ray casting to the surface, derivative image similarity computation and so on. The current refinement scheme minimized the energy in fixed step-width and fixed iteration times. We also consider about replacing it by automatic adapted parameters and dynamic iteration times.

Mesh triangle size: In our implementation, the input mesh models are suggested to have triangle size like two pixels. A dynamic simplification could be an interesting direction to explore. The adaptive triangulation could improve the performance and decrease the noise.

Vegetation: Usually, vegetation exists in the test sites. Because the MVS satellite images are collected in different dates and seasons, the vegetation introduce errors to the similarity measurement and reduce the accuracy of the generated models. To improve the results, we can classify the MVS imagery and mask out the vegetation.

Texture-less surface: Being based on image similarity, it is however still challenged by repetitive texture and non-Lambertian surfaces. To further diminish artefacts of our method in such areas, one important direction is to endow 3D surface models with more a-priori knowledge about the reconstructed scene. In particular, our current implementation includes only a simple thin-plate regulariser to favour low curvature. Elementary priors like the preference for piece-wise constant heights (extensively used in dense matching and 2.5D model fusion), or a preference for right angles, are still missing and need to be ported to the world of 3D mesh-based reconstruction.

In this work, the satellite MVS 3D reconstruction algorithms are mainly applied for the extraction of geometric information. As a hot spot in the past years, the development of deep learning algorithms has promoted the researches on the semantic segmentation of the images or point clouds.

[Brown et al., 2018] and [Goldberg et al., 2018] publish the commercial MVS satellite imagery benchmarks and promote more researches on the semantic segmentation of satellite image data. With the latest benchmarks, [Leotta et al., 2019] combine the MVS reconstructed point cloud and the textured mesh models to get the 3D models with sharper edges. [Romanoni et al., 2017] show some impressive mesh refinement result with the help of semantic information on the airborne MVS dataset. The semantic information is an efficient weapon to tackle the problems like the blurred building edges caused by shadows and trees. In future work, we want to investigate the pipeline, that start the satellite data processing from the photogrammetric MVS reconstruction, then apply semantic information to extract regularized surface and at last recover the details and 3D structures with our mesh refinement algorithms.

We have shown that our proposed mesh refinement algorithm can reconstruct further detail information and improves the height accuracy. This may provide a sight of detailed true 3D reconstruction from space-borne data, and we hope it can trigger more investigations in this area. The current mesh refinement algorithm can hardly recover the building edges effected by shadows or vegetation. Latest deep learning algorithms are acquiring more and more accurate semantic information, which can be applied to constrain the shape of objects in point clouds or DSMs. By combining mesh-refinement algorithm with the fast-developed deep learning algorithms, it is promising that the 3D models generated from satellite MVS imagery can have not only more details but also crisper edges. The resolution of the satellite optical MVS imagery could be improved to higher level in the next decades. The large data volume and time-consuming processing can be obstacles to the application of the satellite MVS imagery. With the help of cloud platforms, the hardware requirement for the satellite data application could be lower. We believe the 3D models reconstructed from satellite MVS data would show more detail structures and plays bigger roles in the global-scale 3D mapping in the future. With the establishment and widespread of 5G wireless systems, Internet-of-Things and smart cities, the application of the 3D models could be boosted to a new era. The satellite MVS would be wider used in the reconstruction of the 3D city models. We believe that, with the growing availability of high-resolution oblique satellite images, the field could move beyond 2.5D elevation models and the quest for full 3D models will gain momentum.

Bibliography

[Ahmadabadian et al., 2013] Ahmadabadian, A. H., Robson, S., Boehm, J., Shortis, M., Wenzel, K., and Fritsch, D. (2013). A comparison of dense matching algorithms for scaled surface reconstruction using stereo camera rigs. *ISPRS Journal of Photogrammetry and Remote Sensing*, 78:157–167.

[Alliez et al., 2007] Alliez, P., Cohen-Steiner, D., Tong, Y., and Desbrun, M. (2007). Voronoi-based variational reconstruction of unoriented point sets. In *Symposium on Geometry processing*, volume 7, pages 39–48.

[Bailer et al., 2012] Bailer, C., Finckh, M., and Lensch, H. P. (2012). Scale robust multi view stereo. In *European Conference on Computer Vision*, pages 398–411. Springer.

[Barnes et al., 2009] Barnes, C., Shechtman, E., Finkelstein, A., and Goldman, D. B. (2009). Patchmatch: A randomized correspondence algorithm for structural image editing. In *ACM Transactions on Graphics (ToG)*, volume 28, page 24. ACM.

[Besse, 2013] Besse, F. O. (2013). *PatchMatch Belief Propagation for Correspondence Field Estimation and its Applications*. PhD thesis, University College London (UCL).

[Bethmann and Luhmann, 2014] Bethmann, F. and Luhmann, T. (2014). Object-based multi-image semi-global matching–concept and first results. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 45.

[Birchfield and Tomasi, 1998] Birchfield, S. and Tomasi, C. (1998). A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):401–406.

[Blaha et al., 2017] Blaha, M., Rothermel, M., Oswald, M. R., Sattler, T., Richard, A., Wegner, J. D., Pollefeys, M., and Schindler, K. (2017). Semantically informed multiview surface refinement. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3819–3827.

[Bleyer et al., 2011] Bleyer, M., Rhemann, C., and Rother, C. (2011). Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pages 1–11.

[Bosch et al., 2016] Bosch, M., Kurtz, Z., Hagstrom, S., and Brown, M. (2016). A multiple view stereo benchmark for satellite imagery. In *Proceedings of the IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–9.

[Bradski, 2000] Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

[Brown et al., 2018] Brown, M., Goldberg, H., Foster, K., Leichtman, A., Wang, S., Hagstrom, S., Bosch, M., and Almes, S. (2018). Large-scale public lidar and satellite image data set for urban semantic labeling. In *Proceedings of Laser Radar Technology and Applications XXIII*, volume 10636, pages 154–167.

[Campbell et al., 2008] Campbell, N. D., Vogiatzis, G., Hernández, C., and Cipolla, R. (2008). Using multiple hypotheses to improve depth-maps for multi-view stereo. In *European Conference on Computer Vision*, pages 766–779. Springer.

[Capaldo et al., 2012] Capaldo, P., Crespi, M., Fratarcangeli, F., Nascetti, A., and Pieralice, F. (2012). Dsm generation from high resolution imagery: applications with worldview-1 and geoeye-1. *Italian Journal of Remote Sensing/Rivista Italiana di Telerilevamento*, 44(1).

[Chang and Chen, 2018] Chang, J.-R. and Chen, Y.-S. (2018). Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418.

[Chen et al., 2019] Chen, B., Qin, R., Huang, X., Song, S., and Lu, X. (2019). A comparison of stereo-matching cost between convolutional neural network and census for satellite images. *arXiv preprint arXiv:1905.09147*.

[Curless and Levoy, 1996] Curless, B. and Levoy, M. (1996). A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312.

[d'Angelo, 2016] d'Angelo, P. (2016). Improving semi-global matching: Cost aggregation and confidence measure. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41:299–304.

[d'Angelo and Kuschk, 2012] d'Angelo, P. and Kuschk, G. (2012). Dense multi-view stereo from satellite imagery. In *Proceedings of 2012 IEEE International Geoscience and Remote Sensing Symposium*, pages 6944–6947.

[d'Angelo et al., 2014] d'Angelo, P., Rossi, C., Minet, C., Eineder, M., Flory, M., and Niemeyer, I. (2014). High resolution 3d earth observation data analysis for safeguards activities. In *Symposium on International Safeguards*, pages 1–8.

[De Franchis et al., 2014] De Franchis, C., Meinhardt-Llopis, E., Michel, J., Morel, J.-M., and Facciolo, G. (2014). An automatic and modular stereo pipeline for pushbroom images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2:49–56.

[Delaunoy and Prados, 2011] Delaunoy, A. and Prados, E. (2011). Gradient flows for optimizing triangular mesh-based surfaces: Applications to 3d reconstruction problems dealing with visibility. *International journal of computer vision*, 95(2):100–123.

[Delaunoy et al., 2008] Delaunoy, A., Prados, E., Piracés, P. G. I., Pons, J.-P., and Sturm, P. (2008). Minimizing the multi-view stereo reprojection error for triangular surface meshes. In *Proceedings of BMVC 2008-British Machine Vision Conference*, pages 1–10.

[Di Rita et al., 2017] Di Rita, M., Nascetti, A., and Crespi, M. (2017). Open source tool for dsms generation from high resolution optical satellite imagery: development and testing of an ossim plug-in. *International journal of remote sensing*, 38(7):1788–1808.

[Dial et al., 2003] Dial, G., Bowen, H., Gerlach, F., Grodecki, J., and Oleszczuk, R. (2003). Ikonos satellite, imagery, and products. *Remote sensing of Environment*, 88(1-2):23–36.

[Dial and Grodecki, 2002] Dial, G. and Grodecki, J. (2002). Block adjustment with rational polynomial camera models. In *Proceedings of ASPRS 2002 Conference, Washington, DC*, pages 22–26.

[Drory et al., 2014] Drory, A., Haubold, C., Avidan, S., and Hamprecht, F. A. (2014). Semi-global matching: a principled derivation in terms of message passing. In *Proceedings of German Conference on Pattern Recognition*, pages 43–53.

[Duan et al., 2016] Duan, Y., Huang, X., Xiong, J., Zhang, Y., and Wang, B. (2016). A combined image matching method for chinese optical satellite imagery. *International journal of digital earth*, 9(9):851–872.

[d'Angelo and Reinartz, 2011] d'Angelo, P. and Reinartz, P. (2011). Semiglobal matching results on the isprs stereo matching benchmark. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38:79–84.

[Esteban and Schmitt, 2004] Esteban, C. H. and Schmitt, F. (2004). Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367–392.

[Facciolo et al., 2015] Facciolo, G., De Franchis, C., and Meinhardt, E. (2015). Mgm: A significantly more global matching for stereovision. In *Proceedings of BMVC 2015-British Machine Vision Conference*, pages 1–12.

[Facciolo et al., 2017] Facciolo, G., De Franchis, C., and Meinhardt-Llopis, E. (2017). Automatic 3d reconstruction from multi-date satellite images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 57–66.

[Fraser et al., 2006] Fraser, C., Dial, G., and Grodecki, J. (2006). Sensor orientation via RPCs. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60(3):182–194.

[Fraser and Hanley, 2003] Fraser, C. and Hanley, H. (2003). Bias compensation in rational functions for ikonos satellite imagery. *Photogrammetric Engineering & Remote Sensing*, 69(1):53–57.

[Fraser and Hanley, 2005] Fraser, C. and Hanley, H. (2005). Bias-compensated rpcs for sensor orientation of high-resolution satellite imagery. *Photogrammetric Engineering & Remote Sensing*, 71(8):909–915.

[Fraser et al., 2002a] Fraser, C., Hanley, H., and Yamakawa, T. (2002a). High-precision geopositioning from ikonos satellite imagery. In *Proceedings ASPRS Annual Meeting, Washington DC*, pages 22–26.

[Fraser et al., 2002b] Fraser, C., Hanley, H., and Yamakawa, T. (2002b). Three-dimensional geopositioning accuracy of ikonos imagery. *The Photogrammetric Record*, 17(99):465–479.

[Fraser and Yamakawa, 2004] Fraser, C. and Yamakawa, T. (2004). Insights into the affine model for high-resolution satellite sensor orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(5-6):275–288.

[Fuhrmann and Goesele, 2011] Fuhrmann, S. and Goesele, M. (2011). Fusion of depth maps with multiple scales. *ACM Transactions on Graphics (TOG)*, 30(6):1–8.

[Fuhrmann and Gösele, 2014] Fuhrmann, S. and Gösele, M. (2014). Floating scale surface reconstruction. *ACM Transactions on Graphics (ToG)*, 33(4):46:1–46:11.

[Furukawa et al., 2015] Furukawa, Y., Hernández, C., et al. (2015). Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148.

[Furukawa and Ponce, 2010] Furukawa, Y. and Ponce, J. (2010). Accurate, dense, and robust multi-view stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376.

[Fusiello et al., 2000] Fusiello, A., Trucco, E., and Verri, A. (2000). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22.

[Galliani et al., 2016] Galliani, S., Lasinger, K., and Schindler, K. (2016). Gipuma: Massively parallel multi-view stereo reconstruction. *Publikationen der Deutschen Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e. V*, 25:361–369.

[Gargallo et al., 2007] Gargallo, P., Prados, E., and Sturm, P. (2007). Minimizing the reprojection error in surface reconstruction from images. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE.

[Gehrig and Franke, 2007] Gehrig, S. K. and Franke, U. (2007). Improving stereo sub-pixel accuracy for long range stereo. In *Proceedings of 2007 IEEE 11th International Conference on Computer Vision*, pages 1–7.

[Ghuffar, 2016] Ghuffar, S. (2016). Satellite stereo based digital surface model generation using semi global matching in object and image space. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3:63.

[Girshick, 2015] Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

[Goesele et al., 2006] Goesele, M., Curless, B., and Seitz, S. M. (2006). Multi-view stereo revisited. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2402–2409. IEEE.

[Goldberg et al., 2018] Goldberg, H. R., Wang, S., Christie, G. A., and Brown, M. Z. (2018). Urban 3d challenge: building footprint detection using orthorectified imagery and digital surface models from commercial satellites. In *Geospatial Informatics, Motion Imagery, and Network Analytics VIII*, volume 10645, page 1064503. International Society for Optics and Photonics.

[Gong and Fritsch, 2016] Gong, K. and Fritsch, D. (2016). A detailed study about digital surface model generation using high resolution satellite stereo imagery. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 3(1).

[Gong and Fritsch, 2017] Gong, K. and Fritsch, D. (2017). Relative orientation and modified piecewise epipolar resampling for high resolution satellite images. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 42.

[Gong and Fritsch, 2018] Gong, K. and Fritsch, D. (2018). Point cloud and digital surface model generation from high resolution multiple view stereo satellite imagery. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42(2).

[Gong and Fritsch, 2019] Gong, K. and Fritsch, D. (2019). DSM generation from high resolution multi-view stereo satellite imagery. *Photogrammetric Engineering & Remote Sensing*, 85:379–387.

[Grodecki, 2001] Grodecki, J. (2001). Ikonos stereo feature extraction–rpc approach. In *ASPRS annual conference St. Louis*.

[Grodecki and Dial, 2003] Grodecki, J. and Dial, G. (2003). Block adjustment of high-resolution satellite images described by rational polynomials. *Photogrammetric Engineering & Remote Sensing*, 69(1):59–68.

[Haala, 2013] Haala, N. (2013). The landscape of dense image matching algorithms.

[Habib et al., 2004] Habib, A., Kim, E., Morgan, M., and Couloigner, I. (2004). Dem generation from high resolution satellite imagery using parallel projection model. In *Proceeding of the XXth ISPRS Congress, Commission*, volume 1, pages 393–398. Citeseer.

[Habib et al., 2005] Habib, A. F., Morgan, M. F., Jeong, S., and Kim, K.-O. (2005). Epipolar geometry of line cameras moving with constant velocity and attitude. *ETRI journal*, 27(2):172–180.

[Hanley and Fraser, 2004] Hanley, H. and Fraser, C. (2004). Sensor orientation for high-resolution satellite imagery: Further insights into bias-compensated rpcs. In *Congress of International Society for Photogrammetry and Remote Sensing (ISPRS), Commission I, Working Group I/2*, pages 12–23.

[Hanley et al., 2002] Hanley, H., Yamakawa, T., and Fraser, C. (2002). Sensor orientation for high-resolution satellite imagery. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(1):69–75.

[Hartley and Saxena, 1997] Hartley, R. I. and Saxena, T. (1997). The cubic rational polynomial camera model. In *Proceedings of the DARPA Image Understanding Workshop*, pages 649–653.

[Heise et al., 2015] Heise, P., Jensen, B., Klose, S., and Knoll, A. (2015). Variational patchmatch multiview reconstruction and refinement. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 882–890.

[Hernández et al., 2007] Hernández, C., Vogiatzis, G., and Cipolla, R. (2007). Probabilistic visibility for multi-view stereo. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE.

[Hirschmüller, 2008] Hirschmüller, H. (2008). Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341.

[Hornung and Kobbelt, 2006] Hornung, A. and Kobbelt, L. (2006). Robust reconstruction of watertight 3 d models from non-uniformly sampled point clouds without normal information. In *Symposium on geometry processing*, pages 41–50. Citeseer.

[ISPRS, 2010] ISPRS (2010). Benchmarking and quality analysis of dem generated from high and very high resolution optical stereo satellite data. `http://isprswgi4.altervista.org/tests_datasets.html#Benchmark`. [Online; accessed 25-February-2010].

[Jancosek and Pajdla, 2011] Jancosek, M. and Pajdla, T. (2011). Multi-view reconstruction preserving weakly-supported surfaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 3121–3128.

[Kazhdan et al., 2006] Kazhdan, M., Bolitho, M., and Hoppe, H. (2006). Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7.

[Kazhdan and Hoppe, 2013] Kazhdan, M. and Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):29:1–29:13.

[Kim, 2000] Kim, T. (2000). A study on the epipolarity of linear pushbroom images. *Photogrammetric Engineering & Remote Sensing*, 62(8):961–966.

[Knapitsch et al., 2017] Knapitsch, A., Park, J., Zhou, Q.-Y., and Koltun, V. (2017). Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4):78:1–78:13.

[Kobbelt et al., 1998] Kobbelt, L., Campagna, S., Vorsatz, J., and Seidel, H.-P. (1998). Interactive multi-resolution modeling on arbitrary meshes. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 105–114.

[Koh and Yang, 2016] Koh, J.-W. and Yang, H.-S. (2016). Unified piecewise epipolar resampling method for pushbroom satellite images. *EURASIP Journal on Image and Video Processing*, 2016(1):11.

[Kuschk, 2013] Kuschk, G. (2013). Large scale urban reconstruction from remote sensing imagery. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40:139–146.

[Kuschk et al., 2017] Kuschk, G., d'Angelo, P., Gaudrie, D., Reinartz, P., and Cremers, D. (2017). Spatially regularized fusion of multiresolution digital surface models. *IEEE Transactions on Geoscience and Remote Sensing*, 55(3):1477–1488.

[Labatut et al., 2007] Labatut, P., Pons, J.-P., and Keriven, R. (2007). Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *2007 IEEE 11th international conference on computer vision*, pages 1–8. IEEE.

[Labatut et al., 2009] Labatut, P., Pons, J.-P., and Keriven, R. (2009). Robust and efficient surface reconstruction from range data. *Computer graphics forum*, 28(8):2275–2290.

[Leotta et al., 2019] Leotta, M. J., Long, C., Jacquet, B., Zins, M., Lipsa, D., Shan, J., Xu, B., Li, Z., Zhang, X., Chang, S.-F., et al. (2019). Urban semantic 3d reconstruction from multiview satellite imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0.

[Li et al., 2016] Li, S., Siu, S. Y., Fang, T., and Quan, L. (2016). Efficient multi-view surface refinement with adaptive resolution control. In *Proceedings of European Conference on Computer Vision*, pages 349–364.

[Li et al., 2015] Li, Z., Wang, K., Zuo, W., Meng, D., and Zhang, L. (2015). Detail-preserving and content-aware variational multi-view stereo reconstruction. *IEEE Transactions on Image Processing*, 25(2):864–877.

[Loop and Zhang, 1999] Loop, C. and Zhang, Z. (1999). Computing rectifying homographies for stereo vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 125–131.

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee.

[Mayer et al., 2012] Mayer, H., Bartelsen, J., Hirschmüller, H., and Kuhn, A. (2012). Dense 3d reconstruction from wide baseline image sets. In *Outdoor and Large-Scale Real-World Scene Analysis*, pages 285–304. Springer.

[Moratto et al., 2010] Moratto, Z. M., Broxton, M. J., Beyer, R. A., Lundy, M., and Husmann, K. (2010). Ames stereo pipeline, nasa's open source automated stereogrammetry software. In *Proceedings of Lunar and Planetary Science Conference*, volume 41, page 2364.

[Morgan et al., 2004a] Morgan, M., Kim, K., Jeong, S.-Y., and Habib, A. W. (2004a). Epipolar geometry of linear array scanners moving with constant velocity and constant attitude. In *XXth Congress of ISPRS*, volume 1024.

[Morgan et al., 2004b] Morgan, M., Kim, K., Jeong, S.-Y., and Habib, A. W. (2004b). Indirect epipolar resampling of scenes using parallel projection modeling of linear array scanners. In *XXth Congress of ISPRS*, pages 12–23.

[Morgan et al., 2006] Morgan, M., Kim, K.-O., Jeong, S., and Habib, A. (2006). Epipolar resampling of space-borne linear array scanner scenes using parallel projection. *Photogrammetric Engineering & Remote Sensing*, 72(11):1255–1263.

[Morgan, 2004] Morgan, M. F. (2004). *Epipolar resampling of linear array scanner scenes*. PhD thesis, University of Calgary, Department of Geomatics Engineering.

[Noguchi et al., 2004] Noguchi, M., Fraser, C. S., Nakamura, T., Shimono, T., and Oki, S. (2004). Accuracy assessment of quickbird stereo imagery. *The Photogrammetric Record*, 19(106):128–137.

[Oh, 2011] Oh, J. (2011). *Novel Approach to Epipolar Resampling of HRSI and Satellite Stereo Imagery-based Georeferencing of Aerial Images*. PhD thesis, The Ohio State University.

[Ono, 1999] Ono, T. (1999). Epipolar resampling of high resolution satellite imagery.

[Ozcanli et al., 2015] Ozcanli, O. C., Dong, Y., Mundy, J. L., Webb, H., Hammoud, R., and Tom, V. (2015). A comparison of stereo and multiview 3-d reconstruction using cross-sensor satellite imagery. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 17–25.

[Pollard and Mundy, 2007] Pollard, T. and Mundy, J. L. (2007). Change detection in a 3-d world. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6.

[Pollard et al., 2010] Pollard, T. B., Eden, I., Mundy, J. L., and Cooper, D. B. (2010). A volumetric approach to change detection in satellite images. *Photogrammetric Engineering & Remote Sensing*, 76(7):817–831.

[Pollefeys et al., 1999] Pollefeys, M., Koch, R., and Van Gool, L. (1999). A simple and efficient rectification method for general motion. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 1, pages 496–501.

[Pons et al., 2007] Pons, J.-P., Keriven, R., and Faugeras, O. (2007). Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *International Journal of Computer Vision*, 72(2):179–193.

[Qi et al., 2017] Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660.

[Qin, 2017] Qin, R. (2017). Automated 3d recovery from very high resolution multi-view satellite images. *Computing Research Repository*, abs/1905.07475.

[Reinartz et al., 2010] Reinartz, P., d'Angelo, P., Krauß, T., Poli, D., Jacobsen, K., and Buyuksalih, G. (2010). Benchmarking and quality analysis of dem generated from high and very high resolution optical stereo satellite data. In *Proceedings of The 2010 Canadian Geomatics Conference and Symposium of Commission I, ISPRS Convergence in Geomatics – Shaping Canada's Competitive Landscape*, volume 38, pages 1–6.

[Romanoni et al., 2017] Romanoni, A., Ciccone, M., Visin, F., and Matteucci, M. (2017). Multi-view stereo with single-view semantic mesh refinement. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 706–715.

[Roth and Mayer, 2019] Roth, L. and Mayer, H. (2019). Reduction of the fronto-parallel bias for wide-baseline semi-global matching. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:69–76.

[Rothermel, 2017] Rothermel, M. (2017). *Development of a SGM-based multi-view reconstruction framework for aerial imagery.* PhD thesis, University of Stuttgart.

[Rothermel et al., 2020] Rothermel, M., Gong, K., Fritsch, D., Schindler, K., and Haala, N. (2020). Photometric multi-view mesh refinement for high-resolution satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166:52–62.

[Rothermel et al., 2012] Rothermel, M., Wenzel, K., Fritsch, D., and Haala, N. (2012). Sure: Photogrammetric surface reconstruction from imagery. In *Proceedings of LowCost 3D Workshop Berlin*, volume 2, pages 1–9.

[Rupnik et al., 2017] Rupnik, E., Daakir, M., and Deseilligny, M. P. (2017). Micmac–a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*, 2(1):14.

[Rupnik et al., 2018] Rupnik, E., Pierrot-Deseilligny, M., and Delorme, A. (2018). 3d reconstruction from multi-view vhr-satellite images in micmac. *ISPRS Journal of Photogrammetry and Remote Sensing*, 139:201–211.

[Scharstein et al., 2017] Scharstein, D., Taniai, T., and Sinha, S. N. (2017). Semi-global stereo matching with surface orientation priors. In *Proceedings of International Conference on 3D Vision (3DV)*, pages 215–224.

[Schönberger et al., 2016] Schönberger, J. L., Zheng, E., Pollefeys, M., and Frahm, J.-M. (2016). Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of European Conference on Computer Vision*, pages 501–518.

[Schöps et al., 2017] Schöps, T., Schönberger, J. L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., and Geiger, A. (2017). A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of IEEE conference on Computer Vision and Pattern Recognition*, pages 3260–3269.

[Seitz et al., 2006] Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., and Szeliski, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 519–528.

[Shean et al., 2016] Shean, D. E., Alexandrov, O., Moratto, Z. M., Smith, B. E., Joughin, I. R., Porter, C., and Morin, P. (2016). An automated, open-source pipeline for mass production of digital elevation models (dems) from very-high-resolution commercial stereo satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 116:101–117.

[Shimizu and Okutomi, 2002] Shimizu, M. and Okutomi, M. (2002). Precise subpixel estimation on area-based matching. *Systems and Computers in Japan*, 33(7):1–10.

[Sinha et al., 2007] Sinha, S. N., Mordohai, P., and Pollefeys, M. (2007). Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE.

[Soatto et al., 2003] Soatto, S., Yezzi, A., and Jin, H. (2003). Tales of shape and radiance in multiview stereo. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 974–981. IEEE.

[Solem and Overgaard, 2005] Solem, J. E. and Overgaard, N. C. (2005). A geometric formulation of gradient descent for variational problems with moving surfaces. In *Proceedings of International Conference on Scale-Space and PDE Methods in Computer Vision*, pages 419–430.

[Stewénius, 2005] Stewénius, H. (2005). *Gröbner basis methods for minimal problems in computer vision*. Citeseer.

[Szeliski and Scharstein, 2004] Szeliski, R. and Scharstein, D. (2004). Sampling the disparity space image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:419–425.

[Treible et al., 2018] Treible, W., Sorensen, S., Gilliam, A. D., Kambhamettu, C., and Mundy, J. L. (2018). Learning dense stereo matching for digital surface models from satellite imagery. *arXiv preprint arXiv:1811.03535*.

[Tyle_ek and Šára, 2010] Tyle_ek, R. and Šára, R. (2010). Refinement of surface mesh for accurate multi-view reconstruction. *International Journal of Virtual Reality*, 9(1):45–54.

[Ummenhofer and Brox, 2015] Ummenhofer, B. and Brox, T. (2015). Global, dense multiscale reconstruction for a billion points. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1341–1349.

[Vogiatzis et al., 2005] Vogiatzis, G., Torr, P. H., and Cipolla, R. (2005). Multi-view stereo via volumetric graph-cuts. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 391–398. IEEE.

[Vu et al., 2012] Vu, H.-H., Labatut, P., Pons, J.-P., and Keriven, R. (2012). High accuracy and visibility-consistent dense multiview stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):889–901.

[Wang and Frahm, 2017] Wang, K. and Frahm, J.-M. (2017). Fast and accurate satellite multi-view stereo using edge-aware interpolation. In *Proceedings of International Conference on 3D Vision (3DV)*, pages 365–373.

[Wang et al., 2016] Wang, K., Stutts, C., Dunn, E., and Frahm, J.-M. (2016). Efficient joint stereo estimation and land usage classification for multiview satellite data. In *Proceedings of IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9.

[Wang et al., 2010] Wang, M., Hu, F., and Li, J. (2010). Epipolar arrangement of satellite imagery by projection trajectory simplification. *The Photogrammetric Record*, 25(132):422–436.

[Wang et al., 2011] Wang, M., Hu, F., and Li, J. (2011). Epipolar resampling of linear pushbroom satellite imagery by a new epipolarity model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3):347–355.

[Wohlfeil et al., 2012] Wohlfeil, J., Hirschmüller, H., Piltz, B., Börner, A., and Suppa, M. (2012). Fully automated generation of accurate digital surface models with sub-meter resolution from satellite imagery. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39:75–80.

[Wolf and Dewitt, 2000] Wolf, P. R. and Dewitt, B. A. (2000). *Elements of photogrammetry: with applications in GIS*, volume 3. McGraw-Hill New York.

[Woodford et al., 2009] Woodford, O., Torr, P., Reid, I., and Fitzgibbon, A. (2009). Global stereo reconstruction under second-order smoothness priors. *IEEE transactions on pattern analysis and machine intelligence*, 31(12):2115–2128.

[Xu et al., 2020] Xu, B., Zhang, X., Li, Z., Leotta, M., Chang, S.-F., and Shan, J. (2020). Deep learning guided building reconstruction from satellite imagery-derived point clouds. *arXiv preprint arXiv:2005.09223*.

[Yezzi and Soatto, 2003] Yezzi, A. and Soatto, S. (2003). Stereoscopic segmentation. *International Journal of Computer Vision*, 53(1):31–43.

[Zach, 2008] Zach, C. (2008). Fast and high quality fusion of depth maps. In *Proceedings of the international symposium on 3D data processing, visualization and transmission (3DPVT)*, volume 1. Citeseer.

[Zach et al., 2007] Zach, C., Pock, T., and Bischof, H. (2007). A globally optimal algorithm for robust tv-l 1 range image integration. In *Proceedings of IEEE International Conference on Computer Vision*, pages 1–8.

[Zeng et al., 2018] Zeng, H., Wu, J., and Furukawa, Y. (2018). Neural procedural reconstruction for residential buildings. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 737–753.

[Zhao et al., 2008] Zhao, D., Yuan, X., and Liu, X. (2008). Epipolar line generation from ikonos imagery based on rational function model. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(B4):1293–1297.

[Zheng et al., 2014] Zheng, E., Dunn, E., Jojic, V., and Frahm, J.-M. (2014). Patchmatch based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517.

[Zheng et al., 2015] Zheng, E., Wang, K., Dunn, E., and Frahm, J.-M. (2015). Minimal solvers for 3d geometry from satellite imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 738–746.

Appendix

The evaluation of our relative RPC compensation method for all the stereo pairs are presented here. Table 5.1 shows the RMSEs of the relative pointing errors before and after the compensation in the Munich test site. The RMSEs of object coordinates of the check points in Munich test site are demonstrated in Table 5.2. For all stereo pairs in the San Fernando test site 1, the evaluation results of the relative pointing error and object coordinates are presented in Table 5.3 and 5.4. The relative pointing error evaluation of the San Fernando test site 2 is shown in Table 5.5. At last, we present the accuracy of the check points object coordinates of San Fernando test site 2 in Table 5.4.

Table 5.1: The relative pointing error evaluation for all stereo pairs in the Munich test site

	Uncompensated [pixel]	Compensated [pixel]
image 1&2	0.339	0.303
image 1&3	0.629	0.563
image 1&4	0.691	0.592
image 2&3	0.635	0.694
image 2&4	0.601	0.557
image 3&4	0.527	0.565

Table 5.2: The check points evaluation for all stereo pairs in the Munich test site

	latitude [m]	longitude [m]	height [m]
image 1&2	0.050	0.095	0.748
image 1&3	0.287	0.109	0.701
image 1&4	0.184	0.069	0.509
image 2&3	0.234	0.085	0.999
image 2&4	0.241	0.073	0.541
image 3&4	0.285	0.496	0.732

Table 5.3: The relative pointing error evaluation for all stereo pairs in the San Fernando test site 1

	Uncompensated [pixel]	Compensated [pixel]
image 1&2	2.42	0.51
image 1&3	2.245	0.534
image 1&4	0.512	0.407

image 1&5	0.466	0.432
image 1&6	7.226	0.518
image 1&7	4.141	0.521
image 1&8	1.858	0.379
image 1&15	2.497	0.43
image 1&16	2.227	0.409
image 1&18	0.599	0.5
image 1&19	5.909	0.278
image 1&25	6.146	0.432
image 1&26	7.396	0.447
image 1&28	5.169	0.386
image 1&29	4.868	0.414
image 1&30	3.646	0.398
image 1&31	2.844	0.401
image 1&32	1.403	0.387
image 1&33	2.143	0.361
image 1&39	3.86	0.515
image 1&40	4.361	0.531
image 1&42	1.419	0.404
image 1&49	1.889	0.58
image 2&4	1.588	0.405
image 2&5	1.919	0.424
image 2&6	9.332	0.3
image 2&7	1.569	0.73
image 2&8	3.914	0.467
image 2&18	4.596	0.815
image 2&32	3.716	0.649
image 2&33	4.729	0.565
image 2&39	7.195	0.589
image 2&49	6.53	0.617
image 3&4	1.538	0.4
image 3&5	1.868	0.441
image 3&6	8.343	0.34
image 3&7	2.192	0.733
image 3&8	3.571	0.616
image 3&18	4.066	0.737
image 3&32	3.18	0.594
image 3&33	4.261	0.499
image 3&39	6.615	0.676
image 3&49	5.569	0.55
image 4&7	2.355	0.754
image 4&15	1.918	0.406
image 4&16	1.093	0.391

image 4&19	4.409	0.707
image 4&25	7.853	0.5
image 4&29	3.559	0.658
image 4&30	4.619	0.747
image 4&33	6.144	0.549
image 4&39	4.589	0.581
image 4&40	6.124	0.86
image 4&42	1.681	0.537
image 4&43	1.766	0.515
image 5&7	1.386	0.812
image 5&15	1.931	0.418
image 5&19	3.479	0.733
image 5&25	6.571	0.542
image 5&29	3.382	0.64
image 5&30	4.235	0.706
image 5&32	4.478	0.414
image 5&33	4.93	0.603
image 5&39	3.335	0.569
image 5&40	4.967	0.911
image 5&42	1.309	0.569
image 5&43	1.804	0.501
image 6&7	3.256	0.525
image 6&8	3.423	0.405
image 6&11	1.398	0.523
image 6&15	3.506	0.419
image 6&16	8.999	0.415
image 6&18	13.307	0.626
image 6&19	6.543	0.583
image 6&25	4.397	0.433
image 6&28	2.554	0.415
image 6&29	0.472	0.396
image 6&30	3.367	0.488
image 6&31	2.798	0.573
image 6&32	2.62	0.514
image 6&33	2.932	0.486
image 6&35	3.354	0.401
image 6&39	5.968	0.563
image 6&40	3.653	0.782
image 6&41	5.021	0.391
image 6&42	5.403	0.343
image 6&43	2.674	0.518
image 6&44	6.556	0.365
image 6&48	1.17	0.41

image 7&15	5.272	0.574
image 7&28	4.918	0.478
image 7&29	5.719	0.495
image 7&35	1.337	0.628
image 7&48	5.797	0.613
image 7&49	6.582	0.581
image 8&15	1.081	0.402
image 8&16	3.326	0.368
image 8&19	1.619	0.579
image 8&25	2.085	0.427
image 8&26	3.246	0.521
image 8&28	1.709	0.495
image 8&29	0.878	0.38
image 8&30	1.926	0.465
image 8&42	1.256	0.429
image 8&43	0.876	0.506
image 8&49	4.216	0.738
image 10&11	1.123	0.736
image 10&19	0.716	0.828
image 10&28	4.553	0.42
image 10&29	3.369	0.609
image 10&30	2.825	0.78
image 10&35	2.701	0.743
image 10&40	2.639	0.832
image 10&41	5.065	0.567
image 10&42	0.409	0.543
image 11&16	2.274	0.462
image 11&19	2.432	0.395
image 11&25	5.145	0.455
image 11&39	1.588	0.594
image 11&40	1.074	0.383
image 11&44	2.603	0.468
image 11&49	0.904	0.533
image 15&16	3.149	0.624
image 15&18	3.547	0.528
image 15&19	4.898	0.542
image 15&31	0.589	0.552
image 15&32	0.431	0.416
image 15&33	1.519	0.439
image 15&39	3.447	0.545
image 15&40	2.018	0.777
image 15&49	0.486	0.461
image 16&18	2.442	0.453

image 16&19	0.419	0.385
image 16&26	3.721	0.302
image 16&29	2.506	0.453
image 16&30	4.721	0.504
image 16&31	1.874	0.455
image 16&35	0.881	0.594
image 16&41	2.053	0.403
image 16&42	4.096	0.519
image 16&43	1.889	0.578
image 16&48	2.355	0.38
image 16&49	0.626	0.496
image 18&19	8.493	0.538
image 18&25	7.181	0.517
image 18&26	8.471	0.403
image 18&28	5.644	0.544
image 18&29	5.335	0.476
image 18&30	7.465	0.505
image 18&35	3.19	0.673
image 18&42	5.847	0.492
image 18&43	2.437	0.508
image 18&48	4.396	0.614
image 18&49	4.576	0.773
image 19&25	1.502	0.465
image 19&28	3.221	0.52
image 19&29	3.262	0.385
image 19&31	0.644	0.462
image 19&32	2.631	0.428
image 19&33	3.218	0.524
image 19&35	6.318	0.416
image 19&41	0.744	0.406
image 19&42	5.936	0.38
image 19&43	3.645	0.386
image 19&44	2.565	0.534
image 19&48	4.144	0.402
image 19&49	0.96	0.532
image 25&26	0.996	0.341
image 25&30	5.883	0.449
image 25&31	4.391	0.446
image 25&32	3.738	0.317
image 25&35	5.183	0.535
image 25&40	1.095	0.566
image 25&41	5.629	0.423
image 25&42	4.599	0.449

image 25&48	3.368	0.319
image 25&49	2.742	0.44
image 26&28	2.306	0.605
image 26&32	2.824	0.653
image 26&33	2.969	0.618
image 26&39	4.592	0.782
image 26&40	1.527	0.765
image 26&41	0.704	0.507
image 26&43	7.114	0.587
image 26&44	9.745	0.601
image 28&30	1.487	0.419
image 28&31	0.567	0.481
image 28&32	0.607	0.479
image 28&40	1.765	0.842
image 28&43	2.636	0.483
image 28&49	1.793	0.409
image 29&31	0.962	0.347
image 29&32	0.87	0.505
image 29&33	0.852	0.398
image 29&39	2.404	0.601
image 29&40	1.125	0.749
image 29&49	2.649	0.553
image 30&32	1.454	0.508
image 30&33	2.743	0.454
image 30&35	3.191	0.222
image 30&39	5.047	0.605
image 30&40	2.661	0.57
image 30&49	3.268	0.546
image 31&33	1.615	0.313
image 31&35	5.087	0.467
image 31&42	1.929	0.364
image 31&48	0.483	0.477
image 31&49	4.518	0.662
image 32&35	5.587	0.508
image 32&42	1.533	0.256
image 32&43	0.805	0.474
image 32&48	0.581	0.396
image 32&49	5.084	0.709
image 33&40	0.812	0.845
image 33&42	0.377	0.356
image 33&43	1.008	0.363
image 33&48	1.639	0.492
image 35&39	3.827	0.63

image 35&42	1.647	0.523
image 35&44	0.762	0.471
image 35&49	5.813	0.348
image 39&41	1.32	0.616
image 39&42	3.18	0.57
image 39&43	2.472	0.644
image 39&48	0.792	0.511
image 39&49	1.963	0.65
image 40&41	0.682	0.266
image 40&42	3.027	0.548
image 40&43	1.292	0.654
image 40&44	3.352	0.656
image 40&48	1.222	0.463
image 40&49	3.828	0.47
image 41&44	7.015	0.336
image 41&48	6.038	0.425
image 41&49	3.189	0.568
image 42&44	0.632	0.327
image 42&49	0.58	0.576
image 43&49	0.982	0.618
image 44&48	2.152	0.348
image 44&49	3.003	0.423

Table 5.4: The check points evaluation for all stereo pairs in the San Fernando test site 1

	latitude [m]	longitude [m]	height [m]
image 1&2	0.468	0.065	0.802
image 1&3	0.463	0.095	0.687
image 1&4	0.33	0.102	0.676
image 1&5	0.422	0.112	0.576
image 1&6	0.108	0.054	0.501
image 1&7	0.39	0.118	0.989
image 1&8	0.386	0.121	0.678
image 1&15	0.6	0.111	0.644
image 1&16	0.148	0.092	0.644
image 1&18	0.299	0.112	0.684
image 1&19	0.185	0.05	0.785
image 1&25	0.183	0.066	0.657
image 1&26	0.276	0.14	0.724
image 1&28	0.474	0.091	0.613
image 1&29	0.34	0.073	0.9
image 1&30	0.573	0.065	0.983
image 1&31	0.264	0.11	0.826

image 1&32	0.325	0.132	0.632
image 1&33	0.36	0.102	0.553
image 1&39	0.347	0.195	0.565
image 1&40	0.462	0.112	0.808
image 1&42	0.32	0.094	0.731
image 1&49	0.307	0.135	0.552
image 2&4	0.445	0.072	0.572
image 2&5	0.445	0.077	0.462
image 2&6	0.142	0.129	0.864
image 2&7	0.307	0.136	0.85
image 2&8	0.245	0.071	0.565
image 2&18	0.235	0.095	0.61
image 2&32	0.268	0.085	0.626
image 2&33	0.343	0.074	0.446
image 2&39	0.304	0.138	0.59
image 2&49	0.315	0.182	0.758
image 3&4	0.496	0.115	0.516
image 3&5	0.53	0.108	0.44
image 3&6	0.144	0.152	0.99
image 3&7	0.303	0.146	0.881
image 3&8	0.324	0.062	0.592
image 3&18	0.276	0.104	0.639
image 3&32	0.299	0.091	0.632
image 3&33	0.4	0.054	0.443
image 3&39	0.343	0.141	0.643
image 3&49	0.327	0.24	0.998
image 4&7	0.264	0.305	0.964
image 4&15	0.366	0.144	0.427
image 4&16	0.164	0.239	0.977
image 4&19	0.108	0.061	0.811
image 4&25	0.222	0.15	0.774
image 4&29	0.2	0.079	0.723
image 4&30	0.259	0.078	0.631
image 4&33	0.285	0.259	0.778
image 4&39	0.424	0.375	0.813
image 4&40	0.465	0.182	0.624
image 4&42	0.233	0.134	0.455
image 4&43	0.436	0.138	0.782
image 5&7	0.286	0.302	0.995
image 5&15	0.409	0.137	0.384
image 5&19	0.087	0.065	0.791
image 5&25	0.215	0.169	0.794
image 5&29	0.217	0.082	0.67

image 5&30	0.301	0.076	0.588
image 5&32	0.277	0.238	0.988
image 5&33	0.274	0.271	0.828
image 5&39	0.418	0.342	0.737
image 5&40	0.46	0.185	0.568
image 5&42	0.246	0.123	0.428
image 5&43	0.46	0.14	0.691
image 6&7	0.121	0.129	0.76
image 6&8	0.141	0.068	0.46
image 6&11	0.167	0.059	0.573
image 6&15	0.134	0.133	0.771
image 6&16	0.099	0.056	0.264
image 6&18	0.119	0.076	0.382
image 6&19	0.001	0.002	0.032
image 6&25	0.125	0.071	0.419
image 6&28	0.149	0.204	0.987
image 6&29	0.124	0.093	0.634
image 6&30	0.091	0.078	0.804
image 6&31	0.09	0.091	0.438
image 6&32	0.129	0.064	0.469
image 6&33	0.156	0.056	0.377
image 6&35	0.174	0.101	0.899
image 6&39	0.19	0.112	0.857
image 6&40	0.236	0.124	0.763
image 6&41	0.139	0.072	0.449
image 6&42	0.128	0.133	0.597
image 6&43	0.124	0.101	0.67
image 6&44	0.119	0.133	0.799
image 6&48	0.125	0.091	0.566
image 7&15	0.377	0.164	0.713
image 7&28	0.328	0.14	0.788
image 7&29	0.262	0.144	0.871
image 7&35	0.222	0.124	0.672
image 7&48	0.175	0.155	0.672
image 7&49	0.251	0.191	0.652
image 8&15	0.405	0.11	0.569
image 8&16	0.169	0.194	0.985
image 8&19	0.117	0.053	0.866
image 8&25	0.158	0.157	0.923
image 8&26	0.196	0.157	0.734
image 8&28	0.328	0.103	0.441
image 8&29	0.227	0.068	0.727
image 8&30	0.341	0.095	0.846

image 8&42	0.267	0.125	0.325
image 8&43	0.405	0.141	0.747
image 8&49	0.243	0.098	0.469
image 10&11	0.328	0.118	0.325
image 10&19	0.143	0.036	0.584
image 10&28	0.326	0.233	0.917
image 10&29	0.301	0.082	0.611
image 10&30	0.273	0.068	0.691
image 10&35	0.293	0.162	0.424
image 10&40	0.346	0.139	0.817
image 10&41	0.259	0.099	0.362
image 10&42	0.271	0.17	0.832
image 11&16	0.173	0.139	0.65
image 11&19	0.118	0.083	0.666
image 11&25	0.216	0.124	0.587
image 11&39	0.36	0.141	0.709
image 11&40	0.577	0.122	0.953
image 11&44	0.313	0.107	0.321
image 11&49	0.32	0.115	0.478
image 15&16	0.159	0.098	0.425
image 15&18	0.321	0.123	0.368
image 15&19	0.193	0.089	0.971
image 15&31	0.412	0.119	0.816
image 15&32	0.321	0.139	0.327
image 15&33	0.402	0.116	0.343
image 15&39	0.335	0.192	0.493
image 15&40	0.46	0.133	0.727
image 15&49	0.333	0.224	0.903
image 16&18	0.161	0.189	0.825
image 16&19	0.111	0.075	0.63
image 16&26	0.175	0.168	0.798
image 16&29	0.128	0.088	0.526
image 16&30	0.134	0.112	0.684
image 16&31	0.138	0.17	0.762
image 16&35	0.142	0.092	0.574
image 16&41	0.161	0.131	0.437
image 16&42	0.109	0.103	0.255
image 16&43	0.136	0.092	0.615
image 16&48	0.173	0.164	0.492
image 16&49	0.173	0.105	0.362
image 18&19	0.142	0.056	0.743
image 18&25	0.188	0.186	0.782
image 18&26	0.189	0.204	0.511

image 18&28	0.351	0.09	0.484
image 18&29	0.28	0.095	0.682
image 18&30	0.466	0.145	0.814
image 18&35	0.136	0.089	0.548
image 18&42	0.405	0.141	0.757
image 18&43	0.404	0.112	0.926
image 18&48	0.149	0.125	0.47
image 18&49	0.238	0.13	0.463
image 19&25	0.145	0.084	0.796
image 19&28	0.16	0.088	0.461
image 19&29	0.167	0.071	0.706
image 19&31	0.189	0.071	0.842
image 19&32	0.148	0.052	0.935
image 19&33	0.134	0.051	0.598
image 19&35	0.13	0.071	0.676
image 19&41	0.127	0.058	0.461
image 19&42	0.241	0.066	0.98
image 19&43	0.184	0.097	0.941
image 19&44	0.114	0.038	0.464
image 19&48	0.138	0.074	0.512
image 19&49	0.186	0.06	0.493
image 25&26	0.233	0.173	0.803
image 25&30	0.195	0.116	0.79
image 25&31	0.176	0.17	0.76
image 25&32	0.183	0.203	0.94
image 25&35	0.211	0.082	0.535
image 25&40	0.359	0.184	0.556
image 25&41	0.24	0.136	0.476
image 25&42	0.178	0.117	0.335
image 25&48	0.327	0.271	0.821
image 25&49	0.241	0.112	0.5
image 26&28	0.434	0.194	0.694
image 26&32	0.224	0.17	0.76
image 26&33	0.251	0.164	0.785
image 26&39	0.26	0.213	0.978
image 26&40	0.452	0.149	0.995
image 26&41	0.328	0.088	0.651
image 26&43	0.32	0.174	0.89
image 26&44	0.236	0.181	0.745
image 28&30	0.446	0.107	0.55
image 28&31	0.302	0.139	0.489
image 28&32	0.308	0.117	0.38
image 28&40	0.466	0.11	0.555

image 28&43	0.498	0.189	0.703
image 28&49	0.354	0.198	0.891
image 29&31	0.193	0.135	0.731
image 29&32	0.146	0.111	0.534
image 29&33	0.184	0.12	0.372
image 29&39	0.199	0.122	0.676
image 29&40	0.384	0.093	0.787
image 29&49	0.365	0.176	0.725
image 30&32	0.269	0.118	0.469
image 30&33	0.313	0.119	0.465
image 30&35	0.181	0.091	0.779
image 30&39	0.267	0.172	0.789
image 30&40	0.454	0.11	0.89
image 30&49	0.293	0.099	0.638
image 31&33	0.285	0.2	0.447
image 31&35	0.162	0.129	0.503
image 31&42	0.406	0.166	0.994
image 31&48	0.113	0.124	0.502
image 31&49	0.213	0.135	0.402
image 32&35	0.18	0.117	0.555
image 32&42	0.313	0.117	0.632
image 32&43	0.398	0.124	0.886
image 32&48	0.143	0.107	0.609
image 32&49	0.215	0.141	0.326
image 33&40	0.416	0.197	0.796
image 33&42	0.308	0.12	0.46
image 33&43	0.376	0.148	0.607
image 33&48	0.21	0.121	0.592
image 35&39	0.353	0.094	0.873
image 35&42	0.153	0.082	0.555
image 35&44	0.358	0.144	0.742
image 35&49	0.286	0.119	0.835
image 39&41	0.361	0.137	0.556
image 39&42	0.237	0.176	0.361
image 39&43	0.299	0.128	0.657
image 39&48	0.297	0.141	0.949
image 39&49	0.273	0.179	0.631
image 40&41	0.59	0.118	0.947
image 40&42	0.408	0.148	0.832
image 40&43	0.456	0.118	0.843
image 40&44	0.404	0.157	0.8
image 40&48	0.528	0.161	0.967
image 40&49	0.403	0.135	0.725

image 41&44	0.188	0.091	0.381
image 41&48	0.404	0.157	0.947
image 41&49	0.286	0.09	0.577
image 42&44	0.172	0.145	0.556
image 42&49	0.294	0.149	0.823
image 43&49	0.316	0.164	0.69
image 44&48	0.164	0.111	0.503
image 44&49	0.275	0.209	0.97

Table 5.5: The relative pointing error evaluation for all stereo pairs in the San Fernando test site 2

	Uncompensated [pixel]	Compensated [pixel]
image 1&2	1.002	0.491
image 1&3	1.011	0.488
image 1&6	6.907	0.522
image 1&7	1.006	0.682
image 1&8	6.07	0.517
image 1&16	2.965	0.546
image 1&18	1.626	0.564
image 1&19	7.892	0.632
image 1&24	3.478	0.391
image 1&25	8.465	0.444
image 1&26	6.863	0.463
image 1&28	2.024	0.63
image 1&34	5.52	0.774
image 1&35	7.158	0.499
image 1&37	0.439	0.46
image 1&44	1.499	0.531
image 2&4	0.924	0.817
image 2&5	0.925	0.817
image 2&6	9.352	0.452
image 2&7	0.669	0.61
image 2&8	5.138	0.429
image 2&18	3.369	0.538
image 2&19	10.306	0.698
image 2&24	3.918	0.693
image 2&26	9.779	0.35
image 2&28	7.651	0.569
image 2&34	5.94	0.91
image 2&37	3.716	0.547
image 2&44	5.412	0.411
image 3&4	0.918	0.815
image 3&5	0.92	0.815

image 3&6	9.354	0.456
image 3&7	0.652	0.603
image 3&8	5.132	0.435
image 3&18	3.364	0.526
image 3&19	10.307	0.703
image 3&24	3.912	0.682
image 3&26	9.779	0.353
image 3&28	7.656	0.567
image 3&34	5.94	0.91
image 3&37	3.716	0.553
image 3&44	5.415	0.413
image 4&15	2.579	0.37
image 4&19	5.397	0.821
image 4&25	8.5	0.954
image 4&35	6.957	0.888
image 5&15	2.579	0.37
image 5&19	5.396	0.821
image 5&25	8.499	0.954
image 5&35	6.956	0.888
image 6&7	4.578	0.501
image 6&8	5.6	0.437
image 6&15	3.51	0.851
image 6&16	8.122	0.383
image 6&18	13.632	0.437
image 6&19	6.101	0.416
image 6&24	5.109	0.551
image 6&25	4.048	0.36
image 6&28	2.02	0.324
image 6&34	5.701	0.531
image 6&37	6.035	0.324
image 6&38	5.266	0.445
image 7&8	3.533	0.558
image 7&15	2.872	0.884
image 7&16	4.518	0.569
image 7&25	9.323	0.594
image 7&28	3.513	0.334
image 7&37	3.265	0.413
image 7&38	2.735	0.547
image 7&44	1.376	0.462
image 8&16	4.911	0.539
image 8&19	1.348	0.623
image 8&25	2.314	0.597
image 8&26	4.272	0.446

image 8&28	3.368	0.392
image 8&34	1.538	0.817
image 8&35	1.704	0.587
image 8&37	1.153	0.439
image 8&38	3.628	0.506
image 8&44	0.966	0.638
image 15&16	2.368	0.938
image 15&44	0.719	0.814
image 16&18	1.943	0.593
image 16&19	1.366	0.354
image 16&24	0.37	0.391
image 16&26	2.039	0.438
image 16&35	5.226	0.394
image 16&37	2.663	0.376
image 16&38	2.343	0.673
image 16&44	3.903	0.414
image 18&19	9.758	0.473
image 18&25	8.548	0.341
image 18&26	7.919	0.468
image 18&28	4.788	0.484
image 18&35	10.727	0.52
image 18&37	5.371	0.416
image 18&38	3.294	0.449
image 18&44	7.687	0.451
image 19&24	1.603	0.41
image 19&25	1.827	0.409
image 19&28	3.406	0.402
image 19&37	6.63	0.354
image 19&38	6.798	0.396
image 19&44	2.936	0.443
image 24&25	4.996	0.426
image 24&28	0.571	0.468
image 24&35	3.637	0.666
image 24&37	0.509	0.442
image 24&38	1.495	0.637
image 24&44	0.61	0.509
image 25&26	0.556	0.525
image 25&37	4.678	0.495
image 25&44	1.839	0.482
image 26&28	3.799	0.459
image 26&34	2.999	0.867
image 26&38	9.907	0.698
image 28&35	0.952	0.602

image 28&37	0.947	0.388
image 28&38	2.389	0.468
image 28&44	4.318	0.553
image 34&37	2.851	0.9
image 34&38	3.657	0.915
image 34&44	2.016	0.536
image 35&38	5.414	0.634
image 35&44	2.732	0.686
image 37&44	1.706	0.374
image 38&44	1.35	0.594

Table 5.6: The check points evaluation for all stereo pairs in the San Fernando test site 2

	latitude [m]	longitude [m]	height [m]
image 1&2	0.445	0.088	0.708
image 1&3	0.456	0.09	0.715
image 1&6	0.003	0.001	0.014
image 1&7	0.343	0.147	0.56
image 1&8	0.497	0.145	0.857
image 1&16	0.133	0.101	0.317
image 1&18	0.249	0.107	0.539
image 1&19	0.182	0.073	0.342
image 1&24	0.419	0.089	0.722
image 1&25	0.152	0.064	0.289
image 1&26	0.263	0.086	0.48
image 1&28	0.174	0.034	0.357
image 1&34	0.288	0.111	0.522
image 1&35	0.149	0.136	0.284
image 1&37	0.366	0.041	0.562
image 1&44	0.162	0.055	0.24
image 2&4	0.416	0.152	0.832
image 2&5	0.416	0.152	0.832
image 2&6	0.041	0.088	0.308
image 2&7	0.236	0.163	0.422
image 2&8	0.318	0.141	0.496
image 2&18	0.183	0.154	0.368
image 2&19	0.196	0.148	0.352
image 2&24	0.373	0.119	0.432
image 2&26	0.205	0.175	0.372
image 2&28	0.493	0.179	0.618
image 2&34	0.236	0.127	0.395
image 2&37	0.484	0.076	0.579
image 2&44	0.206	0.15	0.335

image 3&4	0.429	0.152	0.838
image 3&5	0.429	0.152	0.838
image 3&6	0.041	0.091	0.306
image 3&7	0.28	0.172	0.426
image 3&8	0.317	0.143	0.498
image 3&18	0.183	0.156	0.367
image 3&19	0.214	0.146	0.363
image 3&24	0.373	0.122	0.436
image 3&26	0.203	0.175	0.375
image 3&28	0.503	0.171	0.615
image 3&34	0.234	0.128	0.4
image 3&37	0.473	0.079	0.552
image 3&44	0.207	0.156	0.337
image 4&15	0.406	0.475	0.413
image 4&19	0.183	0.133	0.952
image 4&25	0.157	0.155	0.604
image 4&35	0.207	0.181	0.816
image 5&15	0.406	0.475	0.413
image 5&19	0.183	0.133	0.952
image 5&25	0.157	0.155	0.604
image 5&35	0.207	0.181	0.816
image 6&7	0.097	0.107	0.633
image 6&8	0.089	0.069	0.422
image 6&15	0.038	0.254	0.697
image 6&16	0.092	0.08	0.487
image 6&18	0.061	0.083	0.441
image 6&19	0.113	0.1	0.418
image 6&24	0.102	0.082	0.411
image 6&25	0.082	0.06	0.329
image 6&28	0.059	0.072	0.569
image 6&34	0.126	0.07	0.416
image 6&37	0.043	0.085	0.609
image 6&38	0.043	0.138	0.404
image 7&8	0.298	0.308	0.941
image 7&15	0.386	0.195	0.88
image 7&16	0.17	0.166	0.929
image 7&25	0.154	0.197	0.899
image 7&28	0.223	0.167	0.498
image 7&37	0.257	0.085	0.817
image 7&38	0.286	0.215	0.582
image 7&44	0.182	0.146	0.508
image 8&16	0.17	0.179	0.527
image 8&19	0.155	0.138	0.608

image 8&25	0.159	0.15	0.627
image 8&26	0.318	0.196	0.813
image 8&28	0.207	0.091	0.502
image 8&34	0.302	0.206	0.944
image 8&35	0.202	0.175	0.641
image 8&37	0.2	0.072	0.531
image 8&38	0.298	0.183	0.813
image 8&44	0.182	0.103	0.394
image 15&16	0.278	0.231	0.625
image 15&44	0.189	0.334	0.893
image 16&18	0.149	0.176	0.645
image 16&19	0.145	0.175	0.63
image 16&24	0.166	0.191	0.666
image 16&26	0.151	0.152	0.924
image 16&35	0.155	0.229	0.877
image 16&37	0.209	0.081	0.526
image 16&38	0.164	0.195	0.371
image 16&44	0.163	0.137	0.479
image 18&19	0.177	0.125	0.818
image 18&25	0.118	0.179	0.656
image 18&26	0.23	0.217	0.789
image 18&28	0.14	0.12	0.38
image 18&35	0.18	0.192	0.677
image 18&37	0.182	0.064	0.632
image 18&38	0.174	0.209	0.63
image 18&44	0.143	0.112	0.438
image 19&24	0.147	0.159	0.983
image 19&25	0.15	0.19	0.702
image 19&28	0.242	0.115	0.433
image 19&37	0.258	0.091	0.95
image 19&38	0.218	0.167	0.554
image 19&44	0.178	0.133	0.529
image 24&25	0.149	0.196	0.823
image 24&28	0.342	0.11	0.345
image 24&35	0.147	0.154	0.661
image 24&37	0.438	0.071	0.658
image 24&38	0.293	0.184	0.565
image 24&44	0.241	0.11	0.425
image 25&26	0.117	0.202	0.639
image 25&37	0.195	0.052	0.465
image 25&44	0.138	0.119	0.407
image 26&28	0.257	0.201	0.598
image 26&34	0.266	0.235	0.746

image 26&38	0.249	0.165	0.694
image 28&35	0.162	0.191	0.779
image 28&37	0.336	0.083	0.627
image 28&38	0.375	0.232	0.688
image 28&44	0.195	0.183	0.68
image 34&37	0.214	0.028	0.547
image 34&38	0.225	0.188	0.465
image 34&44	0.197	0.104	0.488
image 35&38	0.155	0.23	0.398
image 35&44	0.11	0.197	0.775
image 37&44	0.194	0.108	0.427
image 38&44	0.233	0.163	0.336

Acknowledgements

During the five years PhD research at Institute for Photogrammetry (IfP), University of Stuttgart, I have received so many supports and helps from numerous people. First of all, I would like to express my deep appreciation to my thesis supervisor Prof. Dr.-Ing. habil. Dieter Fritsch, who has given me continuous supports, advice, inspirations and encouragements. I also would like to express my gratitude to Prof. Dr.-Ing. Konrad Schindler, who has providing many very constructive suggestions to improve this thesis. I am appreciate to all the colleagues of IfP for their cooperation and help. Particularly, I am grateful to Dr. Michael Cramer for his advice in orientation area and Prof. Dr.-Ing. Norbert Haala for his knowledge in Computer Vision area. I thank the team of nFrames GmbH not only for the codes and license provided from them but also the joyful two years collaboration. I am especially appreciate to Dr. Mathias Rothermel. We have started our cooperation since my Master thesis. As a friend and colleague, he has given me a lot of invaluable advice in my research career. I greatly benefited from being associated the research teams of Deutschen Zentrums für Luft- und Raumfahrt (DLR), Oberpfaffenhofen and Institute of Geodesy and Photogrammetry, ETH Zürich. I am thankful to not only the data but also the experience provided by them. I would like to give my most appreciation to my parents who give me unconditional support and selfless love. Finally, no words can express my gratitude and love to my wife, Lin Cheng. Without her company and support, I would not have completed this work.

Curriculum Vitae

Ke Gong

Address: Institut für Photogrammetrie, Geschwister-Scholl-Str. 24D, 70174, Stuttgart
Birthday: January 18, 1991
Birth Place: Jingzhou, China
Email:KeGong@ifp.uni-stuttgart.de

EDUCATIONAL BACKGROUND

09/2008–06/2012	**Bachelor**	School of Remote Sensing, Wuhan University
09/2012–06/2015	**Master**	GeoEngine Program, University Stuttgart
10/2015–	**PhD**	Institute for Photogrammetry, University Stuttgart

EXPERIENCE

09/2010–09/2011	**Research assistant**	Wuhan University Innovative Project
10/2015–02/2017	**Student assistant**	nFrames GmbH
01/2020–03/2020	**Student assistant**	Gyrolog Project