Universität der Bundeswehr München

Scalable 3D Surface Reconstruction by Local Stochastic Fusion of Disparity Maps

Andreas Kuhn

Vollständiger Abdruck der von der Fakultät für Informatik der Universität der Bundeswehr München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Promotionsausschuss:

Vorsitzender: Univ.-Prof. Dr.-Ing. Wolfgang Reinhardt

1. Gutachter: Univ.-Prof. Dr.-Ing. Helmut Mayer

2. Gutachter: Prof. Daniel Scharstein, Ph.D. (Middlebury College)

Prüfer: Univ.-Prof. Dr. Peter Hertling
Univ.-Prof. Dr.-Ing. Mark Minas
Univ.-Prof. Dr. Oliver Rose

Die Dissertation wurde am 26.06.2014 bei der Universität der Bundeswehr München eingereicht und durch die Fakultät für Informatik angenommen. Die mündliche Prüfung fand am 28.07.2014 statt.

Willst du dich am Ganzen erquicken,
so musst du das Ganze im Kleinsten erblicken.
- Johann Wolfgang von Goethe (1749 - 1832)

Danksagung

Diese Arbeit basiert auf einigen Ideen und Verfahren, welche von einer Vielzahl an Personen erarbeitet wurden, denen ich zu Dank verpflichtet bin.

An erster Stelle danke ich vielmals Helmut Mayer, Leiter der Arbeitsgruppe für Visual Computing am Institut für Angewandte Informatik an der Universität der Bundeswehr. Durch seine intensive Arbeit und Unterstützung ermöglicht er seinen Mitarbeitern interessante Forschungen in einem angewandtem Umfeld. Er lehrte mich, neben der fachlichen Arbeit, insbesondere das wissenschaftliche Denken, Lesen und Schreiben.

Sehr gefreut habe ich mich über die Bereitschaft von Daniel Scharstein das Zweitgutachten an dieser Arbeit zu übernehmen. Seine weitreichenden Kenntnisse und Erfahrungen im Bereich der Computer Vision haben mir sehr geholfen.

Diese Arbeit entstand im Rahmen eines Kooperationsprojektes mit dem Deutschen Zentrum für Luft- und Raumfahrt. Ohne die Betreuung von Heiko Hirschmüller vom Institut für Robotik und Mechatronik und dessen richtungweisendes Wirken wäre diese Arbeit nicht möglich gewesen.

Des Weiteren haben viele Kollegen und Angehörige am Deutschen Zentrum für Luft- und Raumfahrt und der Universität der Bundeswehr durch Ihre Arbeiten, Ideen und Kritiken diese Arbeit geprägt und vorangebracht.

Mein besonderer Dank gilt meiner Partnerin Julia, meinen Eltern, meiner Schwester und Freunden welche auf mich unterstützend, richtungweisend, verständnisvoll, und ermunternd gewirkt und die Arbeit dadurch erst ermöglicht haben.

Kurzfassung

Digitale dreidimensionale (3D) Modelle sind in vielen Anwendungsfeldern, wie Medizin, Ingenieurswesen, Simulation und Unterhaltung von signifikantem Interesse. Eine manuelle Erstellung von 3D-Modellen ist äußerst zeitaufwendig und die Erfassung der Daten, z.B. durch Lasersensoren, ist teuer. Kamerabilder ermöglichen hingegen preiswerte Aufnahmen und sind gut verfügbar. Der rasante Fortschritt im Forschungsfeld Computer Vision ermöglicht bereits eine automatische 3D-Rekonstruktion aus Bilddaten. Dennoch besteht weiterhin eine Vielzahl von Problemen, insbesondere bei der Verarbeitung von großen Mengen hochauflösender Bilder. Zusätzlich zur komplexen Formulierung, die zur Lösung eines schlecht gestellten Problems notwendig ist, besteht die Herausforderung darin, äußerst große Datenmengen zu verwalten.

Diese Arbeit befasst sich mit dem Problem der 3D-Oberflächenrekonstruktion aus Bilddaten, insbesondere für sehr große Modelle, aber auch Anwendungen mit hohem Genauigkeitsanforderungen. Zu diesem Zweck wird eine Prozesskette zur dichten skalierbaren 3D-Oberflächenrekonstruktion für große Bildmengen definiert, bestehend aus Bildregistrierung, Disparitätsschätzung, Fusion von Disparitätskarten und Triangulation von Punktwolken. Der Schwerpunkt dieser Arbeit liegt auf der Fusion und Filterung von durch Semi-Global Matching generierten Disparitätskarten zur Bestimmung von genauen 3D-Punktwolken.

Für eine unbegrenzte Skalierbarkeit wird eine Divide and Conquer Methode vorgestellt, welche eine parallele Verarbeitung von Teilräumen des 3D-Rekonstruktionsraums ermöglicht. Die Methode zur Fusion von Disparitätskarten basiert auf lokaler Optimierung von 3D Daten. Damit kann eine komplizierte Fusionsstrategie für die Unterräume vermieden werden. Obwohl der Fokus auf der skalierbaren Rekonstruktion liegt, wird eine hohe Oberflächenqualität durch mehrere Erweiterungen von lokalen Optimierungsmodellen erzielt, die dem Stand der Forschung entsprechen.

Dazu wird die wegweisende lokale volumetrische Optimierungsmethode von Curless and Levoy (1996) aus einer probabilistischen Perspektive interpretiert. Aus dieser Perspektive wird die Methode durch eine Bayes Fusion von räumlichen Messungen mit Gaußscher Unsicherheit erweitert. Zusätzlich zur Bestimmung einer optimalen Oberfläche ermöglicht diese probabilistische Fusion die Extraktion von Oberflächenwahrscheinlichkeiten. Diese werden wiederum zur Filterung von Ausreißern mittels geometrischer Konsistenzprüfungen im 3D-Raum verwendet.

Eine weitere Verbesserung der Qualität wird basierend auf der Analyse der Disparitätsunsicherheit erzielt. Dazu werden Gesamtvariation-basierte Merkmalsklassen definiert, welche stark mit der Disparitätsunsicherheit korrelieren. Die Korrelationsfunktion wird aus ground-truth Daten mittels eines Expectation Maximization (EM) Ansatzes gelernt. Aufgrund der Berücksichtigung eines statistisch geschätzten Disparitätsfehlers in einem probabilistischem Grundgerüst für die Fusion von räumlichen Daten, kann dies als eine stochastische Fusion von Disparitätskarten betrachtet werden. Außerdem wird der Einfluss der Bildregistrierung und Polygonisierung auf die volumetrische Fusion analysiert und verwendet, um die Methode zu erweitern.

Schließlich wird eine Multi-Resolution Strategie präsentiert, welche die Generierung von Oberflächen aus räumlichen Daten mit unterschiedlichster Qualität ermöglicht. Diese Methode erweitert Methoden, die den Stand der Forschung darstellen, durch die Berücksichtigung der räumlichen Unsicherheit von 3D-Punkten aus Stereo Daten.

Die Evaluierung von mehreren bekannten und neuen Datensätzen zeigt das Potential der skalierbaren stochastischen Fusionsmethode auf. Stärken und Schwächen der Methode werden diskutiert und es wird eine Empfehlung für zukünftige Forschung gegeben.

Abstract

Digital three-dimensional (3D) models are of significant interest to many application fields, such as medicine, engineering, simulation, and entertainment. Manual creation of 3D models is extremely time-consuming and data acquisition, e.g., through laser sensors, is expensive. In contrast, images captured by cameras mean cheap acquisition and high availability. Significant progress in the field of computer vision already allows for automatic 3D reconstruction using images. Nevertheless, many problems still exist, particularly for big sets of large images. In addition to the complex formulation necessary to solve an ill-posed problem, one has to manage extremely large amounts of data.

This thesis targets 3D surface reconstruction using image sets, especially for large-scale, but also for high-accuracy applications. To this end, a processing chain for dense scalable 3D surface reconstruction using large image sets is defined consisting of image registration, disparity estimation, disparity map fusion, and triangulation of point clouds. The main focus of this thesis lies on the fusion and filtering of disparity maps, obtained by Semi-Global Matching, to create accurate 3D point clouds.

For unlimited scalability, a Divide and Conquer method is presented that allows for parallel processing of subspaces of the 3D reconstruction space. The method for fusing disparity maps employs local optimization of spatial data. By this means, it avoids complex fusion strategies when merging subspaces. Although the focus is on scalable reconstruction, a high surface quality is obtained by several extensions to state-of-the-art local optimization methods.

To this end, the seminal local volumetric optimization method by Curless and Levoy (1996) is interpreted from a probabilistic perspective. From this perspective, the method is extended through Bayesian fusion of spatial measurements with Gaussian uncertainty. Additionally to the generation of an optimal surface, this probabilistic perspective allows for the estimation of surface probabilities. They are used for filtering outliers in 3D space by means of geometric consistency checks.

A further improvement of the quality is obtained based on the analysis of the disparity uncertainty. To this end, Total Variation (TV)-based feature classes are defined that are highly correlated with the disparity uncertainty. The correlation function is learned from ground-truth data by means of an Expectation Maximization (EM) approach. Because of the consideration of a statistically estimated disparity error in a probabilistic framework for fusion of spatial data, this can be regarded as a stochastic fusion of disparity maps. In addition, the influence of image registration and polygonization for volumetric fusion is analyzed and used to extend the method.

Finally, a multi-resolution strategy is presented that allows for the generation of surfaces from spatial data with a largely varying quality. This method extends state-of-the-art methods by considering the spatial uncertainty of 3D points from stereo data.

The evaluation of several well-known and novel datasets demonstrates the potential of the scalable stochastic fusion method. The strength and the weakness of the method are discussed and direction for future research is given.

Contents

1. Introduction **1**
- 1.1. Problem Statement . . . 3
- 1.2. Motivation . . . 4
- 1.3. Thesis Outline . . . 9

2. Basics **10**
- 2.1. Image Basics . . . 10
  - 2.1.1. Matching Costs . . . 11
  - 2.1.2. Total Variation . . . 13
- 2.2. Geometric Basics . . . 14
  - 2.2.1. Image Registration . . . 16
  - 2.2.2. Stereo Matching . . . 20
  - 2.2.3. Polygonization of Point Clouds . . . 23
- 2.3. Stochastic Basics . . . 25
  - 2.3.1. Distributions . . . 26
  - 2.3.2. Expectation, Variance, and Entropy . . . 28
  - 2.3.3. Machine Learning . . . 29
  - 2.3.4. Binary Bayes Theory . . . 33

3. State of Research **36**
- 3.1. Generation of Point Clouds . . . 36
- 3.2. 2.5D Modeling . . . 37
- 3.3. 3D Surface Reconstruction . . . 39
  - 3.3.1. Local Methods . . . 39
  - 3.3.2. Global Methods . . . 43
  - 3.3.3. Scalable 3D Modeling . . . 46

4. Probabilistic Framework **49**
- 4.1. Volumetric Fusion of Spatial Data . . . 49
- 4.2. Linear Fusion . . . 50
- 4.3. Bayesian Fusion of Gaussians . . . 53

5. Fusion and Filtering of Disparity Maps **58**
- 5.1. Semi-Global Matching . . . 58
- 5.2. Unlimited Scalability . . . 61
- 5.3. Error Models . . . 63
  - 5.3.1. Stereo Error Model . . . 64

5.3.2. Registration Error 66
5.3.3. Disparity Error 69
5.4. Consistency Checks 74
5.4.1. Image Consistency 74
5.4.2. Geometrical Consistency 75
5.4.3. Probabilistic Consistency 76

6. Multi-Resolution Computation 77
6.1. Data Structures 78
6.2. Adaptation to Error Models 79
6.3. Scale Space Analysis 82
6.4. Meshing 85

7. Evaluation and Validation 86
7.1. Datasets 87
7.2. Large Models and Large Scenes 89
7.3. Multi-Resolution 96
7.4. Probabilistic Fusion 98
7.5. Disparity Uncertainty 104
7.6. Registration Uncertainty 108
7.7. Scale Space 109

8. Summary, Conclusion, and Future Work 112
8.1. Processing Chain 112
8.2. Summary and Conclusion 114
8.3. Future Work 115

A. Results 118

Bibliography 124

Chapter 1.

Introduction

Digital three-dimensional (3D) models of the real-world can be reconstructed from multiple measurements. In general, sensors for this task produce a set of measurements that describe the distances to a surface. The sensor techniques can be classified into two categories: active and passive sensors.

Active sensors emit signals into the 3D space that are later detected. The resulting signal information can be used to estimate the geometric distance. E.g., laser sensors and time of flight sensors emit light with a specific frequency in one or multiple directions. The time between emission and detection is used for distance estimation because they are highly proportional. The use of phase information and triangulation can improve the accuracy of laser measurements. Passive sensors do not emit signals and make use of the available signal, e.g., light.

Cameras detect light from the environment. Given data from multiple positions of the sensor, pertaining to the same environment, distances can be obtained indirectly by triangulation in scene geometry. Active illumination, e.g., used in the Kinect sensor, such as light stripes can improve the available information especially in weakly textured scenes. Images from camera sensors are not useful without suitable lighting and sufficiently textured surfaces, which limits their reliability. Nonetheless, cameras have advantages such as the cheap acquisition and the high availability.

Modeling the world in 3D based on two-dimensional (2D) images is a fascinating field of computer vision that achieved great progress in the last decades; however many problems remain unsolved. The challenge is to manage a set of images captured in complex configurations to obtain 3D models with maximum accuracy and completeness. The modeling process should also entail small memory requirement and short processing time, as well as limit the resulting data to essential 3D structures.

The generation of a 3D surface from images can be performed in many different ways using different types of optimization. Additional prior information can improve the quality of 3D surfaces concerning accuracy and completeness. Priors can represent knowledge regarding the smoothness of surfaces, or even semantic knowledge regarding the modeling of a specific type of object. Surface shading can also provide additional information regarding surface geometry, especially when multiple light sources exist. In this thesis, a method is presented that allows for the reconstruction of fine surface details, without prior semantic information, because it is not always available.

A crucial characteristic of the 3D modeling problem is that main parts are ill-posed because the goal is to reconstruct a projection of 3D information. An infinite number of solutions for surface reconstruction exists using information from an image set. The progress of recent methods is on one hand due to the vastly increased power of computer systems in terms of runtime performance and memory resources. On the other hand, machine learning approaches have improved the solution to many computer vision problems, or even made some possible for the first time. Stochastic methods in machine learning are suitable for computer vision, because ill-posed problems can be explained in a probabilistic and statistical way and solved by finding the most probable solution. Nevertheless, an optimal mathematical formulation for solving the problem of 3D modeling using images is still an unsolved challenge.

Structure from Motion (SfM) methods directly estimate 3D geometry by sparse modeling without knowledge of the configuration. To this end, a significant amount of unknown parameters has to be estimated simultaneously. The corresponding optimization problem cannot be formulated in a linear or convex manner without simplified assumptions. Furthermore, variational solutions require an approximate initial solution. Fortunately, the problem of 3D modeling from image sets can be divided into a chain of subproblems:

Image Registration is the first step in reconstructing scene geometry, often considering prior knowledge regarding a certain camera model (calibration). Matching and tracking points in overlapping images is well studied and works well for dense configurations. A 3D position can be obtained from points in multiple images using triangulation based on knowledge regarding the scene geometry. Over a complete set, a sparse 3D point cloud can be obtained by tracking the features. On one hand, the sparse point cloud is useful for generating a first approximation of a 3D model. On the other hand, the sparse point cloud allows for an accurate estimation of camera poses using robust bundle adjustment.

Stereo Matching is concerned with the estimation of the pixelwise relative distance to the surface (depth) regarding two images. The disparity is defined by the distance between two corresponding pixels on the epipolar line in an image pair and has a unique relation to the corresponding depth. With knowledge regarding scene geometry, dense disparity maps can be generated by searching for correspondence along the epipolar line. Stereo methods obtain accurate pixelwise depth information in stereo configurations by considering prior relationships of neighboring pixels. The results are dense disparity maps that can be partially noisy or even inaccurate.

Depth Fusion is the fusion and filtering of noisy and inaccurate dense disparity maps, and it is the main focus of this thesis. Whereas sparse modeling requires a complex re-optimization of the surfaces, the challenge of dense modeling is to eliminate redundancy, outliers, and noise. In this work, a novel probabilistic approach is presented that assumes 3D points that correspond to trivariate Gaussians. The Gaussian parameters are partly statistically estimated by machine learning methods and partly probabilistically derived by geometric knowledge. This thesis combines the power of both assumptions, leading to a stochastic method that is highly scalable through the local optimization of 3D points in octree data structures.

Triangulation entails the transformation of point clouds to connected sets of polygons. It has been shown that a set of triangles is well suited for representing surfaces. Different methods have been proposed for generating meshes of triangles from point clouds. Among them are methods that can manage noisy data, especially when considering the constraint of watertightness. For clean point clouds from Depth Fusion, local methods can be used for fast and efficient processing of possibly large amounts of surface data.

Optimization of the surfaces can be performed optionally as a postprocessing step. An optimization step is required for detailed surfaces when only an approximate initial 3D model is available. Re-optimization can also comprise a reduction in the number of triangles with the least possible loss in surface quality. By reducing this amount of 3D information, color information can possibly be lost. An optimized texturizing of the surfaces is reasonable for many applications. Texture optimization can be extremely complex.

1.1. Problem Statement

The basic problem considered in this thesis is the fusion of disparity maps into clean, accurate point clouds by filtering outliers and fusing noisy data. The fusion of disparity maps by multi-view stereo (MVS) methods is a well-known problem and has a wide range of practical solutions. Nonetheless, the problem is still an open challenge in computer vision. In particular, for scalable 3D modeling from noisy disparity maps, there are several problems to manage:

1. Measurement noise,
2. Outliers from (multi-view) stereo,
3. Uncertainty caused by preprocessing steps,
4. Multiple point redundancy,
5. Hardware limitations,
6. Runtime performance.

This thesis provides solutions to all specified problems with a focus on scalability.

Points 1 to 3 focus on the uncertainty of the 3D points caused by noise, outliers, and misperceptions in the preprocessing steps. First, stereo matching as an ill-posed problem obtains a non-negligible set of outliers. Furthermore, noise caused by numerical, physical, or algorithmic inaccuracy is inevitable. The quality of disparities has to be considered for 3D modeling because it can vary depending on the image configuration and environmental influence. Multiple error models for 3D points from disparity maps already exist, yet all models contain a constant disparity error. The uncertainty of disparity depending on the stereo methods is unknown. Preprocessing steps, such as image registration or stereo matching, are based on prior assumptions that lead to further noise and errors. Hence, considering the different quality of obtained data seems to possess a high potential to improve the fusion of disparity maps.

Points 4 to 6 describe an important, but complex, attribute of 3D modeling: scalability. The progress of chip quality in cameras leads to high-resolution images with high quality even in consumer cameras. Furthermore, 3D modeling can be performed by considering a large amount of images. Image sets can provide highly redundant information, because in dense configurations many images show the same part of a scene. Finally, even in mobile systems, an extremely large amount of information can be stored in compressed images. All this leads to a large amount of data that does not fit in main memory, even in large computer systems. Sequential data processing can lead to high runtime, and requires adapted algorithms.

In order to provide a better impression of the limitation dimensions, they are discussed based on the example shown in Fig. 1.1. Image registration is performed for 823 images with ten-megapixel resolution acquired from an octocopter and from the ground. The images shown provide an impression of the detail that can be obtained through this configuration. In addition, corresponding dense disparity maps are provided.

After registration and stereo estimation, there are over four billion valid disparities estimated in the complete image set. Because of dense image configuration, multiple cameras capture the same part of the scene. Considering 24 bytes per 3D point, a memory space of approximately 90 GB is required simply to read the data. Therefore, processing the entire data on a desktop PC seems impossible. Fortunately, hardware exists with a hundred or even a thousand CPU cores that have the power and memory required to process big data in parallel. Yet, parallel computing means that the data has to be divided and merged by algorithms that process independent areas. This thesis offers a method for processing big data in parallel through local optimization, thus avoiding complex fusion strategies, and yet offering high quality surfaces by a novel stochastic fusion process.

1.2. Motivation

3D modeling of the real world has many practical applications with a variety of requirements for data acquisition. In addition to the question of data processing, it is important to discuss the procurement of such data.

In contrast to most alternative sensor data, such as laser data, images from cameras are publicly available on several platforms, especially images of famous public places. This is

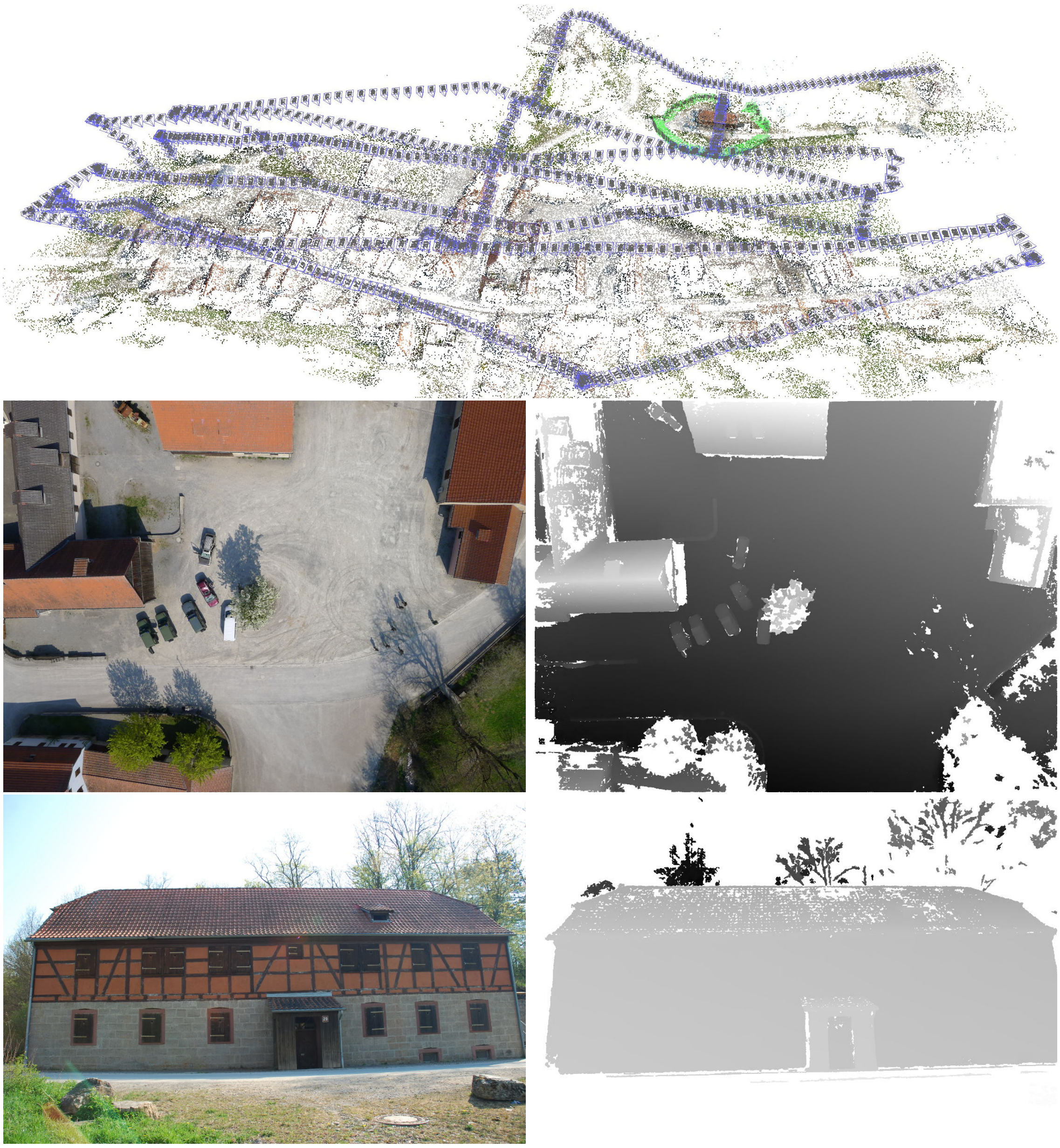


Figure 1.1.: Large image set with 823 images that show a complete village with varying detail. The upper image shows the registered image set with all camera poses and a reference point cloud. The blue pyramids represent camera poses captured from an unmanned aerial vehicle (UAV), whereas the green camera poses represent images captured from the ground. The middle and bottom images show two detailed views of the image set with the corresponding disparity map. The complete set of disparity maps obtains billions of 3D points.

because of the rising numbers of mobile cameras, particularly the cameras contained in mobile phones. It has been shown that the large amount of data available for these places of interest can be used to model such places (Pollefeys et al. 2008). Agarwal et al. (2011) (Agarwal et al. 2009) and Frahm et al. (2010) published methods that work with community photo collections from the Internet by processing millions of images on possibly small systems within a reasonable time. The generation of 3D models using community photo collections is well suited for applications such as digital tourism or even cultural heritage. Two 3D models obtained from community photo collections are shown in Fig. 1.2.

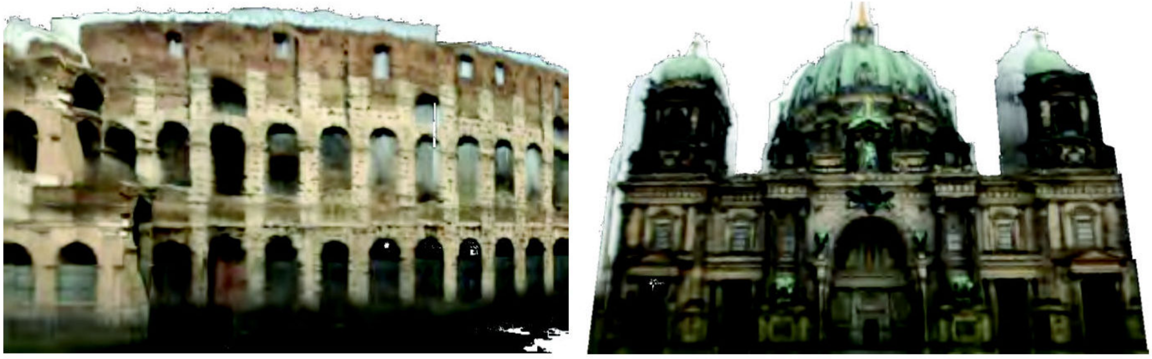


Figure 1.2.: Textured 3D models from community photo collections show the Colosseum in Rome and the Dome in Berlin. The models were computed in less than 24 hrs from subsets of photo collections of 2.9 million and 2.8 million images. (Frahm et al. 2010)

Yet, most of the world is not captured by cameras at all. Methods that focus on community photo collections usually have problems managing sparse sets of images.

For a long time, aerial imaging was the standard method for capturing coherent image sets for modeling urban regions or landscapes. Nowadays, aerial images are captured with digital cameras with hundreds of megapixels. In addition, satellite images are commercially available with a maximum resolution of 0.5 meters. In both cases, large parts can be covered simultaneously. In particular, with digital aerial cameras, dense image sets are often acquired. The availability of Global Positioning System (GPS) and Inertial Navigation System (INS) information make direct registration possible. Accurate image registration allows for pixelwise depth estimation with stereo matching methods, such as Semi-Global Matching (SGM) (Hirschmüller 2005, Hirschmüller 2008). Dense depth maps from aerial configurations allow for the reconstruction of 2.5-dimensional (2.5D) models through intuitive fusion of the disparity maps.

2.5D models expand 2D planes by adding one additional information per 2D point describing the distance from a ground plane. In aerial image configurations with nadir looking cameras, 2.5D modeling provides fascinating surface models of landscapes or even complete cities (cf. Fig. 1.3).

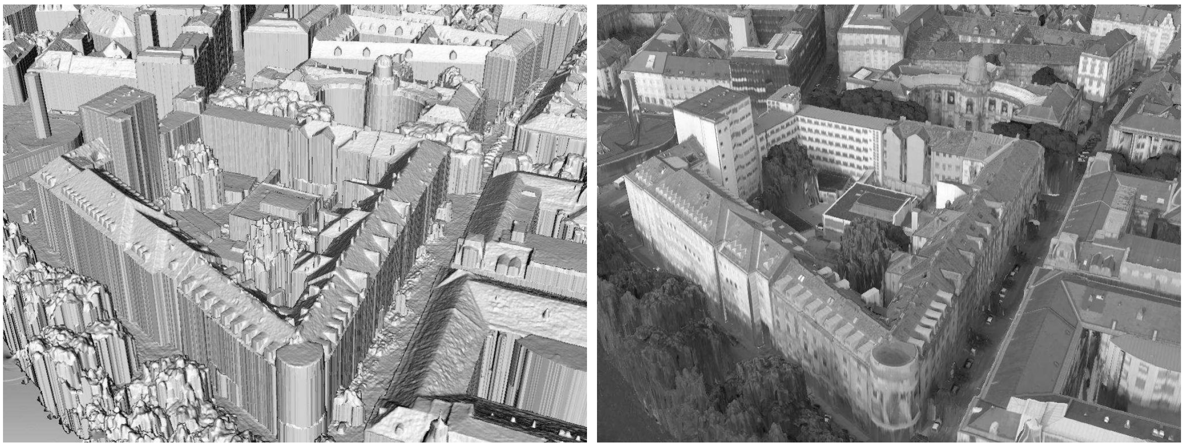


Figure 1.3.: Shaded and textured 2.5D model from Graz. The model results from SGM disparity maps on aerial images. (Hirschmüller 2008) © 2008 IEEE

2.5D models have an extensive range of practical applications because they comprise important information. Next to architecture planning, nowadays numerous museums show models of fascinating natural and urban areas. Even for navigation or agricultural planning 2.5D models are extremely suitable.

Manually or semi-automatically reconstructed 3D models are a good basis for applications that use real-time rendering, such as web applications like Google Earth (Google 2014). Image information can be compressed effectively and transmitted over networks. In contrast, complex geometric surface models of significant size are more challenging and not well studied, yet. For instance, Google Maps shows several geometrically simplified models from urban regions with mapped texture (cf. Fig. 1.4). The models are simplistic and even complex buildings are modeled through a limited number of planes, which leads to possibly large incorrect areas in the visualization.

In the past decade, UAVs have become an attractive alternative for closing the gap between aerial and terrestrial images. Multicopter systems can carry light high-quality consumer cameras. The combination of images captured from varying perspectives offers a wide range of novel practical applications. Complex objects can be covered from all directions, leading to a higher completeness of the 3D model. Furthermore, the configuration allows the capture of images from a shorter distance, which results in high quality details in the 3D model (cf. Fig. 1.5).

The novel image configurations from UAVs are particularly suitable for applications that require a high level of detail. Cultural heritage applications are becoming extremely important. Examples can be found in the field of museums, churches, castles and historical artifacts. For complex and tall buildings, a combination of ground and UAV images is extremely advantageous. Detailed scalable 3D modeling of complete urban regions is

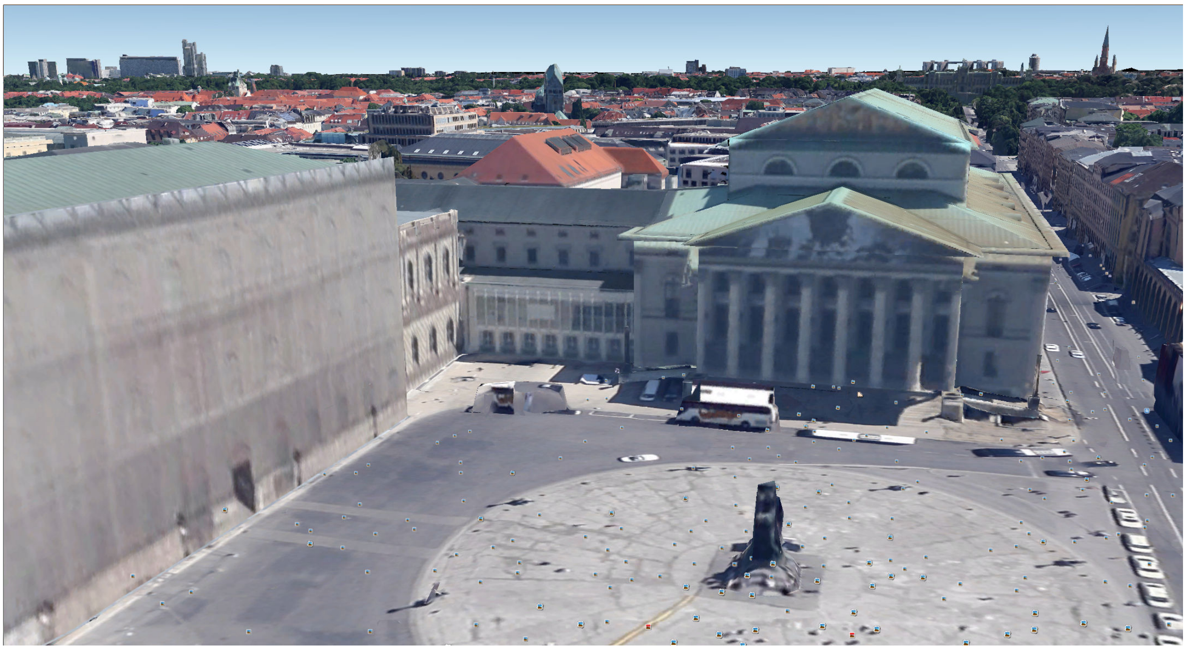


Figure 1.4.: Screenshot of a view in Google Earth from the opera in Munich. The simple model of the building comprises a small set of planes. The simplicity affords fast rendering of large scenes but discards complex geometries. Details such as the pillars are mapped onto a plane. © 2014 Google, © 2014 DigitalGlobe

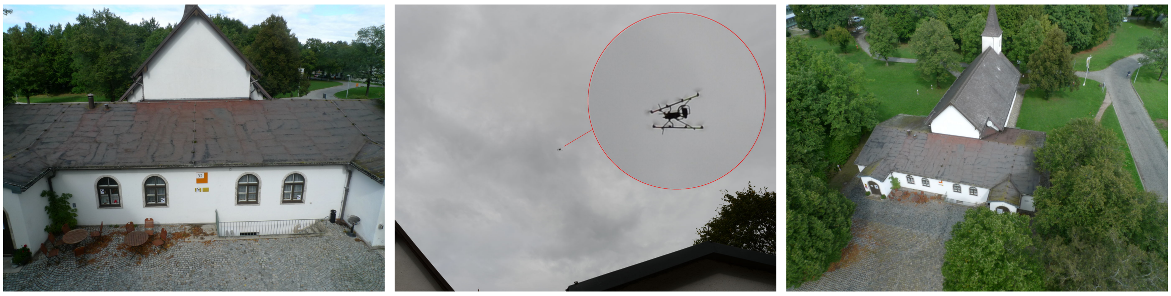


Figure 1.5.: Highly detailed images captured with an octocopter system from varying distances. The middle (zoomed) image shows the octocopter from a height of approximately 100 meter.

useful for municipal planning, as well as for police and military use.

These novel practical applications have requirements in 3D modeling that cannot be fulfilled by 2.5D modeling. However, the step from 2.5 to 3D is far from trivial. The reconstruction algorithms cannot be extended easily because the image configurations are often much more variable and no dominant direction exists. Such challenging configurations lead to highly differing qualities of the spatial data that have to be considered. The new requirements give rise to significant challenges in the fusion of disparity maps.

1.3. Thesis Outline

This thesis focuses on the problem of fusing disparity maps into a detailed and complete 3D point cloud. The proposed solution consists of a novel stochastic framework that extends existing geometric approaches by focusing on scalability. In Chapter 2, foundations are laid in the form of relevant basics. Geometric basics comprise a processing chain from images to 3D surface models. This thesis supplements the reconstruction chain through a novel method for fusing disparity maps. The fusion process involves stochastic methods, whose basics are also described in Chapter 2. Chapter 3 provides an overview of the state of research in 3D modeling. In particular, adaptability to large sets of images that concerns the scalability of state-of-the-art methods is discussed. Chapter 4 addresses the volumetric fusion of spatial data. To this end, a local method and an extension to a novel probabilistic approach is presented. Chapter 5 specifies the fusion and filtering on disparity maps. To guarantee unlimited scalable 3D reconstruction a Divide and Conquer method is presented that allows for parallel processing. Different types of error models are discussed that consider registration and disparity errors and their influence on data fusion. To this end, the disparity error is analyzed and learned from ground-truth data. Chapter 6 is devoted to the problem of multi-resolution computation, discussing data structures and the adaption to error models. Chapter 7 analyzes the progress through the novel proposed methods concerning accuracy and completeness by presenting results from a wide range of datasets. Chapter 8 summarizes and concludes the thesis. Finally, an outlook on future work is presented.

Parts of the reconstruction chain (cf. Section 2.2), the probabilistic approach (cf. Chapter 4), and the multi-resolution computation (cf. Chapter 6), have been previously published in:

Bartelsen, J., Mayer, H., Hirschmüller, H., Kuhn, A. and Michelini, M. (2012): Orientation and Dense Reconstruction of Unordered Terrestrial and Aerial Wide Baseline Image Sets, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume 1, 25–30.

Kuhn, A., Hirschmüller, H. and Mayer, H. (2013): Multi-Resolution Range Data Fusion for Multi-View Stereo Reconstruction, 35th German Conference on Pattern Recognition.

Mayer, H., Bartelsen, J., Hirschmüller, H. and Kuhn, A. (2011): Dense 3D Reconstruction from Wide Baseline Image Sets, 15th International Workshop on Theoretical Foundations of Computer Vision.

Chapter 2.

Basics

This thesis addresses the problem of scalable 3D modeling from image sets focusing on the fusion of disparity maps to accurate point clouds. The fusion of depth maps by limiting the optimization area allows for fast parallel processing of large datasets (Kuhn et al. 2013, Steinbrücker et al. 2013). Depth information can be obtained by stereo methods or active sensors. The reconstruction of accurate surfaces by local fusion of depth values requires comprehensive knowledge about the uncertainty of 3D points. The error of a 3D point results from physical, algorithmic, and numeric uncertainties that can be described in a stochastic manner.

In this chapter, an overview of a geometric reconstruction pipeline and its relationship to the uncertainty in 3D space is given in Section 2.2. Because the 3D information is obtained from 2D images, important basics of image processing are presented in Section 2.1.

In the 3D space, the trivariate uncertainty is difficult to estimate because there are multiple influences that cannot always be derived directly. A statistical machine learning method is presented in this thesis to allow learning the disparity error. Given the uncertainty of a 3D point there is a need in a probabilistic framework for the fusion of several points. For these reasons, the following general stochastic basics are described in Section 2.3: probability distributions, machine learning methods, and fusion theory.

2.1. Image Basics

For 3D surface reconstruction from 2D images, image processing methods are important foundations. Image registration and stereo matching have a need for information regarding corresponding points in two or more images. The estimation of corresponding points is not unambiguous. Nonetheless, comparing pixels or pixel neighborhoods by means of matching costs only provides assumptions for possible correspondences.

For detecting noise in images or disparity maps, the local frequency behavior of intensities is important. The mathematical concept Total Variation (TV) is shown to be suitable for such purpose and for surface reconstruction. Hence, the theoretical description is discussed.

2.1.1. Matching Costs

A crucial task in image processing is the definition of the matching cost C for corresponding pixels. This measurement describes how strongly two pixels, or areas around the pixels, are in correspondence, and hence, come from the same area of the scene. To this end, the intensities I are compared using a specific scheme. The comparison area can consist of a single pixel p or a set of neighboring pixels p_i bordered by polygons, such as the rectangle-based windows shown in Fig. 2.1.

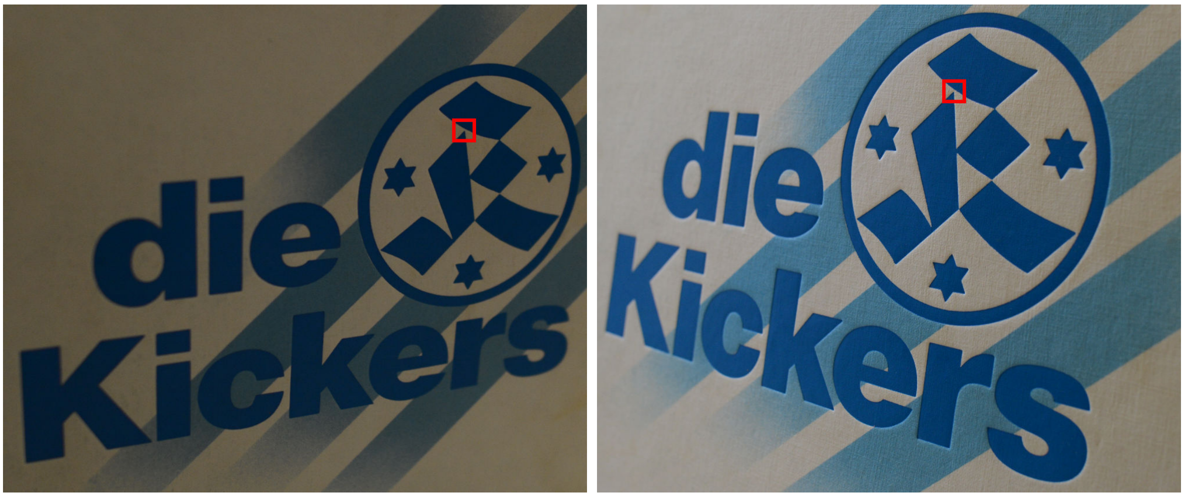


Figure 2.1.: Cover page of a book captured from two perspectives with differing lighting conditions. The pixel in the left image is matched to the right one by including the neighboring pixels in the windows.

In this section, a brief introduction for the most important and popular matching costs is provided. In addition, problems that appear when modeling the real world are discussed. Illumination changes, lack of texture, and perspective deformation are the most critical problems. Furthermore, correspondences are not unambiguous because repeating structures can appear; no correspondence exists when occlusions occur. Those problems cannot be solved based on matching costs.

Absolute Intensity Differences (SAD) (Kanade and Okutomi 1994) is defined by the summed absolute differences of neighboring pixels p_i of pixel p. A comparison between individual pixels at the relative same position is performed using the L_1 norm:

$$C_{SAD}(p) = \sum_i |I_1(p_i) - I_2(p_i + u)| \; , \tag{2.1}$$

where vector u describes the 2D transformation (shift) from the point in image I_1 to the corresponding point in image I_2. When using the L_2 norm, the method is called Sum of Squared Differences (SSD) (Matthies et al. 1989):

$$C_{SSD}(p) = \sum_i (I_1(p_i) - I_2(p_i + u))^2 \ . \tag{2.2}$$

The computationally fast methods SAD and SSD do not account for radiometric differences. Hence, they are suited for fast processing of data captured simultaneously or in laboratory settings. For scenes with changing lighting conditions, SAD is not suitable, even though the L_1 norm is quite robust against outliers. Because of fixed geometry, even small changes in the perspective cause SAD and SSD to fail.

Census (Zabih and Woodfill 1994) is a simple but powerful cost term based on binary decisions regarding illumination differences. Neighboring pixels p_i bordered by a rectangle window are transformed to a binary stream. To this end, the intensity differences to pixel p are coded by a binary relationship:

$$B(p_i) = \begin{cases} 0 & \text{if } I(p_i) \leq I(p) \\ 1 & \text{if } I(p_i) > I(p) \end{cases} \ . \tag{2.3}$$

For the comparison of two regions, the particular binary strings, coding illumination changes in the neighborhood, are compared by the Hamming distance:

$$C_{CENS}(p) = \otimes_i [B_1(p_i), B_2(p_i + u)] \ . \tag{2.4}$$

Census transform was shown to produce stable matching costs when the illumination changes (Hirschmüller and Scharstein 2009). Furthermore, it is suitable for fast processing as the Hamming distance can be coded in logic. On the negative side, Census also has a fixed geometry of the compared pixels, leading to unstable results concerning perspective differences.

Mutual information (MI) (Kim et al. 2003, Campbell et al. 2008) is an expensive but accurate pixelwise matching cost considering the entropy h of the intensities in two images as well as their joint entropy:

$$C_{MI}(p) = h_{I_1,I_2}[I(p), I(p+u)] - h_{I_1}[I(p)] - h_{I_2}[I(p+u)] \ . \tag{2.5}$$

By subtracting the entropy from the joint entropy, the significant information of the images is compared. In general, entropy and joint entropy are obtained from probability distributions over the intensities of both images:

$$H_I = -\int_0^1 P_I(i) \ \log P_I(i) \ di \ , \tag{2.6}$$

$$H_{I_1,I_2} = -\int_0^1 \int_0^1 P_{I_1,I_2}(i_1,i_2)\ \log P_{I_1,I_2}(i_1,i_2)\ di_1 di_2\ , \tag{2.7}$$

where i is an intensity from the range of the random variable I.

The calculation can be performed as a sum over pixels using a Taylor expansion (Kim et al. 2003). This leads to an entropy defined by the sum of data terms depending on the corresponding intensities of pixel p. The data terms h_I and h_{I_1,I_2} can be estimated as the logarithm of smoothed probability distributions P_I and P_{I_1,I_2} with Gaussian kernels. Hirschmüller (2008) gives a more detailed account of the numerical derivation for stereo matching.

MI is powerful because it considers particularly relevant information and obtains a pixelwise matching cost. Nevertheless, it is time consuming and requires warped images because the entropy is calculated on the entire image. Hence, there is a need for an initial guess of the geometric correspondence. Furthermore, normalization of the illumination does not consider local illumination changes that appear in complex scenes.

Normalized cross-correlation (NCC) (Hannah 1974) is also known as normalized sliding dot product. From the point of view of continuous functions, a cross-correlation is similar to a convolution. For image processing, normalization is suitable for cross-correlation because radiometry through lighting changes can vary immensely when capturing images. For the normalization of such influences, the mean of a neighborhood is subtracted and the term that equals the covariance is divided by the standard deviation of illumination. For 2D images with intensities I_1 and I_2, the NCC can be written as:

$$C_{NCC}(p) = \frac{\sum_i (I_1(p_i) - \bar{I}_1)(I_2(p_i + u) - \bar{I}_2)}{\sqrt{\sum_i (I_1(p_i) - \bar{I}_1)^2}\sqrt{\sum_i (I_2(p_i + u) - \bar{I}_2)^2}}. \tag{2.8}$$

with $\bar{I}$ denoting the average intensity of the neighboring pixel intensities. NCC has a substantially high processing time, but it can manage difficult local lighting changes. For noisy low contrast regions, NCC can fail because the normalization has a singularity in non-contrast areas. However, it has been shown to be suitable for perspective differences (Bartelsen 2012).

2.1.2. Total Variation

In the field of computer vision, TV was first used by Rudin et al. (1992) as a regularization term for nonlinear image denoising. Considering a TV term, noise can be detected and removed using specific smoothing methods. In general, TV describes a local oscillation behavior for continuous or discrete functions. For one-dimensional (1D) discrete signals, TV is defined as:

$$TV(x) = \sum_i |I(x_i) - I(x_{i+1})| \ , \tag{2.9}$$

using the L_1 norm. An L_2 norm can also be employed that squares the term of the sum. For a discrete 2D function, such as an image with intensities I, the TV that considers an L_2 norm can be written as:

$$TV(p) = \sum_{i,j} \sqrt{|I(p_{i+1,j}) - I(p_{i,j})|^2 + |I(p_{i,j+1}) - I(p_{i,j})|^2} \ , \tag{2.10}$$

where i and j are the index for x- and y- dimension, respectively.

The choice for the norm to use depends on the application. Rudin et al. (1992) argue that the L_1 norm results appear better visually than the L_2 norm results for image denoising. This can be explained by the robustness against outliers. Using optimization methods such as variational or convex optimization, the L_2 norm can be more suitable because the L_1 norm is nonlinear and computationally complex.

2.2. Geometric Basics

Obtaining 3D information from 2D images is a challenging task because the problem is inverse, and hence, ill-posed. Expecting a pinhole camera model, vector triangulation from two camera centers to the corresponding 2D points p_1 and p_2 on the image planes allows for the estimation of an approximate 3D point (cf. Fig. 2.2).

The estimation of two corresponding points p_1 and p_2 without prior knowledge is complex and computationally expensive because the search area is large. Yet, even if accurate correspondences are available, two lines through the camera centers and image points do not intersect exactly because of image noise.

3D surface reconstruction requires further parameters in addition to the image points p_1 and p_2. First, considering a world coordinate system, the camera positions T_1 and T_2, as well as the camera rotations R_1 and R_2, have to be determined. The rotation matrix and translation vector describe the transformation of a 3D point P_c in camera coordinates to the 3D point P_w in world coordinates:

$$P_w = RP_c + T \ . \tag{2.11}$$

The parameters that describe the geometry of relative camera poses are called outer parameters of the image set.

Second, the pinhole camera model does not fit the real world because images have radial or even tangential distortions caused by the camera lenses. Furthermore, the physical processes cannot be modeled with simplified camera models, resulting in reconstruction noise. The equations that define the distortion models can have a variable

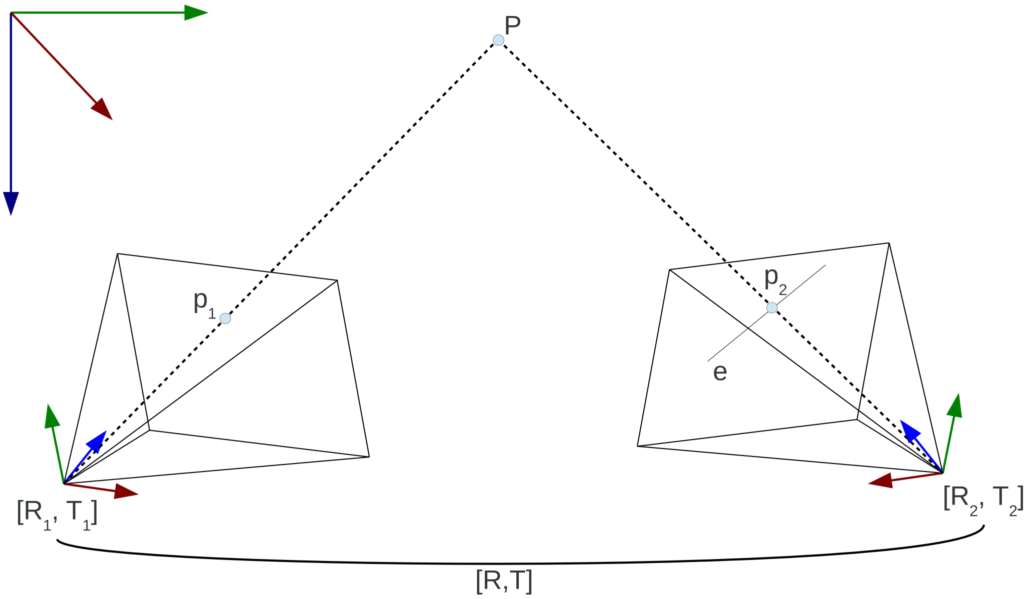


Figure 2.2.: Epipolar geometry with two pinhole cameras in a world coordinate system with absolute pose R_i, T_i for camera i and transformation R, T between the relative poses. Hence, a camera pose is defined by a rotation and a translation. Point P is a 3D point projected on the camera plane as 2D point p_i. The right camera shows the epipolar line e with respect to the left camera.

number of parameters. A relatively general model is given by Bradski and Kaehler (2008):

$$\begin{aligned} x_{rad} &= x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)\ , \\ y_{rad} &= y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)\ , \end{aligned} \tag{2.12}$$

$$\begin{aligned} x_{tang} &= x + (2 l_1 y + l_2 (r^2 + 2x^2))\ , \\ y_{tang} &= y + (l_1 (r^2 + 2y^2) + 2 l_2 x)\ . \end{aligned} \tag{2.13}$$

where r is the distance of the pixel from the origin, and x and y are the coordinates on the image plane. Eq. (2.12) defines the correction of a radial distortion parameterized by k_1, k_2, and k_3 whereas Eq. (2.13) defines the correction of a tangential distortion parameterized by l_1 and l_2.

The projection from pixel coordinates to the image plane can be defined by a 3×3 camera calibration matrix C that contains a focal length f, optical center (principal point) c, and a skew of pixels s. From the camera coordinate system, the 3D point P_c can be transformed to pixel coordinates p using the calibration matrix:

$$p = C^{-1} P_c \text{ , with } C = \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \; . \tag{2.14}$$

The set of parameters defined by the camera models are called the inner parameters of the image set.

When considering the relative orientation and relative distance $R = R_2^T R_1$ and $T = T_2 - T_1$ between two images, the search area of feature correspondence in the images can be limited by the epipolar line e (cf. Fig. 2.2). Hence, stereo matching can be simplified by finding the corresponding pixel only on one line when the relative pose is available. Nonetheless, stereo estimation remains a difficult task because at least one parameter, namely the disparity, i.e., the position on the epipolar line, has to be obtained for all pixels.

The rotation between two coordinate systems can be defined by a quaternion with only four parameters, or three angles instead of nine matrix elements. The number of parameters for the quaternion can be reduced to three because the quaternion is over-parameterized. Furthermore, multi-view reconstruction experience shows that the distortion parameters k_1 and k_2 are representative enough to obtain the required accuracy. Parameter k_3 and the tangential distortion parameters l_1 and l_2 are often not necessary for obtaining quality in the range of a couple of the tenths of a pixel. Estimating the relative poses of the cameras, for one image pair, a further parameter can be omitted because the scale is not fixed. In summary, there have to be at least seven parameters for all cameras and five parameters for all images to be estimated in the image registration step, followed by a disparity estimation for all pixels on the epipolar lines using stereo methods.

In the following sections, a possible processing pipeline is shown, describing surface estimation from arbitrary image sets without prior information, but an approximate calibration of the camera. A brief introduction to the 3D modeling process is provided, excluding the fusion of disparity maps because this is the main focus of this thesis and will be discussed in the following chapters. The remaining steps consist of image registration, stereo estimation, and polygonization of point clouds.

2.2.1. Image Registration

Accurate image registration is an important and sufficient task for dense 3D surface reconstruction. The registration process equals SfM because image registration involves the simultaneous estimation of 3D geometry (structure) and relative camera poses (motion). 3D geometry is particularly important for the estimation of unknown parameters in the image registration process. Robust bundle adjustment allows for accurate parameter estimation considering the obtained 3D geometry. The input of the image registration usually consists of a calibrated or even an uncalibrated set of images.

A suitable model for camera calibration consists of multiple parameters such as distortion, focal length, skew, and optical center (cf. Section 2.2). The number of calibration parameters can be maximum of ten per camera, depending on the distortion model (cf. Eqs. (2.12) - (2.14)). The outer parameters count seven per image, four parameters for the rotation quaternion and three parameters for the translation vector. As already mentioned, the four rotation parameters overparameterize a rotation and can be reduced to three in the process of parameter estimation.

Concerning the estimation of inner parameters, several approaches and calibration toolkits exist. Additional information, such as calibration grids, is suitable for the estimation of focal length, camera center, skew, and distortion parameters, even in the range of hundredth of a pixel. In several applications, the parameters have to be obtained directly from the image set because no initial calibration is known. Based on an initial (educated) guess, parameter optimization of inner parameters can be performed simultaneously with the estimation of outer ones.

In the following paragraphs, an image registration process is described that does not need prior information, besides the image set and possibly an approximate calibration. In an initial step, the images are checked for overlap with further images of the set. The feature points in the 2D images are matched to corresponding pixels also in challenging configurations. There is a large amount of methods for feature extraction that detect different types of structures (FÖRSTNER and GÜLCH 1987, HARRIS and STEPHENS 1988, LOWE 2004, FÖRSTNER et al. 2009, TOLA et al. 2010). By considering scale-space properties, feature matching is becoming scale-invariant, which is a requirement for scenes in challenging configurations. Isotropic filters allow for rotation invariant feature descriptions, but can lead to weakness concerning perspective deformations. The most popular and powerful feature extractor is the Scale-Invariant Feature Transform (SIFT) (LOWE 2004). In addition to scale and rotation-invariant feature descriptions, a matching method is presented. This method is based on the comparison of feature vectors that represent the characteristics of a specific area around a pixel. Yet, it was shown that other matching methods (cf. Section 2.1.1) can be more stable for perspective deformations (BARTELSEN 2012).

The robust bundle adjustment employed obtains a locally optimal solution for unknown parameters. Thus, an approximate solution for the inner and the outer parameters is required that is close to the globally optimal set of parameters.

The image correspondences, extracted by feature matching, can be used to obtain 3D geometry also suitable for camera calibration. Considering a camera pair whose relative poses consist only of rotations around the viewing direction, camera calibration can be performed as described by DE AGAPITO et al. (1998) or FRAHM and KOCH (2003). An initial estimation of the inner parameters can even be performed for general motion (POLLEFEYS et al. 1999) when image pairs with strong 3D information covering a large depth range are available. Whereas calibration based on image data is feasible, the assumption that needs to be fulfilled often does not hold. Yet, this is mostly not a problem in practical applications, because in many cases, an extremely good approximation for camera parameters is available from prior experiments with the same camera. Even if such is not the case, one can derive a suitable approximation from information available in the Exchangeable Image File Format (EXIF) description of the image (focal length) and on the camera (pixel size) on the Internet.

Considering correspondences in two or multiple images that show the same region, a relative pose can be estimated. It has been shown that only five corresponding points are necessary to obtain the relative pose between two calibrated images (Nistér 2003). For this five-point algorithm, no approximation of the model is necessary. The relative pose is defined by rotation R and translation T between the camera coordinate systems (cf. Fig. 2.2).

Unfortunately, stable and accurate estimation of five corresponding points in two images is not trivial because the estimation of corresponding pixels can lead to wrong and inaccurate matches. Random algorithms, such as Random Sample Consensus (RANSAC) (Fischler and Bolles 1981) allow the consideration of these uncertainties. RANSAC is an iterative algorithm for the estimation of parameters by fitting a set of measurements under uncertainties and outliers. In the case of the five-point algorithm, the measurements used are five corresponding points, which are chosen randomly. After estimation of the relative pose, the complete set of measurements is checked for consistency with the obtained pose. The RANSAC algorithm continues considering further five random points until a solution with probability is obtained. Experiments underscore that randomized methods such as RANSAC can obtain correct results even for more than 90% outliers.

Employing the relative poses of all camera pairs, the image features can be tracked over multiple images on paths over the complete set. This is theoretically feasible; however, in the employed approach (Mayer et al. 2011) RANSAC using two times the five-point algorithm is employed also for triplets because it was found that this makes the solution much more robust. For this, triplet paths are estimated through graph algorithms that model the cameras as nodes and the overlapping images as edges (Bartelsen et al. 2012). Such tracking results in a set of n 3D points with information about camera visibility (cf. Fig. 2.3).

For a globally optimal solution of the outer and inner parameters, at least one bundle adjustment has to be conducted. In practice, many intermediate bundle adjustments are employed to avoid the solution from drifting too far from the compact solution. In detail, the bundle adjustment uses the n 3D points x_i^W in world coordinates that appear in o images. Furthermore, the 2D coordinates on the image plane in camera j from point i are known. The tracked 3D point also can be transformed in the image plane using Eqs. (2.11) to (2.14). Unfortunately, this transformation is neither linear nor convex. The bundle adjustment solves this complex problem by minimizing the squared

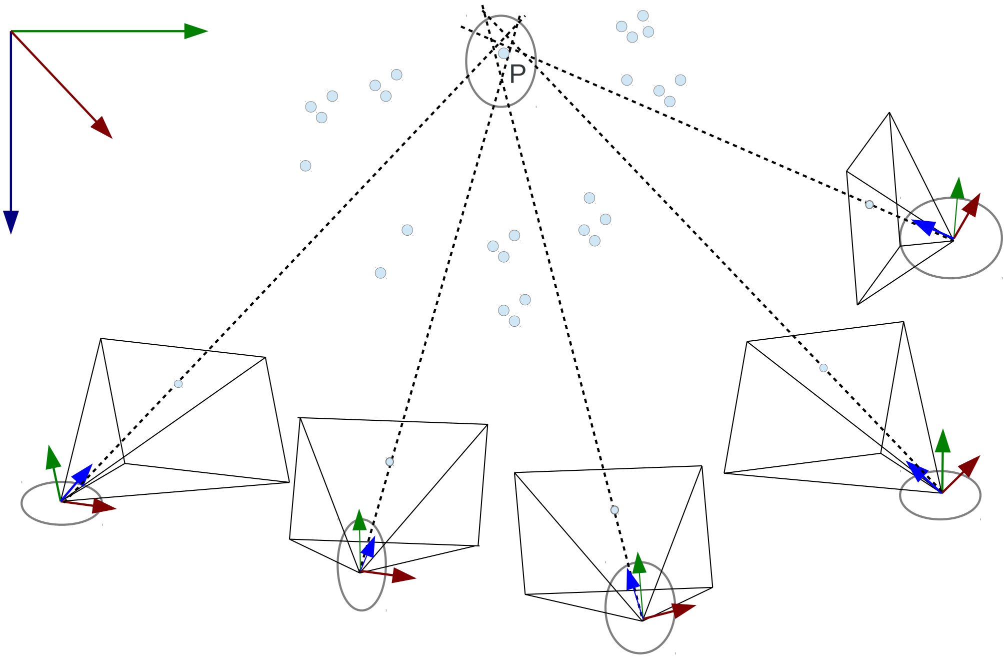


Figure 2.3.: A Point cloud extracted by SfM. Point P, obtained by observations from five cameras, has ten different measurements that define an uncertainty. The uncertainty of the 3D points and the resulting uncertainties of the camera poses can be estimated during image registration.

re-projection error:

$$E_{rp} = \sum_{i=1}^{n} \sum_{j=1}^{o} [d_{ij}]^2 \ , \tag{2.15}$$

which denotes the sum of squared Euclidean distances d between the 2D points from feature extraction, and the re-projected optimized 3D points. To solve the nonlinear problem, Eq. (2.15) is linearized by means of the first order Taylor expansion. It has to be noted that this linearization violates the terms of finding a global solution. In practice, the linearized solution obtains an optimum close to the global optimum under the premise that the initial set of parameters is in the range of the global optimum. The linear system of equations consists of a sparse matrix derived from the Jacobian of the measurements with information on the cameras, 3D points, and covariances that represent the uncertainty measurement. The resulting system can be optimized, e.g., using the Levenberg-Marquardt algorithm. Because there are outliers in the data even after RANSAC filtering, a robust re-weighting that employs an M-estimator is used in the employed approach (Mayer et al. 2011).

The processing pipeline of the employed complex image registration process is presented in Fig. 2.4 which is a description of the method published by Mayer et al. (2011).

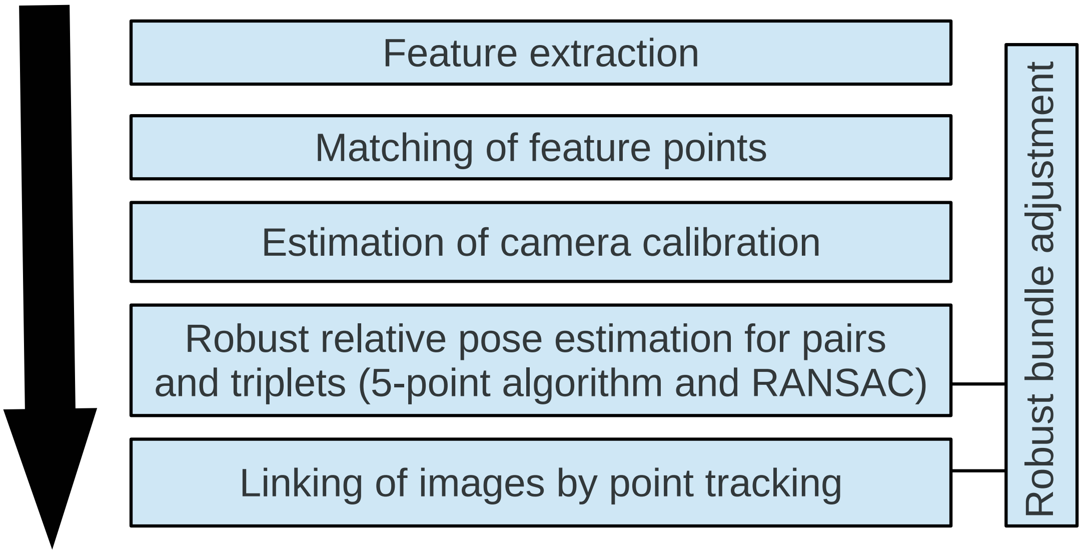

Feature extraction
Matching of feature points
Estimation of camera calibration
Robust relative pose estimation for pairs and triplets (5-point algorithm and RANSAC)
Linking of images by point tracking
Robust bundle adjustment

Figure 2.4.: Image registration process chain based on robust bundle adjustment.

2.2.2. Stereo Matching

The main tasks of (multi-view) stereo methods are the estimation and fusion of pixelwise disparities. The MVS configuration differs from the standard stereo configuration in that multiple images are considered instead of only one image pair. The disparity is defined by the distance of two pixels, from two images with different camera positions, on the corresponding epipolar lines. When the cameras are calibrated, i.e., the plane at infinity is known, the calibration is considered to normalize the position on the epipolar lines. The smaller the disparity, the bigger is the depth of the point that corresponds to the distance from the pixel to the point. The relationship is an inverse ratio, i.e., twice the disparity means half the depth. Considering fixed stereo cameras or registered image sets, the relative camera geometry of the cameras is known. Hence, it is sufficient to find the corresponding pixel on the epipolar line (cf. Fig. 2.2). For MVS or the combination of multiple stereo disparity maps the disparities have to be weighted depending on the baseline between camera pairs to achieve global consistency. An example of a disparity map is shown in Fig. 2.5.

Considering an accurately registered image set, all outer and inner parameters of the scene configuration are known. Image registration with subpixel accuracy allows for pixelwise matching on the epipolar line. Unfortunately, stereo matching remains a complex problem because it has to consider occlusions and repeating structures as well

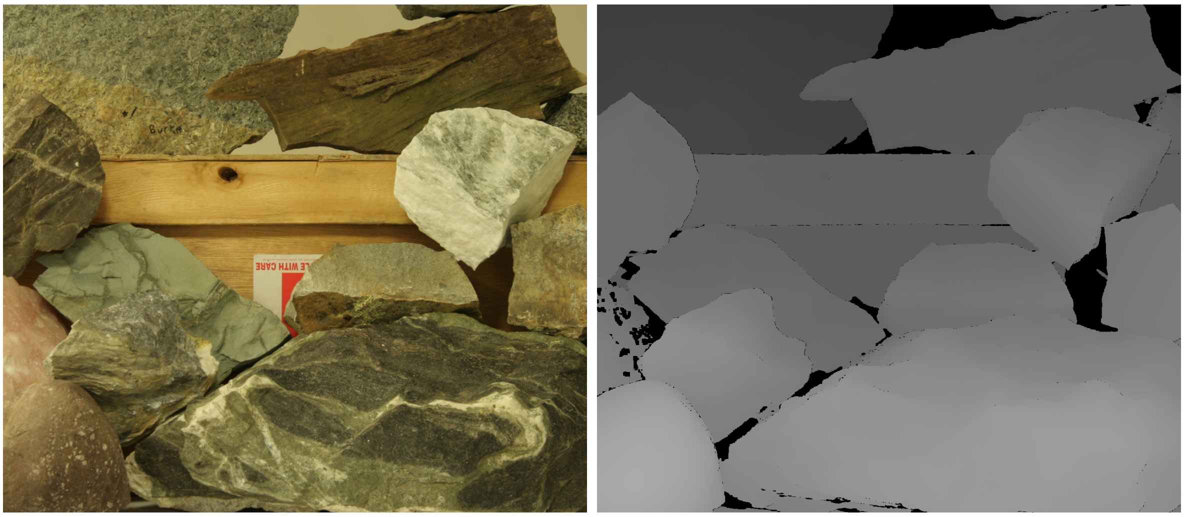


Figure 2.5.: An image from the Middlebury stereo benchmark (Scharstein 2014b). On the right, the disparity ground truth is shown. The disparities are coded from white (large disparities) to dark grey (small disparities). Missing pixels are marked in black.

as geometric and physical deformations.

A naïve method of disparity estimation for a pixel is to compare all pixels on the epipolar line in the second image, considering a specific type of pixel matching cost (cf. Section 2.1.1). Unfortunately, the pixel with minimum cost does not generally describe true disparity because of the ill-posedness of the problem. Nevertheless, several approaches exist that tackle the problem by considering prior information. In addition to the step of Cost Calculation, a step of Cost Aggregation (Scharstein and Szeliski 2002), considering additional information, is integrated in most methods. For an overview of the numerous methods published presently, see the Middlebury Stereo Vision Page (Scharstein 2014b).

The stereo matching problem can be formulated as an optimization problem. Hence, local and global stereo methods can be distinguished. Local methods operate on a limited area in the pixel neighborhood, whereas global methods seek a globally optimal solution over all pixels. For providing a deeper insight in the general idea, a brief introduction to the most important local and global stereo methods is given.

Local methods aggregate the matching cost by optimization within a limited support region. This region can be either defined only in the 2D image or in a 3D space with an additional dimension defined by disparities (xy-d space) (Szeliski 2011). In addition to both image dimensions, the third dimension of the xy-d space discretely describes the cost of pixels obtained with the Cost Calculation step. Two representative methods for the class of local optimization are Plane Sweep (Collins 1996) and Patch Match (Bleyer et al. 2011).

Plane Sweep (COLLINS 1996) defines a virtual camera for multi-view or stereo configurations. The choice of virtual camera geometry is application dependent. In general, it should represent a basic pose for all cameras. From the view of the virtual camera, fronto-parallel planes are evaluated along the viewing direction. The pixel intensities from each image are projected onto the sweeping plane. Similar intensities for different images on the sweeping plane are assumed to mean correspondence. The matching costs are determined with methods such as SSD (cf. Section 2.1.1). Finally, the disparity that corresponds to the plane with minimum cost is chosen for the resulting disparity map. Plane Sweep obtains accurate disparity maps fast, but only for special configurations. In particular, the use of only one set of parallel planes can lead to inaccurate surfaces for general configurations.

Patch Match (BLEYER et al. 2011) is a local method for obtaining high accuracy even for complex configurations. Operating in the 2D image with additional plane parameters, and not in the disparity space, reduces the amount of memory required. For all pixels, a disparity and a plane are obtained. Using one plane per point frees Patch Match from fronto-parallel priors. Yet, considering planes leads to a larger amount of parameters to optimize. The optimization of the disparities and planes f at pixel p is done iteratively by minimizing a cost function:

$$C_{PM}(p, f) = \sum_{q \in \mathcal{N}} w(p, q)\ \rho(q, q - [a_f q_x + b_f q_y + c_f])\ , \tag{2.16}$$

where the parameters a_f, b_f and c_f describe the 3D plane that corresponds to pixel p. $\mathcal{N}$ defines the spatial neighboring points of p considering plane f. The weight function $w(p, q)$ is a parameterized function that exponentially penalizes intensity differences. Function ρ computes the pixel dissimilarity between p and its relationship to plane f. The iteration steps include pixelwise processing of spatial propagation, view propagation, and plane refinement. As local method, Plane Sweep has problems in weakly textured areas.

Global methods usually employ an aggregation step that minimizes a global cost function. The strength of global optimization consists in the regularization prior that allows for the finding of a surface that represents the entire data. In particular, unreliable regions can be accurately reconstructed by considering a reliable and accurately determined neighborhood. Global methods can be computationally expensive and are limited concerning parallelization.

Graph Cuts (KOLMOGOROV and ZABIH 2002) solutions are used in many successful stereo matching methods and are representative of global methods. The *xy-d* space is transformed to graphs where the nodes represent the discrete states and the edges an estimated cost. Within an edge costs, e.g., slant surfaces are penalized introducing a fronto-parallel bias. The min-cut based graph cuts solution allows for the modeling of an optimal surface between front and back in the viewing direction. Global graph-based methods can reach accurate surface quality, but are limited in runtime performance.

Belief Propagation (BP) (Sun et al. 2003) improves graph-based methods. It combines a set of Markov Random Fields (MRFs) into a Markov network. The optimization is performed with a Bayesian approach. Belief Propagation methods are among the best in stereo concerning accuracy (Scharstein 2014b). However, concerning scalability and runtime performance, they are clearly limited.

A hybrid between the local and global methods is **Semi-Global Matching** (SGM) (Hirschmüller 2008) which combines the surface regularization of global methods and the runtime performance of local methods. Dynamic programming methods are used that optimize per line in the image. SGM uses several optimization paths with different directions. As for the fusion presented in this thesis, disparity maps derived by SGM are used, a more detailed description is given in Section 5.1.

More details on matching costs and stereo methods are provided, e.g., in the computer vision textbook by Szeliski (2011).

2.2.3. Polygonization of Point Clouds

The transformation of 3D point clouds to spatial surfaces can be defined as a geometric optimization problem. Whereas point clouds convey information in three space dimensions, connected sets of polygons provide additional information on the topological connection. Surfaces can be represented as a set of connected polygons, e.g., triangles. As for the stereo matching in Section 2.2.2, again, there are global and local methods for the solution. Focusing on scalable 3D modeling, local polygonization methods are of particular interest because they allow for parallelization. Nonetheless, global methods have to be discussed because they obtain more accurate results.

Usually, the optimization considers the 3D points and corresponding normal vectors because the latter is especially important for the topology. If there is no information available on normal vectors, they can be estimated considering neighboring points. A general method for normal estimation is based on the covariance information of point distributions. The distribution of points that describe a surface usually spread in two directions. Hence, the smallest axis of the covariance describes the normal direction. The lengths of the axes correspond to the eigenvalues of the covariance matrix. A more detailed description of the theoretical background is given in Section 2.3.2. The covariance matrix C of point P can be calculated by:

$$C_P = \frac{1}{n}\sum_{i=1}^{n}(P_i - \bar{P})(P_i - \bar{P})^T \ , \tag{2.17}$$

where P_i are the neighbors of P and $\bar{P}$ is the average point over all neighbors. The normal vector can only be estimated if the smallest eigenvalue is considerably smaller than the remaining two eigenvalues. If the eigenvalues are similar, the point cloud is not accurate enough, e.g., it can be caused by noise and outliers of disparity maps. Otherwise, the point cloud is not flat enough at this point because it is part of a 3D edge or corner. In this thesis, a fusion process is presented that obtains extremely detailed point clouds from disparity maps derived by means of SGM.

Global triangulation methods can manage noisy point clouds particularly well because they can consider prior information concerning the smoothness of a surface or its topology. Poisson reconstruction (Kazhdan et al. 2006) is a popular polygonization method that is representative of global optimization methods. Its input is the point cloud and an initial guess of the normal vectors. Hence, it is not feasible for point clouds with extremely large noise or a significant amount of outliers. Poisson reconstruction generates watertight surfaces based on the assumption of closed surfaces that define objects. Poisson reconstruction attempts to fit an indicator function $\mathcal{X}$ defined as zero outside and one inside the object. The gradient of the indicator function is zero, except on the surface. This optimization problem is performed solving a Poisson equation:

$$-\nabla\mathcal{X} = \delta V \ , \tag{2.18}$$

where vector field $\overrightarrow{V}$ is an initial pointcloud with normal vectors. Solving the Poisson equation is well-known and can be approximated by a least squares solution. For discrete realization, the point cloud is transformed to octrees that represent the volumetric units of the 3D space with variable size. The estimated characteristic function assigns values to the volumes. Subsequently, a discrete level set defines the surface represented by a set of volumes. For the polygonization of voxels, the local Marching Cubes method (Lorensen and Cline 1987) is used.

Local Triangulation methods consider only a limited neighborhood for surface reconstruction. Hence, the area of influence is limited for all polygons. In this thesis it is shown that a limitation is suitable for an unlimited scalability of surface reconstruction.

A popular local triangulation method is Marching Cubes (Lorensen and Cline 1987). It uses spatial volumes instead of 3D point clouds as input data. In a volumetric grid, surfaces can be represented by considering varying values assigned to the volumetric elements. When having positive values behind and negative values in front of a surface, the zero level set defines the surface. For spatial volumes, Marching Cubes analyzes the eight neighboring voxels for possible level set surfaces. The analysis is based on the assumption that a limited number of triangle configurations exists that are stored in a look up table. The limitation allows for extremely fast and real-time applicable surface reconstruction having volumetric data.

Several local methods that operate on 3D point clouds exist to transform the processing to 2D meshing, e.g., with Delaunay triangulation. Bodenmüller (2009) proposed an incremental triangulation method that has a limited area of influence (cf. Fig. 2.6).

For all incrementally added points, the neighboring points and triangles are projected on a plane defined by the normal vector. The new point is connected with all neighboring points in a local area. If a new edge intersects an old one, the longer one is removed, avoiding degenerated triangles.

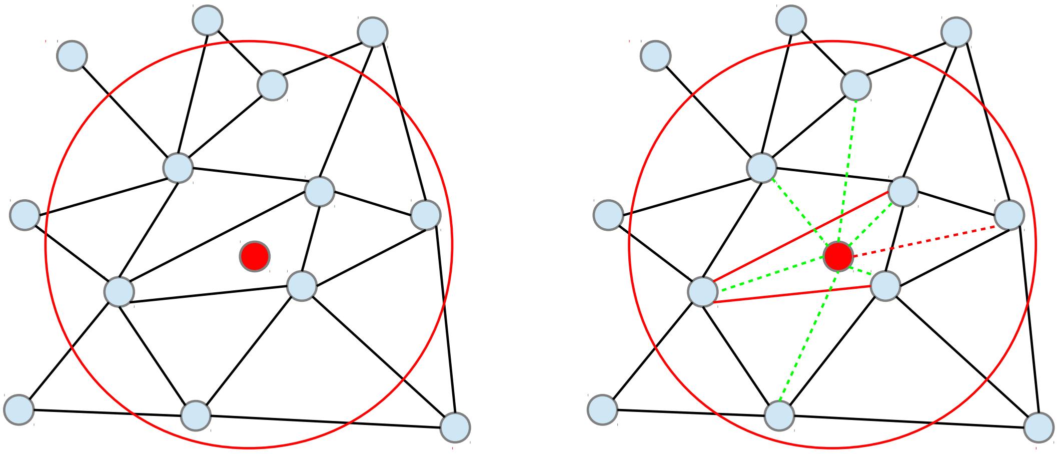


Figure 2.6.: Local update of the meshing for a new vertex according to Bodenmüller (2009). Left: New vertex v (red point) and projected neighboring vertices (circle). Right: New candidate edges from vertex v to all neighboring vertices. The green lines are part of the new triangulation; the red lines are removed because they do not comply with the conditions.

Local methods have advantages in runtime performance and scalability. Global methods obtain a higher accuracy and can manage sparse point clouds. Yet, the latter also means that they tend to produce ghost regions.

2.3. Stochastic Basics

Stochastic is the mathematical science that studies randomness, and it is a generic term for statistics and probability theory. Probability theory is concerned with random processes that are influenced by expected relative frequencies following a probabilistic axiom system. In statistics, measured data can be used to derive knowledge regarding unknown probabilities. Many problems in computer vision are ill-posed, for which no solution can be estimated directly. A stochastic description of geometric properties is extremely useful for obtaining the most probable solution. Hence, it is of high importance to consider the accuracy of measurements. Because image-based methods generate depth measurements indirectly, the description is even more complex. In this thesis, a statistical approach is presented that allows learning the disparity quality considering ground-truth data. Furthermore, a probabilistic sound fusion method for 3D information is proposed. This combination characterizes the method as stochastic fusion of disparity maps. For a better understanding of the employed stochastic methods, a brief theoretical overview is given. The overview focuses especially on probabilistic distributions, statistical learning, and fusion theory.

2.3.1. Distributions

Discrete probabilities represent relative frequencies of events and are in the range of $[0, 1]$ following the Kolmogorov axioms. It is important to consider probability densities as parameterized functions. On one hand, they can represent a big set of probabilities with a small number of parameters. On the other hand, they are suitable for obtaining general densities. Therefore, in the n-dimensional case, the density is described by a non-negative integrable function p with:

$$\int\limits_{\mathbb{R}^n} p(x_1, ..., x_n)\ dx_1\ ...\ dx_n = 1\ . \tag{2.19}$$

Because function p is a probabilistic density function, a corresponding probabilistic cumulative function P is defined as:

$$P(x_1, ..., x_n) = \int\limits_{-\infty}^{x_1,...,x_n} p(t_1, ..., t_n)\ d_{t_1}\ ...\ d_{t_n}\ . \tag{2.20}$$

In the following paragraphs, density and cumulative functions are discussed based on the univariate uniform and normal (Gaussian) distribution. Both functions are important for disparity map fusion, as shown in Section 4.

The **Uniform distribution** is the arrangement with density p with equal probabilities inside the interval $[a, b]$ and zero probability outside:

$$p(x) = \begin{cases} \frac{1}{b-a} & \text{if } x \in [a, b] \\ 0 & \text{else} \end{cases}\ . \tag{2.21}$$

Hence, by integrating the density function, the cumulative function is given by:

$$P(x) = \begin{cases} 0 \text{ if } x < a \\ \frac{x-a}{b-a} \text{ if } x \in [a, b] \\ 1 \text{ if } x > b \end{cases}\ . \tag{2.22}$$

P is a linear function in the dimension of x and interval $[a, b]$. Fig. 2.7 (left) shows the corresponding graphs for the density and cumulative function.

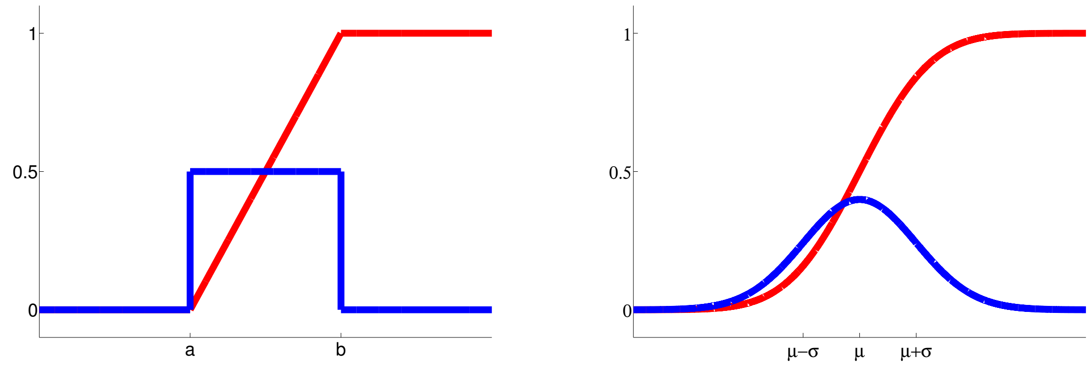


Figure 2.7.: Left: A uniform probability density function (blue) and the corresponding cumulative function (red). Right: The Gaussian probability density function (blue) and the cumulative distribution function (red).

Gaussian distribution or normal distribution $\mathcal{N}(\mu, \sigma)$ fulfills the axioms of probabilistic density functions and considers an exponential rise of the probability. A Gaussian is parameterized by the expected value μ where $f(x)$ is maximum and σ^2 describes the quadratic standard deviation. The univariate Gaussian probabilistic density (PDF) function is written as:

$$\mathcal{N}_{PDF}(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \ . \tag{2.23}$$

The Gaussian is widely applicable because the exponential law of errors represents many physical and numerical assumptions in a stable way. Unfortunately, there is no closed form of the integral, required for the Gaussian cumulative distribution function (CDF):

$$\mathcal{N}_{CDF} = \int\limits_{-\infty}^{x} \mathcal{N}_{PDF}(t)\ dt = \frac{1}{\sigma\sqrt{2\pi}} \int\limits_{-\infty}^{x} \exp(-\frac{(t-\mu)^2}{2\sigma^2})dt \ . \tag{2.24}$$

Fortunately, the Gaussian CDF can be accurately approximated by considering numerical properties. SACHS and HEDDERICH (2006) propose the following transformation for Eq. (2.23):

$$\frac{X-\mu}{\sigma} = Z \ , \tag{2.25}$$

where X is the random variable. By substitution of $u = \frac{t-\mu}{\sigma}$ and using $\frac{du}{dt} = \frac{1}{\sigma}$, all

Gaussians can be transformed into the standard Gaussian $\theta = \mathcal{N}(0,1)$:

$$\theta(x) = \frac{1}{\sqrt{2\pi}} \int\limits_{-\infty}^{x} \exp\left(-\frac{t^2}{2}\right) \, dt \ . \tag{2.26}$$

In other words, for all Gaussians, a simple transformation of the expected value by means of Eq. (2.25) exists, transforming the Gaussian to a standard Gaussian with $\mu = 0$ and $\sigma = 1$. The numerical values for the standard Gaussian can be stored in a table for practical applications.

Fig. 2.7 (right) shows the corresponding graphs for the PDF and CDF.

Mixture distributions combine a set of probability functions and are suitable for applications with multiple influences on the expected value. Multiple measurements can lead to multiple maximums of the probability function. To distinguish the different influences, e.g., a mixture of Gaussians or even a mixture of Gaussians with uniform distributions can be used:

$$p(x) = \begin{cases} \alpha\mathcal{N}(x) + \beta\frac{1}{b-a} & \text{if } x \in [a,b] \\ \alpha\mathcal{N}(x) & \text{else} \end{cases} , \tag{2.27}$$

where $\alpha + \beta = 1$ considering the probabilistic axioms. Because mixture functions combine single functions by a scaled sum, the cumulative case can also be derived by the sum of the individual cumulative functions:

$$P(x) = \begin{cases} \alpha\mathcal{N}_{CDF}(x) + \beta\frac{x-a}{b-a} & \text{if } x \in [a,b] \\ \alpha\mathcal{N}_{CDF}(x) & \text{else} \end{cases} . \tag{2.28}$$

2.3.2. Expectation, Variance, and Entropy

The expectation and the covariance of distributions are of high importance for managing probability distributions. The expectation $\mathbb{E}[f]$ describes the average value of function $f(x)$ that is related to the probability distribution $p(x)$. Considering a random variable X with probability distribution $p(x)$, the expectation is defined as:

$$\mathbb{E}[X] = \int x \; p(x) \; dx \ . \tag{2.29}$$

For a discrete distribution, the integral can be replaced by the sum:

$$\mathbb{E}[X] = \sum x \; p(x) \ . \tag{2.30}$$

The variance for a random variable from $f(x)$ is defined by:

$$var[f] = \mathbb{E}[(X - \mathbb{E}[X])^2] \; = \int (x - \mathbb{E}[X])^2 \; p(x) \; dx, \tag{2.31}$$

providing the information on the variability of the values around the expectation $\mathbb{E}[X]$.

For the Gaussian from Eq. (2.23), the expectation and the variance are the parameters μ and σ^2 that define the infinite set of Gaussians.

For two random variables X and Y the covariance is important. It describes the strength of the correlation of two random variables, i.e., how much they vary together. It is defined by:

$$cov[X, Y] = \mathbb{E}_{X,Y}[X, Y] - \mathbb{E}[X]\mathbb{E}[Y] \tag{2.32}$$

The entropy H of a probability distribution is also of interest for computer vision (cf., e.g., Section 2.1.1). This entropy is related to probability distributions and has a similar notation to the expectation and the covariance:

$$H[X] = - \int p(x) \log p(x) \; dx \; . \tag{2.33}$$

Entropy measures the uncertainty in values selected randomly from a distribution.

2.3.3. Machine Learning

Arguably, machine learning has become one of the most important fields in computer vision. Many applications require a stochastic interpretation of data. For instance, machine learning methods are suitable for the estimation of a set of parameters θ of a specific probability distribution by considering training samples $\mathcal{D}$. In general, this means a representation and generalization of actual measurements. Even extremely noisy measurements with outliers can be generalized using statistical methods.

Several machine learning methods are important for data fusion. The most important for disparity map fusion are: Maximum Likelihood (MLE), Maximum A Posterior (MAP), Bayesian Parameter Estimation (BPE), and Expectation Maximization (EM). Many machine learning methods for parameter estimation make use of the Bayes rule:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta) \; p(\mathcal{D})}{p(\theta)} \; , \tag{2.34}$$

relating posterior $p(\theta|\mathcal{D})$, likelihood $p(\mathcal{D}|\theta)$, prior $p(\mathcal{D})$ and evidence $p(\theta)$. For the remaining portions of this section, it is assumed that a set of sample data $\mathcal{D} = d_1, ..., d_n$ from n independent measurements is available. Because it is employed in this thesis, Binary Bayes theory is discussed in Section 2.3.4.

MLE is a powerful method for learning parameters given measurements that follow the distribution of a specific model. In general, the likelihood part from the Bayes rule is optimized by MLE. The other parts of the Bayes rule are not considered, and regarded as static. All the measurements have to result from a probability function p with an unknown set of parameters θ. As a first step, the joint density function of the measurement probabilities is defined as a likelihood:

$$p(\mathcal{D}|\theta) = \prod_{i=1}^{n} p(d_i|\theta) \ . \tag{2.35}$$

To estimate a global maximum of the likelihood function, it can be useful to consider the logarithmic likelihood function:

$$l(\theta) = \log p(\mathcal{D}|\theta) = \sum_{i=1}^{n} \log p(d_i|\theta) \ , \tag{2.36}$$

because the logarithmic function has the same global maxima and usually the derivations for the summands can be solved more easily. Hence, the optimal set of parameters $\hat{\theta}$ derived by MLE optimization can be written as:

$$\hat{\theta}_{ML} = \arg\max_{\theta} p(\mathcal{D}|\theta) \ . \tag{2.37}$$

This can be obtained directly by setting the gradient $\nabla l(\theta) = 0$.

In the case of the univariate normal distribution from Eq. (2.23), the MLE estimation considering the log-Gaussian:

$$l(\mu, \sigma) = \log(\mathcal{N}(\mu, \sigma)) = -\log(\sigma\sqrt{2\pi}) - \frac{(x-\mu)^2}{2\sigma^2}, \tag{2.38}$$

leads the gradient formulation that considers $\theta_1 = \mu$ and $\theta_2 = \sigma^2$ into:

$$\nabla l(\theta) = \begin{pmatrix} \frac{(x-\theta_1)}{\theta_2} \\ -\frac{1}{\theta_2} + \frac{(x-\theta_1)^2}{2\theta_2} \end{pmatrix} = 0 \ , \tag{2.39}$$

which can be solved linearly resulting in two equations for the parameters μ and σ:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \ , \ \hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2 \ . \tag{2.40}$$

A more detailed derivation of ML and the Gaussian case is provided by Duda et al. (2000).

A class related to MLE is MAP, where the posterior is optimized instead of the likelihood:

$$\hat{\theta}_{MAP} = \arg\max_{\theta} p(\theta|\mathcal{D}) \ . \tag{2.41}$$

The parameters θ that maximize $l(\theta)p(\theta)$ are estimated, where $p(\theta)$ describes the prior probability of the parameters. The evidence only operates as a scaling factor and can be ignored for maximum estimation. With good initial knowledge of the parameters, MAP can be more accurate and stable than MLE, especially for limited training sets.

Parameter estimation by means of the Bayes Parameter Estimator (BPE) differs from MLE and MAP because the set of parameters is not assumed to be fixed. The parameters θ are represented as random variables considering prior distributions over these parameters. The Bayesian estimation is defined by the expected value of the posterior density:

$$\hat{\theta}_{\text{BPE}} = E[\theta|\mathcal{D}] = \int \theta \ p(\theta|\mathcal{D}) \ d\theta \ . \tag{2.42}$$

As an example, consider the univariate Gaussian case that uses random variables for the parameters $p(\theta) = \mathcal{N}(\mu, \sigma)$; such a case consists of the likelihood formulation:

$$p(\mathcal{D}|\theta) = \prod_{i=1}^{n} p(d_i|\mu) = \frac{1}{(2\pi\sigma^2)^{n/2}} \ \exp(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(d_i - \mu)^2) \ . \tag{2.43}$$

For simplicity, the variance σ^2 is supposed to be fixed. The prior of the expected value is assumed to follow a Gaussian:

$$p(\mu) = \mathcal{N}(\mu|\mu_0, \sigma_0^2) \ . \tag{2.44}$$

As scaled product of the prior distribution and the likelihood a Gaussian also follows for the posterior distribution:

$$p(\mu|\mathcal{D}) = \mathcal{N}(\mu|\mu_n, \sigma_n^2) \ , \tag{2.45}$$

by manipulations that involve the completion of the square in the exponent. The mean μ_n can be computed as:

$$\mu_n = \frac{\sigma^2}{n\sigma_0^2 + \sigma^2}\mu_0 + \frac{n\sigma_0^2}{n\sigma_0^2 + \sigma^2}\mu_{MLE}, \tag{2.46}$$

where μ_{ML} is the maximum likelihood solution for μ (cf. Eq. (2.40)). For a detailed deviation and a more general description, see the textbook of Duda et al. (2000).

Hence, the Bayesian parameter estimation in the Gaussian case is a weighted mean of the prior mean μ_0 and the mean μ_{MLE} estimated from the measurements. The fewer measurements are available, the more the expected value tends to the prior value.

MLE, MAP, and BPE assume that the measurement follows a specific probabilistic distribution. For real-world data, a problem can appear where more than one source is responsible for the measurement, e.g., caused by outliers. This can be modeled by mixture functions that combine a set of distributions (cf. Section 2.3.1). The problem of learning parameters from mixture functions is obvious: when data follow different functions, the estimation of the parameters of one function can be influenced by data that belong to another, i.e., to the wrong function. For the estimation of parameters from different functions, the Expectation Maximization (EM) algorithm can be used.

In case of different sources, EM follows the basic idea of iteratively estimating the log-likelihood that considers assignment to the functions $\mathcal{A}$ of the data. The log-likelihood optimizes the probability function $p(\mathcal{D}, \mathcal{A}|\theta)$ (Dempster et al. 1977). As expectation (E) steps the expected value of the log-likelihood, under the current optimal set θ_i, is calculated:

$$\mathcal{Q}(\theta, \theta_i) = \mathbb{E}_{\mathcal{A}|\mathcal{D},\theta_i}[\log p(\mathcal{D}, \mathcal{A}|\theta)] \ . \tag{2.47}$$

The maximization (M) step uses the probabilistic assignment of the expectation step for further optimization of the parameter set. This can be done by MLE weighting the data obtained by the probabilistic assignment.

The EM algorithm can be described in general as (Duda et al. 2000):

1. Choose an initial set of the parameters $\theta_i = \theta_0$.
2. E step: Compute the expected influence of the data on the specific variables $\mathcal{Q}(\theta; \theta_i)$.
3. M step: Maximize the parameters θ over the influencing subsets:
   $\arg\max_{\theta} \mathcal{Q}(\theta; \theta_{old})$.
4. Confirm whether the distribution fits the expectation; otherwise return to step 2 with: $\theta_{old} = \theta$.

An example is learning the sum of a uniform function and a Gaussian from Eq. (2.27) (cf. Fig. 2.8). A first guess of the parameters $\theta = \{\mu, \sigma, \alpha, \beta\}$ is necessary for EM

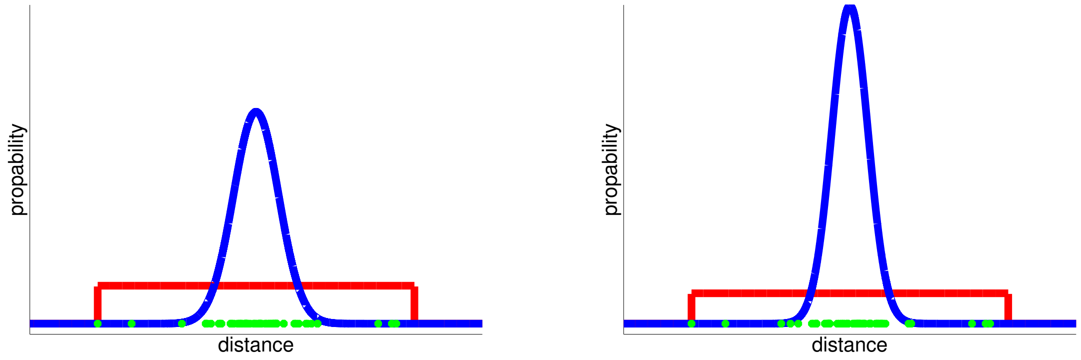


Figure 2.8.: One Expectation Maximization step. Left: Initial Gaussian and uniform distribution. The points show the measurements that are weighted by the probability distribution. Right: The Gaussian and uniform distributions after the M-step.

optimization. In the E step, the measurements can be weighted considering the initial set of parameters. This assignment can also be done by a Bayesian Classifier (Duda et al. 2000), which assigns the measurements to the function that is more probable. After the assignment, the M step can by performed by MLE.

A problem concerned with EM is that the initial set of parameters has to be approximately accurate. By uninformed or even random setting of the parameters, the optimization tends to run into a local maximum.

2.3.4. Binary Bayes Theory

With a set of coherent probability distributions with known parameters, it can be of interest to combine them. To this end, a theoretical framework of probabilistic fusion is necessary. In general, for data fusion, such as a set of measurements from active or passive sensors, a Bayes Filter can be employed. The Bayes Filter is based on the well-known Bayes rule that relates conditional properties of the type $p(A|B)$ to its inverse $p(B|A)$. A and B can be general events. The rule combines prior, posterior, likelihood, and evidence:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} \; . \tag{2.48}$$

This means that a posterior probability can be obtained by the product of likelihood and prior probability. The evidence in the denominator is a normalizer of this function, which is, depending on the application, more or less important. The calculation of the evidence or the prior is not trivial and can have high computational costs. For a continuous function, it has to be integrated over all values of A.

$$p(B) = \int p(B|A)\; p(A)\; dA. \tag{2.49}$$

If there is a finite number of events A_i with $i = 1, ..., n$, the prior can be formulated by the sum of all events:

$$p(B) = \sum_{i=1}^{n} p(B|A_i)\; p(A_i). \tag{2.50}$$

Proofs of the Bayes rule that are partly based on the law of total probability are well-known. The expensive calculation of the evidence is unsuitable for practical applications. Fortunately, in the case of surface estimation, the set of events can be restricted to the binary case.

For events that relate to a set of other events, instead of a single event, the Bayes Rule can be formulated as:

$$p(A|B_1, ..., B_i, ..., B_n) = \frac{p(B_i|A, B_1, ..., B_n)\; p(A, B_1, ..., B_n)}{p(B_1, ..., B_i, ..., B_n)} \; , \tag{2.51}$$

which directly follows from the definition of the Bayes rule.

In practice, the posterior can describe a specific state z estimated from measurements $\mathcal{D} = d_1, ..., d_t$. This state can be time dependent considering a recursive formulation and might be conditioned on all past states and all measurements $p(z_t|z_{0:t-1}, d_{0:t})$. Because conditional independence is assumed for the measurements, it follows:

$$p(z_t|z_{0:t-1}, d_{0:t}) = p(z_t|z_{t-1}, d_t) \ . \tag{2.52}$$

This recursive formulation is an important assumption for real-time applications or for the processing of large amount of data that cannot be contained completely in memory. Yet, the restriction of conditional independence could be a limiting factor. Applications focused on in this thesis are discussed in Section 4.3. The time dependent formulation of state probabilities is known as the Bayes Filter.

Using the Bayes rule, the Bayes Filter can be re-formulated as follows:

$$p(z_t|z_{t-1}, d_t) = \frac{p(z_{t-1}|z_t, d_t)p(z_t|z_{t-1})}{p(z_{t-1}|d_t)} \ . \tag{2.53}$$

Eq. 2.53 remains expensive to solve, especially considering runtime performance.

Fortunately, the volumetric reconstruction of 3D surfaces can be formulated as an estimation considering binary states. A certain area of space can be classified either as in front or behind a surface. Two neighboring areas that are complementary in the binary state define a surface. Hence, the problem can be addressed by the Binary Bayes Filter.

The Binary Bayes Filter is based on the theory of the Bayes Filter; however, according to the binary assumption, the filter considers only two states, z and $\neg z$. The probabilistic ratio of these two states can be written as:

$$\frac{p(z)}{p(\neg z)} = \frac{p(z)}{1 - p(z)} \tag{2.54}$$

For further derivation of the theory, the definition of a new expression, the log odds l, is required (Thrun et al. 2005):

$$l(z) := \log \frac{p(z)}{1 - p(z)} \ . \tag{2.55}$$

Returning to the problem of estimating the probability of the state given a set of measurements, one obtains:

$$p(z|d_{1:t}) = \frac{p(d_t|z, d_{1:t-1})p(z|d_{1:t-1})}{p(d_t|d_{1:t-1})} = \frac{p(d_t|z)p(z|d_{1:t-1})}{p(d_t|d_{1:t-1})} \ . \tag{2.56}$$

By applying the Bayes rule for $p(d_t|z)$ it follows:

$$p(z|d_{1:t}) = \frac{p(z|d_t)p(d_t)p(z|d_{1:t-1})}{p(z)p(d_t|d_{1:t-1})} \tag{2.57}$$

Because there are only two states, the complementary state can be written as:

$$p(\neg z|d_{1:t}) = \frac{p(\neg z|d_t)p(d_t)p(\neg z|d_{1:t-1})}{p(\neg z)p(d_t|d_{1:t-1})} \tag{2.58}$$

The division of Eq. (2.56) and Eq. (2.57) leads to:

$$\begin{aligned} \frac{p(z|d_{1:t})}{p(\neg z|d_{1:t})} &= \frac{p(z|d_t)}{p(\neg z|d_t)} \frac{p(z|d_{1:t-1})}{p(\neg z|d_{1:t-1})} \frac{p(\neg z)}{p(z)} \\ &= \frac{p(z|d_t)}{1 - p(z|d_t)} \frac{p(z|d_{1:t-1})}{1 - p(z|d_{1:t-1})} \frac{1 - p(z)}{p(z)} \end{aligned} \tag{2.59}$$

Considering Eqs. (2.59) and (2.55), Eq. 2.60 can be written as log odd in a recursive formulation:

$$\begin{aligned} l_t(z) &= \log(\frac{p(z|d_t)}{1 - p(z|d_t)}) + \log(\frac{p(z|d_{1:t-1})}{1 - p(z|d_{1:t-1})}) + \log(\frac{1 - p(z)}{p(z)}) \\ &= l_t(z) + l_{t-1}(z) + l_0(z) \ . \end{aligned} \tag{2.60}$$

The last summand describes the prior of the state and can be set to 0 if the binary state is a uniformly distributed $p(z) = p(\neg z) = 0.5$. With this assumption, the recursive update process can be defined by only one sum:

$$l_t(z) + = \log(\frac{p(z|d_t)}{1 - p(z|d_t)}) \ . \tag{2.61}$$

For the reverse estimation of the probability from the logit state, Eq. (2.55) can be reversed as:

$$p(z) = 1 - \frac{1}{1 + e^{l_t}} \ . \tag{2.62}$$

For a more detailed background of the BBF, see the textbook by Thrun et al. (2005).

Chapter 3.

State of Research

There are numerous methods for MVS reconstruction that differ in their algorithm and in their requirements. Some methods work best, or even only, on specific datasets that consider specific attributes. There are only a few methods that are devised for cluttered outdoor scenes, especially for high-resolution images or a large number of images. A detailed overview of recent 3D reconstruction methods from images that capture cluttered outdoor scenes is provided by Vu et al. (2012).

Assumptions such as having only Lambertian surfaces, or additional information such as object silhouettes or even semantic classification, can improve the reliability and accuracy of the results. This thesis provides methods for general surface reconstruction under complex configurations without limitation in scalability or a need of further information because it is not always available.

In this chapter, an overview of the state of research for 3D reconstruction from image sets is provided. First, in Section 3.1 a brief introduction to methods that generate point clouds motivates further fusion to 3D models. Because this thesis is concerned with the processing of image-based point clouds, the difference in quality and density among stereo methods is discussed.

For surface reconstruction of large models, the state-of-the-art technology is 2.5D modeling that is suitable especially for specific image configurations. A relationship between 2.5D and 3D modeling is provided and discussed with respect to scalability in Section 3.2.

In Section 3.3, important work that is concerned with unconstrained 3D modeling is discussed differentiating local and global methods that are based on local and global optimization. This can be crucial for the quality and adaptability to large sets of images. Finally, an overview on those 3D modeling methods that have potential for processing of large scenes by considering scalability is provided.

3.1. Generation of Point Clouds

There are two classes of technologies for the generation of 3D point clouds in particular: active measurement, e.g., by laser-based distance measurements, and passive methods based on images. Laser-based sensors achieve stable and high accuracy quality, but they are expensive and the data acquisition is usually complex. Furthermore, the density of the measurements can be inhomogeneous, depending on the configuration. Passive stereo estimation is limited because it requires lighted textured areas. Furthermore, image-based reconstruction has a complex error behavior depending on the captured objects. However, images are highly available, costs are low, and the density can be controlled reasonably. This thesis focuses on image-based point clouds. Its goal is a higher quality concerning accuracy and completeness.

There are two main quality criteria for MVS: accuracy and completeness of the disparity maps. The quality of the resulting point cloud can strongly differ depending on the stereo method used. When known, the uncertainties in the 3D point cloud can be considered during the step of surface reconstruction by fusion of disparity maps.

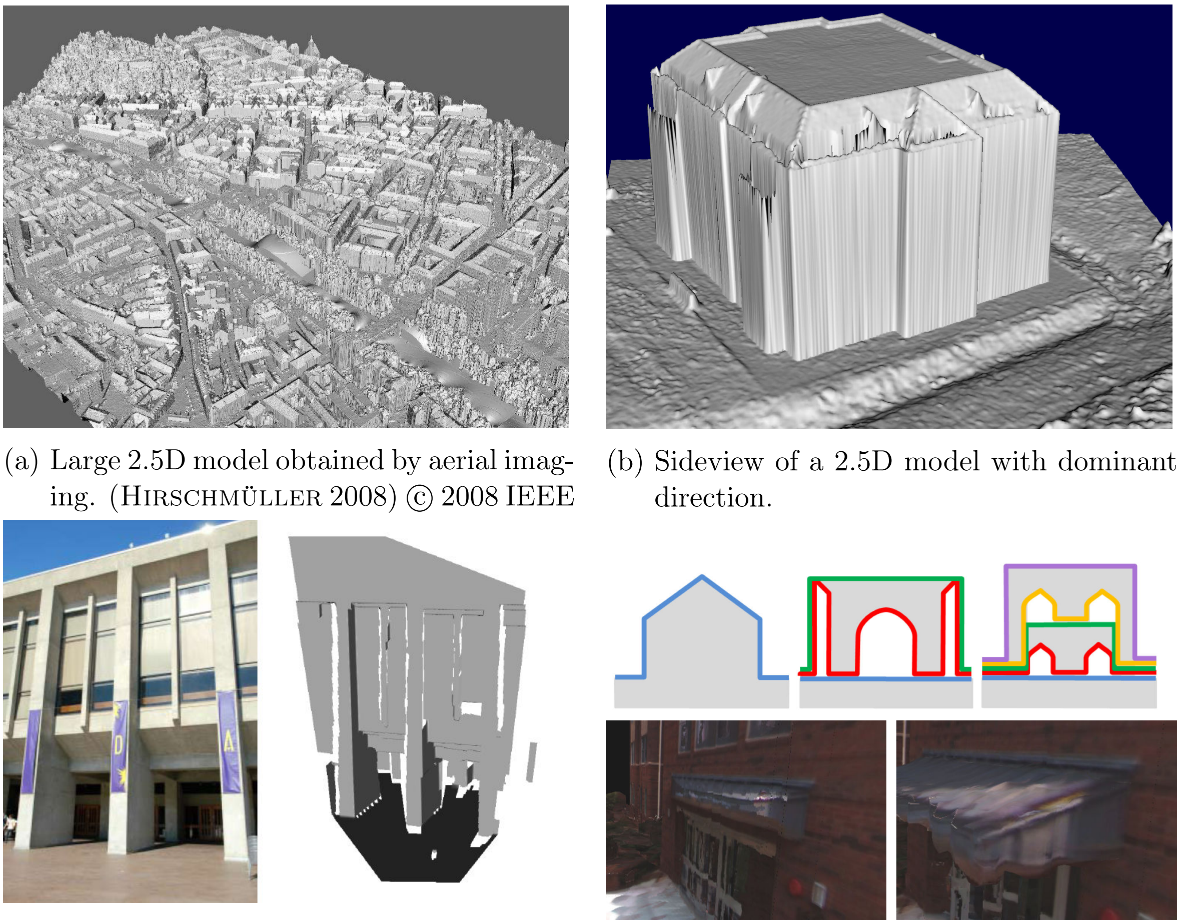
A further important criterion of MVS is scalability. The requirements in scalability for stereo estimation are not as strong as for 3D surface reconstruction. Disparity estimation is independent for all images because only two images have to be considered during the stereo process. Furthermore, image size is limited for specific cameras, whereas the number of images can rise arbitrarily. In practice, disparity maps can be processed in parallel using a multiplicity of CPU cores. The memory configuration and runtime performance can be adapted to specific MVS methods. Nonetheless, memory requirements can rise polynomially or even exponentially with the size of the image. In addition to memory requirements, runtime performance can explode relative to image size. Processing extremely large images such as those from aerial imaging is not necessarily feasible, particularly on small systems.

Local methods usually do not have such problems because they only consider a small neighborhood. Hence, the entire image is not required to be in memory at once. Yet, local stereo methods are not suitable for general configurations. Lack of texture or repeated textures causes missing or wrongly estimated disparity regions that global methods can manage more effectively. An overview of local and global stereo methods is provided in Section 2.2.2. SGM combines local and global methods, and thus presents a trade-off in scalability and accuracy; moreover, because it is employed as the stereo method of this thesis, it is discussed in more detail in Section 5.1.

In summary, there exist stereo methods that combine accuracy and scalability. Nonetheless, such methods obtain a varying quality of the point cloud, unlike point clouds from Light Detection And Ranging (LIDAR) that are characterized by more or less constant noise. The range of error of disparity maps has to be considered in the fusion process.

3.2. 2.5D Modeling

Digital surface models (DSMs) (cf. Fig .3.1a) have been the standard 2.5D representation for decades. A DSM represents a 2D map with height information as attribute. DSMs have the basic limitation of reconstruction in only one dominant direction (cf. Fig. 3.1b). They are especially suitable for fusion of disparity maps from satellite or aerial imaging. For those images, the viewing direction is oriented to the ground. Because for all 2D points/cells only one value for 3D information (height) exists the representation cannot model overhanging structures or multiple objects on top of another. In the case of urban modeling, structures such as bridges, balconies, canopies, or trees cannot be represented in 2.5D models. Nevertheless, DSMs have advantages in that no complex topology has to be considered, thus making the reconstruction considerably easier. Furthermore, interpolation in areas with no information allows for the modeling of watertight surfaces with minimum complexity.



(a) Large 2.5D model obtained by aerial imaging. (Hirschmüller 2008) © 2008 IEEE

(b) Sideview of a 2.5D model with dominant direction.

(c) Model considering the Manhattan world assumption. (Furukawa et al. 2009) © 2009 IEEE

(d) N-Layer heightmaps considering multiple 2.5D planes. (Gallup et al. 2010b)

Figure 3.1.: 2.5D and N-layer models showing urban regions with varying dominant directions. The modeling of all is limited to dominant directions.

By means of 2.5D models, configurations with images captured from the ground can be managed also by considering alternative dominant directions. The method through which specific dominant directions can be obtained in images was shown in studies by Coughlan and Yuille (1999), among others. Coughlan and Yuille (1999) proposed a constraint called the Manhattan World assumption that considers scenes that consist of piece-wise planar surfaces with dominant directions. The Manhattan World assumption is suitable for areas such as urban regions because symmetric buildings and streets can define a 3D space. Furukawa et al. (2009) used the Manhattan World assumption for stereo estimation to obtain accurate surfaces for urban and indoor modeling (cf. Fig. 3.1c). Gallup et al. (2010a) proposed to differ between planar and non-planar regions by image segmentation. Nevertheless, these methods are not suitable for unconstrained high quality 3D reconstruction because planar regions are modeled without maintaining small details.

Expanding 2.5D modeling, that only can model one height per cell, N-Layer heightmaps that use n instead of only one 2.5D model were proposed by Gallup et al. (2010b) (cf. Fig. 3.1d). The constraint of modeling in only one dominant direction remains the same. In urban modeling, taking the vertical direction is suitable. For urban ground images, the Manhattan World assumption is suitable. Complex configurations that combine images from the ground, from UAVs, and from aerial images cannot be modeled well based only on one dominant direction. Such configurations require a novel 3D surface reconstruction that considers complex topology.

3.3. 3D Surface Reconstruction

Methods for surface reconstruction without directional constraints are of main interest in this thesis. The algorithms concerned with unconstrained 3D surface reconstruction are often posed as variational problems that minimize an error function. In general, 3D reconstruction methods can be differentiated into two classes: either a local cost function that considers a limited part of the data or a global cost function over all the data can be minimized. In this section, the two classes are discussed in detail. The reason for the distinction between local and global methods is differences in scalability. In particular, scalable 3D modeling is discussed at the end of this section because it is the focus of this thesis.

3.3.1. Local Methods

Methods based on local optimization of the surface are of big interest for scalable 3D reconstruction. The main motivation for local methods is the ability for parallel processing. 3D points are optimized and connected considering only a limited neighborhood. This restriction allows for independent processing of all points using a multi-core system, e.g., a Graphics Processing Unit (GPU). Parallel processing is useful for online processing on real-time systems. Furthermore, large datasets can be processed on multi-core systems in reasonable time. A disadvantage of local methods is that an adjustment of the global uncertainty of 3D point clouds is not feasible. Point clouds derived from disparity maps can have strongly differing quality depending on the camera configuration or attributes of the scene. Obtaining better knowledge regarding the uncertainty of disparity maps and using it to improve the quality of 3D points is part of this thesis.

There are several methods based on local optimization of point clouds from disparity maps (Alexa et al. 2003, Ohtake et al. 2003, Furukawa and Ponce 2010). Furukawa and Ponce (2010) and Ohtake et al. (2003) used global optimization in postprocessing after initial local optimization; thus, the methods are discussed in Section 3.3.2. Moving Least Squares (Alexa et al. 2003) locally fits polynomial functions to the point cloud. To this end, a distance-weighted tangent plane is estimated for a point p within a local neighborhood $\mathcal{N}$ by least-squares fitting. The used polynomial regression considers the parameterized tangent plane. Least squares optimization generally makes use of the L_2 norm that is not robust to outliers. Hence, noisy and defective disparity maps can generally not be fused without preprocessing. Furthermore, methods that work on point clouds are limited because, depending on the configuration, the redundancy in the point cloud can be high, leading to higher densities and high memory requirement. Redundancy can be processed efficiently through the use of volumetric methods.

Local volumetric methods for MVS reconstruction are based on a discretization of the 3D space to a set of neighboring spatial volume elements such as voxels. The most promising volumetric methods are based on the fusion of cumulative distance functions (cf. Section 2.3.1) in the volumetric space (cf. Fig. 3.2).

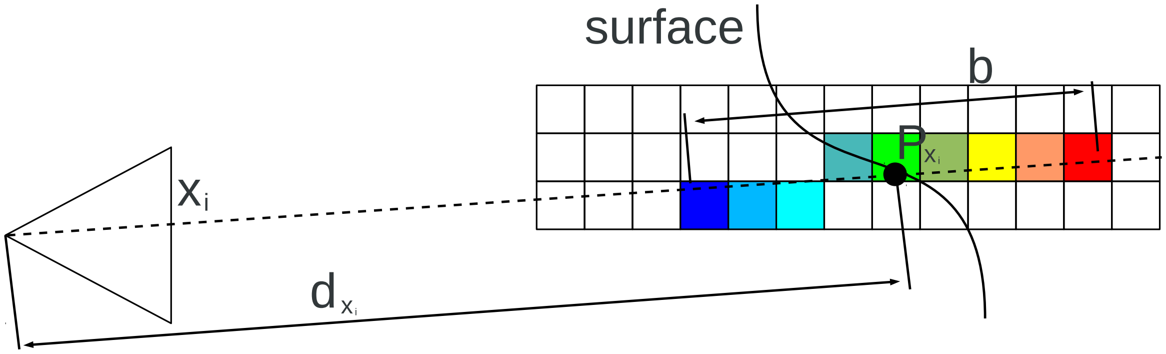


Figure 3.2.: Volumetric description of the position of a 3D point P based on the pixel coordinate x_i at distance d_{x_i}. The discretized elements along the line of sight in a specific area b are assigned a value that follows a distance function.

Hilton et al. (1996) first proposed the fusion of linear signed distance functions for implicit surface representation. The idea is to fuse functions that result from depth values with negative values in front and positive values behind the surface measurements. Combining several functions from the same direction, the optimized position can be derived as the zero-set of the resulting discrete function. The zero-set is defined by the set of two neighboring elements that have positive and negative values. Curless and Levoy

(1996) extended to volumetric fusion for a set of range images from laser data. They added the direction of sensor uncertainty, an incremental update scheme, space efficiency and postprocessing for hole filling. In addition, Curless and Levoy (1996) introduced a weighting function that allows for the penalization of 3D points with lower quality, e.g., depending on the slant of the surface or missing measurements. The distance and weighting function are shown in Fig. 3.3. The zero level set of volumetric space can be transformed to a set of polygons using the Marching Cubes method (Lorensen and Cline 1987). Marching cubes extracts a polygonal mesh of triangles from the iso-surface of numerical voxels (cf. Section 2.2.3). Goesele et al. (2006) showed the adaptability of volumetric fusion for disparity maps instead of range images from laser data.

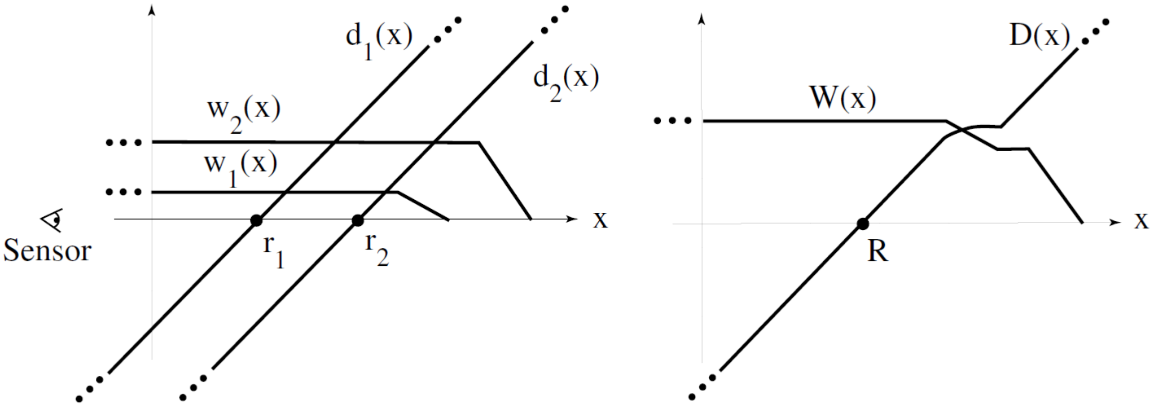
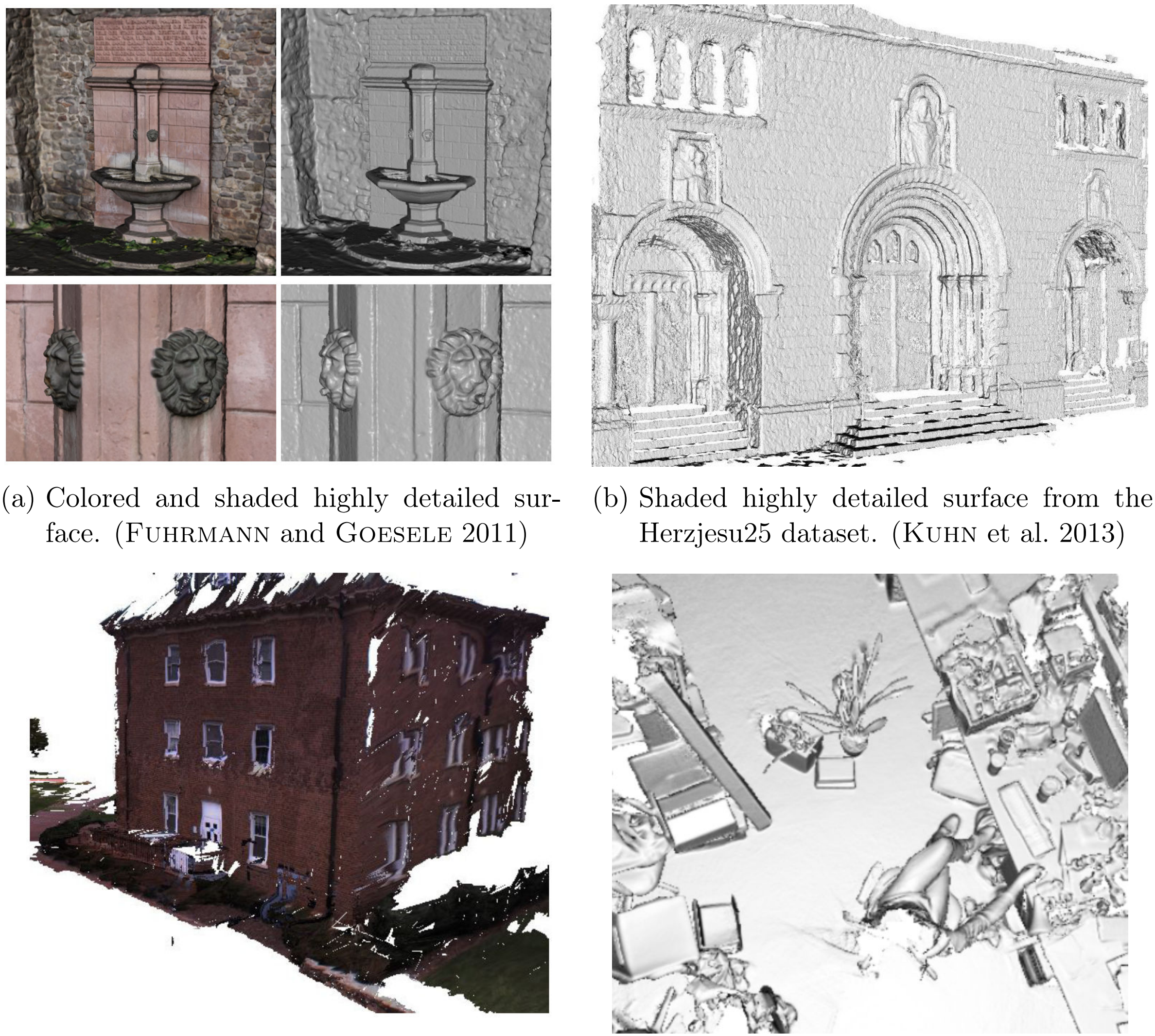


Figure 3.3.: Signed distance and weight functions. The distance functions d have negative values in front and positive values behind the surface measurements r. The weighting functions w penalize values behind the surface measurements. The right image shows the weighted combination of the two functions presented in the left image. (Curless and Levoy 1996)

Volumetric methods for 3D surface reconstruction can have a high computational cost and large memory requirements, especially for large scenes, because the number of elements rises with the third power. Furthermore, the constant size of volume elements precludes the efficient processing of points with varying quality. Hilton and Illingworth (1997) proposed the use of efficient data structures for reducing memory consumption and showed that volumetric methods are suitable for large scenes by using octrees at different levels. In particular, the octree level of occupied voxels is adapted to the surface curvature that bounds the approximation error. Fuhrmann and Goesele (2011) expanded the use of octrees by considering the geometric uncertainty of 3D points enabling the scalability of the method towards large scenes (cf. Fig. 3.4a). Finally, Kuhn et al. (2013) showed that local volumetric methods are not limited concerning scalability when dividing the reconstruction space in independent subsets (cf. Fig. 3.4b). This work is also part of this thesis. Furthermore, in order to process online the fusion of depth data such as RGB-D images, volumetric methods were shown to be suitable (Newcombe et al. 2011, Sturm et al. 2013) (cf. Fig. 3.4d). The method by Curless and Levoy (1996) was demonstrated to be able to fuse Kinect data even in real-time (Steinbrücker et al. 2013).



(a) Colored and shaded highly detailed surface. (Fuhrmann and Goesele 2011)

(b) Shaded highly detailed surface from the Herzjesu25 dataset. (Kuhn et al. 2013)

(c) Colored outdoor surface model from a building captured by a camera from the street. (Merrell et al. 2007) © 2007 IEEE

(d) Shaded surface from indoor reconstruction using the active Kinect sensor. (Newcombe et al. 2011) © 2011 IEEE

Figure 3.4.: Colored and shaded results for 3D surface reconstruction from local state-of-the-art methods.

Local fusion can also be done by comparing the quality of triangles that were directly derived from depth maps (Turk and Levoy 1994, Pito 1996). Turk and Levoy

(1994) proposed the first method that extracts polygon meshes from the range image. After an alignment over all images, the triangle meshes are fused in a local area. First, redundant triangles are removed from an overlapping region. Second, the remaining triangles are mutually clipped, resulting in new triangles that are partly removed considering topological constraints. Pito (1996) proposed an advanced classification of the polygon quality considering the geometric configuration of the sensors. In summary, these methods are fast and intuitive, but cannot manage noisy data and disparity maps in general configurations.

Direct optimization on the depth map (Merrell et al. 2007, Bailer et al. 2012) is another means for local fusion of disparity maps (cf. Fig. 3.4c). Individually filtering and optimizing single disparities over all disparity maps is not local. However, local regions of other disparity maps can be considered iteratively. Hence, only two disparity maps, or their regions, are compared at one time. It has to be mentioned that the complete 3D modeling process proposed by Bailer et al. (2012) contains global estimation steps. Furthermore, disparity map filtering does not solve the problem of unlimited scalability because the possible reduction is limited.

3.3.2. Global Methods

Surface reconstruction by global methods guarantees a global optimum over all input data. Global methods are well established; therefore numerous approaches exist. Optimization can be performed using variational methods (Lhuillier and Quan 2005, Pons et al. 2007, Cremers and Kolev 2011), volumetric methods (Ohtake et al. 2003, Kazhdan et al. 2006, Zach et al. 2007), and graph-based methods (Kolmogorov and Zabih 2002, Boykov and Kolmogorov 2004, Vogiatzis et al. 2005, Hornung and Kobbelt 2006, Vogiatzis et al. 2007, Mücke et al. 2011). In general, global methods obtain high quality, but are limited concerning memory and runtime performance.

Variational methods are based on the minimization of a global error function. For specific functions, such as the Euler–Lagrange equation (Lhuillier and Quan 2005) or the Poisson equation (Kazhdan et al. 2006), mathematical solutions are well studied. Unfortunately, there is no practical solution for directly solving those equations. Simplifications or iterative optimization strategies exist that avoid extremely high processing costs even for small datasets. Unfortunately, simplifications cause variational methods to fall into the local minimum of cost functions. Based on an accurate initial guess of the solution, globally optimal results can be obtained. Nevertheless, it can be expensive to obtain a good initial guess. A convex formulation of the error function can be feasible for specific formulations of the cost function, and has become important over the last decade.

Convex optimization expands classic variational optimization by more efficient and stable strategies for finding global maxima. Methods of convex optimization have become suitable for surface reconstruction and powerful solutions are proposed (Zach et al. 2007) (cf. Fig. 3.6c). Nonetheless, 3D surface reconstruction from images remains a non-convex problem, and convex formulation is either simplistic or additional prior information is necessary. Methods were proposed that consider the visual hull of objects (LAURENTINI 1993) for an initial solution (CREMERS and KOLEV 2011). For real world data the use of a visual hull is not yet possible because an automatic segmentation is not possible in a stable way.

Similar to local volumetric methods (cf. Section 3.3.1), global **volumetric methods** are based on the decomposition of the reconstruction space. The idea is to label the set of volumetric elements as either not occupied or occupied, and hence, define a surface. This can be mathematically described, for instance, by solving a Poisson equation (KAZHDAN et al. 2006). ZACH et al. (2007) used an initial solution from a volumetric evaluation (CURLESS and LEVOY 1996). In their method, global optimal solution of a surface is obtained by convex optimization considering a regularization term. Through TV of the 3D surface (cf. Section 2.1.2), the latter has been shown to improve the 3D surface quality concerning accuracy and completeness. The formulation of a regularization term for local processing is arranged differently and is discussed in Section 6.3.

Graph Cut-based surface reconstruction was formulated by KOLMOGOROV and ZABIH (2002). It is based on a max-flow/min-cut solution on a graph modeling spatial information (cf. Fig. 3.5). Point clouds from disparity maps can be transformed to a 3D grid in which the occupied cells contain one or multiple 3D points (HORNUNG and

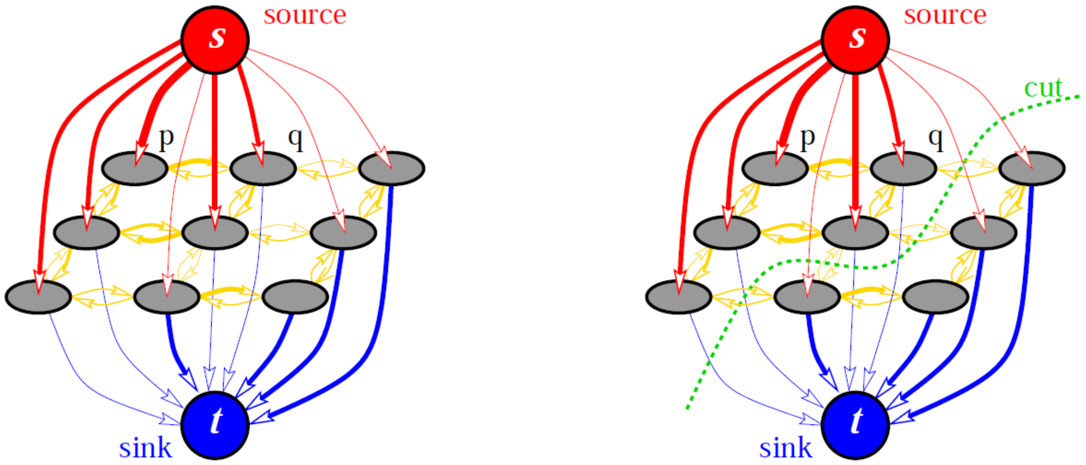
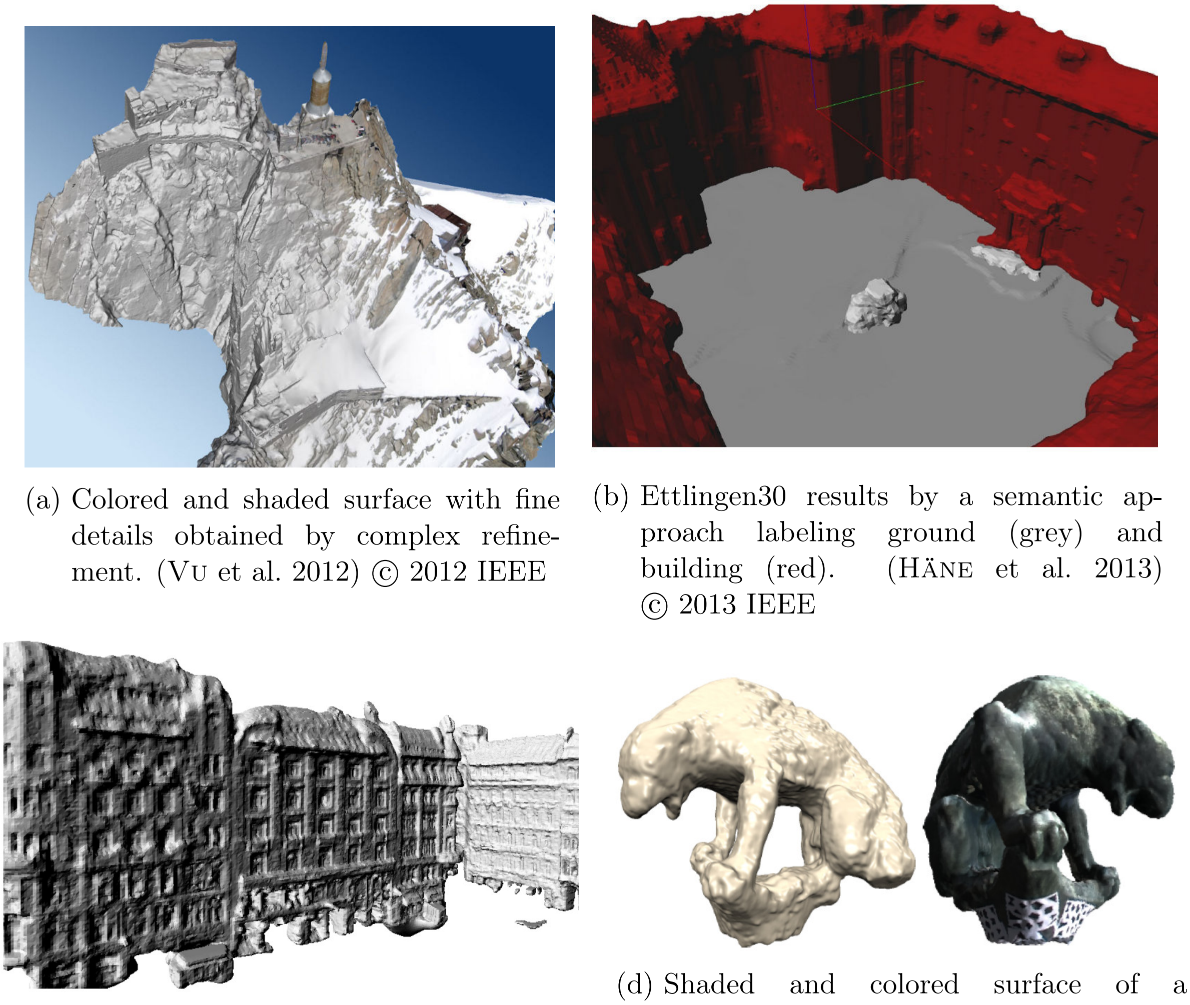


Figure 3.5.: A graph for global surface reconstruction. The edge costs are visualized as arrow thickness. A surface is reconstructed by cutting the graph in front/back (source/sink) using a max-flow/min-cut optimization. The right image shows the cut on the graph that separates the front and back of a surface, resulting in an optimal surface. (BOYKOV and KOLMOGOROV 2004) © 2004 IEEE

Kobbelt 2006) (cf. Fig. 3.6d). Vogiatzis et al. (2007) proposed to connect the cells of the corresponding points that have a high photo-consistency (Vogiatzis et al. 2005, Vogiatzis et al. 2007). The minimum cut of the graph consists of a set of connections that define a manifold surface that separate the inside and outside of an object. Mücke et al. (2011) expanded the method by considering different surface levels that allow for the reconstruction of surfaces with varying quality.



(a) Colored and shaded surface with fine details obtained by complex refinement. (Vu et al. 2012) © 2012 IEEE

(b) Ettlingen30 results by a semantic approach labeling ground (grey) and building (red). (Häne et al. 2013) © 2013 IEEE

(c) Terrestrial city modeling by a convex optimization approach. (Zach et al. 2007) © 2007 IEEE

(d) Shaded and colored surface of a statue by a graph-cut based approach. (Hornung and Kobbelt 2006) © 2006 IEEE

Figure 3.6.: 3D models generated by global surface reconstruction methods showing real world objects.

Further methods extend global optimization through semantic information. Bao et al. (2013) devised a method that learns priors for semantic categories that control the regularization of the object shape. To this end, geometric information from the SfM process is also considered. The manner in which learning priors for real world data can improve model quality was recently demonstrated by Häne et al. (2013) (cf. Fig. 3.6b). They use a joint image segmentation and semantic classification for a specific regularization of classes, such as vegetation and ground. In this thesis, a novel prior for local 3D reconstruction is presented based only the disparity map information because reliable semantic information can be difficult to obtain.

Tyleček and Sara (2010) presented 3D models reconstructed from real world data (Tyleček and Sara 2009). Their method can manage inaccurate camera calibration by adjusting camera parameters. To this end, the camera parameters were adjusted in a global SFM problem. Surface reconstruction and optimization is performed by a variational method that minimizes a global error function, including photo-consistency. The minimization is performed with a gradient-based approach that obtains high accuracy, but demonstrated limited completeness.

Arguably, the best results for surface reconstruction from real world images are currently obtained by Vu et al. (2012) (cf. Fig. 3.6a). They defined a complete reconstruction pipeline that generates a sparse or semi-dense point cloud that is transformed into a set of tetrahedron by means of Delaunay triangulation. The tetrahedra are examined concerning global visibility consistency. Neighboring tetrahedra labeled as inside or outside lead to triangles. The initial model is refined by photo-consistency optimization. The variational optimization is performed based on a global image-based matching score (Pons et al. 2007). Nevertheless, this approach follows complex strategies that are difficult to implement and are limited in scalability concerning runtime performance.

3.3.3. Scalable 3D Modeling

Because this thesis is concerned with scalable 3D reconstruction, those methods that have potential for surface reconstruction from large real world datasets have to be discussed in particular. Published scalable methods for 3D surface reconstruction follow different ideas: camera clustering (Zaharescu et al. 2008, Jancosek et al. 2009, Mauro et al. 2013) and Divide and Conquer (Vu 2011) (cf. Fig. 3.7). To this end, volumetric fusion can be feasible (Fuhrmann and Goesele 2011, Kuhn et al. 2013).

By means of **camera clustering**, redundant information is removed during the preprocessing step. Clustering prevents all images from being considered simultaneously. Hornung et al. (2008) proposed an image selection scheme that relies on coverage and visibility cues. Critical regions are detected by an estimation of the local photo-consistency and are re-optimized by optionally adding more pictures. Methods that focus on Internet Community Photo Collection often use camera clustering because they have a significant amount of redundant data. Furukawa et al. (2010) built a graph from filtered cameras. The graph is subsequently clustered through a normalized-cut. Finally, scene coverage is considered by overlapping the clusters.

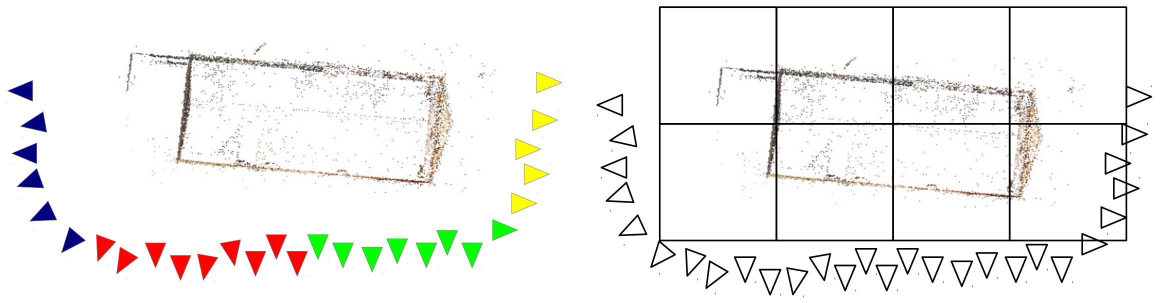


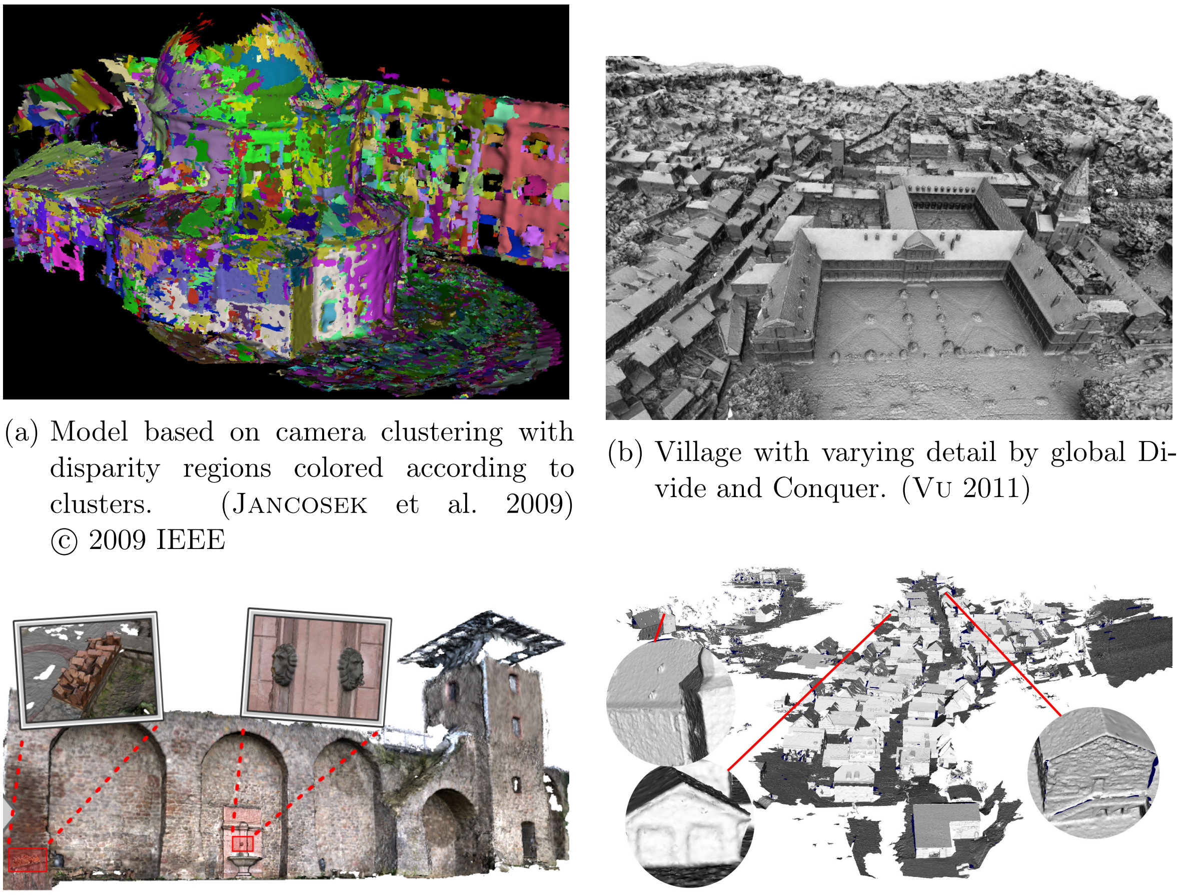
Figure 3.7.: Two general strategies for scalable 3D modeling. Left: Camera clustering for filtering unimportant data. Right: Divide and Conquer for dividing the space into smaller subspaces.

A graph-based method was also described by Pavan and Pelillo (2007). They searched for dominant sets that generalize maximum complete subgraphs to edge-weighted graphs. The assumption is that similarity among internal nodes is higher than between external and internal nodes. Dominant sets can be found through simple algorithms, such as replicator equations (Weibull 1997). Mauro et al. (2013) expanded the method for finding clusters of images using sparse point clouds from the image registration. In general, the camera clustering step can be performed using disparity maps or sparse 3D feature information. Camera clustering is useful for scalable 3D reconstruction because it reduces the amount of data by a particular factor. Mauro et al. (2013) described an increase of six because the amount of data was reduced approximately by this factor. Using high frequency rates for image capturing, e.g., as is performed with a video camera, the factor can be much higher, but it will always be limited for larger scenes.

Jancosek et al. (2009) proposed a method similar to camera clustering (cf. Fig. 3.8a). This method works as a filter on a limited number of images at a time. However, camera clustering does not solve the problem of scalability because image sets are not limited concerning their number.

Divide and Conquer methods operate on a subset of the reconstruction space. The 3D space can be divided into subspaces such as those shown in Fig. 3.7 (right). The individual parts can be reconstructed independently assuming that they are not correlated. Unlike local processing, global methods can produce varying surfaces when not considering all data. Hence, for scalable 3D modeling that uses global optimization, a complex merging postprocessing is required. Vu (2011) proposed a merging strategy for divided surfaces from global reconstruction based on Delaunay tetrahedralization and graph cuts (Vu et al. 2012) (cf. Fig. 3.8b).

Local optimization methods allow for parallel processing. Fuhrmann and Goesele (2011) showed that reconstruction can be made scalable using octrees (cf. Fig. 3.8c). Yet, their method employs a global tetrahedralization for surface reconstruction that restricts



(a) Model based on camera clustering with disparity regions colored according to clusters. (Jancosek et al. 2009) © 2009 IEEE

(b) Village with varying detail by global Divide and Conquer. (Vu 2011)

(c) Building by local fusion of disparity maps. (Fuhrmann and Goesele 2011)

(d) Village from aerial images by local Divide and Conquer. (Kuhn et al. 2013)

Figure 3.8.: Large 3D models obtained by scalable surface reconstruction.

the scalability. For Divide and Conquer methods, complex fusion can be avoided using local volumetric optimization (Kuhn et al. 2013) (cf. Fig. 3.8d). This is because the divided subvoxel of the reconstruction space can be defined with overlap of the neighboring subvoxels. It can be guaranteed that in neighboring spaces, the reconstructed surface is equal. This will be discussed in Section 5.2 because it is part of this thesis.

Furukawa and Ponce (2010) published one of the first methods with the potential to process large scenes (Furukawa and Ponce 2007, Furukawa and Ponce 2010). For this a semi-dense set of patches is generated. The patches describe a point cloud with normal vectors and density information, which is filtered and optimized based on photometric consistency. This method is global because no local surface reconstruction is proposed, but Poisson reconstruction (Kazhdan et al. 2006) is used.

Chapter 4.

Probabilistic Framework

In this chapter, a framework is described that is based on the volumetric fusion of spatial data. The fusion is described in general because the specific adaption to disparity map fusion is discussed in Chapter 5. Volumetric methods discretize the 3D space to a set of voxels (cf. Section 3.3). The algorithm and data structure for efficiently handling a set of volumes are discussed in Chapter 6.

After an introduction to general volumetric fusion in Section 4.1, the seminal method for volumetric fusion of spatial data, proposed by CURLESS and LEVOY (1996), is discussed and a novel probabilistic interpretation is introduced in Section 4.2. Finally, in Section 4.3 the probabilistic perspective is the basis to improve the method by means of a Bayesian fusion of Gaussian distributions.

The probabilistic reinterpretation of volumetric approaches opens new perspectives for 3D modeling. In addition to an optimal surface obtained from multiple measurements, a probability for their validity can be derived. In turn, this information can be used for filtering outliers and for surface quality assurance.

4.1. Volumetric Fusion of Spatial Data

Fusion of measurements in 3D space is a significant challenge, especially when considering a large amount of data. A standard way for local volumetric surface reconstruction is based on the fusion of cumulative distance functions. There are three main problems for this type of fusion considering spatial measurements: 1. Continuous representation of data, 2. Discretization of the continuous function, and 3. Fusion of multiple measurements.

The idea of implicit surface reconstruction combining signed distance functions was first mentioned by HOPPE et al. (1992). CURLESS and LEVOY (1996) expanded the method by introducing a sound theoretical framework for range images from laser measurements. The adaptability for considering disparity maps from stereo images was shown by GOESELE et al. (2006). KUHN et al. (2013) expanded the method by introducing a probabilistic framework allowing for further filtering. This expansion is part of this thesis.

In general, the position of depth measurements can be represented by distance functions and optional additional weight functions. For simplification, these functions are usually considered as univariate. An extension to trivariate distance functions is feasible, but computationally and algorithmically expensive. For the univariate function, a line through the 3D space has to be defined on which the function is propagated. To this end, the line of sight (Goesele et al. 2006) or the normal vector (Schroers et al. 2012), estimated from the depth map, is appropriate. The intersection of this line with the discretized volumetric space defines a set of voxels (cf. Fig. 4.1).

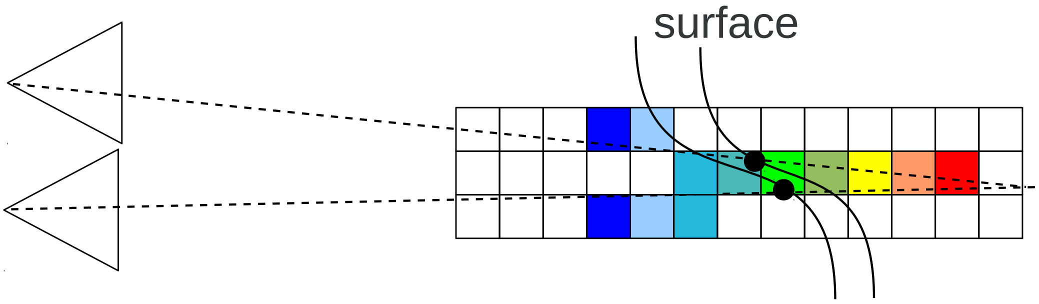


Figure 4.1.: Volumetric description of the position of a 3D point seen from two cameras. The surface is measured differently from different camera positions. The volumetric description allows for the fusion of multiple measurements. For a more detailed explanation that considers individual measurements, see Fig. 3.2.

Discretized values from the distance and weight function are assigned to intersected voxels. In general, a negative value is assigned to those voxels in front of a measured distance, whereas a positive value is assigned to those voxels behind. By this means, the processing of only one measurement at a time is possible, thus allowing limited memory resources. Considering multiple measurements, the values have to be fused. To allow for an incremental processing of the images, a recursive formulation of the fusion process is advantageous. Depending on the distribution used, this can be performed by probabilistic or analytic fusion.

By assigning negative values in front and positive values behind the surface, the zero crossings of neighboring negative and positive values define a possible surface. This set of neighboring voxels is called the zero level set. As shown in Fig. 4.1, the surface is measured differently from all images, because of measurement noise. Hence, values have to be fused to obtain a combined zero crossing in order to achieve an optimized surface according to the measurements.

4.2. Linear Fusion

The linear framework for fusion is based on the method presented by Curless and Levoy (1996) named Volumetric Range Image Processing (VRIP). It employs a continuous implicit function d on the line of sight considering the measured depth z. While VRIP was described for the fusion of laser data, Goesele et al. (2006) showed that the approach is also suitable for (multi-view) stereo data. The function used defines values with linear increase that are negative in the range of $]-\infty, z[$ and positive in the range of $[z, \infty[$. In addition, a second function w is proposed to weight the values of the signed distance function.

The choice of weight function depends on the sensor technique. Curless and Levoy (1996) proposed to decimate the weight of a measured depth value behind the measured distance. This is appropriate because the sensors used do not obtain information behind a surface. The distance and weight function are shown in Fig. 4.2.

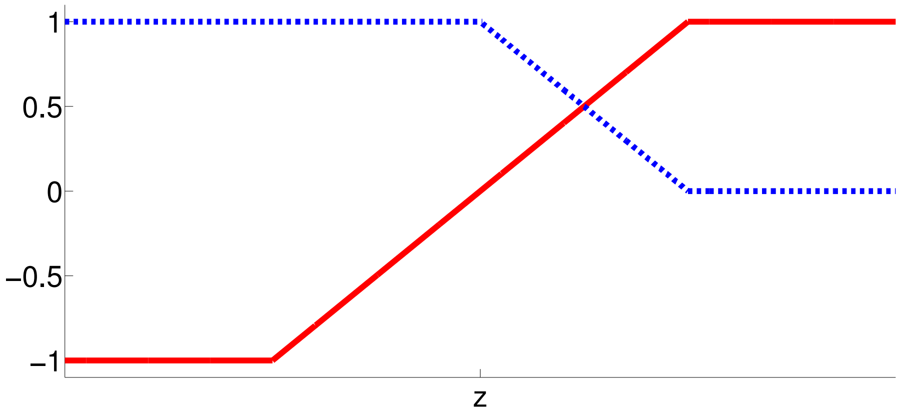


Figure 4.2.: Graphs of the linear cumulative distance function (solid-red) and the weight function (dashed-blue). The weight function penalizes values behind the estimated surface at depth z.

The update process has to employ a recursive scheme to render possible an incremental processing of the disparity maps (cf. Section 4.1). The volumetric update process for voxels on the line of sight with the linear cumulative functions follows two incremental equations (Curless and Levoy 1996):

$$W_{i+1}(v) = W_i(v) + w_{i+1}(v) \ , \tag{4.1}$$

$$D_{i+1}(v) = \frac{W_i(v) + D_i(v) + w_{i+1}(v)d_{i+1}(v)}{W_{i+1}(v)} \ . \tag{4.2}$$

The discretized values $d_i(v)$ and $w_i(v)$ are calculated for voxel v at time i from the distance and weight functions. To this end, the value of the functions at the position that corresponds to the distance between the relevant voxel and the measured depth is assigned. The time i is increased per measurement. The resulting values $D_i(v)$ and $W_i(v)$ characterize the current discrete representation of the fused functions at voxel v. From Eqs. (4.1) and (4.2), it follows that the values are limited in the range of $[-1, 1]$. Curless and Levoy (1996) propose to adapt the scaling of the weight function depending on the angle between the line of sight and the normal vector of the surface, as well as depending on the distance to the next missing measurements in the depth map.

Curless (1997) showed that under a certain set of assumptions, the proposed fusion is optimal in the least squares sense. The assumptions comprise a Gaussian distribution and independently distributed range images. The estimated surface is derived as optimal in terms of a Maximum Likelihood minimization.

In the following paragraphs, a new perspective of the linear fusion is obtained by considering a novel probabilistic point of view. The probabilistic interpretation is theoretically sound, but contradictory to the original formulation. In Chapter 2 the probabilistic theory of the uniform distribution is given. Fig. 2.7 (left) shows the probability density function p and the corresponding cumulative distribution function P. When comparing the cumulative distance function from Fig. 4.2, it is obvious that both functions are equal besides the range of P. It can be shown that the range of the linear fusion can also be defined in a way that maintains the range $[0, 1]$ and considers the 0.5 level set by a scaled sum:

$$d_i^* = \frac{1}{2}(d_i + 1) \ . \tag{4.3}$$

As proof, the non-incremental equations that correspond to Eqs. (4.1) and (4.2) can be obtained (Curless and Levoy 1996):

$$W(v) = \sum_i w_i(v) \ , \tag{4.4}$$

$$D(v) = \frac{\sum_i w_i(v) d_i(v)}{\sum_i w_i(v)} \ . \tag{4.5}$$

The function D can be propagated considering d^*:

$$\begin{aligned} D(v)^* &= \frac{\sum w_i(v) d_i^*(v)}{\sum w_i(v)} = \frac{\sum w_i(v) \frac{1}{2}(d_i(v) + 1)}{\sum w_i(v)} \\ &= \frac{1}{2} \frac{\sum w_i(v) d_i(v) + \sum w_i(v)}{\sum w_i(v)} \\ &\Rightarrow 2(D(v)^* - \frac{1}{2}) = D(v) \ . \end{aligned} \tag{4.6}$$

Hence, by transformation from Eq. (4.6), a level set of 0.5 instead of zero follows. The values are scaled by a factor of 2, which has no influence on the level set or on the corresponding estimated surface. The fusion scheme remains the same considering the interval $[0, 1]$.

With the motivation of reconstructing surfaces from level sets, the voxel assignment can be interpreted as modeling the probability that the voxel lies behind the surface. Thus, the 0.5 level set exactly defines the area where neighboring voxels separate into outer and inner regions. In this probabilistic sense, the linear function from VRIP corresponds to a uniform cumulative distribution function. According to the probabilistic density function, this implies a uniform distribution of the measurement.

The interpretation of the fusion process from Eq. (4.2) in a probabilistic way corresponds to a weighted sum over all surface probabilities. From the weighted sum, it follows that all probabilities are averaged, which is not a probabilistic fusion in the usual sense. Nonetheless, an averaging can be reasonable because measurement data are usually correlated.

It is important to discuss the weight function, penalizing values behind the estimated depth, in a probabilistic fusion. Decreasing the influence of probabilities behind the measurement is meaningful to reduce the influence on possible neighboring surfaces. Nevertheless, the values in front of the measured depth should be decreased also to consider outliers. Furthermore, favoring values measured in front of the surface can theoretically lead to a shift of the surface in the viewing direction. Specifically, these types of configurations could lead to a shrinking of the reconstructed objects.

As already mentioned, Curless (1997) showed that the linear fusion is optimal in the least squares sense when considering uncorrelated data and Gaussian uncertainty. This section has shown that following the novel probabilistic interpretation, data are fused under the assumption that depth values are uniformly distributed. The fusion process is maintained as averaging of the values, which is meaningful for highly correlated data. The probabilistic interpretation is contradictory to the original one; however, this does not imply that either interpretation is wrong because the perspective has changed from least squares to probabilistic optimization. Considering Gaussian uncertainty and uncorrelated measurements a reinterpretation from a probabilistic perspective is important and is discussed in the following section.

4.3. Bayesian Fusion of Gaussians

The preceding section has shown that the original volumetric approach by Curless and Levoy (1996) propagates uniform distributions with correlated fusion as seen from the novel probabilistic perspective. In the following paragraphs, a probabilistic reinterpretation is provided that considers uncorrelated fusion of Gaussians within the probabilistic perspective.

As for the linear approach, a value is allocated to those voxels that lie on the line of sight to the measured depth z. Hence, instead of a linear cumulative function that corresponds to a uniform distribution, a Gaussian distribution is considered. To this end, the uncertainty of the point that corresponds to the measured depth z on the line of sight is assumed to follow a Gaussian distribution with the measured depth as expected value and uncertainty σ:

$$p(z_x) = \mathcal{N}_{PDF}(z, \sigma) \ . \tag{4.7}$$

The uncertainty σ for MVS is discussed in Section 5.3.

Within the probabilistic view, a weight function that reduces the weight behind the measurement is not suitable. Therefore, a novel binary weight function is introduced. It can be seen as an indicator function that limits the area of influence. On one hand, this avoids the influence of outliers, and memory requirements are preserved because a smaller number of voxels is involved. On the other hand, limiting the area can lead to multiple surfaces that have to be considered for reconstructing a surface. It has to be mentioned that without limitation of the area also multiple surfaces appear, but in decimated number. An unambiguous level set is theoretically feasible by fusion of cumulative distance functions, but not guaranteed by MVS reconstruction, because disparity maps generally are incomplete.

For volumetric propagation, a probability $p(v_i^1)$ has to be assigned that models the voxel v_i to lie behind a surface. To this end, the probabilities of the Gaussian PDF have to be integrated, which can be done by using the Gaussian CDF. Unfortunately, there is no closed form of the Gaussian integral. Hence, the CDF has to be estimated numerically. The graphs of the probabilistic distance and weight function are shown in Fig. 4.3.

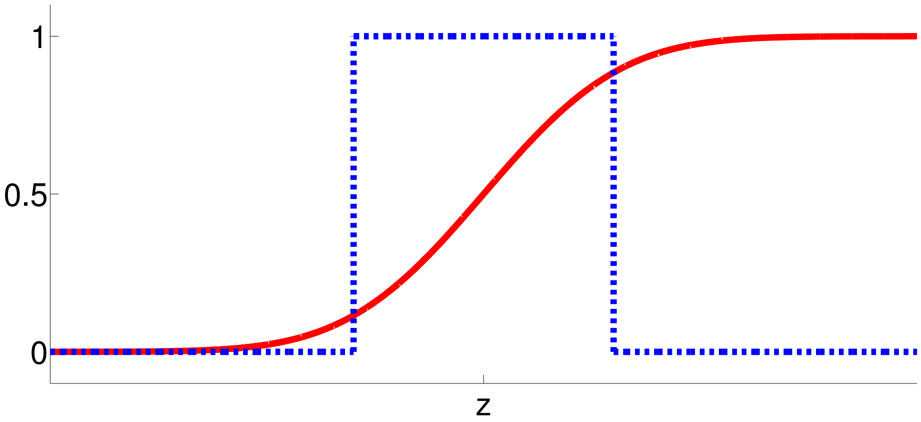


Figure 4.3.: Visualization of the Gaussian cumulative distance function (solid-red) of an estimated depth z. The weight function (dashed-blue) represents an indicator function limiting the area of influence.

In Section 2.3.1, it is shown that the Gaussian CDF can be numerically estimated considering the standard Gaussian with $\mu = 0$ and $\sigma = 1$. For the transformation, the Gaussian parameters have to be shifted and scaled. In practice, this can be performed with use of the Gaussian error function that is available, e.g., in Matlab or the C++ standard library:

$$\mathrm{erf}(x) = 1 - \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-\tau^2}\, d\tau. \tag{4.8}$$

A numerical lookup table is used to provide values for the Gaussian CDF with standard deviation of one. With the scaling parameters μ and σ, the infinite set of Gaussian CDFs can be written as:

$$\mathcal{N}_{CDF}(\mu,\sigma)(x) = \frac{1}{2}(1 + \text{erf}(\frac{x-\mu}{\sigma\sqrt{2}})) \ . \tag{4.9}$$

Hence, as in the linear framework, the probability distribution from one 3D point can be propagated directly and assigned to the voxels:

$$p(v_i^1)(d) = \mathcal{N}_{CDF}(z,\sigma)(d) \ , \tag{4.10}$$

where d is the distance between the measured depth and voxel v_i.

Probabilities obtained from different images or pixels that fall into the same voxel are averaged by the VRIP method. Opposed to this, a probabilistically sound solution would include a probabilistic fusion, especially for uncorrelated data. In addition, it is important to repeat that a recursive formulation of the update process is extremely suitable in scalable MVS reconstruction.

An incremental update process can be defined by employing conditional probabilities $p(v_t^1|v_{t-1}^1, \mathcal{D})$. To this end, the probability at time t for voxel v to lie behind the surface is derived based on the probability at time $t-1$ and all measurements $\mathcal{D}$ assigned to this voxel. The fusion of conditional probabilities can be performed using the Bayes rule. In particular for the binary case with only two states, Bayes fusion can be efficiently solved by a recursive formulation (cf. Section 2.3.4). Hence, the Binary Bayes Theory, which is well-known in the robotics community, e.g., employed by KONOLIGE (1997) for building a map of a robot's environment, is the suitable framework for volumetric fusion of surface probabilities. A voxel can be labeled as behind the surface or not behind the surface. This binary assumption makes the Binary Bayes fusion feasible.

The Bayes theorem that assumes independent measurements can be written as:

$$p(v^1|D = d) \propto p(v^1) \prod_{j \in 1,\ldots,n} p(D_j = d_j|v^1) \ . \tag{4.11}$$

As described in Section 2.3.4, the logit formulation follows:

$$l(v) = log\frac{p(v^1)}{p(v^0)} = log\frac{p(v^1)}{1-p(v^1)} = \sum_j log\frac{p(v_j^1)}{1-p(v_j^1)} \ . \tag{4.12}$$

The logit can be formulated recursively:

$$l_t(v) = l_{t-1}(v) + \log(\frac{p(v^1|d_t)}{1-p(v^1|d_t)}) \ , \tag{4.13}$$

and back-projected to surface probabilities after the fusion process:

$$p(v_t^1) = 1 - \frac{1}{1+e^{l_t(v)}} \ . \tag{4.14}$$

The novel probabilistic fusion extends the linear fusion concerning two basic assumptions: The depth uncertainty is assumed to be Gaussian instead of uniform and the probabilistic fusion assumes uncorrelated measurements. For multiple measurements, the fusion leads to a coherent surface from neighboring voxels (cf. Fig. 4.4).

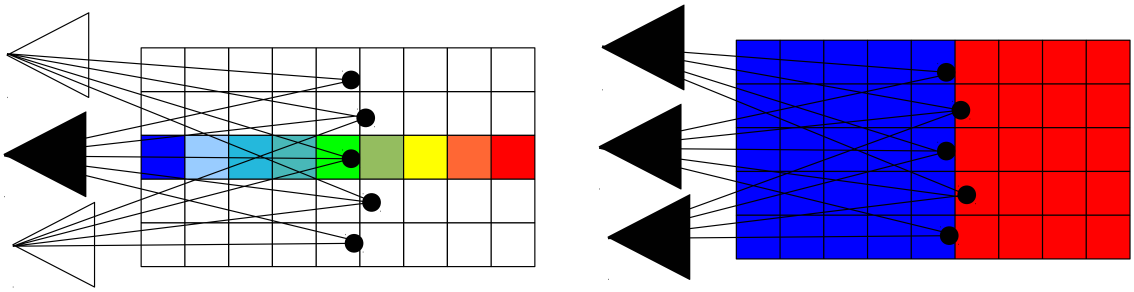


Figure 4.4.: Volumetric propagation of surface probabilities. The black points represent measurements from three cameras. In the left image, one point from one camera is propagated to the volumes. The right image shows the volumes after propagation of all points from all cameras defining a probable surface.

For surface generation from volumetric representation, Curless and Levoy (1996) propose the use of Marching Cubes (cf. Section 2.2.3). This is adequate when there are sufficient measurements and the zero level set is unambiguous. In particular, in complex image configurations multiple surfaces can appear because disparity maps can have holes. For this, a filtering step is necessary to consider surface probabilities.

In the probabilistic space, the surface is obtained from neighboring voxels for which the probability that one is in front of the surface and the other is behind the surface is high. In addition to polygonization, the probabilistic space can be used for the optimization of the surface based on information from the disparity maps. A volumetric representation of the optimized point cloud is advantageous because filtering on triangle meshes is not as efficient as filtering on volumes (cf. Section 5.4.2). The volumetric representation of triangle meshes is also feasible; yet, Marching Cubes only work on regular voxels, and thus, is not feasible for the multi-resolution setup presented in Chapter 6.

(Fuhrmann and Goesele 2011) proposed the transformation to a volumetric representation of the surface voxels, which can indeed be used for surface reconstruction. Their multi-resolution solution is based on global tetrahedralization. Yet, this does not comply with a method that guarantees unlimited scalability (cf. Section 5.2). Nonetheless, a transformation in a volumetric point space is proposed that is suitable for local triangulation either.

For the transformation in this thesis, the estimated 3D point from disparity is shifted along the line of sight, e.g., in a limited area in both directions. Subsequently, all voxels I in the area are examined and two neighbors are taken with a maximum product of probabilities that one is in front of the surface and the other is behind:

$$\arg\max_{i \in I}(p(v_i^0)\ p(v_{i+1}^1)) \ . \tag{4.15}$$

This optimization is advantageous as for complex configurations the 3D probabilistic space can consist of multiple level set surfaces due to the measurement noise.

To obtain the optimized position with subvoxel accuracy, a Gaussian is fitted to the neighboring voxels of v_i and v_{i+1}. For this Gaussian regression, a Maximum Likelihood estimation is employed considering four voxels:

$$d_n = \frac{1}{4} \sum_{j=i-1}^{i+2} d_j \frac{e^{l_j}}{1 + e^{l_j}} \ , \tag{4.16}$$

The result of an optimized 3D point can be supplemented by the surface probability that one is in front and the other is not:

$$p_v = p(v_i^0)\ p(v_{i+1}^1) \ . \tag{4.17}$$

This probability can be used again when determining the voxel to be filtered, thus leading to a more robust filtering.

Chapter 5.

Fusion and Filtering of Disparity Maps

In this chapter, the probabilistic framework presented in Chapter 4 is adapted to the specific case of fusing disparity maps. The quality of disparity maps depends on the stereo method used. In this thesis, SGM is used for MVS reconstruction because it allows for fast processing of large images while maintaining small details. Hence, a brief introduction to SGM is given in Section 5.1.

In addition to the processing of high-resolution images, this thesis focuses on image sets that can be extremely large concerning the number of images. To deal with this, a Divide and Conquer approach is introduced in Section 5.2 that allows for an unlimited 3D surface reconstruction.

In Section 5.3, the influence of uncertainties from image registration and stereo matching on the uncertainty of the 3D points is discussed. In particular, an error model for stereo estimation is extended to a variable disparity error that is statistically learned from ground-truth data.

Finally, the filtering and fusion of noisy 3D points is presented in Section 5.4. For this, the surface probabilities obtained by the probabilistic fusion are used.

5.1. Semi-Global Matching

In this thesis, SGM is used for the estimation of disparity maps from (multiple) image pairs. In Section 2.2.2, a comparison of stereo methods classified into local and global methods is provided. In summary, SGM combines into a semi-global method the advantages from local and global methods because it maintains small details due to pixelwise matching, and has low processing time and memory requirements even for high-resolution images. Because this thesis is concerned with scalable 3D modeling, including the processing of high-resolution images, SGM is an extremely suitable basis and is used for the experiments. For SGM, GPU (Ernst and Hirschmüller 2008) and Field Programmable Gate Array (FPGA) (Hirschmüller 2011) implementations exist that have real-time potential and low cost, even for large datasets.

Similar to most stereo methods, SGM consists of two steps: cost calculation and cost aggregation. In cost calculation, a discrete 3D space with x-, y- and d-dimensions that represent pixel coordinates and disparity is built. For all pixels in the reference image, the possible corresponding pixels on the epipolar line in the second image are compared using matching costs C, such as MI or census (cf. Section 2.1.1). Cost calculation generates a noisy 3D space that defines multiple pixelwise surfaces because wrong matches on the epipolar line can easily have a lower cost than correct ones (cf. Section 2.2.2). By considering prior information it is possible to obtain clean surfaces by means of cost aggregation.

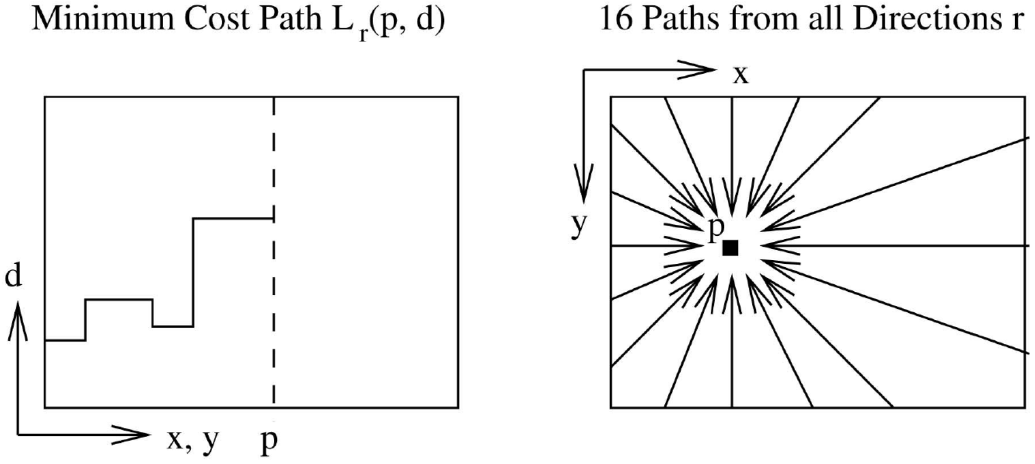
SGM considers a constraint that supports the smoothness of surfaces. Within this constraint, changes of neighboring disparities are penalized. This is defined by an energy function, which is minimized:

$$\begin{aligned} E(D) = \sum_{p} (C(p, D_p) + \sum_{q \in \mathcal{N}_p} P_1 \, T[|D_p - D_q| = 1] \\ + \sum_{q \in \mathcal{N}_p} P_2 \, T[|D_p - D_q| > 1]) \ , \end{aligned} \tag{5.1}$$

with the operator T, which results in one if the argument is true; otherwise, the result is zero. The energy function combines three cost terms that penalize different classes of errors. The first term is the sum of matching costs over all pixels p considering the disparity D_p. The second term counts all pixels in the neighborhood $\mathcal{N}_p$ for which the disparities have small differences of one pixel. The third term counts neighboring pixels that have larger differences in the disparities. The second and third term are scaled by the parameters P_1 and P_2. P_2 has to be larger than P_1 because large differences in the disparities are to be penalized. In the experiments of this thesis, the parameters were fixed to $P_1 = 28$ and $P_2 = 30$. Hirschmüller (2008) proposed to adapt P_2 depending on the intensity contrast of the neighboring pixels $P_2^* = \frac{P_2}{I_p - I_q}$ for neighboring pixels p and q. This is also employed in this thesis. In general, the cost terms represent a fronto-parallel bias that favors surfaces parallel to the image plane.

The global minimization of an error function similar to Eq. (5.1) is NP-hard for 2D images (Boykov et al. 2001). In contrast, the minimization along single 1D rows or columns of the image can be solved in polynomial time using Dynamic Programming (Meerbergen et al. 2002). The optimization in only one direction leads to the problem of having to fuse individual lines that were estimated independently. In SGM, this fusion is solved using multiple paths in different directions and aggregating the cost as a sum of the individual paths. Fig. 5.1 shows an example with 16 paths and the calculation of the minimum cost path. The paths through the disparity space correspond to straight lines in the base image, but in general to non-straight lines in the second image. The cost along a path in the direction r can be recursively formulated:

$$\begin{aligned} L_r(p, d) = C(p, d) + \min(L_r(p - r, d), \\ L_r(p - r, d - 1) + P_1, \\ L_r(p - r, d + 1) + P_1, \\ \min_i L_r(p - r, i) + P_2) - \min_k L_r(p - r, k) \ . \end{aligned} \tag{5.2}$$



Minimum Cost Path $L_r(p, d)$

16 Paths from all Directions r

Figure 5.1.: Aggregation of costs in disparity space. Left: The minimum cost path. Right: Example with 16 individual paths through the image. (Hirschmüller 2008) © 2008 IEEE

Hence, only the cost of the path is required, and not the path itself.

After cost aggregation, the maximum for all pixels can be calculated independently. For this, the pixelwise minimum cost in the disparity dimension is chosen. To obtain subpixel accuracy, the two neighboring disparity values are considered also for the regression of a parabola. The minimum of the quadratic function corresponds to the subpixel value for the estimated disparity.

Subsequently, a left right check is performed to filter outliers. To this end, for both images of the stereo pair, the disparity map is calculated and compared. If there is a disparity difference in corresponding pixels that exceed a threshold (e.g., of one pixel), the disparity is removed from the disparity map.

SGM processes only image pairs, instead of n-images. The algorithm can be extended to matching n-pairs by calculating a pixelwise matching cost that considers multiple images. However, the problem of occlusions in image configurations would have to be solved on the pixel level that was found to be extremely unstable. Hence, for MVS by SGM, it seems advantageous to fuse disparity maps considering n-pairs after a pairwise filtering of occluded areas by the left right check.

Let the disparity D_k result from matching the reference image and image k. The disparity images are scaled differently because they can have different baselines that influence the depth from disparity. In a registered image set, the differences of the relative baselines are known. Hence, the disparities can be re-scaled by a factor t that has linear correspondence with the baseline and possible differences in the camera constants. The disparities have to be normalized by $\frac{D_{kp}}{t_k}$. This normalized disparity is not per definition a disparity that defines the pixel distance

The fused disparity is calculated by a weighted mean of the n disparities. The estimation of a specific representative value, such as the median of all disparities, results in the filtering of outliers. The disparity D_p is calculated by the weighted mean considering the factor t:

$$D_p = \frac{\sum_{k \in V_p} D_{kp}}{\sum_{k \in V_p} t_k} \ . \tag{5.3}$$

Median based filtering of disparities that have a larger error than one pixel can be written as:

$$V_p = k \mid \left| \frac{D_{kp}}{t_k} - \mathrm{med}_i \frac{D_{ip}}{t_i} \right| \leq \frac{1}{t_k} \ . \tag{5.4}$$

The expansion of 2.5D filtering of disparity maps to more than the n images of the set would be advantageous. Particularly, the filtering of outliers could become more stable, especially when considering varying disparity qualities (cf. Section 5.3.3). Unfortunately, filtering requires the comparison of all disparity maps to all other disparity maps, leading to a quadratic processing time in terms of the number of images. This is unsuitable for the modeling of extremely large scenes, even with parallel processing on GPUs. Limitations of the image sets would be helpful, e.g., by camera clustering or Divide and Conquer methods; however, this is beyond the scope of this thesis.

5.2. Unlimited Scalability

SGM renders the processing of high-resolution images feasible. The key question for a scalable 3D surface reconstruction, which is the main goal of this thesis, is how to fuse a non-limited number of possibly high-resolution disparity maps. There are ways to reduce the amount of data for scalable 3D modeling by camera clustering (cf. Section 3.3.3). In general, those approaches do not solve the problem of unlimited scalability because they reduce data, but cannot manage extremely large reconstruction spaces.

A feasible way to guarantee unlimited scalability is to divide the reconstruction space in subareas that limit the size of data to be processed. In addition to limiting the amount of data, the division in subspaces allows for parallel computing of the subareas, assuming independence in the optimization of the subareas.

A Divide and Conquer strategy was already proposed by Vu (2011) with a result of impressive 3D surface models. In this method, the 3D space is divided in subareas that are reconstructed by means of global optimization and are fused afterwards. The fusion obtains triangle meshes that follow a complex fusion strategy employing graph cuts. The global optimization in the subareas leads to two further problems: ambiguous surfaces and restricted parallelization. Furthermore, the complex fusion strategy is expensive to compute, and the results can differ depending on the subdivision position.

In this thesis, the Divide and Conquer strategy is combined with local optimization of the subareas. By means of local optimization, complex fusion strategies can be avoided.

This is guaranteed by defining an overlap between neighboring subareas, with the size of the overlap being at least twice the size of the largest local optimization area of the surface parts. Complex fusion is avoided because the resulting point cloud in neighboring subareas is guaranteed to be equal inside the border region of half the overlap (cf. Fig. 5.2). This border can simply be cut off when fusing or visualizing the data, because in the critical region of the subarea the points are guaranteed to be equal.

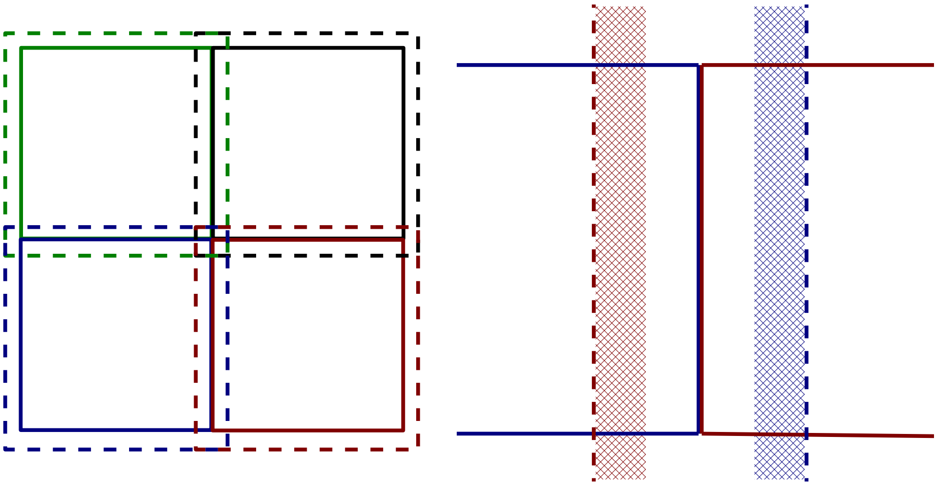


Figure 5.2.: 2D representation of space division by Divide and Conquer. Left: The reconstruction space is divided in four (3D: eight) neighboring subspaces represented by the colored continuous squares. The subspaces have an overlap that defines a larger subspace represented by the dashed squares. Right: Two zoomed neighboring subspaces. Because the overlap is twice the maximum local optimization area, the critical area at the border is guaranteed to be equal. The hatched fields show the incorrect areas.

In addition to point clouds, the unlimited scalability is valid for triangle meshes if they are optimized only in a limited local area. In this thesis, a local method for triangulation of point clouds is adapted to this requirement (cf. Section 6.4). A disadvantage of this strategy is that multiple equal triangles appear in the overlap area. Fortunately, the problem of removing equal triangles is not difficult to solve because the corresponding points can be filtered based on equal coordinates.

In summary, by local optimization, it can be guaranteed that the 3D points or triangles in neighboring areas that are equal inside the border areas can simply be cut. This allows for fast parallel processing of the surface.

A subdivision of the reconstruction space in subareas with similar size is adequate only for spatial data with constant density. Complex scenes and camera configurations with different distances to the scene cause varying densities of the point clouds. A regular decomposition of the reconstruction space would lead to possibly large varying computational efforts for the subspaces. For a dynamic allocation of the subarea size, sparse point clouds from the registration step can be employed. Alternatively, the space can be iteratively split by counting the points in the current area. If the number of points exceeds a threshold, the reconstruction space is split in 3D into eight neighboring subspaces (cf. Fig. 5.3). Through an iterative algorithm, the space can be split at runtime when the memory resources are not sufficient. The sum of the points in the subareas is scaled by the expected redundancy. This is because multiple points can be fused in the same voxels if they represent the same surface. This information can be derived from the disparity map by the number of matching images per reference image.

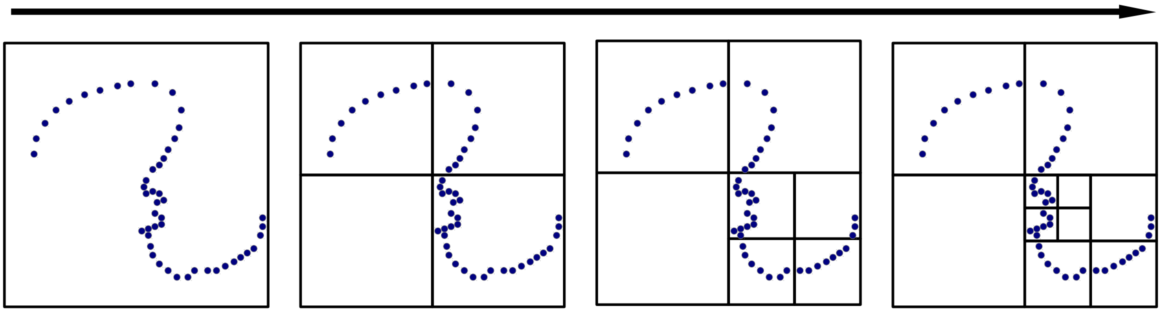


Figure 5.3.: Dynamic iterative division of the reconstruction space (2D). Individual subspaces contain a large amount of data and are split depending on the density. The entire space is split in ten subareas that can be processed in parallel.

The subspace overlap leads to a multiple processing of small parts. In addition, the fact that the overlap has to be twice the maximum local optimization area can lead to problems when having points with strongly varying quality. For dense parts of the point cloud, the algorithm divides the reconstruction space into small areas. If these areas contain points with low quality, the overlap can be larger than the reconstruction space itself. Because this is not meaningful, the low quality data has to be discarded. On one hand this can theoretically lead to the elimination of important data; on the other hand, in practical application, the low quality data are mostly not of interest.

5.3. Error Models

Methods based on local 3D reconstruction do not yet reach the quality of global methods (Scharstein 2014a). In addition to unsolved difficulties in the complex formulation, the influence of stereo uncertainties on disparity maps is not well studied. The continuous formulation of measurements in the local volumetric fusion method described in Section 4.3 considers an uncertainty of the measurements that has to be provided. In the following sections, three main influences on the 3D uncertainty are discussed. First, a derivation of an ellipsoidal error model of 3D space based on the stereo configuration is presented. Second, a derivation and an analysis of the registration error influence on the ellipsoidal error model are given. Third, the uncertainty of disparities is discussed and a statistical learning scheme is presented that allows for uncertainty-classification, depending on the local oscillation behavior of the disparity map.

5.3.1. Stereo Error Model

Disparity maps from a registered image set allow for the reconstruction of 3D point clouds. Errors of the disparities result in 3D reconstruction errors. In the following paragraphs, an ellipsoidal model is discussed that propagates the influence of the disparity uncertainty to the uncertainty of the 3D point.

The basics for the transformation of a 3D point P in camera coordinates onto the image plane are given in Section 2.2. Assuming fixed parameters for camera calibration, the transformation can be reduced by accounting radial distortion in the images. Furthermore, the equations can be further simplified by moving the origin of the image coordinate center to the center of the image (Hirschmüller 2003). Considering equally oriented cameras that only differ in the baseline, the following equations can be obtained:

$$p_{1x} = f \frac{P_x}{P_z} \ , \tag{5.5}$$

$$p_{2x} = f \frac{P_x - t}{P_z} \ , \tag{5.6}$$

$$p_{1y} = p_{2y} = f \frac{P_y}{P_z} \ . \tag{5.7}$$

where t is the length of the baseline between two images, f is the focal length, and p_1 and p_2 are the image coordinates (cf. Fig. 5.4).

The assumption of equally oriented cameras is a simplification that is not generally valid. Nevertheless, it is essential because the general case leads to large equation systems in error propagation that cannot be solved directly, and all practically important stereo configurations can be reduced to this case.

By inversion of the linear system of Eqs. (5.5) to (5.7) and further substitutions, the final result is (Hirschmüller 2003):

$$P_x = \frac{t \ p_{1x}}{p_{1x} - p_{2x}} \ , \tag{5.8}$$

$$P_y = \frac{(p_{1y} + p_{2y}) \ t}{2(p_{1x} - p_{2x})} \ , \tag{5.9}$$

$$P_z = \frac{f \ t}{p_{1x} - p_{2x}} \ . \tag{5.10}$$

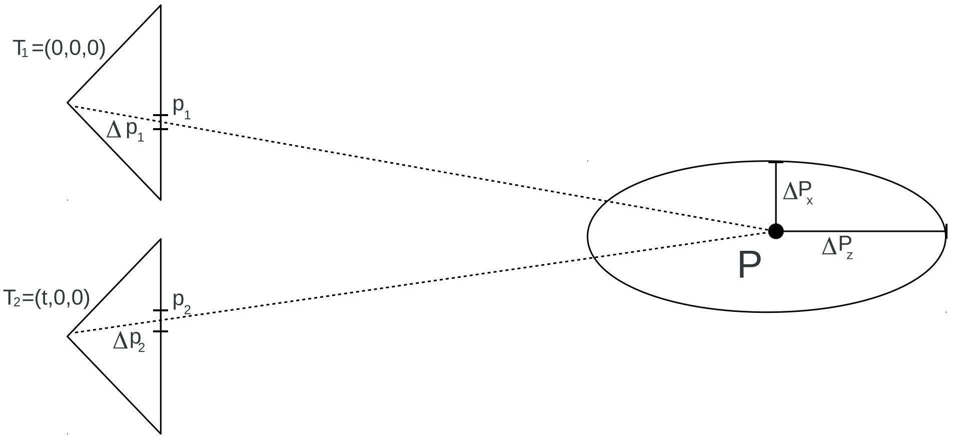


Figure 5.4.: An equally oriented camera pair. The uncertainty of the disparities Δp in the image leads to an ellipsoidal uncertainty of the 3D point.

The linear equation system of Eqs. (5.8) to (5.10) is important for error propagation because it describes the 3D point depending on the corresponding 2D image points of both images. For this, the partial derivatives from Table 5.1 have to be considered.

$f(p_{1x}, p_{1y}, p_{2x}, p_{2y}) =$	$P_x = \frac{t\ p_{1x}}{p_{1x}-p_{2x}}$	$P_y = \frac{(p_{1y}+p_{2y})\ t}{2\ (p_{1x}-p_{2x})}$	$P_z = \frac{f\ t}{p_{1x}-p_{2x}}$
$\frac{\delta f}{\delta p_{1x}} =$	$\frac{-t\ p_{2x}}{(p_{1x}-p_{2x})^2} = \frac{P_z(t-P_x)}{f\ t}$	$\frac{-t\ p_{1y}}{(p_{1x}-p_{2x})^2} = \frac{-P_z P_y}{f\ t}$	$\frac{-f\ t}{(p_{1x}-p_{2x})^2} = \frac{-P_z^2}{f\ t}$
$\frac{\delta f}{\delta p_{1y}} =$	0	$\frac{t}{2(p_{1x}-p_{2x})} = \frac{P_z}{2f}$	0
$\frac{\delta f}{\delta p_{2x}} =$	$\frac{t\ p_{1x}}{(p_{1x}-p_{2x})^2} = \frac{P_z P_x}{f\ t}$	$\frac{t\ p_{1y}}{(p_{1x}-p_{2x})^2} = \frac{P_z P_y}{ft}$	$\frac{f\ t}{(p_{1x}-p_{2x})^2} = \frac{P_z^2}{f\ t}$
$\frac{\delta f}{\delta p_{2y}} =$	0	$\frac{t}{2(p_{1x}-p_{2x})} = \frac{P_z}{2f}$	0

Table 5.1.: Jacobian with partial derivatives from Eqs. (5.8) to (5.10)

Considering a Gaussian error Δp with the same standard deviation σ_p in all coordinates p_{1x}, p_{1y}, p_{2x}, and p_{2x} and the partial derivatives, the propagated error is obtained, e.g., in the x- direction:

$$\Delta P_x = \sqrt{(\frac{\delta P_x}{\delta p_{1x}}\Delta p)^2 + (\frac{\delta P_x}{\delta p_{1y}}\Delta p)^2 + (\frac{\delta P_x}{\delta p_{2x}}\Delta p)^2 + (\frac{\delta P_x}{\delta p_{2y}}\Delta p)^2}\ . \tag{5.11}$$

Substituting the derivatives from Table 5.1 for ΔP_x, ΔP_y, and ΔP_z results in Eqs. (5.12) to (5.14). In addition to (Hirschmüller 2003), the error propagation was reported by numerous Photogrammetric textbooks, Molton and Brady (2000), and

Matthies (1992), who used $\Delta P_z \sim \Delta p \frac{P_z^2}{ft}$, which is an approximation.

$$\Delta P_x = \Delta p \frac{P_z}{ft} \sqrt{(t - P_x)^2 + P_x^2} \tag{5.12}$$

$$\Delta P_y = \Delta p \frac{P_z}{ft} \sqrt{2P_y^2 + \frac{t^2}{2}} \tag{5.13}$$

$$\Delta P_z = \Delta p \frac{P_z^2}{ft} \sqrt{2} \tag{5.14}$$

An important aspect to consider the error in 3D is the quadratic increase in the z-direction. This is discussed for scalable disparity map fusion in Section 6.2.

5.3.2. Registration Error

In the preceding section, a stereo error model is given for equally oriented cameras and constant disparity error in all directions. In addition to disparity errors, the 3D point depends on the errors in the camera parameters (cf. Fig. 5.5). In the following paragraphs, a joint consideration of the registration and the disparity error is discussed.

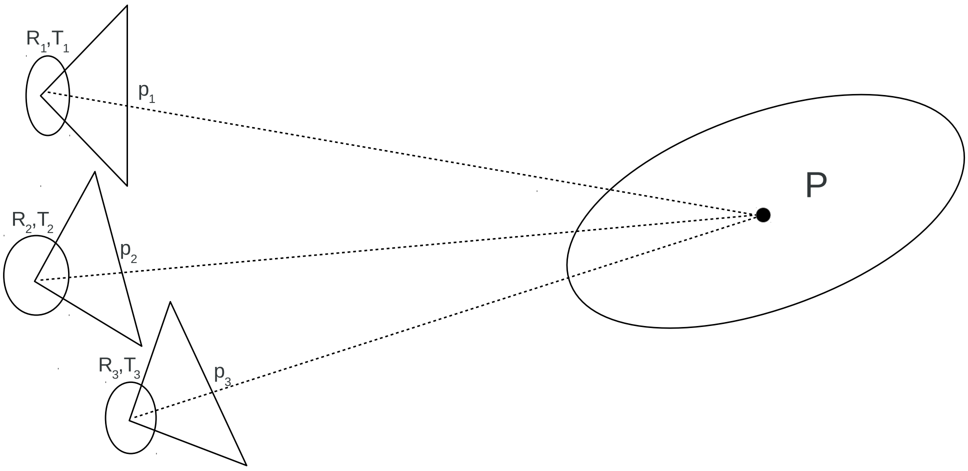


Figure 5.5.: The uncertainty of three camera poses influences the uncertainty of the 3D point. In addition to the disparity uncertainty shown in Fig. 5.4, six further parameters per camera that describe the pose and possibly additional inner parameters for the cameras and their covariance information have to be considered.

As described in Section 2.2.1, image registration corresponds to the estimation of the set of camera parameters. Both the inner camera parameters that describe calibration and distortion, and the rotation matrices or quaternions and translation vectors for all images are obtained by a robust bundle adjustment.

As described in Section 2.2.1, image registration estimated the 3D points from the complete set of images. Hence, for a 3D point, multiple measurements from n cameras exist. From those, variances and covariances of all inner and outer parameters are obtained. For simplification, only the covariances of the outer parameters are considered for error propagation represented by a 6×6 matrix that contains the variances and covariances of three rotation and three translation parameters.

Similar to the stereo case presented in Section 5.3.1, the first step is the generation of an equation system that describes the 3D points depending on all the geometric parameters of all cameras. By obtaining quaternions q with parameters q_1, q_2, q_3, q_4 instead of rotation matrices, the equation system from Eq. (2.11) for camera i can be written as:

$$P_w = q_i P_{ci} q_i^{-1} + T_i \ . \tag{5.15}$$

Unfortunately, in contrast to the simplified Eqs. (5.8) to (5.10), a joint analytical formulation that considers a set of cameras is difficult to derive, it is complex, and thus, even if available, it is difficult to analyze.

Fortunately, a numerical propagation by infinitesimal changes of the values can simulate a model that assumes linear error propagation. Yet, the propagation of the parameter uncertainty is expensive for multiple camera pairs. Nevertheless, it seems feasible for parallel systems.

For all 3D points from n images, a numerical propagation has to be done considering partial derivations. To this end, the Jacobian matrix J_p has to be calculated containing the partial derivation of the camera parameters and the disparity d:

$$J_P^i = \begin{pmatrix} \frac{\delta P_x}{\delta q_1} & \frac{\delta P_x}{\delta q_2} & \frac{\delta P_x}{\delta q_3} & \frac{\delta P_x}{\delta t_x} & \frac{\delta P_x}{\delta t_y} & \frac{\delta P_x}{\delta t_z} & \frac{\delta P_x}{\delta d} \\ \frac{\delta P_y}{\delta q_1} & \frac{\delta P_y}{\delta q_2} & \frac{\delta P_y}{\delta q_3} & \frac{\delta P_y}{\delta t_y} & \frac{\delta P_y}{\delta t_y} & \frac{\delta P_y}{\delta t_z} & \frac{\delta P_y}{\delta d} \\ \frac{\delta P_z}{\delta q_1} & \frac{\delta P_z}{\delta q_2} & \frac{\delta P_z}{\delta q_3} & \frac{\delta P_z}{\delta t_z} & \frac{\delta P_z}{\delta t_y} & \frac{\delta P_z}{\delta t_z} & \frac{\delta P_z}{\delta d} \end{pmatrix} \ , \tag{5.16}$$

For the quaternion, the largest value of the normalized quaternion is maintained fixed, and thus, covariance information is only estimated for the remaining three parameters.

Disparity maps are obtained by MVS considering n images. Hence, an overall Jacobian has to be defined:

$$J_P = (J_P^1 \ \ldots \ J_P^n) \ . \tag{5.17}$$

In practice, Jacobians can be derived from Eq. (5.15) by infinitesimal change of the parameters and evaluation of the corresponding changes of the 3D point P. The additional disparity uncertainty has a direct influence on the depth p_z in camera coordinates. It can be considered, e.g., as half a pixel and transformed to the uncertainty of P by Eq. (5.14). The uncertainty of the disparity is discussed in detail in Section 5.3.3.

From the image registration for all cameras, 6×6 covariance matrices C_i are obtained that contain the uncertainty of rotation R and translation T parameters. For relative pose estimation, one camera pair obtains only a 5×5 covariance matrix (cf. Section 2.2.1). The covariance matrices can be extended by a further dimension adding the disparity error. For further simplification, the disparity error is assumed to be uncorrelated with the error of the camera parameters:

$$C_i = \begin{pmatrix} \sigma_{r_1^2} & \sigma_{r_1 r_2} & \sigma_{r_1 r_3} & \sigma_{r_1 t_1} & \sigma_{r_1 t_2} & \sigma_{r_1 t_3} & 0 \\ \sigma_{r_2 r_1} & \sigma_{r_2^2} & \sigma_{r_2 r_3} & \sigma_{r_2 t_1} & \sigma_{r_2 t_2} & \sigma_{r_2 t_3} & 0 \\ \sigma_{r_3 r_1} & \sigma_{r_3 r_2} & \sigma_{r_3^2} & \sigma_{r_3 t_1} & \sigma_{r_3 t_2} & \sigma_{r_3 t_3} & 0 \\ \sigma_{t_1 r_1} & \sigma_{t_1 r_2} & \sigma_{t_1 r_3} & \sigma_{t_1^2} & \sigma_{t_1 t_2} & \sigma_{t_1 t_3} & 0 \\ \sigma_{t_2 r_1} & \sigma_{t_2 r_2} & \sigma_{t_2 r_3} & \sigma_{t_2 t_1} & \sigma_{t_2^2} & \sigma_{t_2 t_3} & 0 \\ \sigma_{t_3 r_1} & \sigma_{t_3 r_2} & \sigma_{t_3 r_3} & \sigma_{t_3 t_1} & \sigma_{t_3 t_2} & \sigma_{t_3^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta p^2 \end{pmatrix} . \tag{5.18}$$

The covariance matrix C of point P can be derived by multiplying the numerical Jacobian J_P from Eq. (5.17) and the covariance matrices C_i from Eq. (5.18) as:

$$C_p = J_P \begin{pmatrix} C_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & C_n \end{pmatrix} J_P^T . \tag{5.19}$$

The probabilistic framework presented in Section 4.3 allows for the fusion of spatial data. Yet, it only considers the uncertainty of the 3D point in one direction. The error propagation that considers registration and disparity uncertainty leads to a trivariate uncertainty of the spatial measurement. Nevertheless, the univariate fusion is feasible because uncertainties in stereo configurations mostly have one dominant direction. This direction can be obtained from the covariance matrix C_p in terms of the maximum eigenvalue as the uncertainty, and the corresponding eigenvector as the dominant direction (cf. Fig. 5.6).

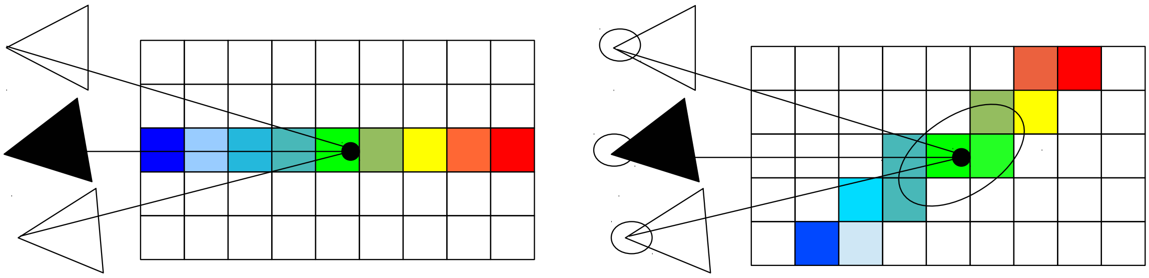


Figure 5.6.: The uncertainty of a 3D point can be estimated considering the covariance matrices from all the cameras that describe the uncertainty of inner and outer parameters. Left: Propagation along the line of sight. Right: The eigenvector with the largest eigenvalue describes a dominant direction that can be used for propagation of the univariate uncertainty.

Image registration is based on the processing of sparse visually salient data. Depending on the images, this and the overlap between images can lead to strongly varying densities of 3D points. In turn, this leads to differing uncertainties in different regions. Furthermore, the numerical error propagation is time-consuming when considering large sets of images. For practical applications, the numerical error propagation is discussed in Section 7.6.

5.3.3. Disparity Error

The stereo error models from Sections 5.3.1 and 5.3.2 allow for an error propagation that considers a disparity error Δp. The suitable ellipsoidal error model with a simplification in one direction described by Eq. (5.14) assumes an equal disparity error in the x- and y- direction of both images. The generalization of a constant disparity error, e.g., half a pixel, can be a suitable assumption when having simple image configurations and well-textured objects. Nevertheless, there are various configurations and scenes that lead to varying disparity errors. In the following paragraphs, the disparity uncertainty is analyzed and a feature is presented that is highly correlated with the disparity error. For the estimation of a function that describes the correlation, a machine learning approach is presented.

Learning quality functions for disparities entails much effort and is difficult because it requires ground-truth data, a stable feature that correlates highly with the uncertainty, and machine learning methods that can manage noisy data with outliers. The disparity quality depends on multiple features, such as the texture strength or the surface slant. The latter results from foreshortening and from prior assumptions in stereo methods that often have a fronto-parallel bias. In this thesis, SGM is used as stereo method. It penalizes neighboring disparity changes, generating fine reconstruction of fronto-parallel surfaces (cf. Section 5.1). SGM is a semi-global method that allows for the estimation of disparities in untextured regions also by considering the textured neighborhood. This leads to dense disparity maps, yet with extremely varying quality of individual disparities. An example for the slant is given in Fig. 5.7.

Most stereo methods obtain subpixel accuracy by considering the matching costs of the neighboring pixels. SGM uses quadratic regression that fits a parabola through the neighboring pixel costs. Depending on the slant, the accuracy of the subpixel estimation can vary as the disparity changes differ in neighborhoods, particularly for non-fronto-parallel regions.

The disparity quality is influenced by many other factors. The registration error and its influence on the 3D geometry have been discussed in Section 5.3.2. There, disparity error and registration error are assumed to be uncorrelated. However, image registration can actually influence the quadratic regression for obtaining subpixel accuracy because its accuracy can lead to varying subpixel uncertainty.

Furthermore, the uncertainty can be influenced by the cameras used. Low quality cameras have a restricted quality because they use small chips and low quality lenses.

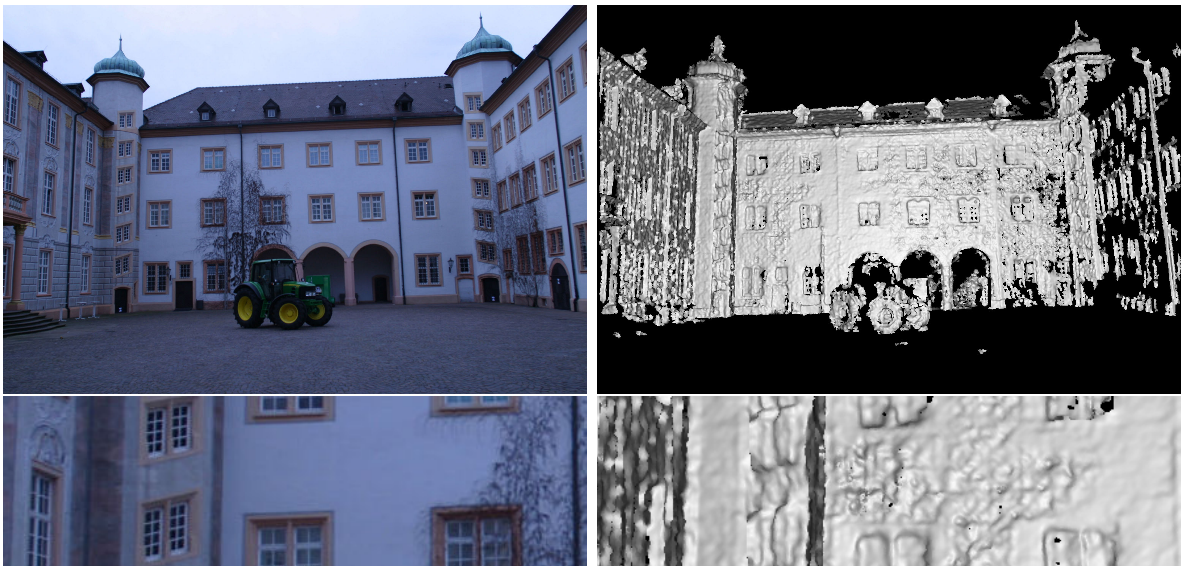


Figure 5.7.: Left: (Zoomed) image of the Ettlingen30 sequence (Strecha et al. 2008). Right: Coded image that shows the slant to the surfaces estimated from the depth map that considers 48 neighboring values. The coding ranges from 0°(white) to 90°(black). In this image, the slant provides a reasonable impression of the reconstruction quality of the surfaces. Untextured and non-fronto-parallel regions are mostly not precisely reconstructed.

Yet, even high quality cameras cannot completely avoid motion blur or out of focus areas.

Naïve learning of the disparity quality could consist in the definition of a multivariate space with feature dimensions that cover the uncertainties. Features such as texture strength or surface slant influence the quality and could be used for learning. Yet, the quality also depends on the camera type, especially on the chip and lens employed. Learning the quality for specific cameras is expensive because ground-truth data is necessary.

The classification of uncertainties in disparity maps that depend on slant and texture could be inexpertly performed by the estimation of a pixelwise normal vector and an image gradient. However, for disparity maps, this is usually unstable because they show an oscillation with unknown frequency, as presented in Fig. 5.8. Therefore, the normal estimation window used could oversmooth the normal; however, it could also obtain wrong measurements by undersampling. Fig. 5.7 shows a slant map that uses a constant neighborhood that considers 48 neighbors. It is obvious that some normal vectors are wrongly oriented in weakly textured regions or areas with large slant. Furthermore, learning multivariate systems is difficult because wrong correlations tend to be estimated when not enough data is available.

Using a single feature that covers both, and possibly more, uncertainties seems a more

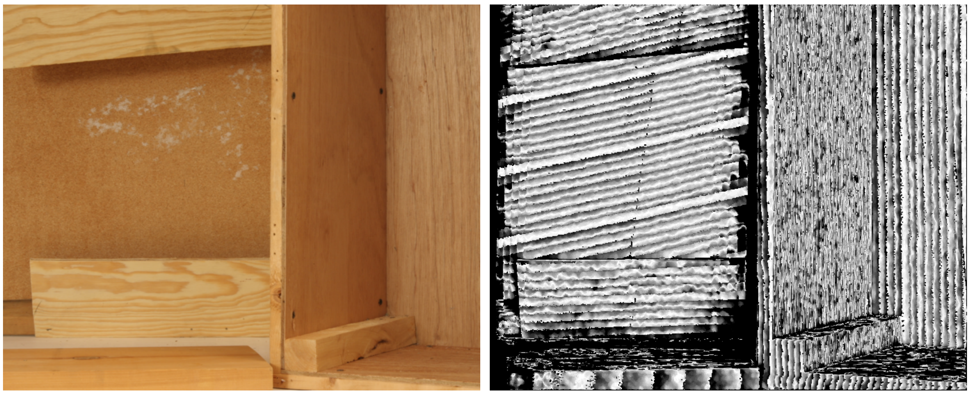


Figure 5.8.: Left: Half-resolution image from the Middlebury stereo data (Scharstein 2014b). Right: Difference between ground truth and disparity map from SGM coded from zero error (white) to half pixel error (dark grey). The surface quality has varying frequencies depending on the slant.

promising way to pursue learning the correlation function. Thus, a TV-based feature is proposed that implicitly covers important aspects of the uncertainty. The TV feature represents the local oscillation behavior of the disparity map and covers uncertainties depending on slant and texture (cf. Section 2.1.2). The L_2 norm was chosen as its norm. In the past, methods that use global optimization were published that also use TV for the regularization of 3D surface reconstruction (Zach et al. 2007, Kolev et al. 2012). They use TV-L_1 in 3D space to regularize surfaces considering outliers. In this thesis, the TV-L_2 norm is used that is not robust against outliers. The TV is estimated from disparity maps, instead of 3D surfaces, with the motivation of separating accurate surfaces from noisy surfaces. Noisy surfaces and outliers cause high TV terms, making the TV-L_2 norm feasible for terms of quality estimation on disparity maps. Because the method works on disparity maps, the original formulation on 2D signals from Rudin et al. (1992) is used (cf. Section 2.1.2).

The simple use of a TV term does not automatically solve the problems of oversmoothing and undersampling. In addition to different frequencies, the TV term should be discretized because for discretization levels, the uncertainty of disparities can be learned directly. A reasonable way to define discretization levels is to determine the TV of varying neighboring pixel sets. To this end, a threshold θ is used that limits the value of the TV sum:

$$\arg\max_n \Big(\sum_{m=1}^{n} \frac{1}{8m} \sum_{i,j \in x_i} \sqrt{|d_{i+1,j} - d_{i,j}|^2 + |d_{i,j+1} - d_{i,j}|^2} < \theta \Big) \; , \tag{5.20}$$

where disparity d and $|x_i| = 8m$. At first, the eight directly neighboring pixels are considered. The resulting sum of the TV-L_2 norm is compared with the threshold. If the sum exceeds the threshold, the discretized value $n = 1$ defines the TV class. Otherwise, TV is calculated considering the next 16 ($8m$, $m = 2$) pixels and added to the sum. When not exceeding the threshold, this is iterated until a maximum $n = 20$ is reached. The calculation is performed in linear time because for $i = 20$, a maximum of 361 pixels are considered. For pixels in the disparity map that do not have valid disparity, a value of ∞ is used for the TV.

The number of pixels considered for level m rises with $8m$ (cf. Eq. (5.20)). Hence, the TV overall sum increases with the level size. This is normalized by a division of the sum by $8m$. In the experiments shown in Section 7.5, this normalization leads to better results. For θ, a value of one is empirically found to be suitable, leading to an oscillation of a maximum of one disparity on average for all directions. Fig. 5.9 provides two examples.

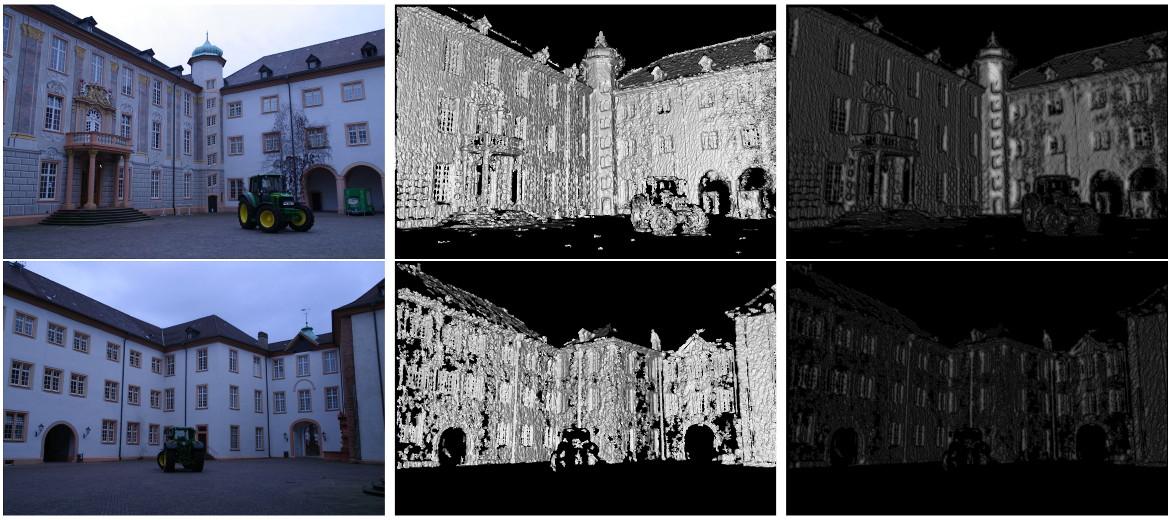


Figure 5.9.: Quality voting for all pixels assigned a disparity. The left images are two images from the Ettlingen30 dataset. The middle images show the surface slant that provides a reasonable impression of the quality. The right images show the quality class based on TV from zero (black) to 20 (white). Frontoparallel and fine textured regions result in high digit classes.

The discretized TV classes $n = [1, 20]$ can be used for the learning of error functions that describe the uncertainty. For this, it is assumed that the individual error follows a set of Gaussians $\mathcal{N}_n(\mu_n, \sigma_n)$ with parameters $\theta_n = \mu_n, \sigma_n$.

Ground-truth data is necessary for learning the function parameters. In this thesis, the Middlebury Stereo datasets (Scharstein and Szeliski 2002, Scharstein and Szeliski 2003, Scharstein and Pal 2007) are used because, for some images, the ground-truth data is made publicly available (Scharstein 2014b). As the ground-truth contains discrete disparity values, the unsigned accuracy is limited to 0.5 pixels. Thus, only the half-sized images were used, allowing for an accuracy of 0.25 pixels.

The TV class can be determined for a set of disparity maps from the valid pixels.

The corresponding Gaussian can be estimated for all classes $0 < n \leq 20$ by an EM method that maximizes the probability of the parameters $\arg\max_{\theta_n} p(\theta_n|\mathcal{D}_n)$. The data $\mathcal{D}_n$ describes the set of differences between the ground truth and the value based on the SGM results assigned to class n. A description of the machine learning methods used here is given in Section 2.3.3.

The reason for using EM instead of a ML or MAP estimation is that mixture functions are considered. For modeling the error from disparity maps, a combination between a Gaussian and a uniform function is a suitable approximation for the error distributions (Vogiatzis and Hernández 2011). A Gaussian represents the disparity uncertainty, whereas a uniform function represents the outliers. Unfortunately, the EM learning method has a need for an initial estimate that is close to the optimum. A modified MAP estimation obtains an initial set of parameters that are suitable for EM. As prior information $p(\theta)$ for MAP, $\mu = 0$ ($p(\mu = 0) = 1$) is used, because the estimated value should be the most probable. The calculation of the variance with standard MAP estimation is:

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(d_i - g_i)^2 \ , \tag{5.21}$$

with disparity d and ground truth g for i measurements.

This function is extremely sensitive concerning outliers because the L_2 norm is used. In disparity maps, multiple outliers can occur. Therefore, a better estimation is provided by the L_1 norm. This modified MAP* formulation leads to an uncertainty of:

$$\sigma = \frac{1}{n}\sum_{i=1}^{n}|d_i - g_i| \ . \tag{5.22}$$

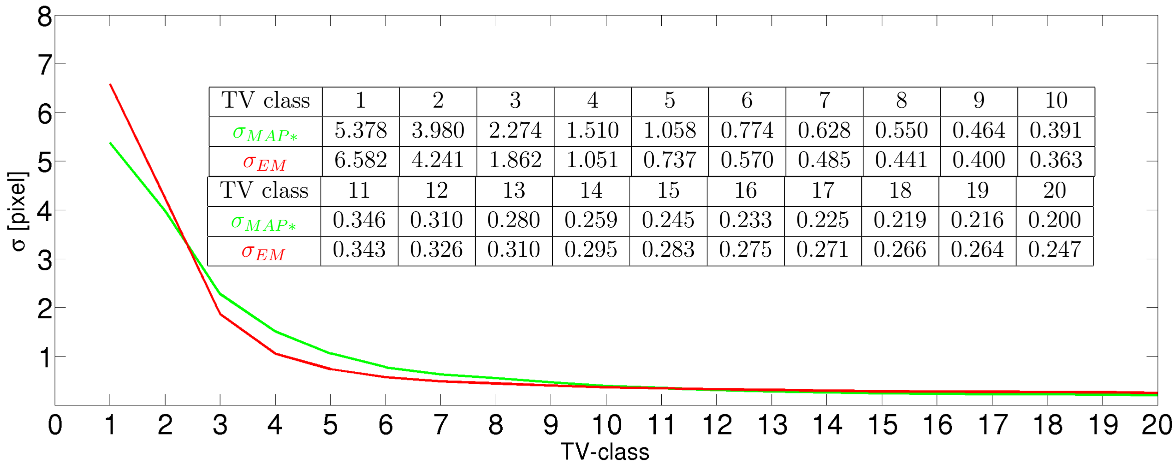
With this function, an adequate initial approximation of the Gaussians that represent the disparity uncertainty is obtained for the TV classes.

To obtain a probability for the outliers, represented as uniform distribution, an additional step is conducted after the MAP* step. There, those measurements are detected which are not in an area of five σ, and hence classified as outliers. The ratio between the number of outliers and the number of all measurements describes the outlier probability.

The estimated Gaussian and uniform distribution are used as initial state for EM. In the E step, only the data assigned to the Gaussian are considered (cf. Section 2.3.3 and Fig. 2.8). Measurements assigned to the uniform distribution are regarded as outliers and are not considered in the E step. For filtered data, the classical MAP function from Eq. (5.21) is used for the M step. Afterwards, a new probability is calculated for the uniform distribution as done after the MAP* step.

The EM process is iterated only one time because experience shows that one step is enough. The resulting standard deviations for the MAP and the optimized EM estimation for the 20 classes are shown in the table in Fig. 5.10.

Obviously, the error rises exponentially for small TV classes. Higher number of TV classes converges to a quarter pixel because this is the best value the estimation can



TV class	1	2	3	4	5	6	7	8	9	10
σ_{MAP*}	5.378	3.980	2.274	1.510	1.058	0.774	0.628	0.550	0.464	0.391
σ_{EM}	6.582	4.241	1.862	1.051	0.737	0.570	0.485	0.441	0.400	0.363
TV class	11	12	13	14	15	16	17	18	19	20
σ_{MAP*}	0.346	0.310	0.280	0.259	0.245	0.233	0.225	0.219	0.216	0.200
σ_{EM}	0.343	0.326	0.310	0.295	0.283	0.275	0.271	0.266	0.264	0.247

Figure 5.10.: Learned standard deviations in pixels for the 20 TV classes. The green graph shows the initial estimation using MAP. The red graph shows the optimized function estimated using EM.

achieve, given the quantization effects of the ground truth.

5.4. Consistency Checks

The probabilistic framework presented in Section 4.3 in combination with error models from Section 5.3 allows for the fusion of noisy spatial data. Nevertheless, outliers can appear, but in reduced numbers. In addition to filtering the depth maps, like in SGM, the novel volumetric fusion allows for further consistency checks in the 3D space. These consistency checks are inevitable for obtaining accurate results for difficult configurations. Methods already exist that consider image consistency (Vu et al. 2012) or geometric consistency (Merrell et al. 2007, Hu and Mordohai 2012) for the optimization and filtering of 3D surface models. In the following two sections, the two types of consistency checks are discussed, considering scalable disparity map fusion. Finally, an extension to volumetric probabilistic consistency checking is introduced.

5.4.1. Image Consistency

For consistency checks in the image space, the colored 3D point cloud or a resulting triangle mesh is optimized in a specific area by checking the consistency in the respective images. To this end, the initial surfaces can be reduced also by triangle merging or increased by a tessellation of the triangle mesh. For the optimization, the geometric parameters in 3D are optimized, e.g., with variational approaches (Pons et al. 2007). Comparing surface and image space, a cost function can be built with costs such as NCC (cf. Section 2.1.1). For the comparison, the surface has to be re-projected to the images.

To avoid variational optimization, local approaches are feasible with an independent processing of the parameters.

In any case, the transformation to the image space is expensive because possibly many images have to be considered simultaneously. Hence, for image consistency checks, it is difficult to guarantee scalability. The direct modeling of locally optimized point clouds with a recursive Bayesian formulation enables a modeling that only considers one image at a time (cf. Section 4.3). By this means, small memory requirements are guaranteed, thus making image consistency checks in general unfeasible.

5.4.2. Geometrical Consistency

Assuming opaque surfaces and by being able to see a 3D point from an image, it follows that the space between a point and the camera center has to be empty. This geometric prior knowledge can also be used for consistency checks on the disparity map or in 3D space.

Merrell et al. (2007) proposed geometric consistency checks on depth maps. In this 2.5D filtering step, depth maps are re-projected to other depth maps considering geometric attributes. Geometric consistency checks can be computationally expensive for large amounts of data. Nevertheless, they are suitable for hardware-accelerated implementations.

For 3D consistency, the point cloud can be represented by a set of ellipsoids estimated based on the uncertainty. Geometric consistency can be achieved by intersection checks between all lines of sight between points and cameras and all ellipsoids (Hu and Mordohai 2012). The size of the ellipsoids can be used again as filter criterion by observing conflicts. Unfortunately, a continuous filtering in 3D is computationally expensive because it rises quadratically with the number of 3D points.

Volumetric methods are more suitable for scalable 3D modeling. Elements of the discretized space are empty or contain information from one or multiple 3D points. Checking for intersections on the line of sight beginning at a 3D point, the intersection of the voxels can be used to calculate the exact set of involved volumes efficiently. The volumetric intersection can even be managed more efficiently considering specific spatial data structures (cf. Section 6.1).

Counting the number of points that fall into the same volume element can help for choosing the correct voxel in case of conflicts by filtering the volume element that contains the smaller number of points. The novel probabilistic fusion presented in Section 4.3 allows for the estimation of surface probabilities. Therefore, the more robust probabilistic background should be considered in the filtering decision.

For unlimited scalability, 3D geometric consistency checks are not intuitively suitable. Ray tracing methods are global because they cast rays from 3D points to cameras. The divided reconstruction spaces described in Section 5.2 do not imply complete rays. Nevertheless, local optimization makes ray tracing feasible because the rays can be limited in a local area. Hence, the 3D outliers that cause reconstruction errors can be filtered. Outliers appear alone or in separate clusters that can be classified, e.g., in the meshing step. Global filtering is not possible and should be performed in 2.5D as a preprocessing step.

5.4.3. Probabilistic Consistency

In Section 4.3, a probabilistic approach is proposed that allows for 3D reconstruction in a volumetric probabilistic space. In this space, the surface is characterized by neighboring voxels for which the probability that one is in front of the surface and the other is behind the surface is high. The probabilistic space is used to generate volumetric surfaces that contain probabilistic information of surface description.

For geometric consistency checks, it is helpful to consider probabilistic information. As shown in Fig. 5.11, it is not obvious which voxel contains only outliers, and hence has to be filtered when detecting one or multiple conflicts. In addition, by casting another ray, the voxel itself could be filtered by another point.

Hence, the filtering decision has to be made only after casting all rays. To this end, rays are cast from all occupied voxels to all cameras from where the voxel was seen. For detected conflicts, the probability of the conflicting voxel is assigned to another voxel. All voxels retain only the maximum of the conflicting surface probabilities. Then, those voxels are filtered whose surface probability is less than the maximum surface probability from conflicting voxels. The filtering criterion can be written as:

$$p_{v_i} \leq \max_{c_t} \; p_{v_{c_t}} s_{c_t, v_{c_t}} \ , \tag{5.23}$$

where s is a binary predicate that voxel v_{c_t} was occupied by camera c_t. The predicate avoids the influence of quantization effects of the voxel space. $p_{v_{c_t}}$ is the probability of the voxels intersected by the ray from voxel v_i to camera c_t.

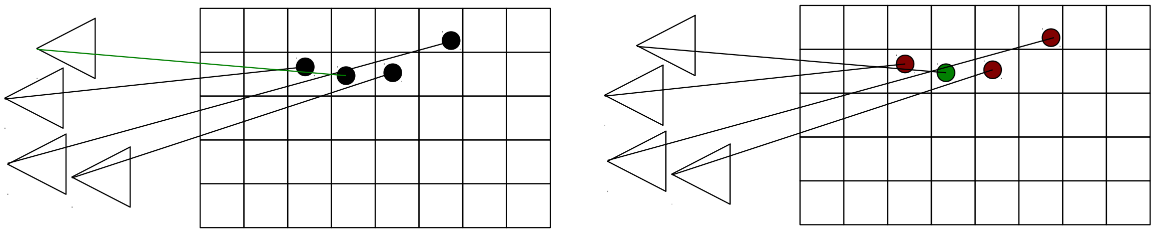


Figure 5.11.: Geometric consistency checks that consider surface probabilities. Left: The second point from the left is cast to the upper camera, resulting in a conflict with the leftmost point. It is not obvious which point has to be filtered. Right: After considering multiple points with surface probabilities, which are all in conflict, only the point with the highest surface probability is maintained.

Chapter 6.

Multi-Resolution Computation

In Chapter 4, a probabilistic fusion method was presented that allows for the fusion of noisy spatial data with similar quality. This chapter focuses on the handling of data with strongly varying quality. The error model presented in Section 5.3.1 describes the uncertainty of the 3D points that can be derived considering camera and disparity parameters as well as their uncertainties. The uncertainty depends on the baseline between the camera positions, the focal length, the disparity error, and the distance to the surface, among other factors. The distance to the surface influences the errors in 3D points even quadratically in a single direction. Having a 20-times larger distance, as shown in the images in Fig. 6.1, an uncertainty difference with a factor of 400 follows. The simple fusion of data without considering the uncertainty will be useless as the low-resolution information disturbs the high-resolution information.

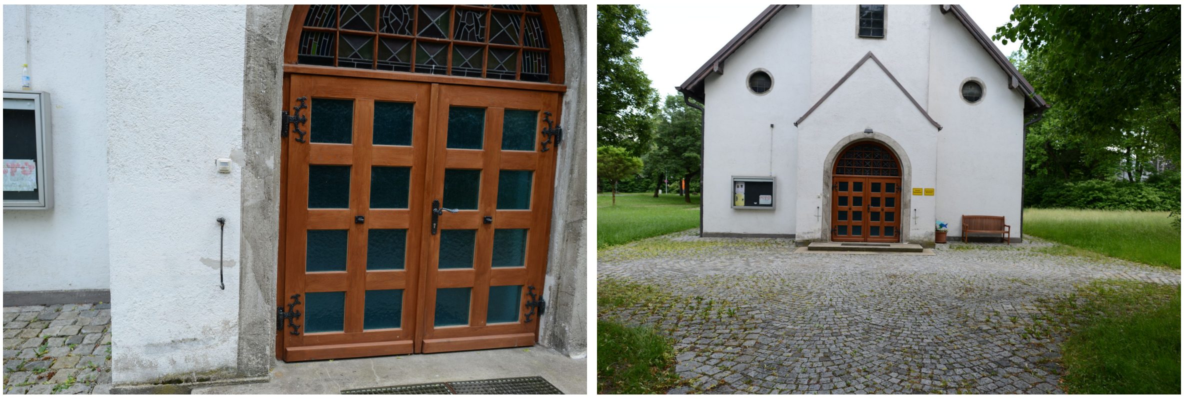


Figure 6.1.: Two images showing partially the same scene. The right image was captured at 20-times the distance to the object than the left image. The difference in quality has to be considered in the fusion process.

In Chapter 3, octrees were brought up as a means for volumetric fusion. In Sections 6.1 and 6.2, data structures and their use in the volumetric fusion of similar and differing disparity quality are discussed and adapted to (multi-view) stereo error models. The use of hierarchical spatial data structures enables a consideration of quality on varying levels. In Section 6.3, a brief analysis of the applicability of scale-space techniques is given. Additionally, for optimizing the point clouds, the method described in this thesis is concerned with the reconstruction of 3D surface models. In Section 6.4, the adaption of a known triangulation method for the processing of spatial data with varying quality is presented.

6.1. Data Structures

Spatial data structures are of high importance in the fields of computer vision and computer graphics. The key idea of the spatial data structures employed herein lies in the division of the 3D space into finite subsets. The subsets are arranged into efficient structures enabling rapid access and manipulation of the elements with small memory requirements. The division can be conducted depending on the density of the point clouds by means of kd-trees (Bentley 1975). Spatial data structures can be used to quickly obtain the relationships among points, e.g., for an efficient estimation of normal vectors from a point cloud considering a fixed number of neighbors. For fusing point clouds, discretization through a space division is suitable because redundant data are to be merged. Octrees allow for the fusion and filtering of points, also with varying quality, and are thus discussed in more detail.

An octree is a hierarchical data structure based on a uniform regular decomposition into single cubic elements, called voxels. Octrees can be used for merging redundant data, such as multiple 3D points from MVS, in the same octree elements. In contrast to kd-trees, the space is divided with respect to the point quality instead of the point density. The transformation of a 3D point into an octree space corresponds to the update in the octree. The basic idea of a point representation in an octree is given in Fig. 6.2.

Beginning with a root node representing the initial reconstruction space, an octree usually has eight children per node, which divides the space into eight subspaces whose center points have equal distances to the center point of the parent node. This subdivision is iterated to a specific point-dependent level. For a new 3D point, a voxel-wise subdivision is conducted until the specific level is reached. This level has to be specified and can be adapted, e.g., depending on the quality of the 3D point. Octrees are particularly suitable for merging redundant 3D points to reduce the memory consumption in dense configurations. Points being assigned to the same voxel can be merged. For this reason, discretization effects should not be neglected.

Standard volumetric methods regularly discretize the complete reconstruction space. In contrast, octrees represent empty areas with large voxels. Furthermore, octrees have a short access time that is logarithmic in complexity. The complexity can even be optimized, e.g., by employing multiple octrees organized in binary trees (Bodenmüller 2009). When inserting a 3D point into a voxel at a specific level, the voxel is assigned a binary state coding the occupancy. In practical applications, the voxels represented by the nodes of the octree can be given attribute information such as the center coordinate, color, surface probability, or camera information.

Octrees with varying voxel size allow for the efficient processing of operations such as ray tracing (Amanatides and Woo 1987). In ray tracing, a line is cast through the

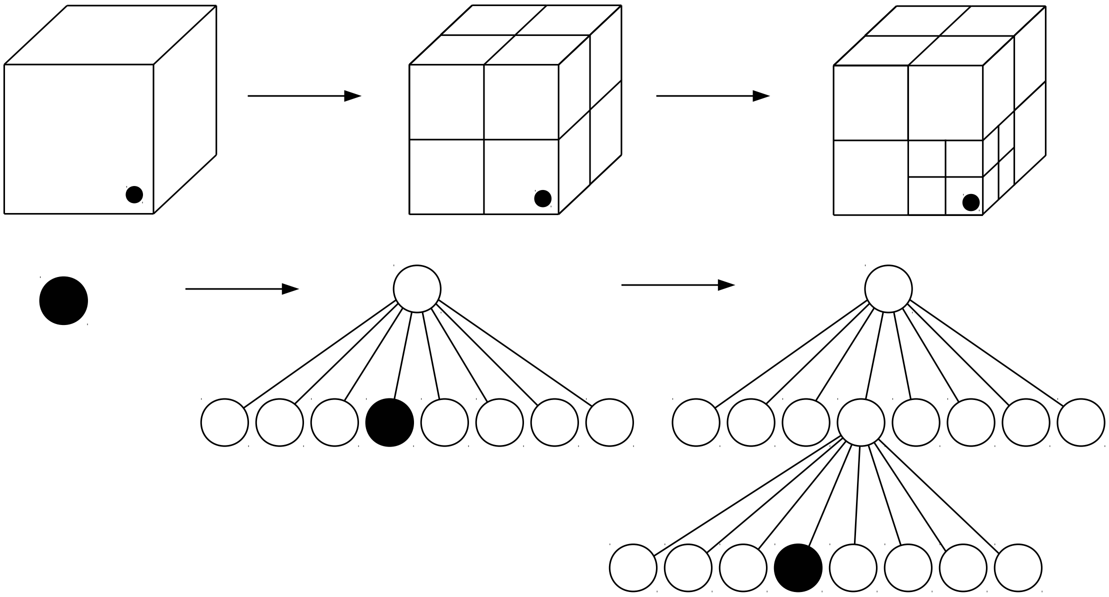


Figure 6.2.: Update of an octree when inserting a spatial point. From the initial node, the space is incrementally divided into eight subspaces modeled by eight children. The space is divided until the given level is reached (e.g., level three). The black nodes are marked as occupied.

3D space determining the intersecting volumetric subspaces. In the case of MVS, ray tracing methods are suitable for fast geometric consistency checks (cf. Section 5.4.2). On the line of sight, intersecting points with the subvolumes are calculated incrementally by considering the topology of the data structure.

6.2. Adaptation to Error Models

Octrees allow for an efficient handling of spatial data on varying levels. This is very suitable for processing point clouds from MVS configurations because they can have highly differing qualities depending on the image configuration. In Section 5.3.1, an ellipsoidal error model is presented, which can be reduced to a dominant uncertainty σ_p of a 3D point in one dimension (cf. Eq. (5.14)). The uncertainty depends, e.g., on the disparity error, focal length, baseline, or distance to the surface. For a calibrated image set, all values other than the disparity error are known. In Section 5.3.3, a feature that allows for an estimation of the disparity error depending on the local oscillation behavior of the disparity map is presented. Considering all available information, a fixed scalar value parameterizing the 3D uncertainty can be derived by representing a Gaussian uncertainty on the line of sight.

An important step for an efficient handling of point clouds with varying density in the octree data structures is the individual choice of the octree level that corresponds to the voxel size v_s. It is intuitive to define a linear correspondence between the voxel size and spatial uncertainty of the 3D point.

A similar approach proposed by Fuhrmann and Goesele (2011) is based on the linear fusion presented by Curless and Levoy (1996) (cf. Section 3.3.1). The choice of voxel size is dependent on the logarithmic relationship between the triangle size and the size of the root node. The triangle is obtained from a single disparity map. Making the choice dependent on an error model is more sound, especially for complex configurations.

For the fusion of point clouds, it is important to merge similar data. For data with strongly varying quality, data with lower quality should be considered for filtering. Octrees are suitable for such fusion and filtering, because algorithms run through the octree from low-resolution to high-resolution voxels (cf. Fig. 6.2). This allows the filtering of low-quality data by setting the nodes with further children to an unoccupied state.

For the fusion of spatial data, only those data that correspond to the same octree level should be considered. Even better is fusion at two neighboring levels for all points to avoid the quantization effects. Hence, spatial data that have at minimum half the quality, and at maximum twice the quality, are fused.

It is suitable to choose the voxel size for individual points with linear dependency on the point uncertainty σ_p :

$$\sigma_p < a v_s < 2\sigma_p \ , \tag{6.1}$$

where a is a regularization parameter and was empirically found to be suitable in the range of $[3, 6]$. For this, the runtime and memory consumption have to be considered. For uncorrelated data, e.g., from multiple sensors or when many images are taken with different cameras, it can be suitable to raise the value of a. For the evaluation of the experiments described in Chapter 7, $a = 6$ is used. For very large models that can disregard the smallest details, $a = 3$ is used for a high runtime performance.

In Section 4.3, two kinds of volumetric processing are presented: the reconstruction of a probabilistic space, for data fusion and the reconstruction of a space of optimized points. The optimized point cloud space renders volumetric consistency checks feasible. Direct triangulation of the probabilistic space with Marching Cubes is unfeasible in data structures with dynamic voxel size (Fuhrmann and Goesele 2011). Therefore, a transformation to a volumetric point cloud is suitable. For the probabilistic and the point space, the same octree depth is used, particularly because 3D points in the point space are optimized and fused by the probabilistic space. For smoothing the optimized point cloud, increasing the voxel size of the point space relative to the probabilistic space may be a suitable method.

When fusing in a probabilistic space, those voxels that are intersected by the line of sight in a specific area are considered (cf. Section 4.1). Such voxels can be efficiently accessed through ray tracing. To this end, a 3D point is integrated into the octree space, and the line of sight through the voxel space is traced in a specific area depending on the uncertainty. Octrees are suitable for ray tracing because only two cubic intersection points have to be calculated. The neighboring voxel assumption follows from the exit point of the ray intersection. The voxels considered are additionally given to their binary occupied state, a probability of being assigned to lie behind the surface, as described in Section 4.3.

In the point space, the points are optimized considering the voxels of the probabilistic space. Though a Gaussian regression, a new optimized point is calculated including the surface probability (cf. Section 4.3). This point is then propagated to the point space octree. Hence, the voxels are given additional information regarding the surface probability and indices for the cameras located at the position where the point was seen. The latter information allows for a probabilistic filtering of the occupied voxels. An example for both spaces on two different levels is presented in Fig. 6.3.

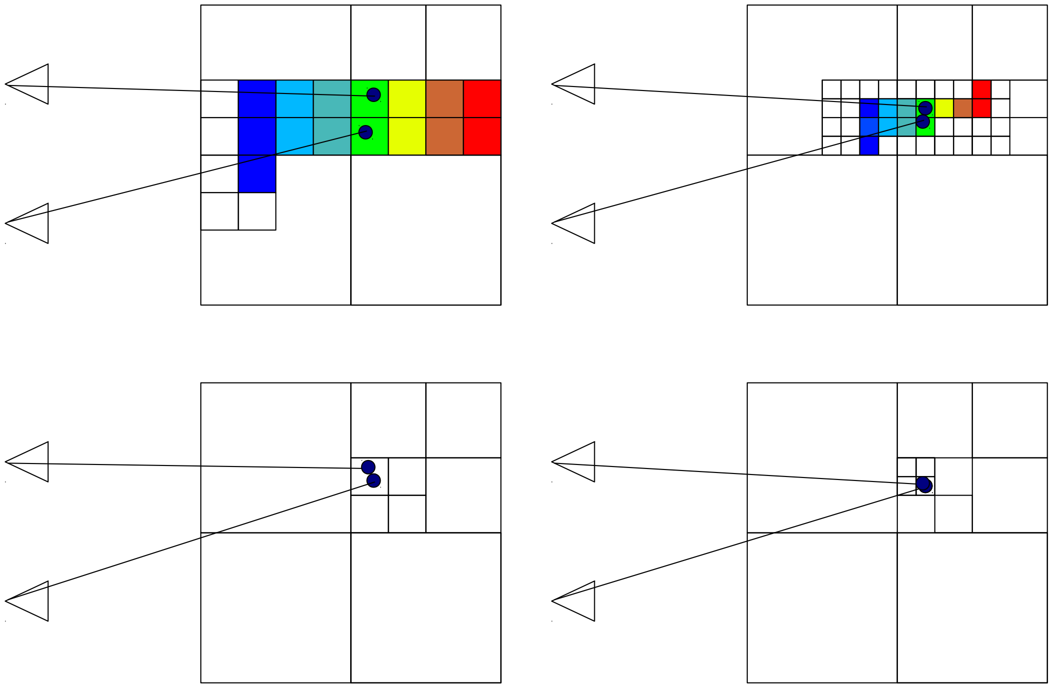


Figure 6.3.: The upper rows show a probabilistic octree space. The left and right images show the fusion for the same octree on different levels. The level of the octree depends on the quality of the 3D point. The lower row presents the point space whose points were fused in the probabilistic space. Using the high-resolution points shown in the right image, the low-resolution points shown in the left image are filtered.

Filtering in terms of the geometric consistency is discussed in Section 5.4 and an adaption, taking into account the surface probability, is also given. Checks through the use of ray tracing can be efficiently handled using an octree. The surface probability obtained using the method proposed in Section 4.3 is crucial for filtering the outliers.

Multi-resolution integration becomes more complex as additional varying surface probability points appear in different solutions. It is hard to determine whether low-resolution points with a higher surface probability can provide important information. In general, an octree is visited through the low-resolution to high-resolution filtering the low resolution points. Hence, for both probabilistic and point spaces, only high-quality data from a high-resolution area remain unfiltered. In Fig. 6.4, ray tracing is shown for the filtering of low-resolution areas.

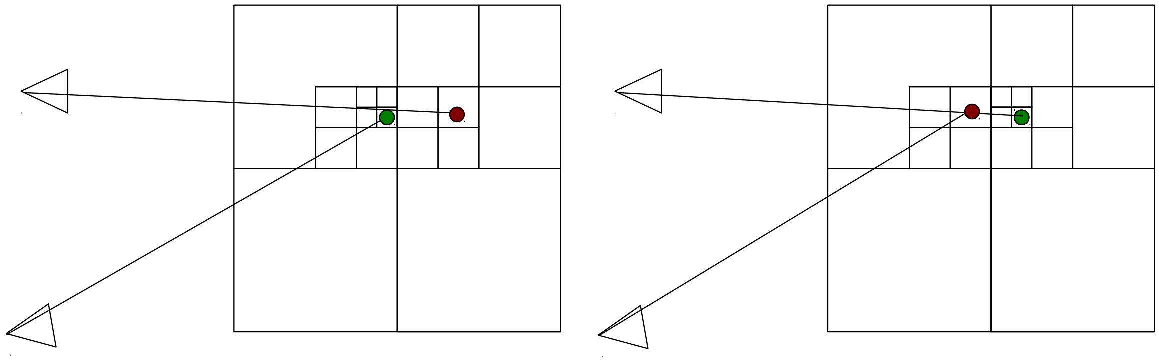


Figure 6.4.: Filtering using ray tracing. The points from the occupied voxels are cast to the cameras. When detecting conflicts with the occupied voxels, the low-resolution points are filtered (red).

For filtering spatial data, the important question regarding multi-resolution integration is whether low-resolution areas can contain more accurate information than high-resolution areas. This has to be discussed in more detail, leading to a consideration of the scale-space aspects.

6.3. Scale Space Analysis

Scale space theory is concerned with multi-scale signals, e.g., for handling image structures of different sizes, i.e., at different scales. Point clouds at different levels of a spatial data structure offer information from varying 3D scales (cf. Fig. 6.5).

There has been a lot of work on scale space theory considering 2D images or 1D signals. Such signals are mainly convoluted with specific kernels in varying configurations leading to artificial scale space representations. Images, e.g., can be smoothed using a Gaussian filter with varying variance leading to a Gaussian-scale space that can be used for feature extraction (Lindeberg 1993).

For surface modeling, regularization in a 3D space leading to a four-dimensional (4D) regularization problem is worth considering. In Section 6.2, an approach for filtering data

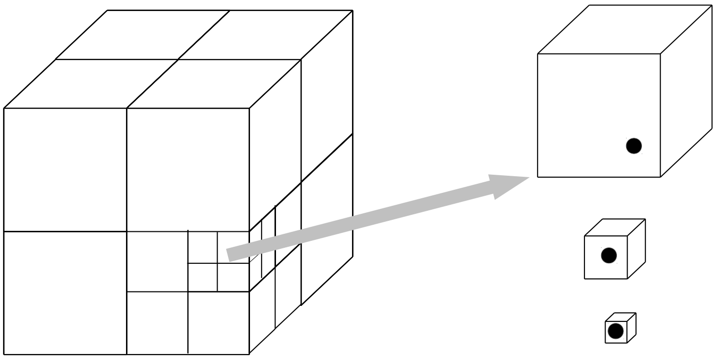


Figure 6.5.: Three 3D points with different quality on three different octree levels. It is important to consider whether low-resolution points should influence the 3D information of high-quality points. This can be seen as a 4D regularization problem.

on different levels is described. Low-resolution points are filtered given high-resolution points. Considering the scale space, the surface probability should also be considered.

The regularization of 3D surfaces from multi-resolution octrees was previously mentioned by Fuhrmann and Goesele (2011). As described in Section 6.2, their method is based on a linear fusion and a volumetric representation of point clouds. For the weight and distance functions from Eqs. (4.1) and (4.2), the weighting is defined considering octree level l and the coarser level $l-1$:

$$\tilde{d_l} = \frac{d_l w_l + d_{l-1}(\tau_0 - w_l)\ \min(1, \frac{w_{l-1}}{\tau_0})}{w_l + (\tau_0 - w_l)\ \min(1, \frac{w_{l-1}}{\tau_0})} \ , \tag{6.2}$$

$$\tilde{w_l} = w_l + (\tau - w_l)\ \min(1, \frac{w_l - 1}{\tau_0})\ , \tag{6.3}$$

where τ_0 is a saturation threshold avoiding oversmoothing (Mitchell 1987). Additionally, a second confidence threshold τ_1 is used that allows voxels to be filtered where $\tau_1 \leq \tau_0$. The proposed regularization may also be suitable for probabilistic fusion, but further analysis of the probabilistic aspects is needed.

In the probabilistic framework proposed in Section 4.3, the regularization of Eq. (6.2) has to be adapted. The regularization of the weight function is not important because binary weighting is used (cf. Fig. 4.3). The voxel probabilities p_l on level l should be fused with the neighboring levels. The probability votes for the voxel to be completely behind a surface. If a voxel at level l lies completely behind the surface, it does not necessarily imply that the voxel at the coarser level $l-1$ also lies completely behind the surface. Nonetheless, the probabilities are correlated, and it could be advantageous to determine whether both voxels tend to lie behind the surface:

$$\tilde{p}_l = p_l \; p_{l-1} \; . \tag{6.4}$$

Results of experiments regarding this first intuitive probabilistic idea of a 3D scale-space evaluation are described at the end of Section 7.7. An accurate framework needs further theoretical work for a 3D scale space. This is an open problem that is beyond the scope of this thesis.

Another alternative way to consider multiple scales without a complex 3D scale space theory would be to estimate disparity maps from images of varying scale (cf. Fig. 6.6). In Section 5.3.3, it was shown that the disparity quality varies from the subpixel level to multiple pixels. Downscaled images can lead to smoother depth maps that tend to have a higher quality. The method presented in this thesis allows for the input of multiple disparity maps from a single image. The algorithm automatically chooses the pixelwise voxel depending on the camera and disparity parameters. The experiments are described in Section 7.7.

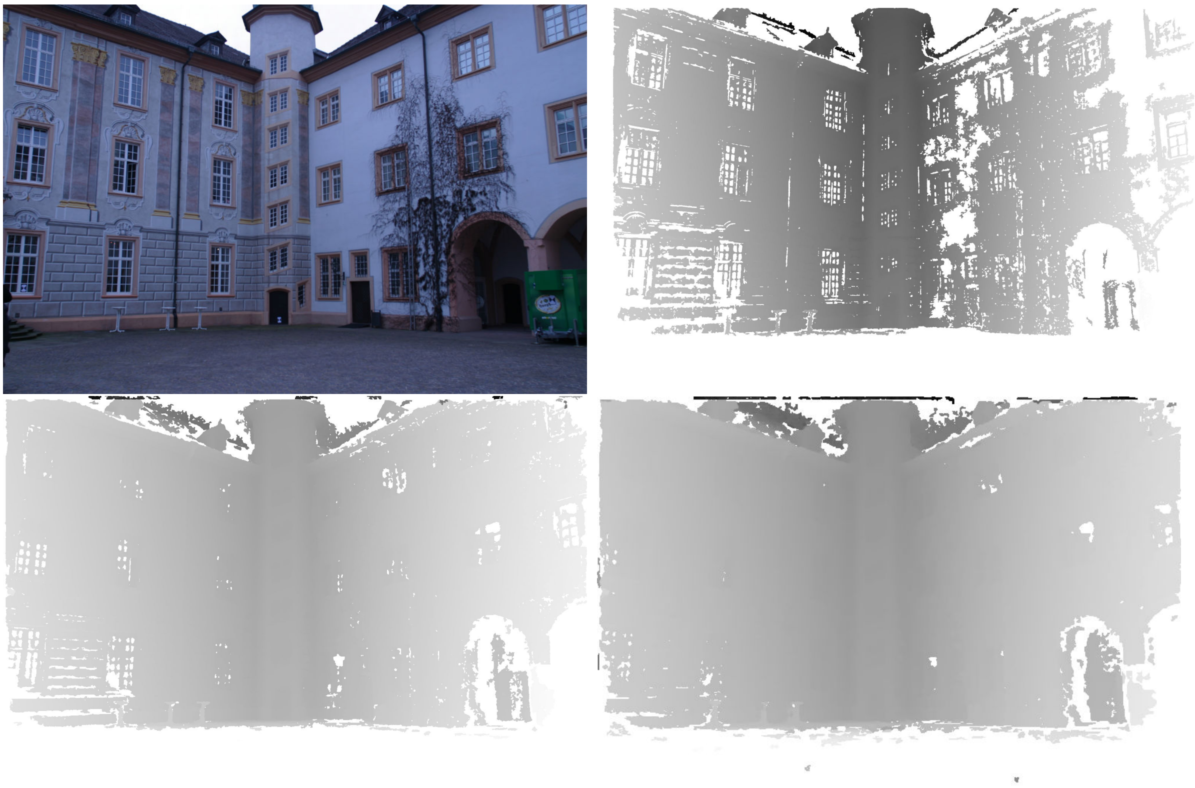


Figure 6.6.: Three disparity maps obtained by images downscaled with factor 1, 2 and 4. In the upper left the original image from the Ettlingen30 sequence is shown. Low resolution image lead to higher densities and can even have better accuracy.

6.4. Meshing

This thesis provides a scalable method resulting in accurate point clouds. For complete 3D modeling, the transformation into connected polygons defining a 3D surface is of high importance.

To guarantee a scalable 3D reconstruction, a local meshing method has to be devised. Additionally, the meshing method has to be fast, but there is no need to optimize the point cloud because this has been previously conducted during the fusion. Local meshing described in Section 2.2.3 is suitable for these conditions. Curless and Levoy (1996) use Marching Cubes for fast polygonization of volumetric spaces. Yet, this does not allow for multi-resolution processing. Therefore, the propagation to a volumetric point cloud space is presented in Section 4.3 that allows for directly processing point clouds.

Fuhrmann and Goesele (2011) proposed a global tetrahedralization for polygonization of volumetric point clouds that is clearly limited in scalability. It is particularly important not to violate the core idea of local processing in combination with the Divide and Conquer assumption (cf. Section 5.2). For this, the locality of the processing has to be limited, i.e., the overlap between subspaces has to be twice the maximum size of the optimization area.

In particular, the incremental algorithm by Bodenmüller (2009) can be adapted considering a varying neighborhood size. Such adaption of the varying distances is part of this thesis, and is shown in Fig. 6.7. The optimized point cloud allows for an accurate estimation of normal vectors from the neighboring points. Considering point P, an incrementally built mesh is projected onto the local plane perpendicular to the normal vector. Only those triangles made up of points lying in a spatial neighborhood defined by a factorized voxelsize av_s from the voxel space are connected. In the experiments, $a = 5$ is used for limiting the neighborhood.

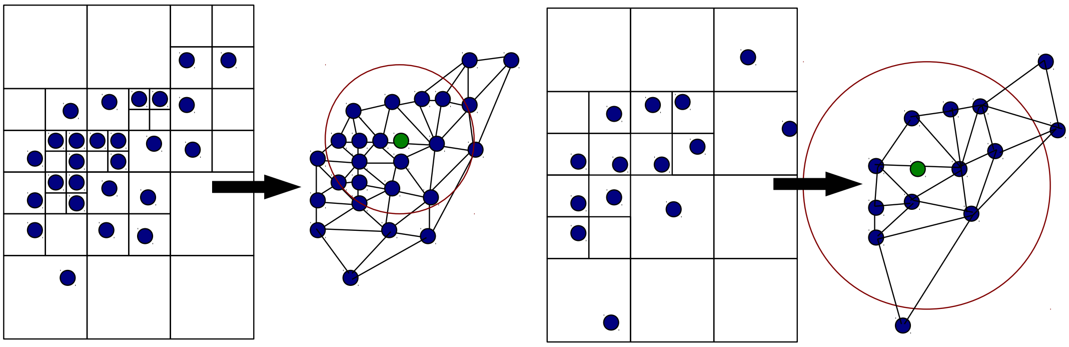


Figure 6.7.: Meshing considering the variable resolutions of points from the octrees. The green point is added incrementally to the initial mesh. The point is connected to all points within the red circle, as described in Section 2.2.3 and shown in Fig. 2.6. The size of the circle is adapted to the factorized voxel size considering the density at the respective resolution.

Chapter 7.

Evaluation and Validation

After presenting the methods for 3D surface reconstruction in Chapters 4, 5, and 6, an evaluation of the results from various datasets is described in this chapter. Additionally, for a qualitative visual comparison of the resulting 3D surface models, a numerical quantitative evaluation has become a necessity. To this end, datasets with ground-truth 3D surfaces are needed. Unfortunately, for large-area 3D surface reconstruction, no datasets with ground truth are available. Nevertheless, there are sets of images with ground-truth surfaces that have either small numbers of images (Strecha et al. 2008) or low-resolution images (Seitz et al. 2006), which is discussed in Section 7.1.

This thesis has a practical goal: the scalable fusion of disparity maps to 3D surface models. Therefore, the number of possible disparity maps is not limited, and the resolution of the disparity maps can be extremely high. In Section 5.2, a Divide and Conquer method was presented that allows for the reconstruction of surfaces without limits to the scalability. The processing of large datasets using the method presented in this thesis, and an evaluation of the results, are presented in Section 7.2.

A multi-resolution computation, as described in Section 6, allows for the fusion of spatial data with varying quality. The evaluation and validation of this extension is discussed in Section 7.3. To this end, multiple 3D surface models resulting from complex real-world image configurations are shown.

Chapter 4 introduces a novel probabilistic framework for the fusion of noisy disparity maps. This allows for an optimization of the point clouds from the disparity maps, and the extraction of surface probabilities. This surface quality again allows for the probabilistic filtering of outliers, thereby avoiding multiple surfaces. An evaluation of these extensions is described in Section 7.4. Additionally, a numerical and visual comparison with the method by Fuhrmann and Goesele (2011) who proposed a similar multi-resolution approach without a probabilistic framework or consideration of a principled error model, is provided.

In Section 5.3.3, a statistical learning scheme allowing for an estimation of the disparity error was presented. This is based on a feature extracted from disparity maps that describes the local oscillation behavior. The improvement in 3D surface reconstruction considering the estimated disparity errors is given in Section 7.5.

In general, for the generation of 3D surface models, the process chain described in Section 2.2, along with the extensions presented in this thesis, is used. The disparity maps are estimated using SGM with Census matching costs and constant parameters. The images are downscaled by a factor of 2, as most of the images are taken by consumer cameras and do not have pixelwise uncorrelated values, e.g., caused by a small pixelsize and a Bayer Pattern in the cameras. How downscaling influences the quality, and whether the loss in quality can be used for scale-space modeling, is discussed in Section 7.7.

For fusion of the disparity maps, the probabilistic framework presented in Section 4, along with the multi-resolution adaptation described in Section 6, is used, taking into account the filtering step presented in Section 5.4. For the error model, the stereo error model with dynamic disparity error is used (cf. Sections 5.3.1 and 5.3.3).

Only the image registration uncertainty (cf. Section 5.3.2) and scale space aspects (cf. Section 6.3) are not considered, which are discussed in Sections 7.6 and 7.7 as final points of this chapter.

7.1. Datasets

It is important to evaluate the quality of the surfaces in terms of the accuracy and completeness. Because humans can use additional semantic information, a visual inspection can help assess the accuracy, and even further, the completeness. Nonetheless, a numerical comparison considering ground-truth data is essential because a visual comparison has a limited guarantee particularly concerning the minor differences. The Middlebury multi-view challenge (Scharstein 2014a) has been quite beneficial for the development of 3D surface reconstruction methods by providing MVS data from laboratory configurations that can be numerically evaluated against ground-truth data. Seitz et al. (2006) introduced multi-view image datasets registered with a laser-scanned surface model. For each of the two models shown in Fig. 7.1, three sets containing a varying number of images are provided.

By experiments in this thesis, it was found that the provided calibration data are not accurate. However, a re-registration leads to an inconsistency of the coordinate systems. Furthermore, a laboratory configuration is unique as the images do not have varying quality typical for real-world data. Nonetheless, an evaluation is important, and the scalable method presented in this thesis has been evaluated on the Temple and Dino datasets (cf. Section 7.5).

Strecha et al. (2008) provided the first numerical evaluation on real-world data for small image sets. The images were registered with LIDAR data, and the accuracy was evaluated relative to the uncertainty of the LIDAR measurements. Unfortunately, the evaluation is no longer provided (Strecha 2014). Nonetheless, for two small sequences, the ground truth is available and can be used for an evaluation of the 3D surface models.

The first dataset, called EttlingenFountain, consists of eleven images showing a fountain, a stone wall, and the ground from a short distance. The second dataset, called Herzjesu8, consists of eight images showing a building from the front. Screenshots of both ground-truth surfaces and example images are given in Fig. 7.2.

The objects visible in these images have similar distances to the camera and are well

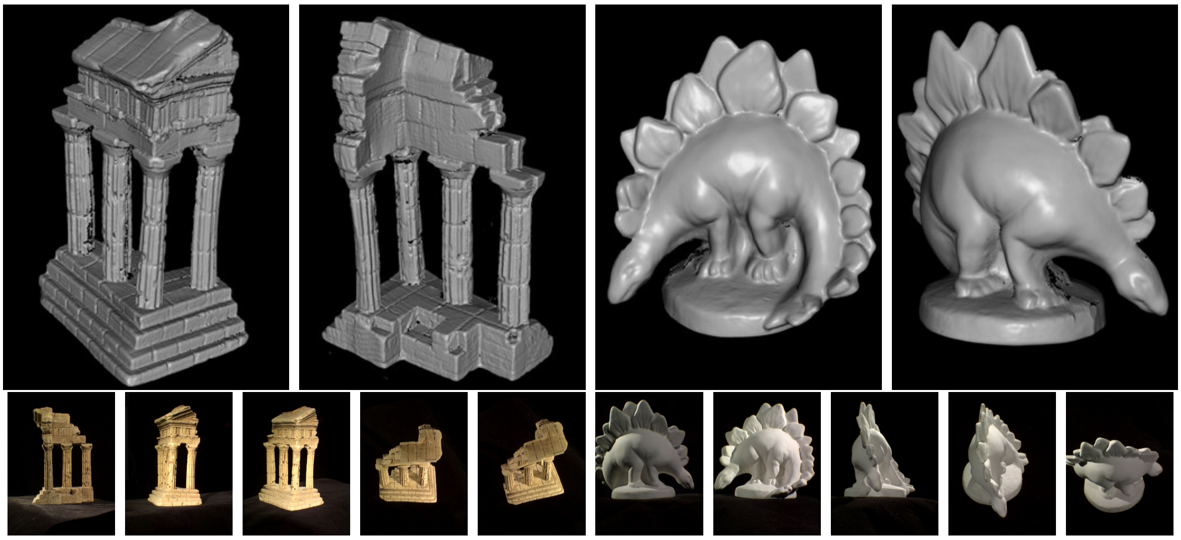


Figure 7.1.: Top: Ground-truth data for the Middlebury Temple and Dino from laser measurements. Bottom: Five images from the complete sets. The six image sets provided contain varying numbers of images with a resolution of 640 × 480: TempleSparseRing (16 images), TempleRing (47 images), Temple (312 images), DinoSparseRing (16 images), DinoRing (48 images), and Dino (363 images)

.

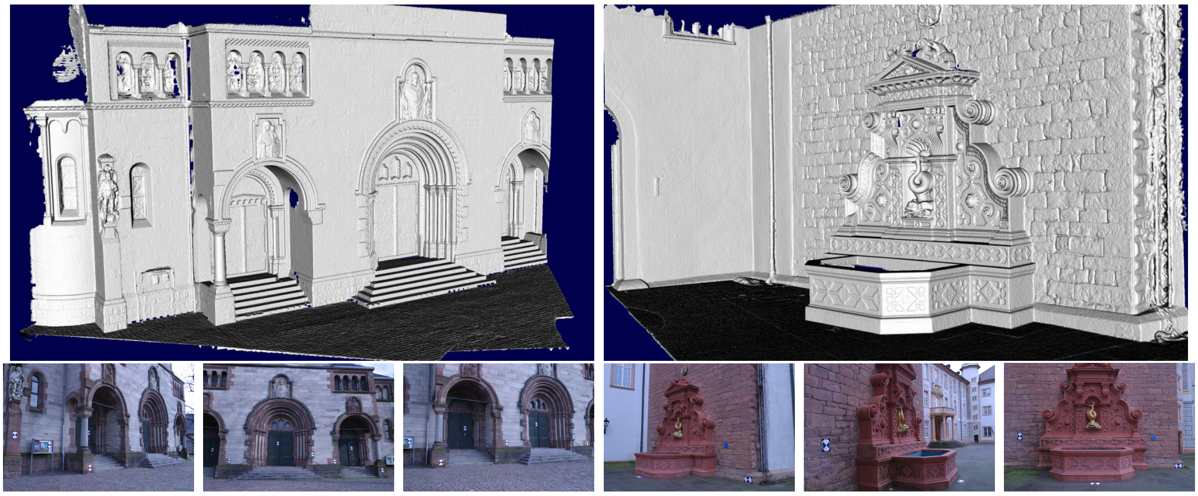


Figure 7.2.: Ground truth for real-world data. Left: Herzjesu8 ground truth. Right: EttlingenFountain ground truth. Bottom row: three of the original eight and eleven images.

textured. The configuration includes varying perspectives, but from similar directions.

By showing that it can model very large scenes in terms of the completeness of the surfaces, the scalability of 3D surface reconstruction is visually demonstrated. To this end, some large datasets were obtained that also partly consider high-resolution images. The datasets are shown in the following sections and Appendix A.

7.2. Large Models and Large Scenes

The main focus of this thesis is the 3D surface reconstruction from large image sets that may contain high-resolution images. A suitable way to evaluate the results is to analyze the completeness of the 3D surface models. This must be done visually because no ground truth is available and is very hard to obtain. For the evaluation, two different types of large-scale datasets were obtained and processed using the pipeline described in Section 2.2.

The camera configuration of the first dataset is shown in Fig. 1.1. The resulting colored and shaded 3D surface of this complete model is provided in Fig. 7.3. The image set consists of 823 images captured from the ground and from a UAV. The problems and potential in reconstructing such complex configurations are presented in Figs. 7.4 and 7.5.

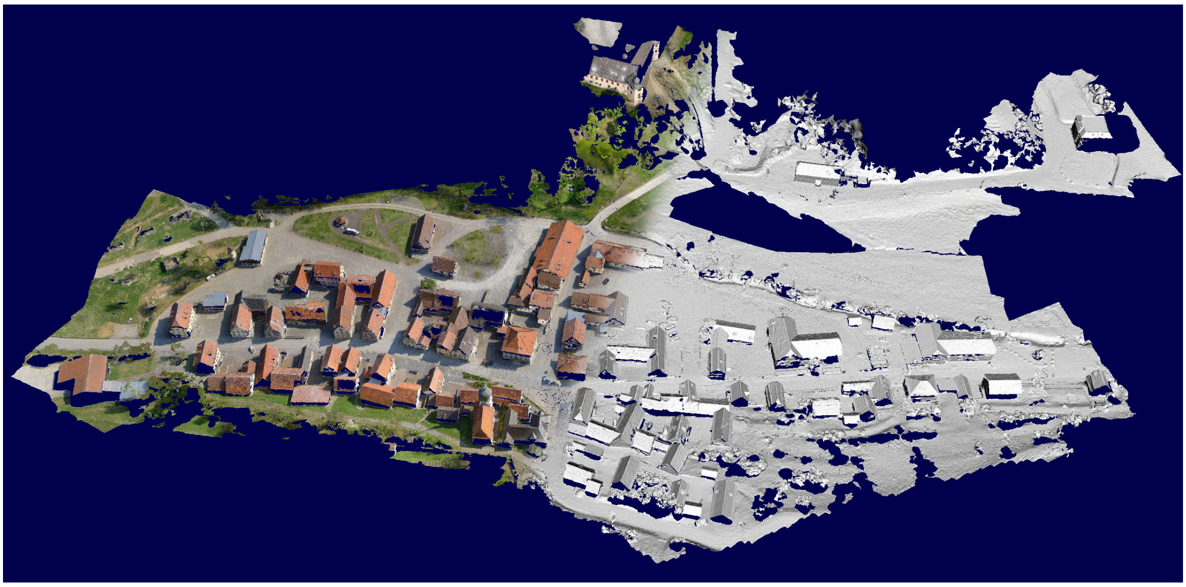


Figure 7.3.: Half-shaded and half-textured 3D model from the Bonnland dataset. The model is based on 823 images captured from the ground and a UAV. The surface consists of one billion triangles. The complete model was processed in a couple of hours using hundreds of CPU cores. The completeness of the model demonstrates the suitability of the method presented in this thesis. The zoomed-in parts of the model are shown in Figs. 7.4 and 7.5.

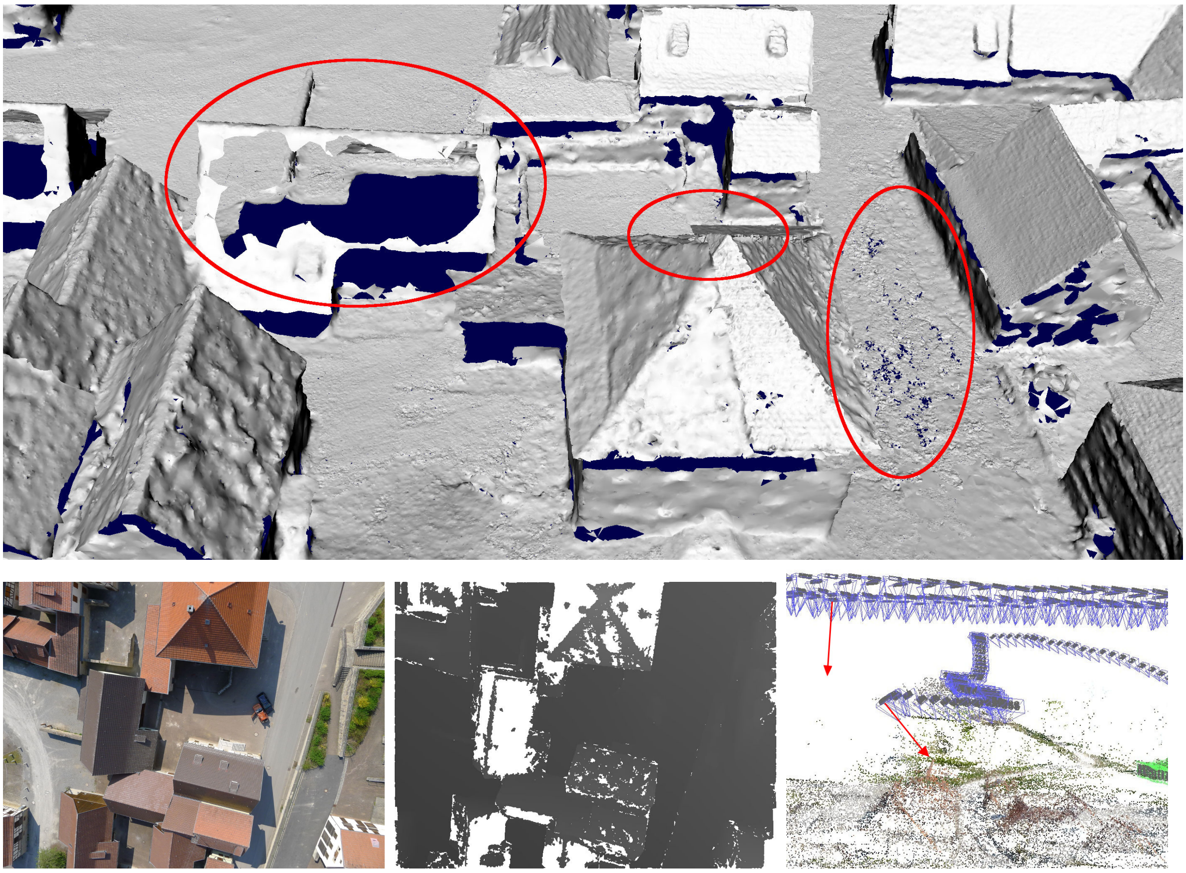


Figure 7.4.: The upper part shows some poorly reconstructed parts of the large Bonnland model (cf. Fig. 7.3). It is clear that parts of the roofs and the walls are missing. For the lower-left image, the corresponding disparity map is shown in the lower-middle. The disparity map has holes because of slanted surfaces. The surface modeling of the roof shown in the middle of the upper image has an offset. The area on the ground next to the building also shows reconstruction errors in the pitted surfaces. This is due to registration uncertainties because, for this part of the scene, strong differences in perspective exist, as demonstrated in the lower-right image. The red arrows represent the viewing direction of the upper and lower cameras, and hence, the strong differences in perspective.

The images show various details of the scene. The registration takes about two hours on a system with 24 CPU cores. The disparity map estimation and fusion of the 823 disparity maps were computed on a cluster with about 100 cores in less than six hours. In particular, 3D surface reconstruction by local fusion of the disparity maps presented in this thesis takes about three hours. For this, the reconstruction space was automatically split into thousands of subspaces with varying size, which were processed in parallel. The

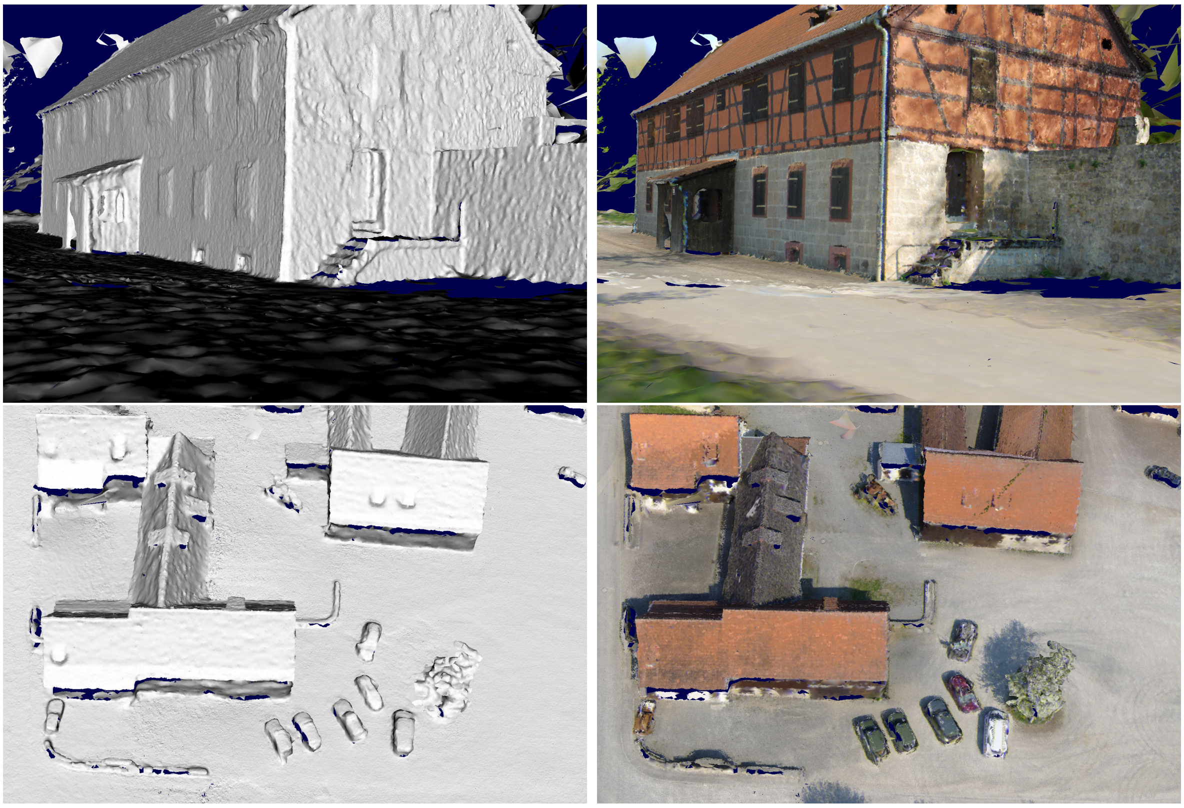


Figure 7.5.: Shaded and textured zoomed-in parts from the Bonnland model (cf. Fig. 7.3). The combination of images from the air and the ground show fine details without a dominant direction. The images captured from the ground even allow for the reconstruction of the overlap from the windows and roofs.

runtime can be further reduced using more cores because the 823 images and thousands of submodels had to be sequentially processed on 100 cores. The final surface consists of almost one-billion triangles.

Visualization of the large models is not possible on standard graphic cards. To visualize the complete model in Fig. 7.3, the model was processed at a quarter-resolution of the images instead of a half-resolution. In this case, the runtime for disparity map fusion falls to about a one-quarter of the runtime given above.

In general, the surface of the 3D model is complete for all areas captured by two images at minimum, in addition to some smaller regions. The missing regions are generally not available in the disparity maps (cf. Fig. 7.4). This incompleteness is due to highly non-fronto-parallel areas or non-static objects such as vegetation. The image configuration is quite dense in general. Nonetheless, because the cameras partly capture the scene from a small distance from the objects, large-perspective deformations may appear in the image pairs, which cannot be handled by SGM.

Fig. 7.4 also shows parts of the model that are not smoothly reconstructed. This is caused by uncertainties in the image registration, which are not considered here, but are discussed in Section 7.6. The lower-right part of Fig. 7.4 shows the image configuration of the specific area. The cameras were partly oriented with large differences in perspective, leading to a large uncertainty in the camera parameters.

The specific characteristic of this image set is the combination of images captured from a UAV and from the ground. This leads to a strongly varying level of detail depending on the local camera configuration. Fig. 7.3 shows an overview of the complete model. In particular, the images from the ground allow for the reconstruction of small details in the 3D surface, as demonstrated in Fig. 7.5. While the reconstruction of the ground and the roofs can also be achieved through 2.5D modeling, the 3D surface modeling presented in this thesis allows for the reconstruction of small details independently from a dominant camera direction.

The overall dataset was processed at one-quarter image resolution to be able to visualize it. When processing the half-resolution images, which is usually conducted, even more details are produced (cf. Fig. 7.6). The visualization of smaller parts of the complete high-resolution model can be achieved using standard graphic cards. Fig. 7.6

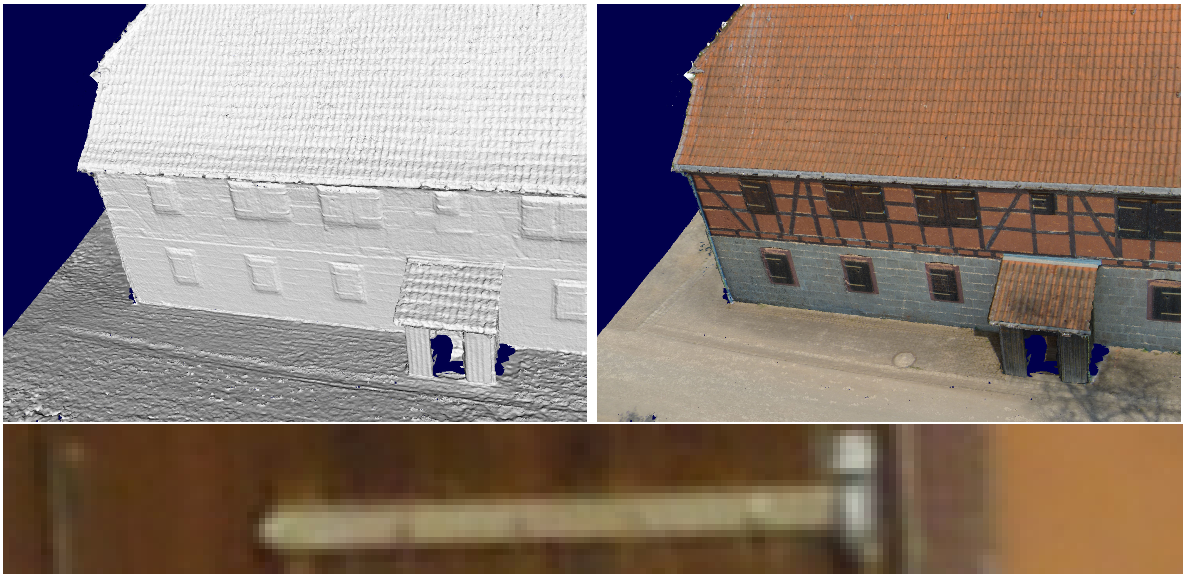


Figure 7.6.: Part of the model shown in Figs. 7.3 and 7.5 with a higher resolution. This small part of the model alone consists of about 15-million triangles. Even small details such as the hinges in the windows are clearly maintained on the surface. The bottom image shows part of an image from the set, illustrating the details of the hinges at a minimum distance. The size of the image space is only a couple of pixels.

shows the building from Fig. 7.5 at a higher resolution. In a 3D model, small details are maintained that correspond to only a couple of pixels in the image space.

The second large dataset (cf. Fig. 7.7) consists of 235 images from three different cameras: two with 10 MP resolution, and one with 36 MP resolution.

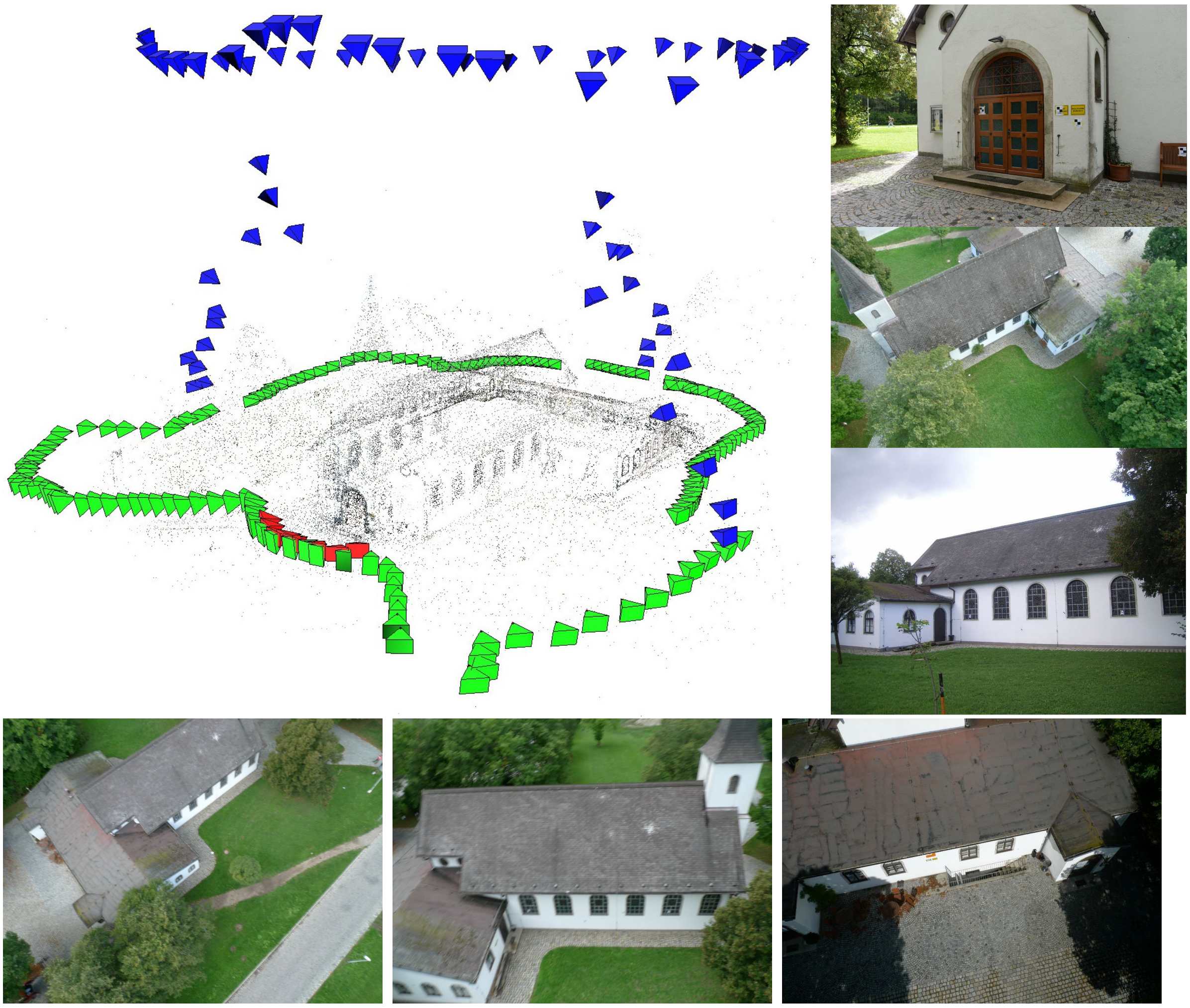


Figure 7.7.: Unikirche dataset. This image set consists of 235 images. Some images were captured from a UAV (blue), and some from the ground (green), using 10 MP cameras. Additionally, images of the door of the building were acquired from a 36 MP camera (red). The sets show strongly varying perspectives and distances to the object. In addition, the lighting conditions changed and some of the images are blurred.

The successful processing of this dataset shows that, along with the ability to deal with large scenes, the proposed method produces, an accurate reconstruction of the surfaces from high-resolution images in possibly difficult configurations. The configuration of the

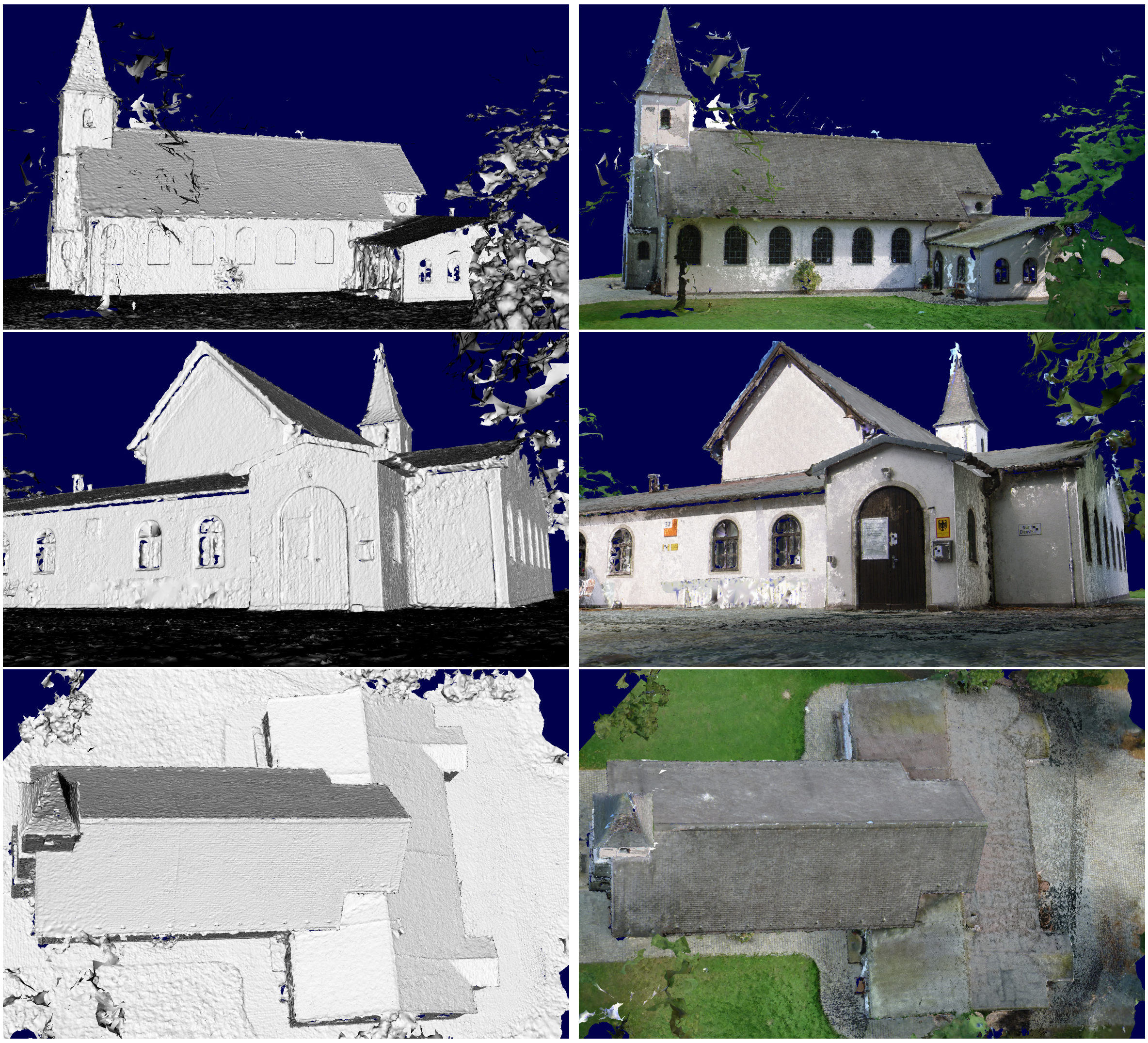


Figure 7.8.: Shaded and textured parts of the 3D surface model obtained from the image set shown in Fig. 7.7. The surfaces were obtained from 10 MP images taken from the air and on the ground. Moving objects such as vegetation were only partly reconstructed.

cameras and resulting parts of the 3D model are shown in Figs. 7.7, 7.8, and 7.9.

The complete image set contains over six-billion pixels. The processing of the 3D modeling was conducted similarly as the Bonnland dataset from Fig. 7.3. The complete process takes about 10 hours.

The resulting surface contains over four-billion triangles representing varying resolutions. The reconstruction space was automatically split into subspaces that differ in their relative size by up to a factor of 128. All parts were processed in parallel on a cluster

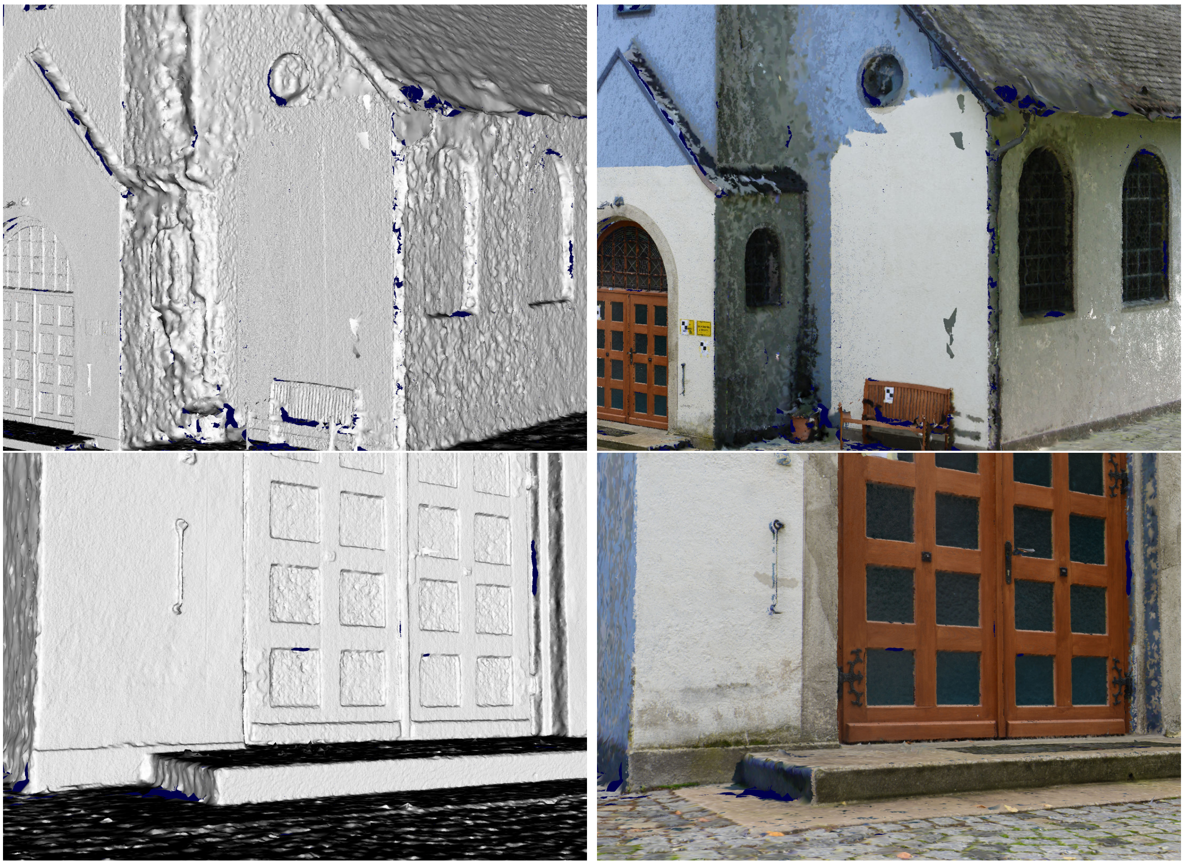


Figure 7.9.: Shaded and textured high- and low-resolution parts of the Unikirche model. The area shown in the upper images combines high-resolution parts with low-resolution parts captured from a larger distance with a lower image resolution. The difference in the texture results from radiometric differences because the images were captured by different cameras under very different lighting conditions. The area shown in the bottom images was acquired with a high-resolution camera. Small details such as the doorknob were reconstructed.

with about 100 CPU cores. When binary coded, the overall model has a size of more than 5 GB. Fig. 7.8 shows the larger parts of the model to illustrate its completeness.

The level of detail obtained depends on the individual surface parts. In particular, the distances to the cameras and the image resolutions have an effect on the surface quality. The details at the front of the building show the general power of image-based 3D modeling (cf. Fig. 7.9). It is clear that the quality strongly differs as parts are captured from lower-resolution cameras at a larger distance. Despite the strong radiometric differences, which are clear from Fig. 7.9, the surface is complete, and even very small details were obtained.

A typical problem for this kind of configuration arises from images from the ground with an untextured background from the sky. On the border of the sky, ghost surfaces, which are very smooth and hence not classified as outliers, can appear in the disparity map. This problem is discussed further in Section 8.3.

7.3. Multi-Resolution

The multi-resolution method described in Section 6.2 allows for an efficient fusion of spatial data with varying quality (Kuhn et al. 2013). Additional to the fusion of noisy data, it is important to filter data with a lower quality. Octrees allow for the processing and representation of spatial data on different levels depending on the quality of the measurement. A multi-resolution representation is also suitable for 3D consistency checks. The results provided in Section 7.2 already show the successful processing of data with varying spatial quality. Nonetheless, it is useful to discuss a multi-resolution fusion in more detail.

A characteristic image configuration leading to highly differing qualities contains varying distances to the object (cf. Fig. 7.10 (left)). The error model from Eq. (5.14) shows that the error increases quadratically with the distance. The images are acquired from the same object as the Unikirche dataset, but in a different configuration. The 54 images show only the front of the building at varying distances. During the multi-resolution reconstruction, only the highest-quality data in the specific areas is considered. The surface maintains small details such as the hinges and the hook.

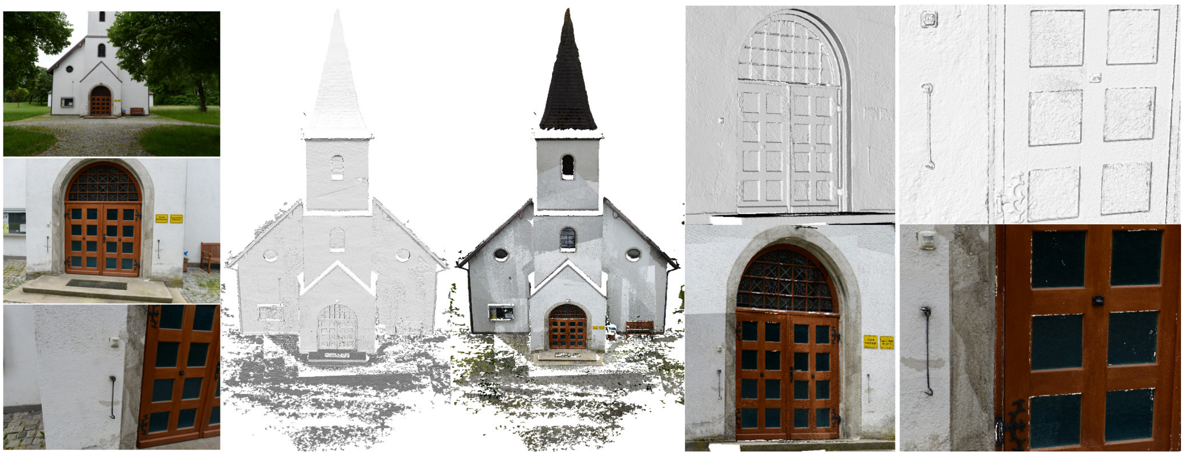


Figure 7.10.: Textured and shaded 3D model reconstructed from 54 images (left column). The images were captured at varying distances to the object. The multi-resolution method retains only the best qualities during the 3D surface reconstruction. The largest voxel is about 200 times larger than the smallest.

For a further visual inspection, the Ettlingen data from Strecha et al. (2008) were all processed into a single model. It consists of the Ettlingen10, Ettlingen30 and EttlingenFountain datasets. To this end, the images from Ettlingen10, Ettlingen30, and EttlingenFountain were all registered using the method presented in Section 2.2.1. The resulting configuration consists of qualitatively varying data, as the distance to the objects differs by a factor of around 20. The resulting 3D model is shown in Fig. 7.11. The surface area above the fountain (left-redbox) combines high-quality parts derived from the fountain sequence and lower-quality parts generated from the Ettlingen30 sequence. The method described in this thesis also allows for a complete and consistent modeling in the border area at different resolutions.

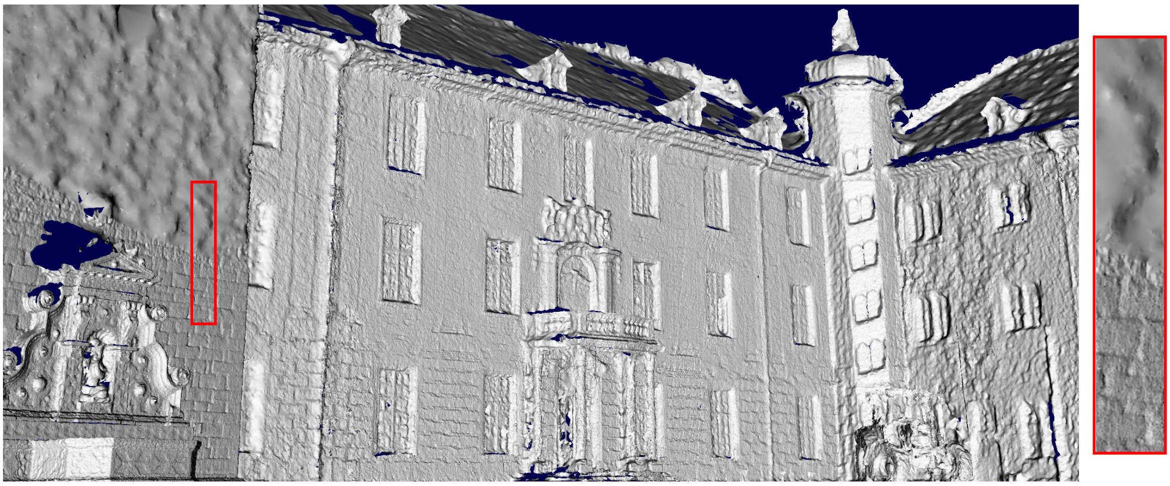


Figure 7.11.: Result from the Ettlingen datasets. The image shows a combination of the Ettlingen10, Ettlingen30, and EttlingenFountain images in a single model. The region to the right of the fountain shows a transition between very different resolutions (cf. also zoomed-in in the right – top: low versus bottom: high).

Another multi-resolution approach was presented by Fuhrmann and Goesele (2011). As with the fusion method, one particular difference between their method and the method presented in this thesis is the choice of the octree depth for fusing similar data (cf. Section 6.2). Fortunately, an implementation of this method is available for a direct comparison. Experiments show, that this implementation does not allow data to be processed with extremely varying distances, e.g., the image set shown in the Results chapter (cf. Fig. A.3). For the model given in Fig. 7.11, the results look rather similar. Hence, the extension of the probabilistic fusion method introduced in this thesis has to be discussed in further detail.

7.4. Probabilistic Fusion

In Section 4.3, a novel probabilistic framework for the fusion of spatial data is presented. This framework allows for a derivation of surface probabilities also for the case of disparity map fusion (cf. Section 5.3). To this end, the framework from Curless and Levoy (1996) for the linear fusion of cumulative distance functions is probabilistically interpreted to derive surface probabilities. Furthermore, a novel Bayesian fusion method considering Gaussian uncertainty is devised. The derived surface probabilities can be used for filtering local outliers by means of geometric consistency checks.

Curless and Levoy (1996) proposed a linear weighting function that penalizes surfaces lying behind the measurement, i.e., the estimated disparity (cf. Fig. 4.2). In this thesis, a binary weighting is used, which only considers values near the measurement. This is probabilistically sound and reduces the memory requirements because less voxels have to be processed.

The probabilistic fusion is modified in two ways: a Gaussian uncertainty is assumed instead of uniform uncertainty and a Bayesian fusion of surface probabilities is used instead of a weighted sum. In this section, the extensions are qualitatively evaluated on several datasets through a visual comparison. The EttlingenFountain and Herzjesu8 datasets are not really suitable at all, because they consist of rather simple image configurations capturing well-textured surfaces. Nonetheless, for these datasets a numerical evaluation is feasible and small parts of the scene are captured from difficult perspectives.

One crucial improvement, rendered possible by obtaining surface probabilities, is the potential to filter outliers in 3D space. In Section 5.4, a ray tracing based filtering method is presented, which has been extended for local filtering. This prohibits outliers disturbing the surface, which is necessary because a large amount of outliers tends to appear near the surface. Single outliers, or small clusters of outliers, can be disregarded later in the meshing. In all experiments, small islands of less than 100 triangles were filtered.

To compare the competing methods, it is important to make use of sound implementations. The implementation from Fuhrmann and Goesele (2011) is used for the evaluation of the probabilistic filtering method. Fuhrmann and Goesele (2011) extend the method by Curless and Levoy (1996) by a multi-resolution approach.

For complex camera configurations, such as shown in Fig. 7.12, multiple outliers appear in the 3D point cloud derived from the disparity maps generated by SGM. For the evaluation, the images had to be downscaled by a factor of 4 because the implementation from Fuhrmann and Goesele (2011) is very memory intensive, and even 180 GB was insufficient for half-resolution images. Furthermore, this implementation does not allow for sequential or parallel processing. A comparison of the two 3D surface models generated from the same image size shows the progress made by probabilistic filtering in 3D space (cf. Fig. 7.12). The ghost surfaces, e.g., on the roof and around the balconies, were successfully filtered through probabilistic geometric consistency checks.

The same tendency is evident when comparing the results for the Middlebury data

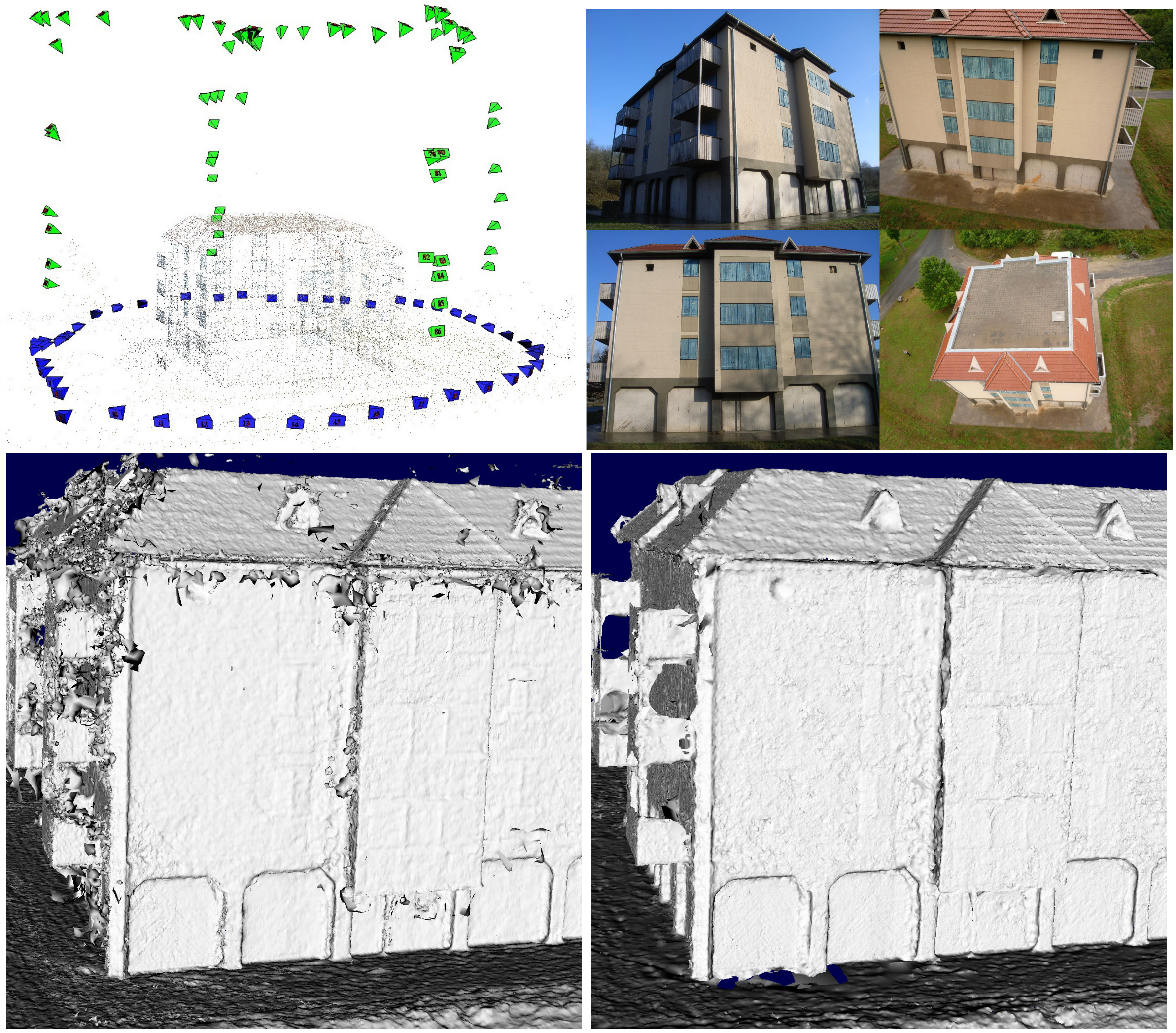


Figure 7.12.: Top: The registration (left) and four of 112 images with 10 MP resolution capturing a building (right) are shown. The building is acquired from strongly varying perspectives leading to outliers, particularly near surfaces. Bottom-left: 3D surface model resulting from the method by Fuhrmann and Goesele (2011). Bottom-right: Probabilistically filtered 3D surface model as presented in this thesis.

(Seitz et al. 2006). The image configurations of Temple and Dino are to that effect complex because the objects are captured from varying perspectives. In particular, the sparse and ring image sets create ambiguous surfaces (cf. Fig. 7.13).

Looking at Figs. 7.12 and 7.13, it is clear that probabilistic filtering is very suitable for 3D surface modeling from complex configurations. Nonetheless, in addition to outlier elimination, the fusion of noisy data considering the quality of the surfaces has also to

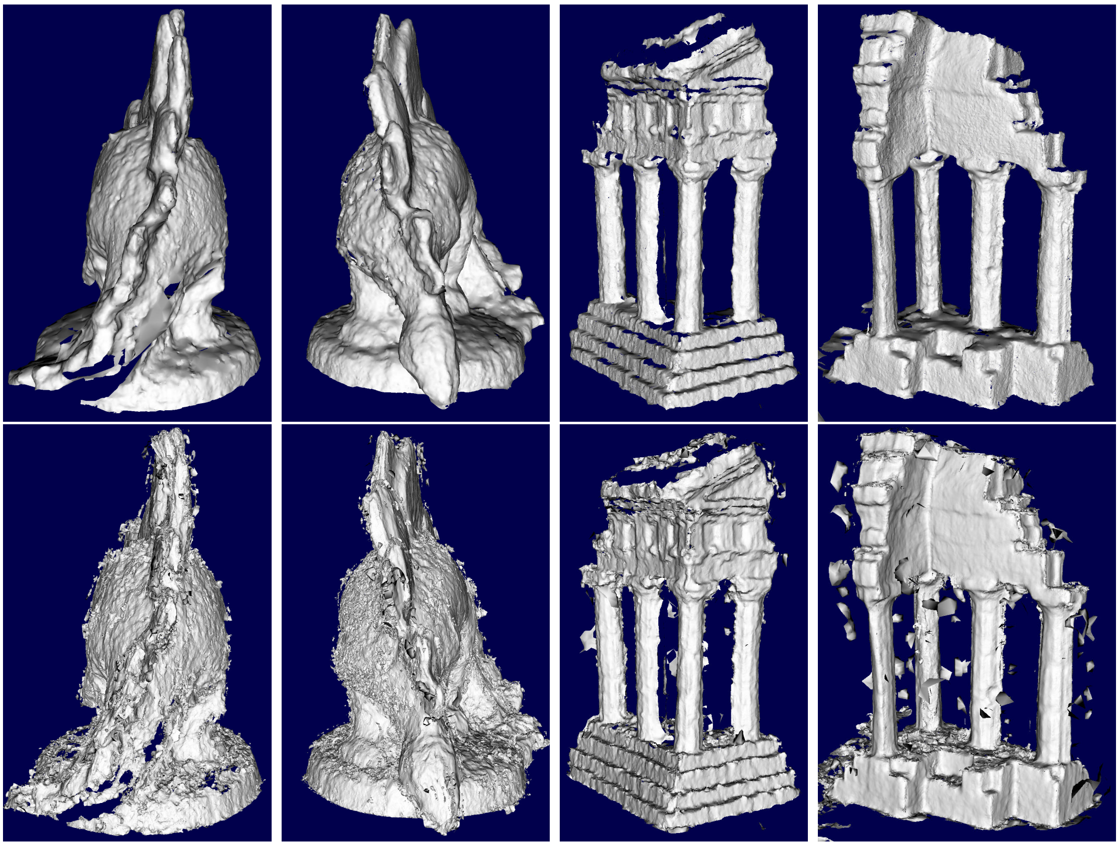


Figure 7.13.: Top: Results of the method presented in this thesis. Bottom: Results of the method by FUHRMANN and GOESELE (2011) based in the same disparity maps. From left to right: DinoSparseRing, DinoRing, TempleSparseRing, and TempleRing.

be discussed in terms of accuracy and completeness.

It is of particular importance to evaluate the reinterpretation of error modeling and fusion theory. CURLESS and LEVOY (1996) showed that a weighted mean fusion of linear cumulative distance functions is optimal in the sense of least squares. For this, Gaussian uncertainty and uncorrelated data are assumed. In Section 4.2, it is shown from a probabilistic perspective that the fusion can be interpreted as a fusion of correlated data that are uniformly distributed. From a probabilistic perspective, a novel framework was presented that fuses uncorrelated Gaussians by means of the Binary Bayes Theory.

For a numerical evaluation, the EttlingenFountain and Herzjesu8 sequences are suitable because ground-truth data from LIDAR measurements are available (cf. Section 7.1). However, for a special configuration with locally similar camera poses and well-textured objects, only a small increase in quality can be shown. As done by the evaluation by STRECHA et al. (2008), which is unfortunately no longer available, for all image pixels, the depth derived from the obtained 3D model is compared against the ground truth. To this end, rays are cast from the images to the model and the ground truth. The distance between the intersecting points corresponds to the absolute error of the surface. In the areas where ground truth is available, the unsigned distance is measured and presented as cumulative functions. An analysis of the signed distance function is not meaningful because all rays are cast from the sensor leading to negative error distances only for multiple surfaces.

In the original evaluation, the absolute error is compared considering the estimated uncertainty of the LIDAR data. Because this information is not publicly available, the absolute error in meters is evaluated in this thesis. To avoid the consideration of incorrect laser data, a template is provided to filter poor correspondences. The templates and error maps for one view for both datasets are shown in Fig. 7.15.

First, the method presented in this thesis is numerically evaluated against the method by Fuhrmann and Goesele (2011). The resulting 3D surface models for the EttlingenFountain and Herzjesu8 datasets are given in Fig. 7.14. The EttlingenFountain and Herzjesu8 datasets are not very suitable for showing the improvements obtained by the method presented in this thesis. Nevertheless, it is evident that difficult areas are more completely reconstructed. This holds, e.g., for the side of the fountain and the walls to the left, above, and right, because these areas are only captured in a non-fronto-parallel manner. The right wall area is not considered in the numerical evaluation because no ground-truth data are available (cf. Fig. 7.15). In addition, difficult areas on the side from the Herzjesu8 dataset are reconstructed well using the method proposed in this thesis.

One advantage of the method proposed in this thesis is that the 3D surface of the EttlingenFountain consists of only 17 Million triangles, whereas the 3D surface of the competing method is made up of 51.8 Million triangles. For the Herzjesu8 models, the difference in size of the data is even larger as it differs by a factor of 10 (3.5 Million - 33.1 Million triangles). Nonetheless, the larger model visually does not give the impression of being more detailed. Hence, the voxel size chosen based on the error model seems to be more suitable.

To compare the qualities statistically, the unsigned cumulative errors of both models are shown in Fig. 7.16. It is clear that the method presented in this thesis is more complete in the low-quality areas, because the unsigned distance function reaches a consistently higher value. In the high-quality area, both methods show virtually the same results. However, this may be due to the ground-truth quality, which is unavailable, not taken into account. It also has to be considered that the EttlingenFountain and Herzjesu8 datasets are not really complex. Furthermore, the method from Fuhrmann and Goesele (2011) takes also global tetrahedralization into account that limits the scalability.

It is important to evaluate the reinterpretation of the fusion method in detail, because the results from Fig. 7.16 may also depend on the implementation and further extensions developed in this thesis, e.g., the TV based disparity error classification. To this end, the EttlingenFountain and Herzjesu8 datasets are again numerically evaluated against the

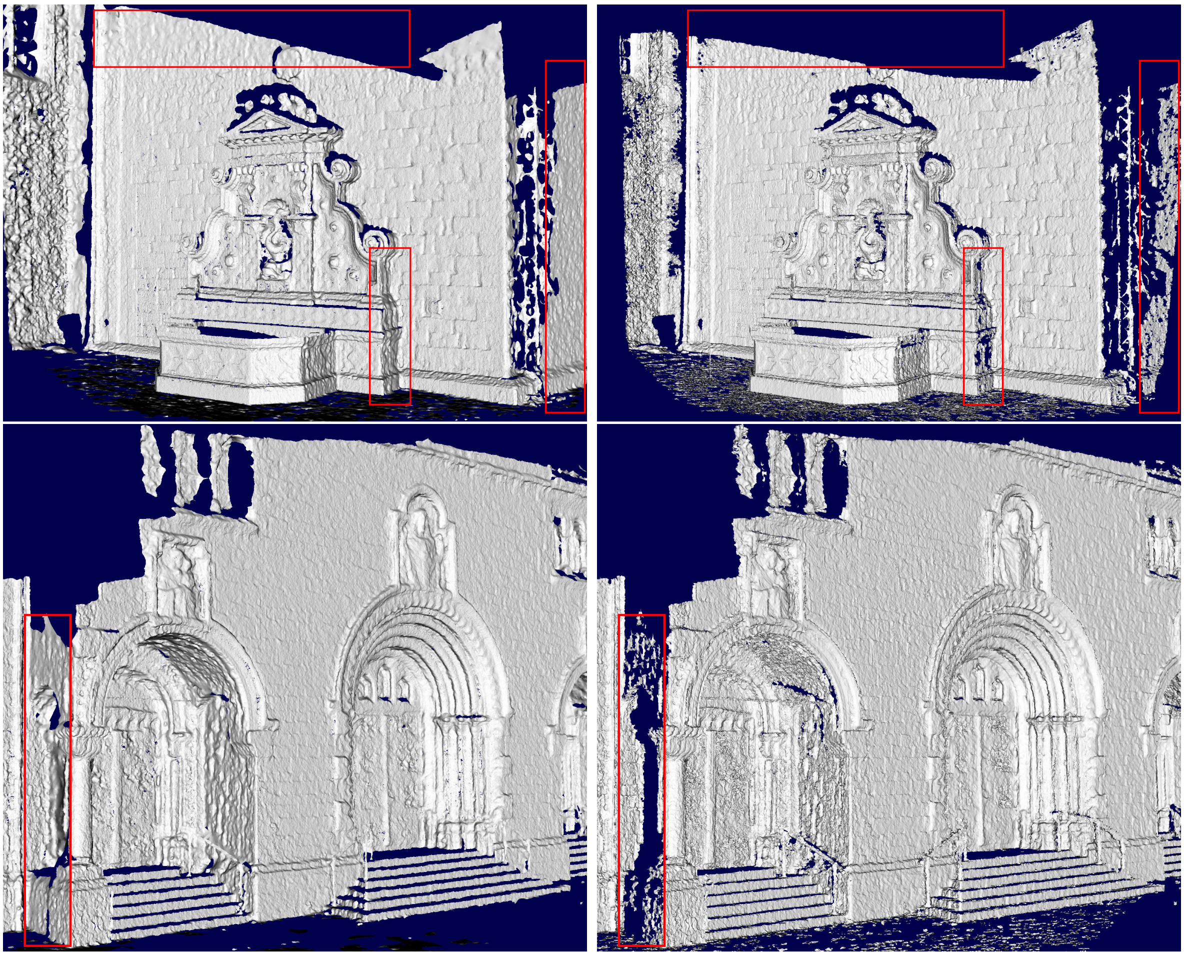


Figure 7.14.: 3D models from the EttlingenFountain (top) and Herzjesu8 (bottom) datasets. Left: 3D surface obtained by the method proposed in this thesis. Right: 3D surface generated using the method by Fuhrmann and Goesele (2011) based on the same disparity maps. The quality looks similar besides in difficult areas such as at the marked side (red box) of the fountain. Because the method proposed in this thesis takes into account the quality of non-fronto-parallel planes, difficult areas can be reconstructed. The surfaces contain varying numbers of triangles: 16990310 (left) and 51796108 (right) for EttlingenFountain and 3462399 (left) and 33067030 (right) for Herzjesu8. In spite of the large number of triangles, the surfaces on the right do not appear to be more detailed.

ground-truth data, but this time using the same implementation described in this thesis. Three surface models were generated based on the linear function and the mean fusion, as proposed by Curless and Levoy (1996), a Gaussian function with mean fusion, and a Gaussian function with Bayes fusion. The uncertainty of the linear function is

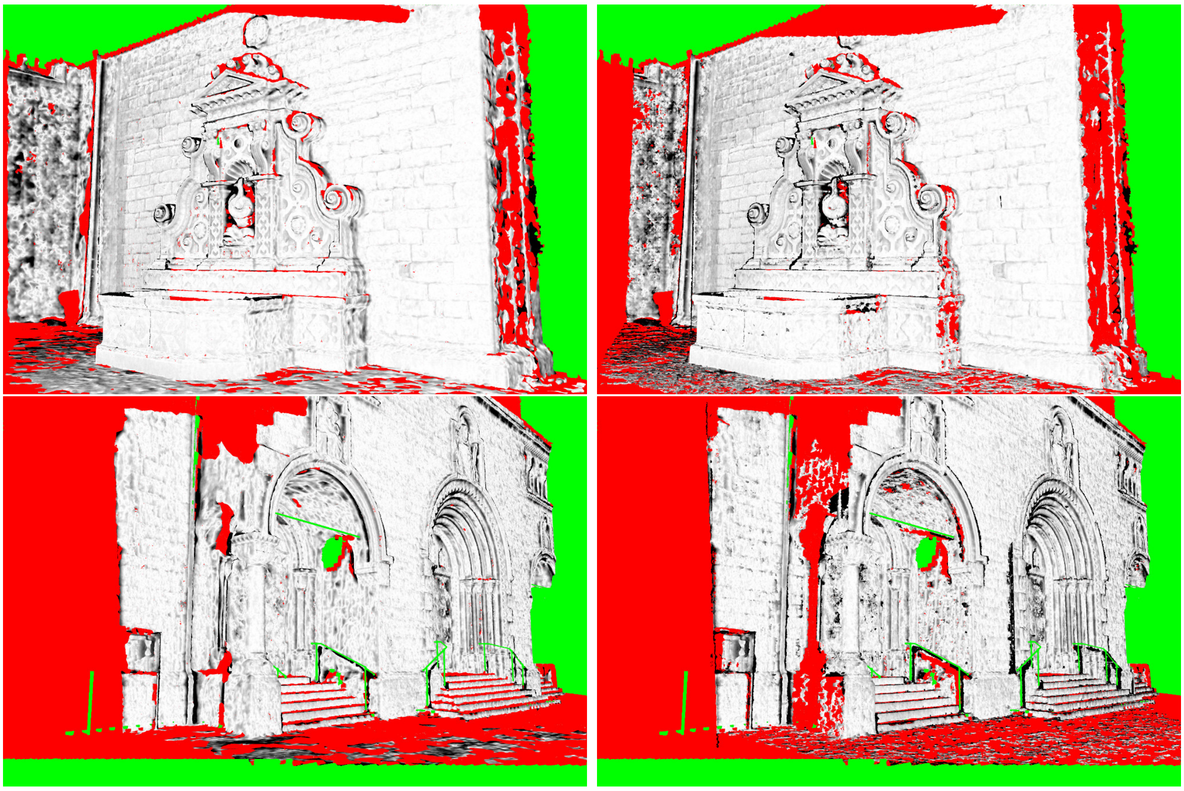


Figure 7.15.: Coded errors of the EttlingenFountain (top) and Herzjesu8 (bottom) surfaces from Fig. 7.14 (left, this thesis; right, (Fuhrmann and Goesele 2011)). The white pixels correspond to more accurate surfaces than black pixels. The red pixels were not reconstructed, and no ground truth is available for green pixels.

considered as 2σ, i.e., $p(-2\sigma) = 0$ and $p(2\sigma) = 1$ (cf. Fig. 2.7). The unsigned cumulative distance functions are shown in Fig. 7.17.

For both datasets, the Bayesian Fusion assuming Gaussian measurements provides the best results in terms of the accuracy and completeness. The assumption of a Gaussian instead of a uniform distribution and Bayes fusion instead of a weighted mean, lead to small but significant improvement. To obtain a consistent spatial fusion of multiple disparity maps, the correlated and uncorrelated data should be distinguished. Comparing the evaluation to the implementation from Fuhrmann and Goesele (2011) (cf. Fig. 7.16) the graphs in Fig. 7.17 confirm the results. The further progress shown in Fig. 7.16 is due to further extensions presented in this thesis concerning the error model and dynamic disparity error.

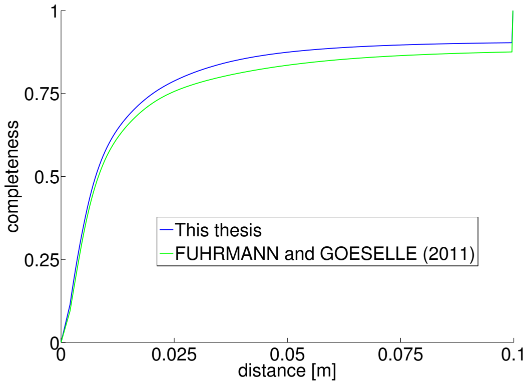


Figure 7.16.: Unsigned error graph for the EttlingenFountain (left) and Herzjesu8 (right) datasets obtained from Fig. 7.15. The method proposed in this thesis (blue) shows higher completeness than the method by Fuhrmann and Goesele (2011) (green), which results from the former method's reconstruction of difficult areas.

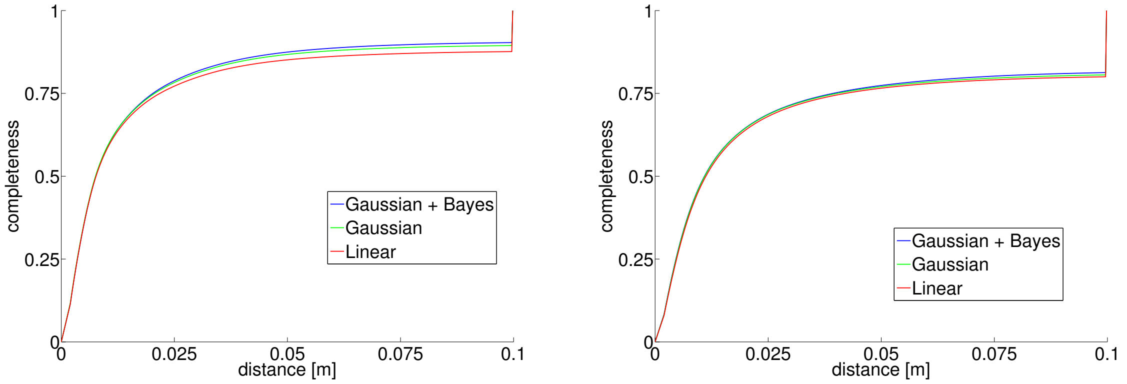


Figure 7.17.: Unsigned error graph considering linear, Gaussian, and combined Bayes Gaussian fusion. The novel fusion method obtains the best quality in terms of the accuracy and completeness. The stepwise improvements when considering Gaussians and Bayes fusion are small but significant. The linear fusion results in the worst level of accuracy. Left, Herzjesu8; Right: EttlingenFountain.

7.5. Disparity Uncertainty

For the estimation of the disparity maps from multiple image pairs, SGM is used in this thesis (cf. Section 5.1). In spite of the term "semi-global", SGM is considered a global estimation method, because the disparities are optimized on paths over the complete image. Global stereo methods allow for disparity estimation in weakly textured regions as prior information from the neighborhood are considered. For the cost, the intensity gradient in the images is considered. Furthermore, SGM has a fronto-parallel bias because differences between neighboring disparities are penalized. The quality of the disparity can thus be very different, and has to be considered in the fusion of disparity maps.

In this thesis, a feature that is highly correlated with the disparity error is provided. This feature is defined based on the local oscillation behavior and leads to 20 classes. Considering a varying disparity quality allows for an improved local depth fusion, e.g., for the varying perspective and strength of the texture. The relationship between the quality classes and the disparity error is learned using ground-truth data (cf. Section 5.3.3).

The error per class is described by the standard deviation of the disparity error ranging from a one-quarter pixel to several pixels. This uncertainty is then considered based on the mean of the error propagation for the stereo model (cf. Section. 5.3.1).

In the following paragraphs, a dynamic disparity error is evaluated against the fusion using a static disparity error with an assumed standard deviation between 0.5 and 4 pixels. The Ettlingen30 dataset is well suited for visually demonstrating a dynamic disparity error because the images have a varying perspective and texture with different strengths. There are two main problems in particular concerning the disparity in quality: a lack of texture owing to the white walls, and a slant to the surface producing increased uncertainties because of the fronto-parallel bias of SGM. In both cases, the TV prior weights the textured and fronto-parallel planes highly. From Fig. 7.18, it is clear, that the TV-derived standard deviation leads to the best quality. This is true for areas with many details, comparable to those areas with a standard deviation 0.5 or 1, and concerning completeness, it is similar to the model with a standard deviation of 4.

The runtime of the fusion process is about 10 minutes. Together with the depth estimation (20 minutes) and meshing (5 minutes), the overall runtime is 35 minutes. The reconstruction space was split into hundreds of subspaces that were computed in parallel on a cluster with about 100 cores.

The dynamic disparity error based on TV is especially suitable for complex image configurations with varying perspectives of the cameras and a lack of texture in the images. Unfortunately, the image sets available for a numerical evaluation do not show these difficulties at all. In spite of this, for the sake of completeness, the dynamic disparity error classification is numerically compared to static disparity errors based on the Herzjesu8 and EttlingenFountain datasets. A statistical evaluation of the five differing models with an assumed error between 0.5 and 4 pixels and the TV based error is shown in Fig. 7.19.

As expected, the TV-based disparity error is best in terms of the overall completeness for both datasets. For the Herzjesu8 dataset, the TV-based disparity error also leads to the highest accuracy. For the TV-based solution, the EttlingenFountain dataset shows a small loss in accuracy compared to a constant uncertainty of $\sigma = 2$ and $\sigma = 4$. However, although it cannot be proved, it is thought that the evaluation results would differ in the high-resolution parts, as evaluated based on the sensor noise from the ground truth,

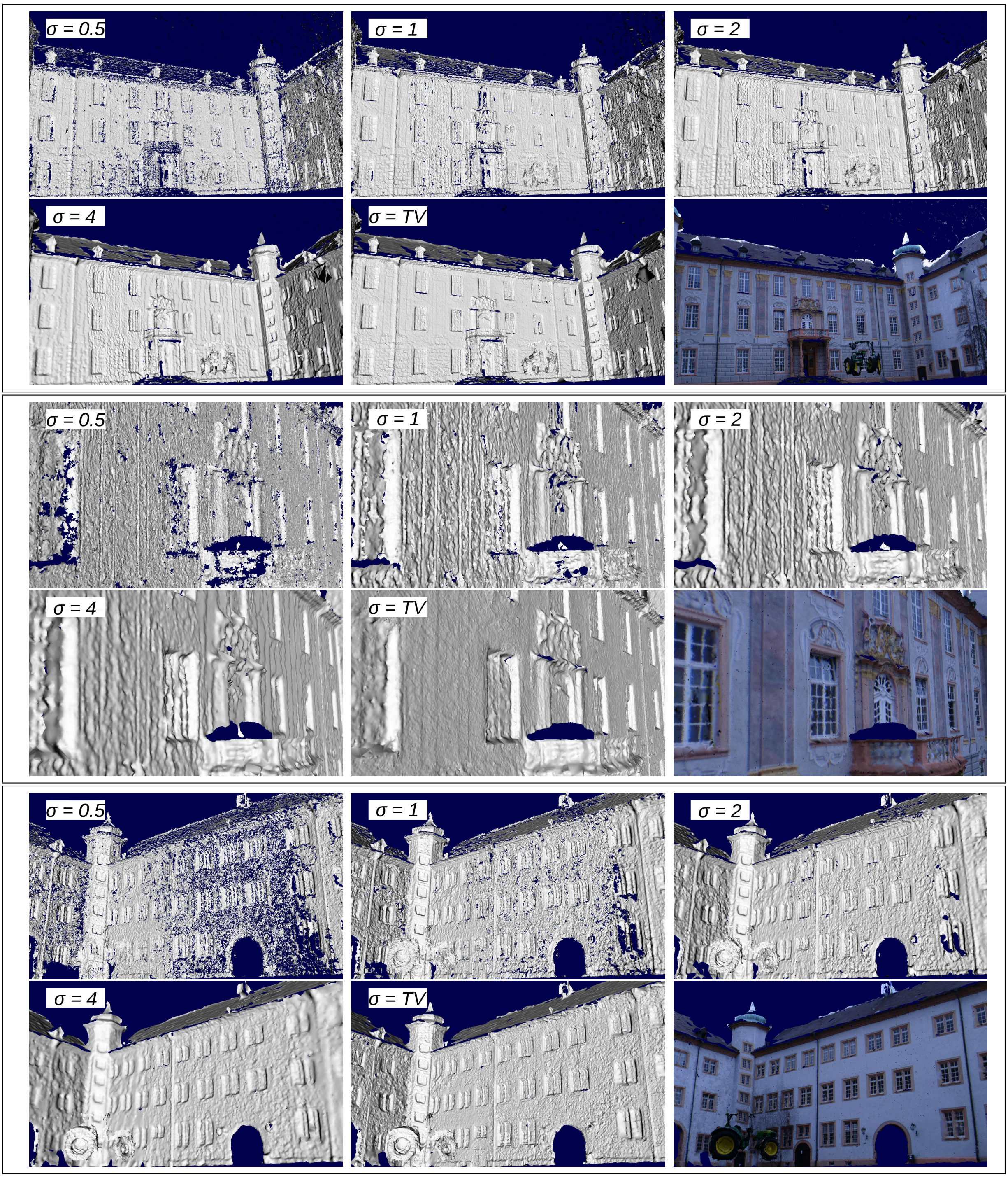


Figure 7.18.: The three boxes show parts of the model derived from the Ettlingen30 dataset. From left to right and top to bottom of each box: models considering assumed standard deviation of: 0.5, 1, 2, and 4, and the TV and textured TV models. The TV solution is best in both completeness and accuracy.

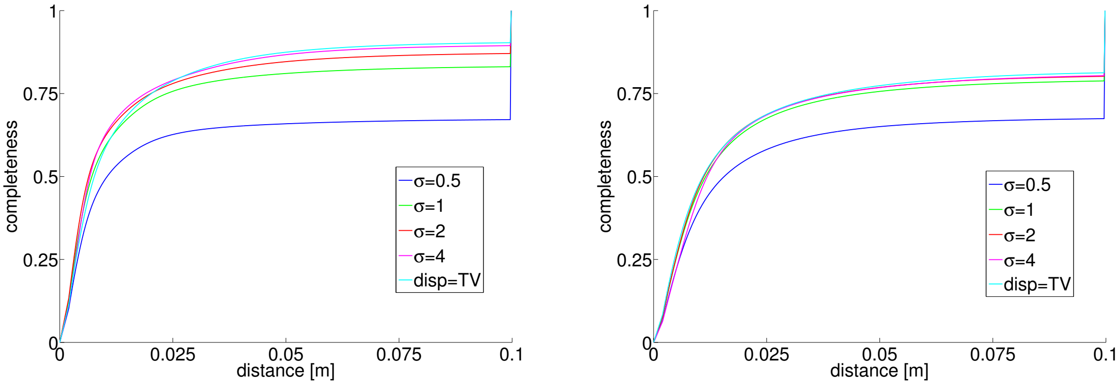


Figure 7.19.: Unsigned error graphs for the EttlingenFountain (left) and Herzjesu8 (right) dataset. The five graphs show the errors for the models when assuming a constant uncertainty in the disparity error (0.5 to 4 pixels) and TV-based variable disparity error.

which is unfortunately not publicly available.

For the sake of completeness, an evaluation on the Middlebury multi-view benchmark is also conducted. The datasets are not suitable for showing the strength when considering different disparity qualities because the objects do not have a variable texture and the perspective is simple. However, the evaluation (cf. Table 7.1) confirms the idea, derived from the visual results, that the TV results are the best in terms of accuracy, and are generally the best in terms of the completeness. The models with a TV prior therefore combine the details with completeness.

	Temple	TempleRing	TempleSparse
acc.	0.55/0.54/0.57/0.73/**0.51**	0.62/**0.6**/0.79/1.86/**0.6**	0.6/0.59/0.76/1.75/**0.56**
compl.	85.1/95.6/**97.5**/95.7/97.4	92.9/**96.3**/92.9/69.4/95.5	79.0/84.7/**86.3**/68.9/84.8
	Dino	DinoRing	DinoSparse
acc.	0.45/0.43/**0.41**/0.46/0.42	0.52/0.47/0.49/0.77/**0.42**	0.57/0.52/0.48/0.75/**0.42**
compl.	61.4/94.8/97.6/**98.7**/97.3	89.4/94.9/**97.1**/94.2/94.4	80.9/89.6/**93.0**/91.1/88.9

Table 7.1.: Evaluation of Dino and Temple with $\sigma = 0.5/1/2/4/TV$ concerning accuracy and completeness. The best value of the individual dataset is marked in bold. In general, the TV results are the best in accuracy and close to the best in terms of the completeness.

7.6. Registration Uncertainty

The image registration method presented in Section 2.2.1 allows for an estimation of the variances and covariances of the camera parameters. When ignoring the inner parameters, the influence of the registration uncertainty on the uncertainty of a 3D point can be described using a non linear equation system. Unfortunately, there is no simplified description of the influence of n cameras on a 3D point available. Nonetheless, the covariance information for the registration can be considered as error propagation can be solved numerically for all pixels. As shown in Section 5.3.2, this is computationally complex, because large matrices have to multiplied. Furthermore, the Jacobian has to be estimated numerically for all 3D points based on the error propagation depending on the image configuration.

The error propagation described in Section 5.3.2 considers covariance matrices for all cameras employed in the stereo matching. The covariance matrices are used to describe the uncertainty of the absolute camera pose in a global coordinate system. Yet, the 3D points obtained by SGM from only a small part of the image set can have a high relative but a low absolute accuracy. Nonetheless, error propagation of the 3D points considering absolute registration uncertainty can be suitable for quality assurance. E.g., for applications like navigation global uncertainty information for surfaces is of high importance. Considering the covariance matrices for relative poses of camera pairs would also be suitable even for improving the disparity quality, but is beyond the work of this thesis.

The runtime increases considerably, to about ten-times the runtime without considering the registration uncertainty. Considering, i.e., five cameras that influence a single 3D point a 30×30 overall covariance matrix is used for the propagation by Eq. (5.19). Because the calculation of the uncertainty is pixelwise independent and mainly consists of matrix operations, it is also suitable for a multi-core implementation on a GPU. The propagation of points can even be substituted by the more efficient propagation of point clusters with similar parameters.

In Fig. 7.4 areas are shown with inconsistencies of the surfaces caused by the registration errors. The corresponding image set was also processed by propagating the registration uncertainty. The univariate error as described in Section 5.3.2 (cf. Fig. 5.6) has been employed and the resulting error is used for the choice of the voxel size as described in Section 6.2.

Fig. 7.20 shows the same part of the scene as Fig. 7.4 but with consideration of registration uncertainty. The inconsistent areas are regularized by means of the registration uncertainty leading to smooth and consistent surfaces. However, because the images are from various perspectives and there is a corresponding highly propagated uncertainty, the small details are lost, i.e., it is not suitable for high-quality surface reconstruction. Nonetheless, the processing is interesting to obtain a global consistent surface.

Hence, it is not recommended to use the registration uncertainty in the 3D fusion process directly. For large uncertainties that exceed one pixel, the disparities will be

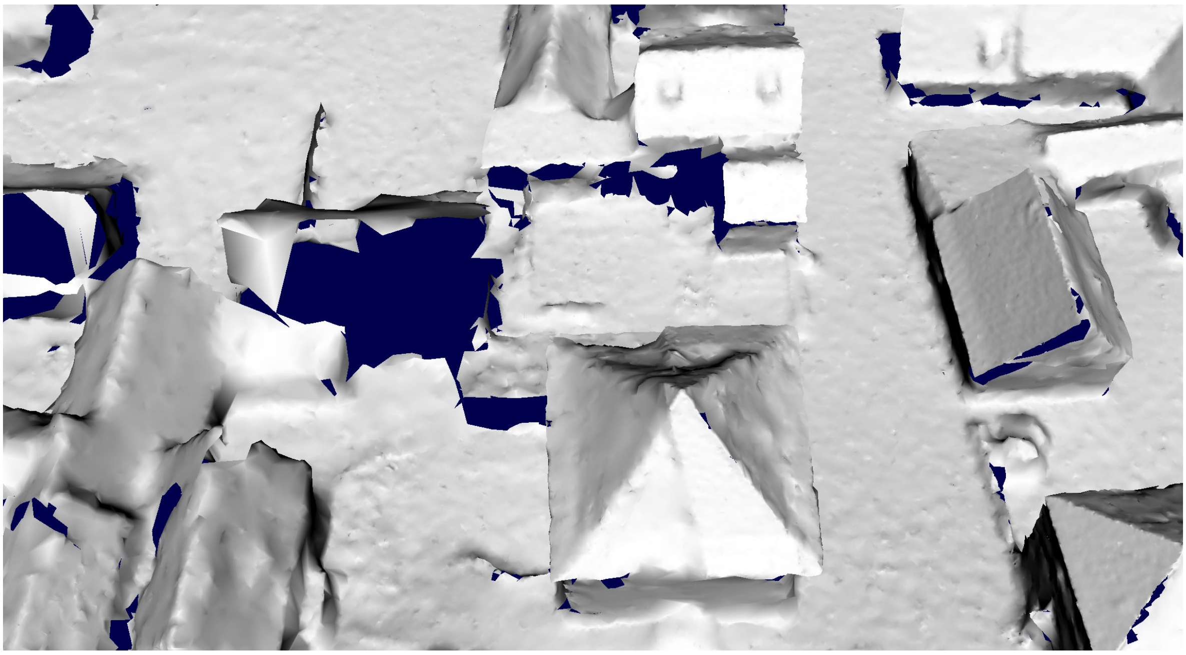


Figure 7.20.: This image shows the same part of the model shown in Fig. 7.4. The registration error was successfully used for regularization because the roof is not hard-edged any more. In addition, the ground is smoothly reconstructed. Because of the large number of images, the resolution is much lower, creating a high loss of accuracy.

filtered out in the fusion of the disparity maps for a single camera. It may be more suitable to consider the information in the stereo matching. This is especially true for relative uncertainties of camera pairs. Nonetheless, application dependent it can be important to obtain global consistent information for all surface parts.

7.7. Scale Space

Scale Space theory (cf. Section 6.3) is not the focus of this thesis. Nevertheless, it is important because the multi-resolution approach deals with data with varying quality at different levels. The adaption of scale space methods is described in Section 6.3 proposing two ideas: the propagation of multiple scale disparity maps, and a simple probability regularization of neighboring octree levels.

Propagation of disparity maps from multiple scales is advantageous, because by making use of that a surface might be textured differently well at different scales can lead to a more complete disparity map, particularly due to the disparity maps from lower resolutions. For a numerical evaluation of the possible quality improvement, the datasets EttlingenFountain and Herzjesu8 were again used. In contrast to the standard fusion, in addition to the disparity map derived from images downscaled by a factor of 2, disparity maps derived from images of the original size and downscaled by a factor of 4 were propagated.

It is important to note that SGM filters disparities that do not comply with the left right check: these disparities were filtered, which have a difference of one pixel or more. Hence, the disparity maps are generally more dense in the downscaled disparity maps.

Fig. 7.21 shows an unsigned graph describing the errors for the individual disparity maps, and the results, considering all disparity maps, in a single fusion process. Using the disparity maps with a downscaling factor the surface quality is best in terms of completeness, but worst in terms of accuracy. The disparity maps of the original size should be the best when concerning the accuracy. Unfortunately, this cannot be evaluated using the given ground truth because the accuracy for the most detailed area is below $1cm$ and there is no uncertainty information for the ground truth. The multiscale approach reached the best quality in terms of completeness, and most likely the best quality in terms of accuracy (cf. Fig. 7.22).

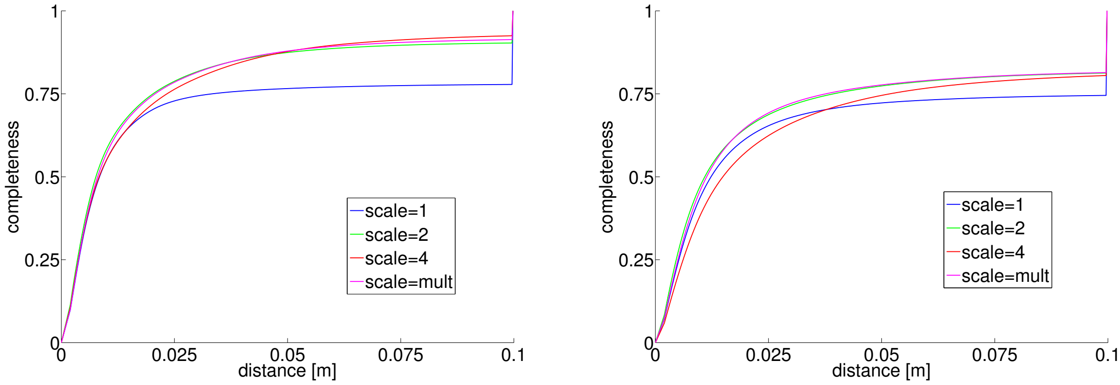


Figure 7.21.: Unsigned error graphs for the Herzjesu8 (left) and EttlingenFountain (right) dataset. The graphs show the errors in the models obtained from the disparity maps from downscaled images, and the model considering all scales. The combination shows a similar level of quality as the best in terms of accuracy and completeness, but is slightly more complex computationally.

Scale space processing can improve both the resulting accuracy and the completeness. Such improvements are usually not very high, but the runtime falls quadratically for each level of resolution. Half the resolution leads to one-quarter of the time required for disparity map estimation. Depending on the application, this overhead may be acceptable.

For the probability regularization of the neighboring octree levels, considering Eq. (6.4), it was found to neither improve nor decline the numerical evaluation of the EttlingenFountain and Herzjesu8 datasets. In addition, only marginal differences were

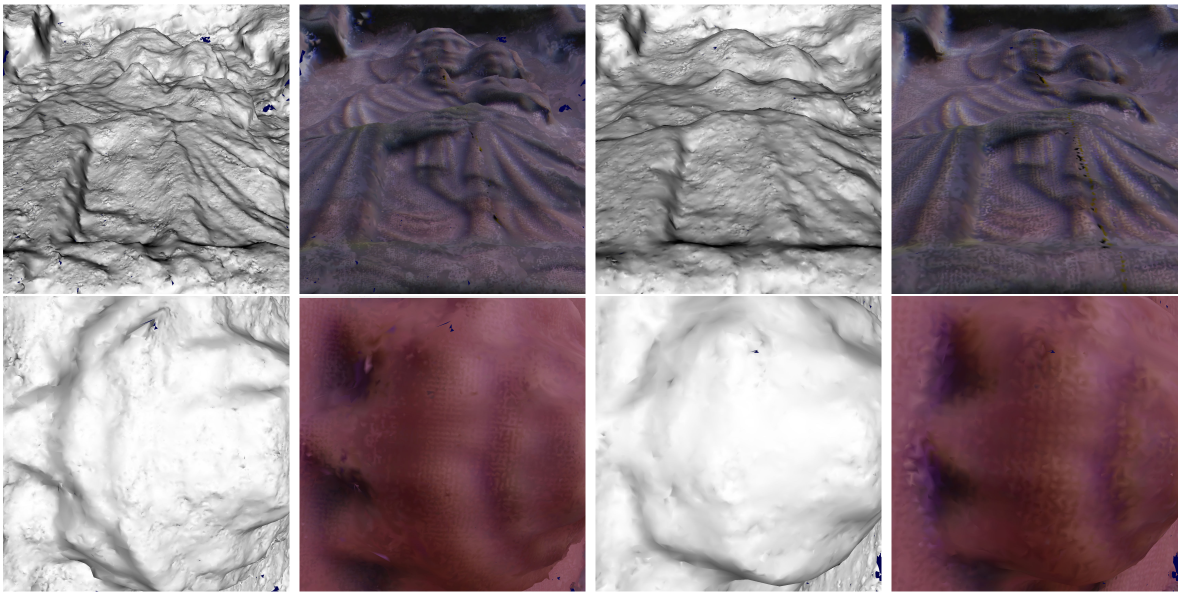


Figure 7.22.: Left two columns: shaded and textured surfaces from multiple disparity maps. Right two columns: shaded and textured surfaces from disparity maps derived from images downscaled by a factor of 2. Top: Small area of the Herzjesu8 surface model. Bottom: Small area of the EttlingenFountain surface model. The left surface shows finer details in certain areas.

detected through a visual comparison. Scale space approaches considering multiple scale measurements will be suitable during the filtering steps. Although, the progress seems to be slight and the theory is complex. Nonetheless, further research would be suitable to guarantee a better stability.

Chapter 8.

Summary, Conclusion, and Future Work

The ultimate goal of this thesis is scalable high-quality 3D surface reconstruction from large image sets. Towards a solution for this goal, five extensions of existing state-of-the-art methods have been presented:

1. A local Divide and Conquer procedure for unlimited scalable 3D reconstruction,
2. Probabilistic interpretation and extension of a volumetric fusion,
3. Statistical estimation of the disparity uncertainty,
4. Probabilistic filtering of spatial data,
5. Multi-resolution scheme considering sound error models.

In this chapter, an overview, a summary, and some concluding remarks of the thesis are given. In Section 8.1, an overview of the 3D reconstruction pipeline is presented emphasizing the contributions of this thesis. In section 8.2, the five main extensions presented in this thesis are summarized and some concluding remarks are given. Finally, in Section 8.3, a look into possible future work for the 3D reconstruction pipeline is provided.

8.1. Processing Chain

For the development and evaluation of the methods presented in this thesis, a processing chain was defined for 3D surface reconstruction from image sets. This chain is illustrated in Fig. 8.1, and a brief description is given below. In particular, the contributions of this thesis in terms of its individual steps are emphasized.

1. Image Registration is concerned with an estimation of the camera poses and possible inner parameters (cf. Section 2.2.1). It was used in some of the experiments described in Section 7, and numerical error propagation considering the parameter uncertainty has been derived in Section 5.3.2.

2. Stereo Matching is about the estimation of pixelwise disparities considering the known camera poses. SGM was used for the experiments and quality estimation derivation described in this thesis (cf. Section 5.1).

3. Quality estimation for the disparities is an important novelty of the method described in this thesis. Feature classes based on the TV were defined in Section 5.3.3,

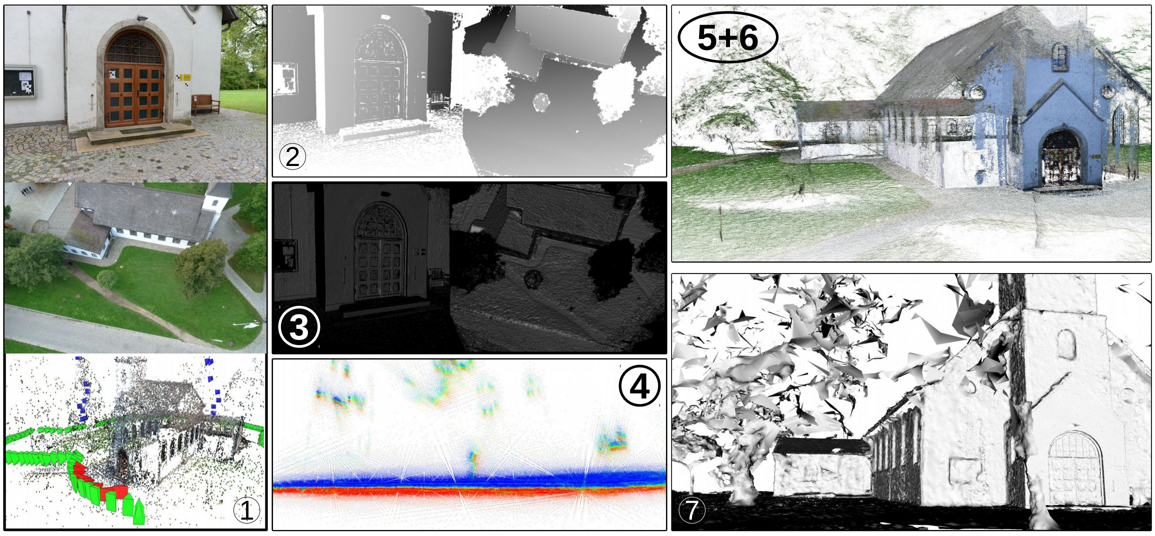


Figure 8.1.: Process chain of the 3D reconstruction method described in this thesis: 1) Image registration, 2) Stereo Matching using SGM, 3) quality estimation for disparities, 4) generation of a probabilistic space [1], 5) point optimization considering the probabilistic space, 6) filtering of the outliers in the point cloud, 7) triangulation of the point cloud. Steps 3 through 6 are the main focus of this thesis.

which are highly correlated with the disparity uncertainty. The correlation function was statistically learned from the ground-truth data by the means of machine learning (cf. Section 5.3.3).

The generated **4. probabilistic space** is used for the **5. optimization of the point cloud**. To this end, a novel probabilistic interpretation for the seminal volumetric fusion method by Curless and Levoy (1996) is given in Section 4.2. The interpretation has then been used in Section 4.3 to derive a novel Bayes fusion method for spatial data considering Gaussian uncertainty.

In addition to an optimization of the point clouds, the novel probabilistic fusion allows for the extraction of surface probabilities. The latter are used for the **6. filtering of outliers** by means of the volumetric ray tracing approach presented in Section 5.4.3.

For the **7. triangulation** of the point cloud, a fast local triangulation method was adapted for connecting points with varying densities, as derived from the volumetric information (cf. Section 6.4).

In summary, steps 3 through 6 are the main focus and comprise the novelty of this thesis. To guarantee unlimited scalability, these steps, along with triangulation in step

[1] The image of step 4 shows the probabilistic space modeling the ground surface. Blue areas have high probabilities to be above and red areas to be below the ground.

7, are conducted on subsets of the reconstruction space. To this end, a Divide and Conquer strategy was presented in Section 5.2. Because the processing of the subspaces only employs local optimization for the fusion and triangulation, a simple merging of the subspaces is possible. For this, the overlap has to be twice the area of the maximum local optimization area.

8.2. Summary and Conclusion

This thesis presented a Divide and Conquer strategy allowing for unlimited scalability and high runtime performance because solutions for subspaces of the reconstruction space can be computed in a highly parallel manner. The strategy extends state-of-the-art Divide and Conquer strategies because a purely local optimization is used, allowing for a simple fusion of the subspaces. Validation on several challenging datasets shows the potential of this novel approach. In general, local optimization for 3D surface reconstruction does not produce a surface quality as high as by global optimization. Therefore, in this thesis, the local optimization was improved through a further deepened analysis of the spatial error.

The seminal method for volumetric fusion of noisy data from disparity maps by CURLESS and LEVOY (1996) was reinterpreted from a probabilistic perspective. This novel perspective allows for a derivation of the surface probabilities, which in turn are used for the filtering of outliers in 3D space considering the geometric consistency. From the probabilistic perspective, the assumption of Gaussian noise in the depth data was found to be violated in the original formulation. Hence, the probabilistic fusion of spatial data requires a novel probabilistic fusion theory. The extended probabilistic framework considers the propagation of spatial points with Gaussian uncertainty. For the fusion of uncorrelated data, a Bayes framework was devised, which is particularly suitable for scalable surface reconstruction because a recursive fusion is possible. This extension was demonstrated to improve the results in terms of both accuracy and completeness. The presented framework assumes the existence of uncorrelated data for the fusion, which is not generally valid for multi-view stereo (MVS) data.

By considering a varying disparity quality that depends on the local oscillation behavior in the disparity maps, this thesis provides another means for an improvement of both the accuracy and completeness of the local methods. To this end, the local oscillation behavior is measured by estimating Total Variation (TV) classes. The correlation function between a disparity error and the TV classes was learned from ground-truth data by means of an Expectation Maximization (EM) approach. In particular, weakly textured and non-fronto-parallel areas are classified as being of low quality. Considering the varying quality of the disparities and propagating them into a 3D space employing sound error models was shown to improve the results on several datasets. The classification is especially important for regularization, leading to complete and accurate surfaces for challenging configurations.

Furthermore, this thesis extends existing state-of-the-art methods by means of consistency checks in 3D space. The surface probabilities propagated by the probabilistic framework are used for the filtering of outliers in 3D space. These geometric consistency checks are particularly suitable to filter outliers close to the surface where they mainly appear. Filtering outliers far from the surfaces does not comply with an unlimited scalability. These outliers are usually a minority, and can be filtered during the meshing step by disregarding surfaces with a small number of triangles.

In the fusion, it is important to account for largely varying qualities of spatial measurements, because low-quality points can disturb high-quality points. This thesis provides a multi-resolution approach for fusing data on different levels employing efficient data structures. To this end, it extends the state-of-the-art methods by means of a sound 3D error model. Furthermore, error models considering uncertainties of the camera parameters and aspects of the scale were discussed.

In summary, this novel method is not limited to scalability. The extensions introduced in the local methods considerably reduce the loss of accuracy. Multiple subareas can be processed in parallel. It even has high potential for massive parallelization, e.g., using GPUs. However, this is beyond the scope of this thesis.

8.3. Future Work

There are multiple issues for further improving the quality towards an optimal solution to scalable 3D modeling.

The assumption of accurately registered images is not always fulfilled. Image registration leads to uncertainties that are hard to model because error propagation cannot be described linearly. Considering the covariances of the global camera poses is insufficient and not directly suitable for high-resolution 3D surface reconstruction. In the stereo estimation, varying relative uncertainty should be considered because subpixel accuracy cannot be guaranteed for all regions. As described in Section 5.1, the disparity estimation by SGM from n image pairs filters and fuses the disparities from the individual image pairs. Hence, not all covariance information from all cameras influences the individual disparities. However, experiments have shown that considering only stereo pairs in 3D also does not improve the quality because the disparity fusion from multiple images is of high importance for the accuracy of the disparity maps. A possibly promising direction for further research may be to consider relative covariances of camera pairs in the step of stereo estimation and disparity fusion.

An extension of the disparity estimation should also scrutinize the fronto-parallel bias. There are several challenging image configurations with highly slanted surfaces leading to no or inaccurate surfaces. An adaption of the fronto-parallel bias can improve the disparity quality. The learning scheme presented in Section 7.5 allows for a classification of the uncertainty of non-fronto-parallel planes. However, such classification is not used for the improvement of the quality of the disparity maps. As the quality drops, especially for slanted surfaces, it may be helpful to consider a bias for multiple planes with different directions instead of only the fronto-parallel plane.

The classification of the disparity quality is derived from a variable pixel neighborhood. However, whether the disparities really describe the same surface has yet to be considered. In certain configurations, stereo matching can obtain spurious results in border areas of the surfaces (cf. Fig. 8.2) that are not guaranteed to be classified as low quality. A combined classification of the disparity maps and segmented images may alleviate this problem. First, progress has been achieved by filtering large segments that have a sparse disparity density. However, this was not considered in the experiments conducted for this thesis.

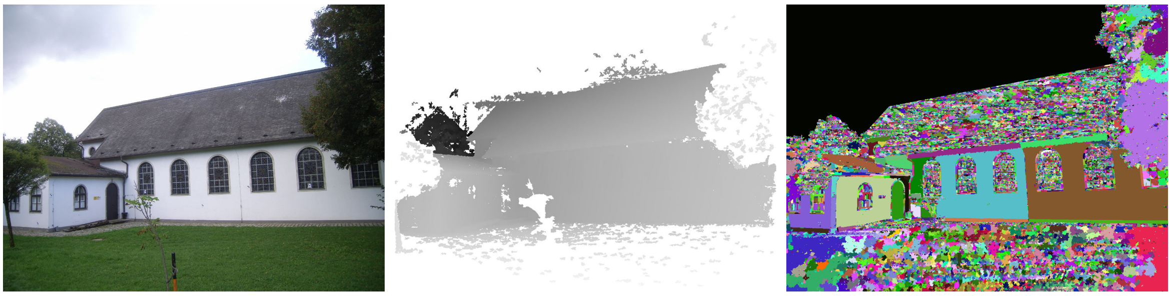


Figure 8.2.: Left: Image from the Unikirche dataset. Middle: Corresponding disparity map. Right: By means of watershed transform segmented image (Meyer 1992). The outliers of the disparity map at the top of the roof can be classified based on information from the segmented image.

Within the Bayesian fusion theory for the propagation of disparity maps to accurate point clouds a very basic assumption are uncorrelated measurements. This is an important source of uncertainty because especially MVS data are usually highly correlated. This is, e.g., due to the use of the same sensor and from deriving the disparity maps from at least two images. For an algorithmic de-correlation, the cameras can be clustered and computed independently. Fusing clusters of disparity maps may lead to more accurate results and at the same time lower runtime because only important spatial data are propagated in 3D.

The Divide and Conquer approach allows for parallel processing of the model parts, e.g., on a cluster system with multiple cores. The local optimization of all point clouds in turn renders parallel processing strategies of the subareas possible. This can even be done on multi-core units such as GPUs. Owing to the difficult implementation, especially of the data structures, this was not realized for this thesis, but is regarded as a promising direction for further research.

Beyond the topic of this thesis there are further 3D modeling problems to be solved. When allowing for an unlimited number of images, the resulting surfaces can become extremely large. For mostly smooth objects, the surfaces can be described using a small fraction of the given amount of triangles. Because this thesis avoids complex fusion strategies and employs parallel processing, a reduction of the polygon set should also employ a local strategy.

This thesis does not deal with the texturization of the surface. Only one color is assigned to each point and resulting triangle. Particularly after a triangle reduction, the texturization is of high importance. Due to the possible radiometric and physical differences within the camera configuration, optimization of the textures can be very complex. Additionally to the use of different cameras, the images might have been captured at different times and under varying lighting conditions, the surfaces might have perspective deformations and the assumption of Lambertian surfaces is often not fulfilled.

Appendix A.

Results

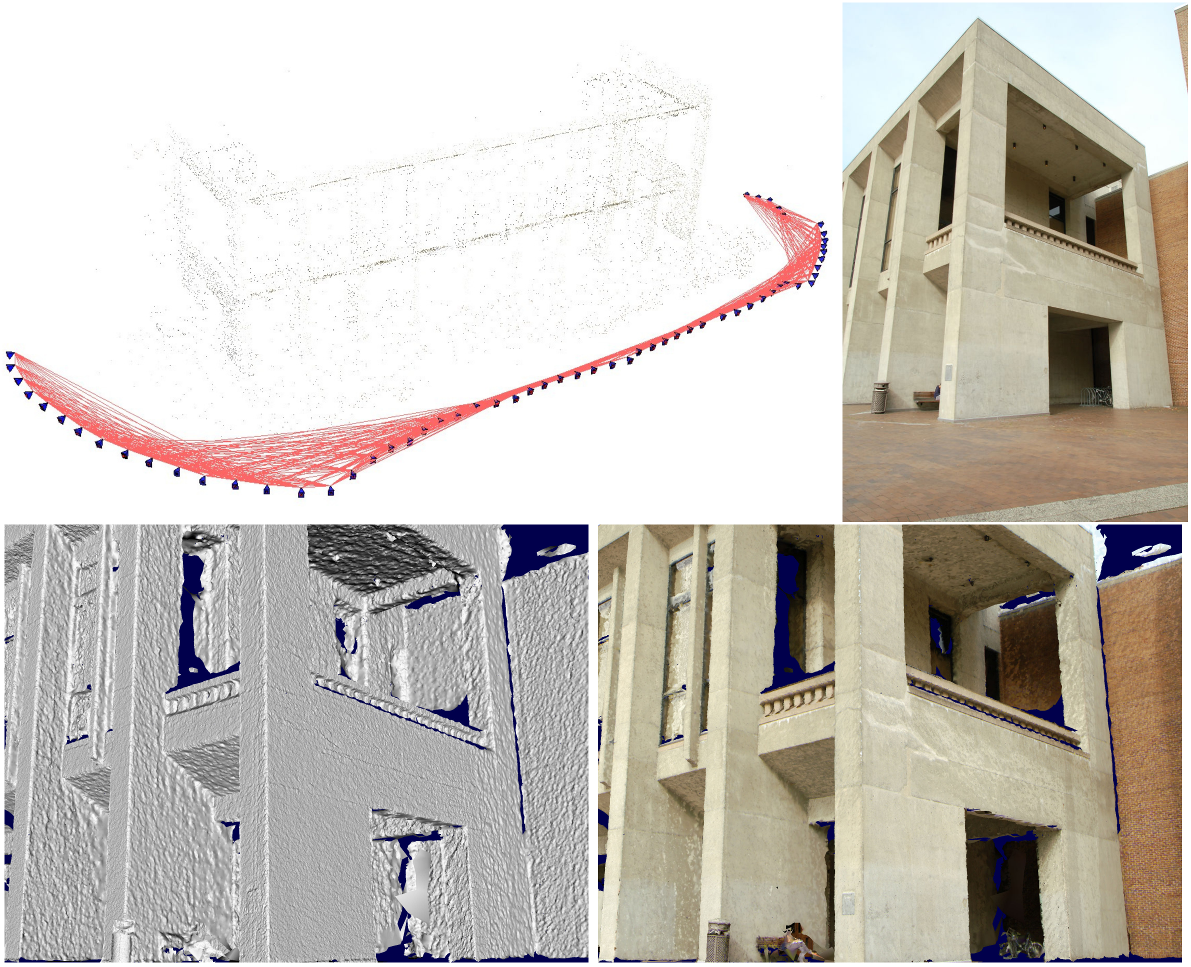


Figure A.1.: Hall dataset (Furukawa and Ponce 2010). The 61 images from a sequence were registered and the surface was reconstructed by the method presented in this thesis. Even small details like the rubbish are maintained in the 3D surface model. The upper left shows the camera poses and a sparse point cloud. On the upper right one image of the sequence is shown. Bottom: The shaded and the textured 3D surface model.

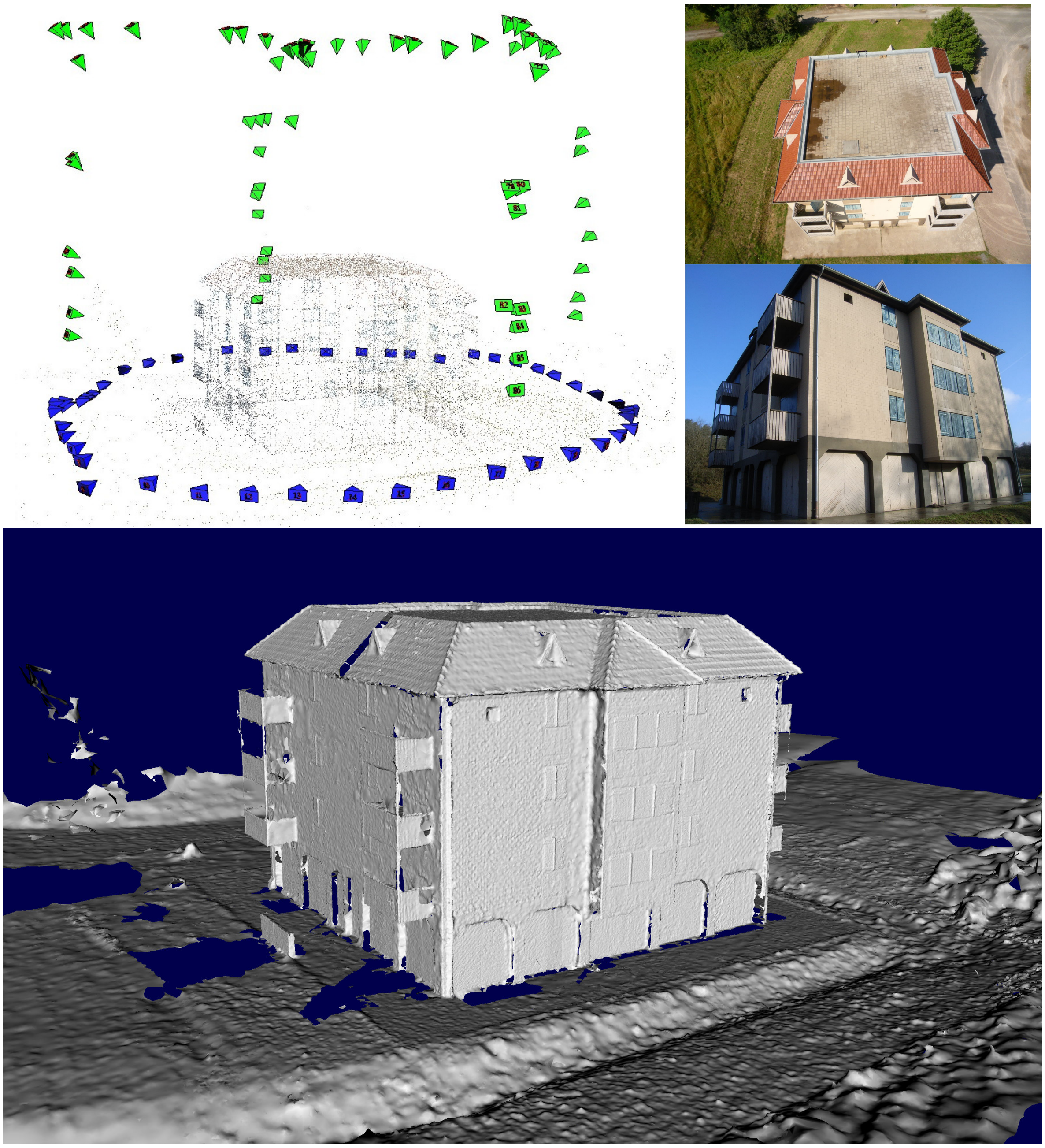


Figure A.2.: Haus51 model obtained from 112 images with a resolution of 10 MP. 48 images were captured from the ground and 64 from the air by a camera on a UAV. In contrast to the models shown in Fig. 7.12 this model considers multiple resolutions of the images as described in Section. 6.3.

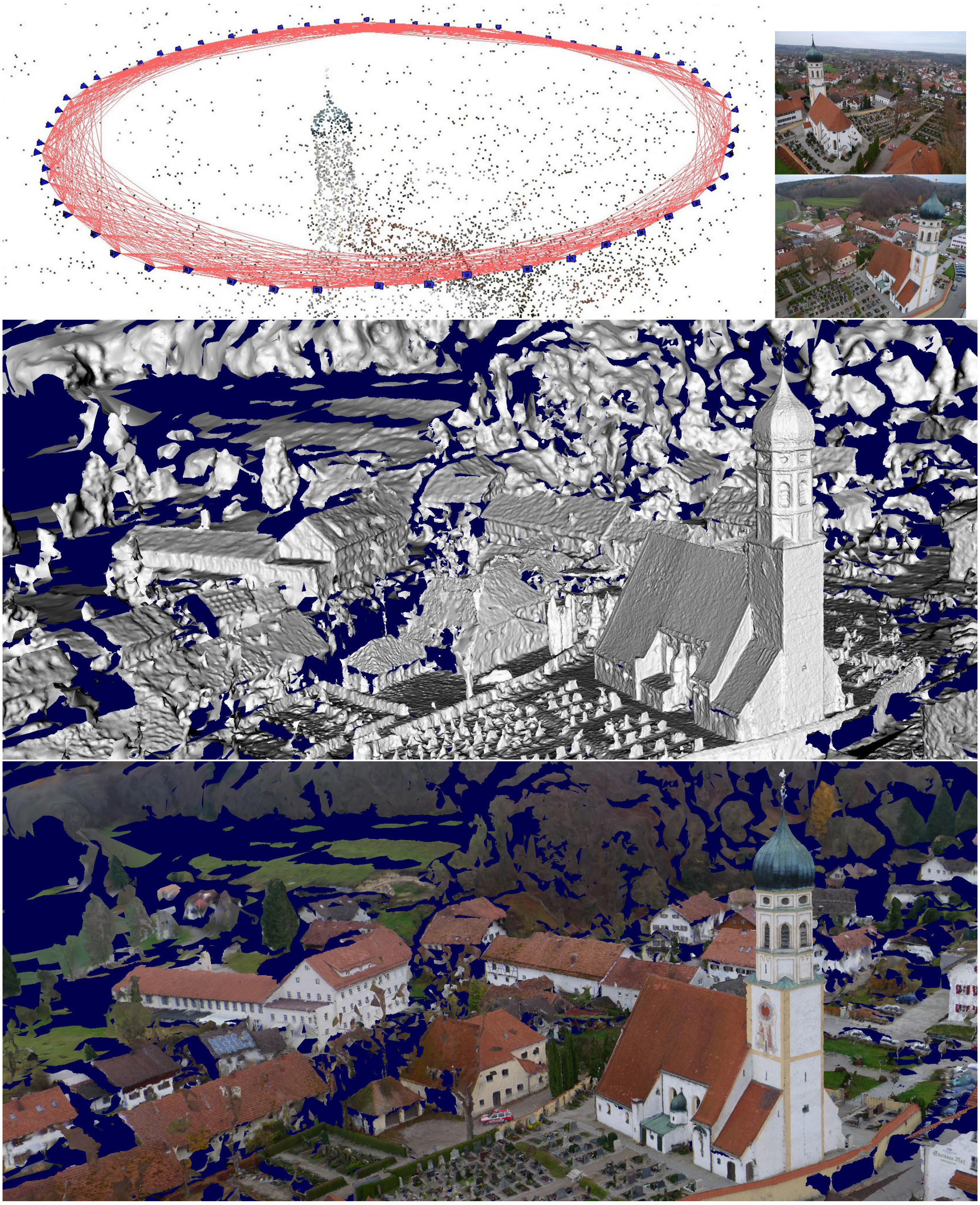


Figure A.3.: Seefeldkirche dataset. 60 images were captured by a camera on a UAV from varying perspectives focusing on the object. © DLR Institute of Robotics and Mechatronics

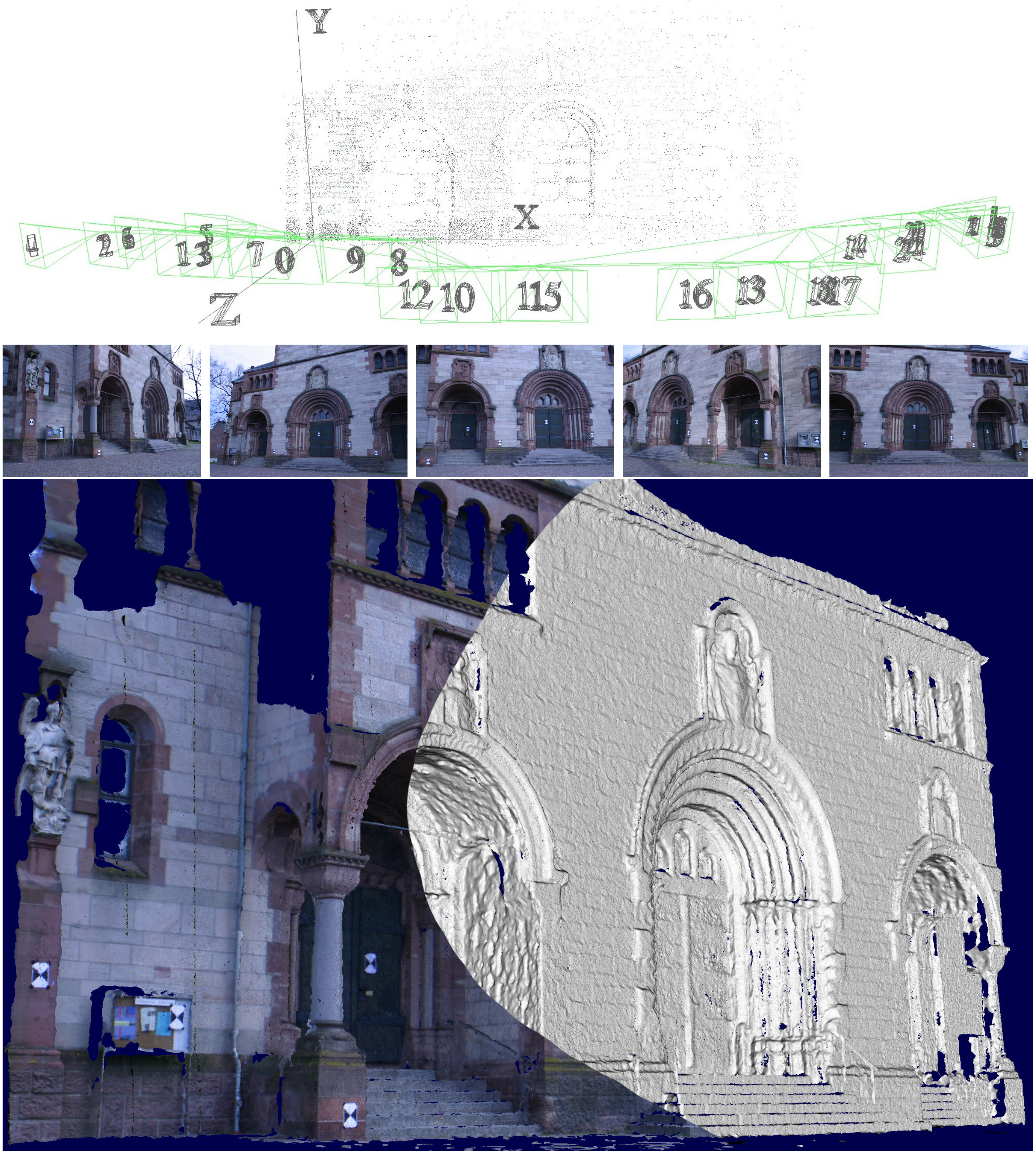


Figure A.4.: Partly shaded and partly textured 3D surface model obtained from the Herzjesu25 sequence. The markers were not taken into account. The 3D surface model contains small details like the stair railings or the metal bars. It consists of 32-million connected triangles.

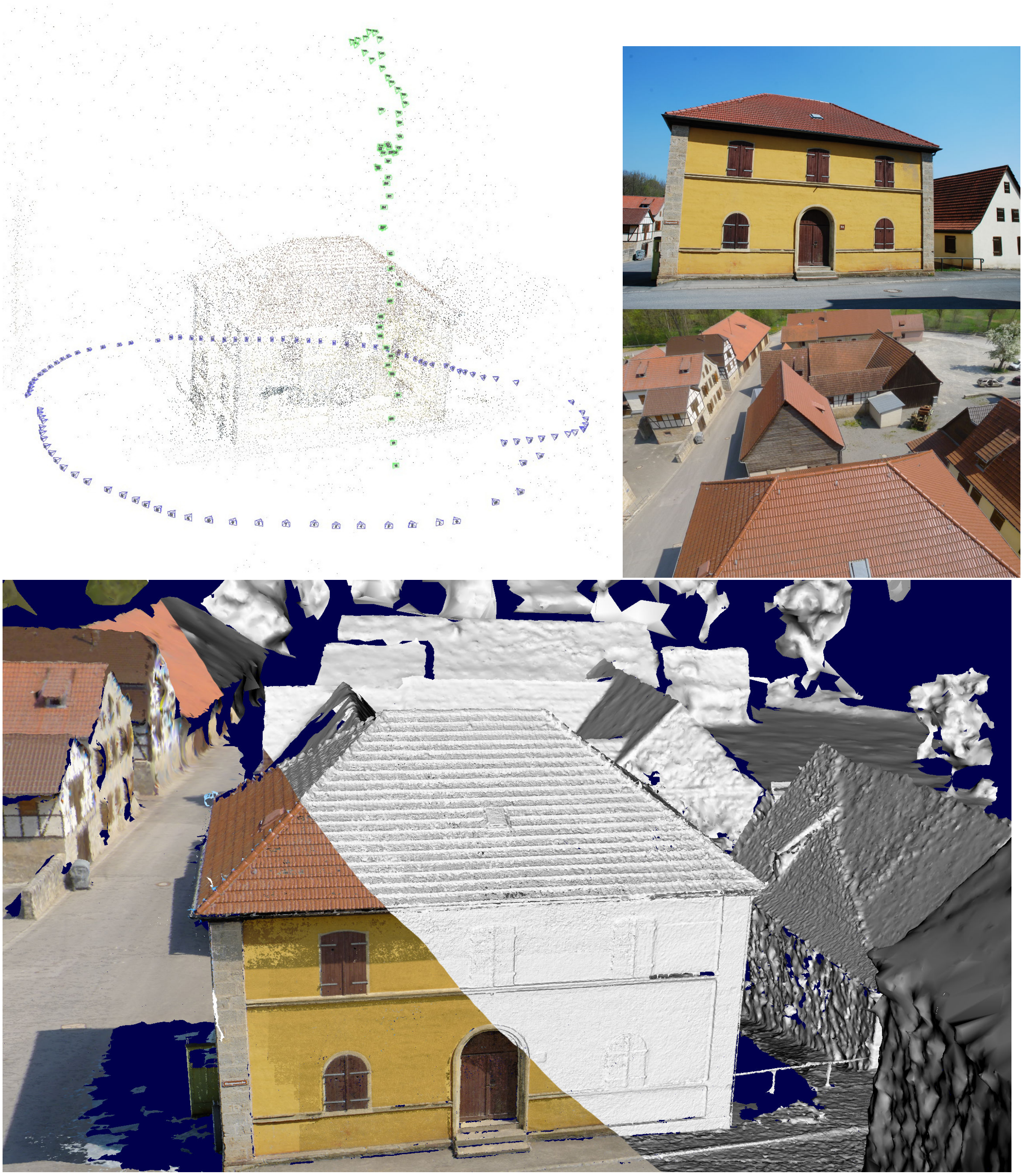


Figure A.5.: Haus35 Model obtained from 150 images from the ground and an UAV. The surface is visualized partly textured and shaded. The front areas are automatically modeled with higher detail depending on the error model.

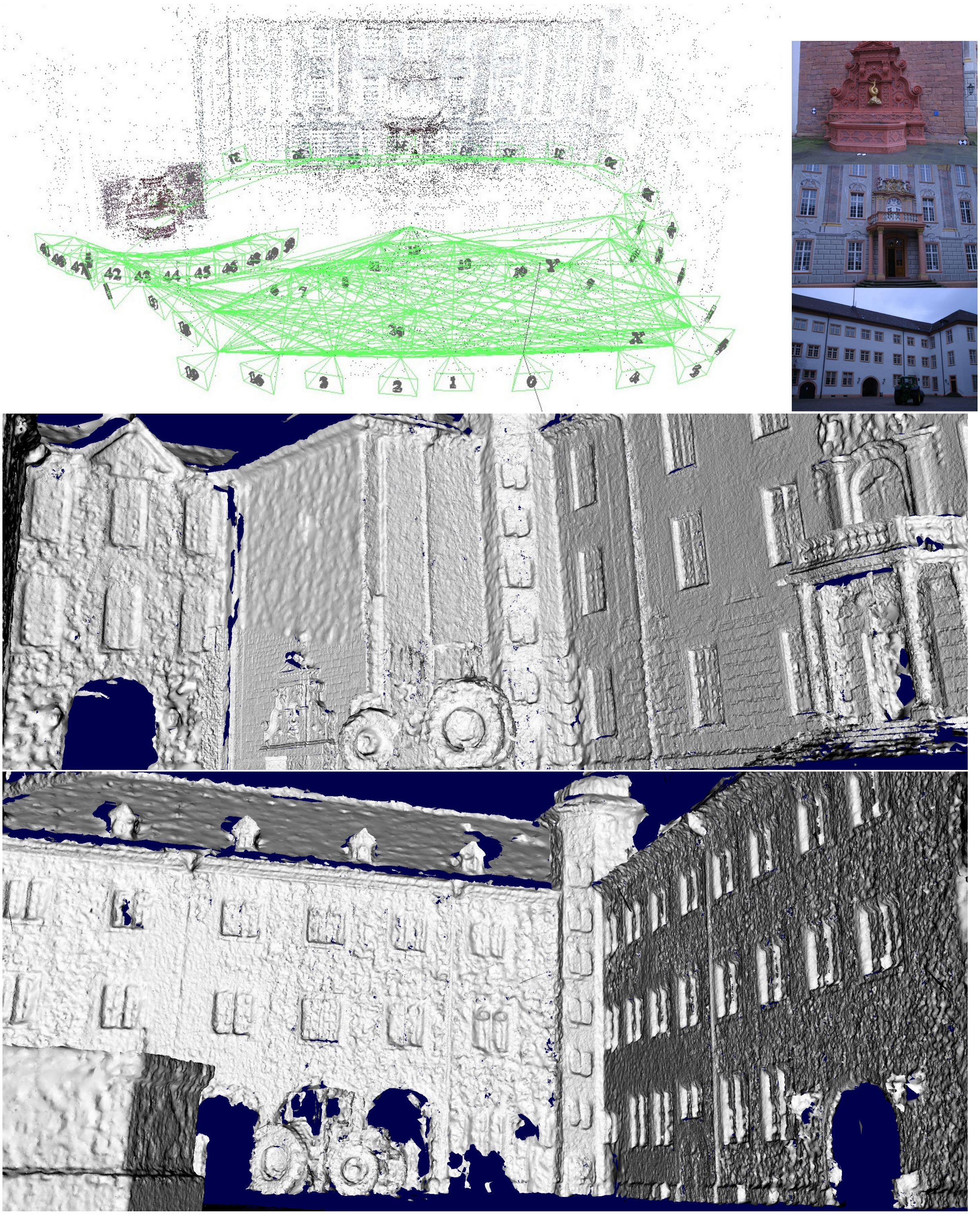


Figure A.6.: Combination of Ettlingen10, Ettlingen30 and EttlingenFountain.

Abbreviations for frequently cited conferences and journals

3DV : International Conference on 3D Vision (formerly: 3DIMPVT)
BMVC : British Machine Vision Conference
CVPR : IEEE Conference on Computer Vision and Pattern Recognition
ECCV : European Conference on Computer Vision
EUROGRAPHICS : Conference of the European Association for Computer Graphics
GCPR : German Conference on Pattern Recognition (formerly: DAGM)
ICCV : IEEE International Conference on Computer Vision
IJCV : International Journal of Computer Vision
PAMI : IEEE Transactions on Pattern Analysis and Machine Intelligence
SIGGRAPH : International Conference and Exhibition on Computer Graphics and Interactive Techniques

Bibliography

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M. and Szeliski, R. (2009): Building Rome in a day, *ICCV*.

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M. and Szeliski, R. (2011): Building Rome in a day, *Communications of the ACM* **54**(10): 105–112.

Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D. and Silva, C. T. (2003): Computing and rendering point set surfaces, *SIGGRAPH*.

Amanatides, J. and Woo, A. (1987): A fast voxel traversal algorithm for ray tracing, *EUROGRAPHICS*.

Bailer, C., Finckh, M. and Lensch, H. P. A. (2012): Scale robust multi view stereo, *ECCV*.

Bao, S. Y., Chandraker, M., Lin, Y. and Savarese, S. (2013): Dense object reconstruction with semantic priors, *CVPR*.

Bartelsen, J. (2012): *Orientierung von Bildverbänden mit großer Basis*, PhD thesis, Universität der Bundeswehr Munich.

Bartelsen, J., Mayer, H., Hirschmüller, H., Kuhn, A. and Michelini, M. (2012): Orientation and dense reconstruction of unordered terrestrial and aerial wide baseline image sets, *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume 1, 25–30.

Bentley, J. L. (1975): Multidimensional binary search trees used for associative searching, *ACM*.

Bleyer, M., Rhemann, C. and Rother, C. (2011): Patchmatch stereo - stereo matching with slanted support windows, *BMVC*.

Bodenmüller, T. (2009): *Streaming Surface Reconstruction from Real Time 3D Measurements*, PhD thesis, Technical University Munich.

Boykov, Y. and Kolmogorov, V. (2004): An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, *PAMI* **26**(9): 1124–1137.

Boykov, Y., Veksler, O. and Zabih, R. (2001): Fast approximate energy minimization via graph cuts, *PAMI* **23**(11): 1222–1239.

Bradski, G. and Kaehler, A. (2008): *Learning OpenCV (Computer Vision with the OpenCV library)*, O'Reilly.

Campbell, N. D., Vogiatzis, G., Hernández, C. and Cipolla, R. (2008): Using multiple hypotheses to improve depth-maps for multi-view stereo, *CVPR*.

Collins, R. T. (1996): A space-sweep approach to true multi-image matching, *CVPR*.

Coughlan, J. M. and Yuille, A. L. (1999): Manhattan world: Compass direction from a single image by Bayesian inference, *ICCV*.

Cremers, D. and Kolev, K. (2011): Multiview stereo and silhouette consistency via convex functionals over convex domains, *PAMI* **33**(6): 1161–1174.

Curless, B. L. (1997): *New Methods for Surface Reconstruction from Range Images*, PhD thesis, Stanford University.

Curless, B. and Levoy, M. (1996): A volumetric method for building complex models from range images, *SIGGRAPH*.

de Agapito, L., Hayman, E. and Reid, I. D. (1998): Self-calibration of a rotating camera with varying intrinsic parameters, *BMVC*, 883–893.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977): Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society, Series B* **39**(1): 1–38.

Duda, R. O., Hart, P. E. and Stork, D. G. (2000): *Pattern Classification*, Wiley.

Ernst, I. and Hirschmüller, H. (2008): Mutual information based semi-global stereo matching on the GPU, *4th International Symposium on Advances in Visual Computing*.

Fischler, M. A. and Bolles, R. C. (1981): Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM* **24**(6): 381–395.

Förstner, W. and Gülch, E. (1987): A fast operator for detection and precise location of distinct points, corners and centres of circular features, *ISPRS intercommission conference on fast processing of photogrammetric data*.

Förstner, W., , Dickscheid, T. and Schindler, F. (2009): Detecting interpretable and accurate scale-invariant keypoints, *ICCV*.

Frahm, J.-M. and Koch, R. (2003): Camera calibration with known rotation, *ICCV*.

Frahm, J.-M., Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., Jen, Y.-H., Dunn, E., Clipp, B., Lazebnik, S. and Pollefeys, M. (2010): Building Rome on a cloudless day, *ECCV*.

Fuhrmann, S. and Goesele, M. (2011): Fusion of depth maps with multiple scales, *SIGGRAPH Asia*.

Furukawa, Y. and Ponce, J. (2007): Accurate, dense, and robust multi-view stereopsis, *CVPR*.

Furukawa, Y. and Ponce, J. (2010): Accurate, dense, and robust multiview stereopsis, *PAMI* **32**(8): 1362–1376.

Furukawa, Y., Curless, B., Seitz, S. M. and Szeliski, R. (2009): Manhattan-world stereo, *CVPR*.

Furukawa, Y., Curless, B., Seitz, S. M. and Szeliski, R. (2010): Towards internet-scale multi-view stereopsis, *CVPR*.

Gallup, D., Frahm, J.-M. and Pollefeys, M. (2010a): Piecewise planar and non-planar stereo for urban scene reconstruction, *CVPR*.

Gallup, D., Pollefeys, M. and Frahm, J.-M. (2010b): 3D reconstruction using an n-layer heightmap, *GCPR*.

Goesele, M., Curless, B. and Seitz, S. M. (2006): Multi-view stereo revisited, *CVPR*.

Google (2014): Google Earth, google.de/intl/de/earth/.

Häne, C., Zach, C., Cohen, A., Angst, R. and Pollefeys, M. (2013): Joint 3D scene reconstruction and class segmentation, *CVPR*.

Hannah, M. J. (1974): *Computer matching of areas in stereo images.*, PhD thesis, Stanford University.

Harris, C. and Stephens, M. (1988): A combined corner and edge detector, *Fourth Alvey Vision Conference*, 147–151.

Hilton, A. and Illingworth, J. (1997): Multi-resolution geometric fusion, *3DV*.

Hilton, A., Stoddart, A. J., Illingworth, J. and Windeatt, T. (1996): Reliable surface reconstruction from multiple range images, *ECCV*.

Hirschmüller, H. (2003): *Stereo Vision based mapping and immediate virtual walkthroughs*, PhD thesis, De Montfort University.

Hirschmüller, H. (2005): Accurate and efficient stereo processing by semi-global matching and mutual information, *CVPR*.

Hirschmüller, H. (2008): Stereo processing by semiglobal matching and mutual information, *PAMI* **30**(2): 328–341.

Hirschmüller, H. (2011): Semi-global matching - motivation, developments and applications, *Photogrammetric Week*.

Hirschmüller, H. and Scharstein, D. (2009): Evaluation of stereo matching costs on images with radiometric differences, *PAMI* **31**(9): 1582–1599.

Hoppe, H., DeRose, T. and Duchamp, T. (1992): Surface reconstruction from unorganized points, *SIGGRAPH*.

Hornung, A. and Kobbelt, L. (2006): Hierarchical volumetric multi-view stereo reconstruction of manifold surfaces based on dual graph embedding, *CVPR*.

Hornung, A., Zeng, B. and Kobbelt, L. (2008): Image selection for improved multi-view stereo, *CVPR*.

Hu, X. and Mordohai, P. (2012): Least commitment, viewpoint-based, multi-view stereo, *3DV*.

Jancosek, M., Shekhovtsov, A. and Pajdla, T. (2009): Scalable multi-view stereo, *3DV*.

Kanade, T. and Okutomi, M. (1994): A stereo matching algorithm with an adaptive window: Theory and experiment, *PAMI* **16**(9): 920–932.

Kazhdan, M., Bolitho, M. and Hoppe, H. (2006): Poisson surface reconstruction, *EUROGRAPHICS*.

Kim, J., Kolmogorov, V. and Zabih, R. (2003): Visual correspondence using energy minimization and mutual information, *ICCV*.

Kolev, K., Brox, T. and Cremers, D. (2012): Fast joint estimation of silhouettes and dense 3D geometry from multiple images, *PAMI* **34**(3): 493–505.

Kolmogorov, V. and Zabih, R. (2002): Multi-camera scene reconstruction via graph cuts, *ECCV*.

Konolige, K. (1997): Improved occupancy grids for map building, *Autonomous Robots* **4**: 351–367.

Kuhn, A., Hirschmüller, H. and Mayer, H. (2013): Multi-resolution range data fusion for multi-view stereo reconstruction, *GCPR*.

Laurentini, A. (1993): The visual hull concept for silhouette-based image understanding, *PAMI* **16**(2): 150–162.

Lhuillier, M. and Quan, L. (2005): A quasi-dense approach to surface reconstruction from uncalibrated images, *PAMI* **27**(3): 418–433.

Lindeberg, T. (1993): Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention, *IJCV* **11**(3): 283–318.

Lorensen, W. E. and Cline, H. E. (1987): Marching cubes: A high resolution 3D surface construction algorithm, *SIGGRAPH*.

Lowe, D. G. (2004): Distinctive image features from scale-invariant keypoints, *IJCV*.

Matthies, L. (1992): Toward stochastic modeling of obstacle detectability in passive stereo range imagery, *CVPR*.

Matthies, L., Szeliski, R. and Kanade, T. (1989): Kalman filter-based algorithms for estimating depth from image sequence, *IJCV* **3**(3): 209–238.

Mauro, M., Riemenschneider, H., Gool, L. V. and Leonardi, R. (2013): Overlapping camera clustering through dominant sets for scalable 3D reconstruction, *BMVC*.

Mayer, H., Bartelsen, J., Hirschmüller, H. and Kuhn, A. (2011): Dense 3D reconstruction from wide baseline image sets, *15th International Workshop on Theoretical Foundations of Computer Vision*.

Meerbergen, G. V., Vergauwen, M., Pollefeys, M. and Gool, L. V. (2002): A hierarchical symmetric stereo algorithm using dynamic programming, *IJCV* **47**(1-3): 275–285.

Merrell, P., Akbarzadeh, A., Wang, L., Mordohai, P., Frahm, J.-M., Yang, R., Nistér, D. and Pollefeys, M. (2007): Real-time visibility-based fusion of depth maps, *CVPR*.

Meyer, F. (1992): Color image segmentation, *4th Conference on Image Processing and its Applications*.

Mitchell, D. P. (1987): Generating antialiased images at low sampling densities, *SIGGRAPH*.

Molton, N. and Brady, M. (2000): Practical structure and motion from stereo when motion is unconstrained, *IJCV* **39**(1): 5–23.

Mücke, P., Klowsky, R. and Goesele, M. (2011): Surface reconstruction from multi-resolution sample points, *International Workshop on Vision, Modeling and Visualization*.

Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohli, P., Shotton, J., Hodges, S. and Fitzgibbon, A. (2011): Kinectfusion: Real-time dense surface mapping and tracking, *International Symposium on Mixed and Augmented Reality*.

Nistér, D. (2003): An efficient solution to the five-point relative pose problem, *CVPR*.

Ohtake, Y., Belyaev, A., Alexa, M., Turk, G. and Seidel, H.-P. (2003): Multi-level partition of unity implicits, *SIGGRAPH*.

Pavan, M. and Pelillo, M. (2007): Dominant sets and pairwise clustering, *PAMI* **29**(1): 167–172.

Pito, R. (1996): Mesh integration based on co-measurements, *International Conference on Image Processing*.

Pollefeys, M., Koch, R. and Gool, L. V. (1999): Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters, *IJCV* **32**(1): 7–25.

Pollefeys, M., Nistér, D., Frahm, J. M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S. J., Merrell, P., Salmi, C., Sinha, S., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G. and Towles, H. (2008): Detailed real time urban 3D reconstruction from video, *IJCV* **78**(2-3): 143–167.

Pons, J.-P., Keriven, R. and Faugeras, O. (2007): Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score, *IJCV* **72**(2): 179–193.

Rudin, L. I., Osher, S. and Fatemi, E. (1992): Nonlinear total variation based noise removal algorithms, *Physica D*.

Sachs, L. and Hedderich, J. (2006): *Angewandte Statistik: Methodensammlung mit R*, Springer.

Scharstein, D. (2014a): Middlebury multiview benchmark, vision.middlebury.edu/mview.

Scharstein, D. (2014b): Middlebury stereo benchmark, vision.middlebury.edu/stereo.

Scharstein, D. and Pal, C. (2007): Learning conditional random fields in stereo, *CVPR*.

Scharstein, D. and Szeliski, R. (2002): A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *IJCV* **47**(1): 7–42.

Scharstein, D. and Szeliski, R. (2003): High-accuracy stereo depth maps using structured light, *CVPR*.

Schroers, C., Zimmer, H., Valgaerts, L., Bruhn, A., Demetz, O. and Weickert, J. (2012): Anisotropic range image integration, *GCPR*.

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D. and Szeliski, R. (2006): A comparison and evaluation of multi-view stereo reconstruction algorithms, *CVPR*.

Steinbrücker, F., Kerl, C., Sturm, J. and Cremers, D. (2013): Large-scale multi-resolution surface reconstruction from RGB-D sequences, *ICCV*.

Strecha, C. (2014): Epfl multiview benchmark, cvlab-www.epfl.ch/ strecha/multiview/.

Strecha, C., von Hansen, W., Gool, L. J. V., Fua, P. and Thoennessen, U. (2008): On benchmarking camera calibration and multi-view stereo for high resolution imagery, *CVPR*.

Sturm, J., Bylow, E., Kahl, F. and Cremers, D. (2013): CopyMe3D: Scanning and printing persons in 3D, *GCPR*.

Sun, J., Zheng, N.-N. and Shum, H.-Y. (2003): Stereo matching using belief propagation, *PAMI* **25**(7): 787–800.

Szeliski, R. (2011): *Computer Vision (Algorithms and Applications)*, Springer.

Thrun, S., Burgard, W. and Fox, D. (2005): *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, The MIT Press.

Tola, E., Lepetit, V. and Fua, P. (2010): Daisy: An efficient dense descriptor applied to wide baseline stereo, *PAMI* **32**(5): 815–830.

Turk, G. and Levoy, M. (1994): Zippered polygon meshes from range images, *SIGGRAPH*.

Tyleček, R. and Sara, R. (2009): Refinement of surface mesh for accurate multi-view reconstruction, *Modeling-3D workshop, ACCV*.

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

Vogiatzis, G. and Hernández, C. (2011): Video-based, real-time multi-view stereo, *Image and Vision Computing* **29**(7): 434–441.

Vogiatzis, G., Hernández, C., Torr, P. H. S. and Cipolla, R. (2007): Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency, *PAMI* **29**(12): 2241–2246.

Vogiatzis, G., Torr, P. H. S. and Cipolla, R. (2005): Multi-view stereo via volumetric graph-cuts, *CVPR*.

Vu, H. H. (2011): *Large-scale and high-quality Multi-view stereo*, PhD thesis, École des Ponts ParisTech, France.

Vu, H.-H., Labatut, P., Pons, J.-P. and Keriven, R. (2012): High accuracy and visibility-consistent dense multiview stereo, *PAMI* **34**(5): 889–901.

Weibull, J. W. (1997): *Evolutionary Game Theory*, The MIT Press.

Zabih, R. and Woodfill, J. (1994): Non-parametric local transforms for computing visual correspondence, *ECCV*.

Zach, C., Pock, T. and Bischof, H. (2007): A globally optimal algorithm for robust TV-L1 range image integration, *ICCV*.

Zaharescu, A., Cagniart, C., Ilic, S., Boyer, E. and Horaud, R. (2008): Camera-clustering for multi-resolution 3–d surface reconstruction, *ECCV*.