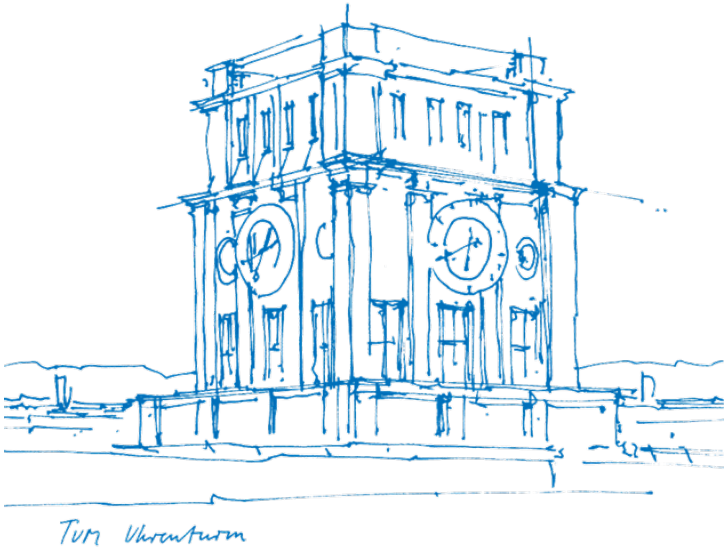
Chair of Geoinformatics
TUM School of Engineering and Design
Technical University of Munich

The Impact of Point Cloud Density and Accuracy on the Quality of 3D Building Reconstruction

Florian L. Faltermeier



Chair of Geoinformatics
TUM School of Engineering and Design
Technical University of Munich

The Impact of Point Cloud Density and Accuracy on the Quality of 3D Building Reconstruction

Florian L. Faltermeier

Thesis for the attainment of the academic degree

Master of Science (M. Sc.)

at the TUM School of Engineering and Design of the Technical University of Munich.

Examiner:
Prof. Dr. Thomas H. Kolbe

Supervisor:
Asst. Prof. Dr. Filip Biljecki (National University of Singapore)

Submitted:
Munich, 01.11.2024

I hereby declare that this thesis is entirely the result of my own work except where otherwise indicated. I have only used the resources given in the list of references.

Florian Faltermeier

Munich, 01.11.2024 Florian L. Faltermeier

Abstract

Three-dimensional (3D) city models are critical in many applications, with 3D building models at their core. To meet the demand for up-to-date digital information on urban changes, the frequency of model updates must increase. However, automatic, large-scale, and accurate 3D building reconstruction remains challenging and is the focus of active research. Quality improvements rely on its precise quantification using suitable metrics. Most research has focused on the comparative evaluation of different reconstruction methods with respect to their output quality. The influence of input data quality, however, has received less attention. LiDAR point clouds are an essential data source for 3D reconstruction. Higher resolution and accuracy can yield better model quality, but the quantitative relationship has not been established. This thesis addresses it by simulating airborne laser scanning on a 3D scene of over 1000 existing building models to produce point clouds at twelve density and eleven accuracy levels, yielding 132 scenarios. Using an established method, the 3D building models are reconstructed from each point cloud at three levels of detail. The reconstruction parameters are adjusted to each scenario: Bayesian optimization is applied to identify values that minimize the geometric distance between reconstructed and input models. Reconstruction quality is evaluated with the input models as reference and further analyzed by building subgroups with distinct geometric and semantic properties. Findings indicate that increased accuracy improves reconstruction quality, with diminishing returns at higher accuracy levels. Point density enhances quality up to 16 pt m^{-2} and more significantly at lower levels. Model complexity, volume, flat roof share, and year of construction influence reconstruction quality, while building use does not. The results are interpreted with attention to the challenge of defining 3D building model quality. The methodology's limitations and directions for future research including further point cloud properties, reconstruction methods, and input data sources are discussed.

Zusammenfassung

Dreidimensionale (3D) Stadtmodelle sind in vielen Anwendungsbereichen von entscheidender Bedeutung, wobei 3D-Gebäudemodelle eine zentrale Rolle spielen. Um die Nachfrage nach aktuellen digitalen Informationen über urbane Veränderungen zu decken, muss die Häufigkeit der Modellaktualisierungen zunehmen. Die automatische, großflächige und zugleich präzise Rekonstruktion von 3D-Gebäudemodellen bleibt jedoch herausfordernd und ist Gegenstand aktiver Forschung. Qualitätsverbesserungen setzen eine genaue Quantifizierung mithilfe geeigneter Metriken voraus. Die meisten Forschungsarbeiten konzentrieren sich auf die vergleichende Bewertung verschiedener Rekonstruktionsmethoden in Bezug auf die Qualität ihrer Ergebnisse. Der Einfluss der Eingabedatenqualität hat hingegen weniger Beachtung gefunden. LiDAR-Punktwolken sind eine wesentliche Datenquelle für die 3D-Rekonstruktion. Eine höhere Auflösung und Genauigkeit können zu einer besseren Modellqualität führen, aber der quantitative Zusammenhang ist bisher nicht geklärt. Diese Arbeit befasst sich mit diesem Problem, indem Airborne Laserscanning einer 3D-Szene mit über 1000 bestehenden Gebäudemodellen simuliert wird, um Punktwolken in zwölf Dichte- und elf Genauigkeitsstufen zu erzeugen, was 132 Szenarien ergibt. Mithilfe einer etablierten Rekonstruktionsmethode werden aus jeder Punktwolke die 3D-Gebäudemodelle rekonstruiert. Die Rekonstruktionsparameter werden für jedes Szenario angepasst: Mittels Bayes'scher Optimierung werden Parameterwerte identifiziert, die den geometrischen Abstand zwischen rekonstruierten und Eingabemodellen minimieren. Die Rekonstruktionsqualität wird in Bezug zu den Eingabemodellen bewertet und weiter analysiert, indem Untergruppen mit unterschiedlichen geometrischen und semantischen Eigenschaften untersucht werden. Die Ergebnisse zeigen, dass eine höhere Genauigkeit die Rekonstruktionsqualität verbessert, wobei der Effekt bei mit höherer Genauigkeit abnimmt. Die Punktdichte verbessert die Qualität bis zu einem Wert von $16\,\mathrm{pt\,m^{-2}}$, mit einem stärkeren Effekt bei niedrigeren Dichten. Modellkomplexität, Volumen, Flachdachanteil und Baujahr beeinflussen die Rekonstruktionsqualität, die Gebäudenutzung hingegen nicht. Die Ergebnisse werden unter Berücksichtigung der Herausforderung interpretiert, die Qualität von 3D-Gebäudemodellen zu definieren. Die Grenzen der Methodik sowie Richtungen für zukünftige Untersuchungen werden diskutiert, einschließlich weiterer Punktwolkeneigenschaften, Rekonstruktionsmethoden und Eingabedatenquellen.

Acknowledgments

I would like to express my sincere gratitude to my examiner, Prof. Dr. Thomas Kolbe, Chair of Geoinformatics at the Technical University of Munich, and my supervisor, Asst. Prof. Dr. Filip Biljecki, head of the Urban Analytics Lab (UAL) at the National University of Singapore (NUS), for their continuous support throughout all phases of this thesis project. Their openness to my research interests, particularly their encouragement to pursue my research visit to UAL, and the guidance they provided in the application processes at NUS and for a DAAD scholarship were invaluable.

Filip Biljecki guided the development of my research topic and generously shared his expertise through numerous, in-depth discussions, providing insights that have shaped this project. Thomas Kolbe offered an essential organizational foundation and gave invaluable feedback at every critical stage.

I am grateful for the opportunity to meet the wonderful members of UAL and the other visiting scholars, who welcomed me so openly and integrated me into their work and activities, which made me feel welcome and at home in Singapore. They enriched this project through many thoughtful discussions.

I am also thankful to the many friends that I have made at different stages of my life—in the wider Passau region, Tübingen, Freising, and Munich, among other places—who are now spread throughout the world, and who have accompanied and supported me throughout the last years and during this last phase before graduation. Their friendship is a source of strength and meaning to me.

Finally, I am deeply grateful to my parents, Iris and Hans, and my sister Miriam, who stood by me at all times and in all circumstances throughout this journey.

Contents

Abstract **vii**

Zusammenfassung **ix**

Acknowledgments **xi**

List of Figures **xv**

List of Tables **xix**

1 Introduction **1**

2 Data and Methods **5**
2.1 Conceptual Basis . . . 5
2.1.1 Foundational Concept . . . 5
2.1.2 Definitions . . . 5
2.2 Data Pipeline . . . 7
2.2.1 Simulation of Airborne Laser Scanning . . . 7
2.2.2 Point Cloud Processing . . . 9
2.2.3 Reconstruction of 3D Building Models . . . 10
2.2.4 Evaluation of the Reconstructed Models . . . 11
2.2.5 Pipeline Implementation . . . 13
2.3 Optimization of the Reconstruction Parameters . . . 13
2.3.1 The Case for an Optimization . . . 13
2.3.2 Optimization Method . . . 14
2.3.3 Parameters and Ranges for Optimization . . . 16
2.4 Experiment Design . . . 17
2.4.1 Input Data Selection . . . 17
2.4.2 Scenarios . . . 21
2.4.3 Pipeline Configuration . . . 22

3 Results and Analysis **23**
3.1 Point Clouds . . . 23
3.1.1 Filtering and Merging of the Point Cloud Strips . . . 23
3.1.2 Point Density and Random Error Variations . . . 23
3.2 Reconstruction Optimization . . . 26
3.2.1 Optimized Reconstruction Parameter Values . . . 26
3.2.2 Optimized Quality Metric . . . 28
3.2.3 Effects of the Single-Objective Optimization on Other Metrics . . . 28
3.2.4 Selection of Parameter Values for the Reconstruction . . . 29
3.3 3D Building Reconstruction . . . 30
3.3.1 Reconstruction Results . . . 30
3.3.2 Evaluation of the Reconstructed 3D Building Models . . . 31
3.3.3 Evaluation of the Reconstructed 3D Building Models by Groups . . . 36
3.3.4 Visualization of the Reconstructed 3D Building Models . . . 37

4 Discussion **43**
4.1 Overview . . . 43
4.2 Technical and Methodological Limitations and Opportunities . . . 43
4.2.1 Design of the Data Pipeline: Limitations of Scale and Variety . . . 43
4.2.2 Characteristics of the Reconstruction Method . . . 44
4.2.3 Evaluation . . . 44
4.2.4 Optimization of the Reconstruction Parameters . . . 45
4.3 Interpretation and Implications of the Results . . . 46
4.3.1 Facets of Quality and Challenges in Their Quantification . . . 46
4.3.2 Research Question and Hypotheses . . . 47
4.3.3 Reconstruction Quality as a Function of Point Cloud Properties . . . 48
4.4 Opportunities for Future Work . . . 48
4.4.1 Point Cloud Properties . . . 48
4.4.2 Reconstruction Methods . . . 49
4.4.3 Data Sources . . . 49

5 Conclusions **51**

References **53**

Appendices **61**

A Figures **61**
A.1 UML Class Diagrams of the Data Pipeline . . . 61
A.2 Reconstruction Parameters . . . 65

List of Figures

2.1 The foundational idea underlying the methodology: A dataset of existing building models is converted into multiple point clouds that differ along the two dimensions of density and accuracy. Based on each of these, a set of new building models is reconstructed such that there is one output model for each input model. This enables a quantitative evaluation of the reconstructed building models in comparison to the original ones. 6

2.2 The components of a simulated airborne laser scanning survey in HELIOS++: Scanner, platform, and scene. Legs define a moving platform's trajectory and enable location-dependent scanner settings. (From Winiwarter et al., 2022; *CreativeCommons CC BY 4.0* 2024) . . . 8

2.3 The steps to 3D building reconstruction taken by the software Geoflow. (From Peters et al., 2022; *CreativeCommons CC BY-NC-ND 4.0* 2024) 11

2.4 A flowchart of the data pipeline outlining the steps, input data, configurations, and output data involved. 14

2.5 The tiles into which the 3DBAG dataset is subdivided for download. (Background map © OpenStreetMap) 19

2.6 Distribution of building use types in the Netherlands according to the Dutch cadastre (BAG). 19

2.7 A map of the city center of Utrecht demonstrating the local subdivision of the 3DBAG dataset into tiles. The pie charts visualize the building use distribution in each tile as computed from the cadastre BAG. Further given are the tile number, the number of buildings per tile, and the standard deviation of the normalized building use distribution (lower numbers indicate a more even distribution of use types). The building models in tile 10/492/594 serve as input dataset. (Background map © OpenStreetMap) 20

2.8 A visualization of the input dataset from the Dutch 3D cadastre (3DBAG) showing the building models in the city center of Utrecht that are located in the 3DBAG tile 10/492/594. . . . 20

3.1 Illustration of the point cloud filtering step of two adjacent swaths. Top: The point clouds of the two swaths as obtained from the ALS simulation with 50 % overlap in coverage. The trajectory of the laser scanner in each cloud runs parallel to the long side at the center of the short side. Bottom: The same point clouds after the filtering step that ensures a constant point density after merging while retaining unique information from both clouds by discarding duplicate points beyond the center line between the trajectories but keeping points that complement the other cloud. The points are colored by the two classes *building* and *ground*. The point density is 32 pt m^{-2}. 24

3.2 The final point cloud from the simulated ALS survey of the input 3D scene after filtering and merging the four individual swaths. The point color indicates the source swath. This point cloud has a density of 32 pt m^{-2}. 25

3.3 Point clouds covering a roof at five different density levels: 4, 8, 16, 32, and 64 pt m^{-2}. These point clouds are free of random error. 25

3.4 Point clouds covering a roof at six levels of random error: 0 %, 20 %, 40 %, 60 %, 80 %, and 100 % of the maximum horizontal and vertical errors, which are 1 m and 0.3 m, respectively. The point density is 16 pt m^{-2}. 25

3.5 The distribution of the optimized values of the seven reconstruction parameters across the point density and error levels. 27

3.6 Results of the optimization of the reconstruction parameters across all scenarios in terms of two metrics: RMS minimum distance, which was the optimization target, and building model complexity. In all cases, the results are based on the reconstruction of the subset of 100 buildings used for the optimization, and the mean across all buildings is presented. The plots to the left show the results obtained in the best performing optimization iteration for each scenario. For comparison, the center plots show the results from a reconstruction of the same subset using the default reconstruction parameters (see table 2.1). The plots to the right demonstrate the difference between the two. Note that the vertical scales of the plots are not aligned to emphasize the relations in the individual plots. 29

3.7 The correlation between the optimization target metric (RMS minimum distance) and building model complexity as obtained from the optimization of one scenario (11.3 pt m^{-2}, 0 % error). The Pareto optimal, best (in terms of the target metric), and selected solutions are highlighted. The selected solution is the one within 10 % of the best one that has the lowest complexity. 30

3.8 The share of the total number of building models present in the input dataset (1079) that was reconstructed successfully in each scenario. 31

3.9 Evaluation of the reconstructed 3D building models in three LODs in terms of three metrics: RMS minimum distance, Hausdorff distance, and building model complexity ratio (output to input model face count). In all cases, the mean across all building models is presented. The error bars show 5 % of the standard deviation in each scenario, which gives an indication of the high spread. 32

3.10 Mean difference in three roof height metrics between the reconstructed and the input 3D building models: Minimum, 70th percentile, and maximum elevation of all points used to reconstruct the LOD 2.2 roof. The 70th percentile height is used as reference of the default LOD 1.2 models. The error bars show 5 % of the standard deviation in each scenario, which gives an indication of the high spread. 34

3.11 Building-level distributions of two quality metrics at two LODs and four point density levels (4, 8, 16, and 32 pt m^{-2}) across all error levels: RMS minimum distance at LODs 1.2 and 2.2, and the building model complexity ratio at LOD 2.2. The labels to the right indicate the respective point densities. The vertical axes represent the kernel density estimate. Note that outliers, defined as values with a modified Z-score greater than 3.5, are not included in the calculation to facilitate the reproduction of the central part of the distributions in each plot. For the same purpose, neither the horizontal nor the vertical axes' scales are aligned. 35

3.12 Building-level distributions of the RMS minimum distance at 11.3 pt m^{-2} density and an error level of 0 %, 30 %, and 60 %, differentiated by groups of properties: Geometric properties of the scanned LOD 2.2 input building models (number of faces, volume, and ratio of flat to total roof area), and semantic properties from the Dutch cadastre BAG (year built and building use). The legend of numerical properties provides only the upper limit of each bin; its lower limit is the upper limit of the previous bin. Slight variations in the overall shape of the distribution result from taking the sum of the groups' kernel density estimates. Note that the overall shape of the distributions in the bottom row differs from the others because many buildings have multiple use types and therefore contribute to more than one group. As before, outliers, defined as values with a modified Z-score greater than 3.5, are not included in the calculation to facilitate a better reproduction of the central part of the distribution. . . 38

3.13 The original building model from the Dutch 3D cadastre (3DBAG) of the building in 28 Hamburgerstraat, Utrecht, that served as input for the generation of the point clouds used to reconstruct the building models presented in figures 3.14 and 3.15. Its identifier in the Dutch cadastre (BAG) is 0344100000069160. 39

3.14 3D building models reconstructed from point clouds at 2, 2.8, 4, and 5.7 pt m^{-2} density and error levels between 0 % and 100 % in steps of 20 %. The numbers to the bottom left of each model indicate its RMS minimum distance [m] (top) from the input building model and its face number (bottom), including the ratio to that of the input model in brackets. The corresponding input building model from the 3DBAG is demonstrated in figure 3.13. 40
3.15 3D building models reconstructed from point clouds at 8, 16, 32, and 64 pt m^{-2} density and error levels between 0 % and 100 % in steps of 20 %. The numbers to the bottom left of each model indicate its RMS minimum distance [m] (top) from the input building model and its face number (bottom), including the ratio to that of the input model in brackets. The corresponding input building model from the 3DBAG is demonstrated in figure 3.13. 41

A.1 Simplified UML diagram of the data pipeline classes and their relationships. Unless specified otherwise, multiplicities are 1. A UML diagram of the classes relating to the Evaluator class is provided in figure A.2. 62
A.2 Simplified UML diagram of the classes developed for the evaluation of the experiment and their relationships. Unless specified otherwise, multiplicities are 1. The class Evaluator links this diagram to the one in figure A.1. 63
A.3 The distribution of the seven reconstruction parameters' values from the optimization that were selected for the experiment, across the point density and error levels. For each scenario, they are taken from the optimization iteration that delivered the lowest mean model complexity among all iterations that achieved a RMS minimum distance within 10 % of the best one. 65

List of Tables

2.1 A subset of parameters of the reconstruction software Geoflow for the configuration of individual reconstruction steps. Since there is no official documentation of the parameters, their descriptions were paraphrased from configuration files on GitHub by Peters and Dukai (2024a,b). The default values are taken from the first of these, according to which they are optimized for a point density of 8 pt m^{-2}. If ranges are given, they were used to optimize the corresponding parameters. 15
2.2 The building use types in the Dutch cadastre (BAG) and their English translation. 18
2.3 The point cloud density levels and corresponding point spacing as set in the scenarios of the experiment. 21
2.4 The levels of random error of the point clouds as set in the scenarios of the experiment: Vertical error $RMSE_V$, horizontal error $RMSE_H$, and corresponding errors in the x and y coordinates. 21

3.1 Point cloud densities as targeted by means of point spacing and corresponding ALS survey parameters, and as measured in the resulting point clouds. 26

1 Introduction

Methods to automatically reconstruct three-dimensional (3D) building models from remotely sensed information were first explored and developed three decades ago (Haala and Hahn, 1995; Fischer et al., 1998; Haala et al., 1998; Maas and Vosselman, 1999). The fact that it is still an active field of research today speaks to the success in the adoption of 3D building and city models in a wide variety of applications. It may also point to ongoing challenges, such as those of scale, efficiency, degree of automation, and quality (Xu and Stilla, 2021), as well as technological innovations opening up new possibilities, both of which give rise to the development of new and improved methods.

Moreover, the notion of urban digital twins has gained momentum in recent years. The usefulness of the term has been debated (Batty, 2018; Tomko and Winter, 2019), and current research is therefore concerned with the identification of an unequivocal definition as well as opportunities and challenges facing their adoption in practice (Ferré-Bigorra et al., 2022; Tzachor et al., 2022; Lei et al., 2023; Ye et al., 2023). However, there is agreement that 3D city models are a core component of urban digital twins, depending on the envisioned use case and implementation.

The applications of 3D city and building models are as numerous as they are manifold, and they have been comprehensively surveyed and categorized in a number of studies over the years (Batty et al., 2000; Billen et al., 2014; Biljecki et al., 2015; Willenborg et al., 2018; Ketzler et al., 2020; Alva et al., 2022). Use cases can be encountered in urban and regional planning (Schrotter and Hürzeler, 2020), architectural and civil engineering (Kolbe and Donaubauer, 2021), environmental analysis and sustainability (Chen et al., 2020; Xu et al., 2021), disaster management and emergency response (Macatulad and Biljecki, 2024), transportation and mobility planning (Beil and Kolbe, 2024), cultural heritage preservation and tourism (Pepe et al., 2021), real estate and land administration (Paasch and Paulsson, 2023), and public health and safety (Padsala et al., 2024).

While 3D city models may contain representations of any objects that are part of an urban environment, 3D building models are a key component. Furthermore, as urban environments continuously evolve, a virtual representation in the shape of a 3D city model must be updated regularly if it is to reflect these changes. The generation and continuous maintenance of a dataset of 3D building models is the task of 3D building reconstruction, which is one reason for the sustained efforts in its advancement.

Methods for 3D building reconstruction can be distinguished along multiple dimensions (Haala and Kada, 2010; Xu and Stilla, 2021), and the data source lies at the foundation. Most established approaches for the reconstruction of detailed building models rely directly two kinds of remotely-sensed information: Either aerial or satellite imagery, or Light Detection and Ranging (LiDAR) point clouds obtained through laser scanning surveys. In the former case, building features can either be identified directly in the images or stereo image pairs can be used to infer 3D information from the two-dimensional (2D) imagery and translate it into a photogrammetric point cloud. Notably, methods have been developed that are independent of using these data sources and rely solely, for instance, on 2D building footprints and semantic information, which allow the inference of a third dimension with good accuracy (Biljecki and Dehbi, 2019).

A further differentiation of reconstruction methods is possible with respect to their algorithmic approach to translate the unstructured input data into a topologically correct and geometrically accurate 3D building model. A common concept distinguishes reconstruction methods by the degree to which they are model-driven or data-driven (Tarsha-Kurdi et al., 2007; Vosselman and Maas, 2014; Jung and Sohn, 2017; Wang et al., 2018), also referred to as top-down and bottom-up approaches, respectively. Model-driven approaches utilize the fact that most buildings share similar features from a limited set of common geometries, mainly with respect to the roof structure. In such methods, the reconstruction is, for instance, based on a set of roof primitives that are fitted to the input data. This works particularly well for simple buildings, but even more complex buildings can be represented by enabling combinations of roof primitives.

Data-driven approaches, on the other hand, impose fewer prior assumptions and therefore fewer constraints on the reconstruction process, which results in building models that are closer to the source data. In the case of point clouds, this implies the segmentation or delineation of basic geometric features like lines or surfaces in the data by means of, for instance, the region growing algorithm, the random sample consensus (RANSAC) algorithm, or the alpha-shape algorithm. Subsequently, the identified features are regularized such that they can be assembled into a consistent 3D building model.

Both approaches have advantages and drawbacks. Model-driven approaches facilitate a better conformity and validity of the resulting models and they are less sensitive to errors in the input data. In contrast, data-driven approaches are more flexible in representing different geometries, which can deliver more detailed and accurate models. However, they are more prone to gaps or outliers in the input data and may reconstruct overly complex models. The distinction between model-driven and data-driven methods is not categorical and, in practice, most reconstruction methods fall along a continuum between these opposing paradigms.

Moreover, 3D building reconstruction methods can be distinguished by characteristics of their output data. First, different approaches to the geometric representation of the reconstructed models are used (Wang et al., 2018; Xu and Stilla, 2021): In Constructive Solid Geometry (CSG), building models are composed of basic volumetric primitives by means of set operations like union and intersection. In boundary representation, on the other hand, 3D solids are represented by their 2D boundaries, which are commonly restricted to planar surfaces. Second, 3D city and building models can be encoded in a multitude of formats and data models. Purely geometric models can be stored in data formats related to Computer-Aided Design (CAD) or 3D mesh model formats. However, many applications require additional information about the represented buildings. The data model CityGML accommodates these requirements by enabling storage and exchange of semantic 3D city models (Kolbe, 2009; Kolbe et al., 2021). Third, the reconstructed 3D building models may differ in their level of detail (LOD). In the context of semantic 3D city models, the concept of the LOD was first established with the specification of CityGML and comprised five different levels: LOD0 to LOD4, in order of increasing detail (Kolbe, 2009). Biljecki et al. (2016) suggested a refined classification that subdivides each LOD into four sub-levels, which is frequently applied.

In the past decades, multiple studies have compared the quality of 3D building reconstruction between different reconstruction methods using the same input data (Kaartinen et al., 2005; Rottensteiner et al., 2012; Rottensteiner et al., 2014). One goal of such studies is to investigate the advantages and shortcomings of different reconstruction methods and, ultimately, to translate this knowledge into the development of reconstruction methods that either produce higher-quality models or can be adapted more specifically to different use cases. This corresponds to a model-centric approach to advance 3D building reconstruction.

Fewer studies exist that explore how the quality of the input data, and of point clouds in particular, affects the reconstruction quality while controlling for the reconstruction method. Often, observations in this respect are merely communicated as a side-product of the investigation of a different research problem. However, several studies focus on the effect of point cloud quality on related applications: Dreek et al. (2020) examined the influence of point cloud density on 2D building outline extraction. They compare results obtained with five different density levels, which are achieved by sub-sampling a point cloud, to reference footprints and find a limited impact. Rebolj et al. (2017) conducted an experiment to establish quality criteria for point clouds for the purpose of automated construction monitoring, which involves comparing an As-Built point cloud with an As-Designed Building Information Model (BIM). They identify quality thresholds in terms of point density, accuracy, and coverage to detect building elements of different sizes. Oude Elberink and Vosselman (2011) explored methods to analyze the quality of 3D building models reconstructed from LiDAR point clouds with two different density levels. However, they only evaluate the model quality with respect to the input point cloud and by means of internal model validity measures. They note therefore that a complete quality assessment should, if possible, include a comparison to reference data, which is difficult to achieve in practice. Notably, they mention data gaps, caused by occlusions, as a third quality criterion besides point density and accuracy.

Despite these efforts, there is a lack of a data-centric perspective on the quality of 3D building reconstruction: Even though it has frequently been observed that the properties of an input point cloud significantly

determine the reconstruction outcome, no study has systematically explored the quality of reconstructed 3D building models as a function of the point cloud quality by means of a quantitative comparison to the ground truth buildings. Therefore, this study aims to address this issue and poses the following research question: How do point cloud density and accuracy affect the quality of 3D building reconstruction?

To further define the research question, guide the development of the methods, and express initial expectations about the results, the following hypotheses are stated:

1. Point cloud density and accuracy are positively correlated to the quality of 3D building reconstruction.
2. The correlation is non-linear, such that a higher 3D reconstruction quality requires a disproportionately higher density or accuracy.
3. Knowledge of the relationship allows an optimization of the point cloud acquisition with respect to a desired 3D reconstruction quality.

A different way to phrase the second hypothesis is that an increasing point cloud density and accuracy are expected to provide diminishing returns with respect to the quality of 3D reconstruction. This relationship motivates the research question from an economic point of view: While point cloud acquisition methods have advanced significantly over the past decades, they remain time-consuming and costly, and they cause immissions into the environment. An adequate answer to the research question could support a well-informed decision-making in survey planning and execution, leading to a further reduction of these costs. This can facilitate more frequent update cycles for 3D city models, responding to the ever-growing demand for up-to-date information in applications such as urban digital twins.

To address the stated problem, point clouds with multiple levels of density and accuracy are generated from a scene of input 3D building models that serve as ground truth. Using an established 3D reconstruction method, new 3D building models are reconstructed from each of the point clouds. This ensures that each reconstructed model can be compared directly to the original model. Subsequently, the quality of the obtained building models is evaluated using multiple metrics to provide a comprehensive answer to the research question. In this context, various quality aspects of 3D building models are discussed.

Following this introduction, the thesis is structured as follows: The methods that were developed to conduct the experiment, the software and datasets that are used, and the experiment design are described in chapter 2. In chapter 3, the results are presented and subjected to a rigorous analysis. A thorough discussion of technical and methodological limitations and opportunities as well as an in-depth interpretation of the results and their implications are given in chapter 4, along with opportunities for future work. Finally, the thesis is concluded in chapter 5.

2 Data and Methods

2.1 Conceptual Basis

2.1.1 Foundational Concept

Before the methodology and the involved steps are described in detail, this section aims to provide an overview of the foundational idea that served as a starting point for the development and therefore conveys the underlying concept well. The impact of point cloud density and accuracy on the quality of 3D building reconstruction can be investigated by conducting an experiment as it is illustrated in figure 2.1: Based on a set of existing building models, multiple point clouds that differ along the two dimensions of density and accuracy are obtained. Subsequently, each of them is used to reconstruct a new dataset of 3D building models that have a one-to-one correspondence to the input models. This enables a comparative evaluation between each reconstructed dataset and the reference dataset. Any identified differences in quality can thus be attributed to the respective point cloud properties.

It is advantageous to generate the required point clouds from a dataset of existing 3D building models for two reasons: First, it provides control over the exact qualities of the point clouds. Second, the input building models can serve as ground truth for a quantitative evaluation. For this purpose, a method capable of generating point clouds from virtual 3D scenes is required. The selected solution HELIOS++ is a software for the simulation of laser scanning surveys that was developed at the 3D Geospatial Data Processing Group at the University of Heidelberg (Winiwarter et al., 2022), and is described in section 2.2.1.

For the 3D reconstruction step, any reconstruction software could be used in principle. The main requirements were that it should be easily available, capable of processing large datasets efficiently, representative of at least its own class of reconstruction methods, and used and proven in practice. Section 2.2.3 provides details about the software Geoflow, developed at the 3D Geoinformation Group at Delft University of Technology in collaboration with the company 3DGI (Peters et al., 2022), that was found to meet these criteria. Other potential candidates that could serve to expand this experiment are discussed in section 4.4.2.

2.1.2 Definitions

To be able to address the research question adequately, it is useful to clearly define the terms it makes use of. In particular, the notions of the *density* and *accuracy* of a point cloud and the *quality* of 3D building reconstruction require clarification. While their meaning is intuitively understandable, a systematic investigation must rely on measurable quantities to represent them.

Point Cloud Density

The *density* of a point cloud refers to the number of points contained within a reference area or volume. It can be computed globally across a dataset or locally to quantify density variations. In the practice of airborne laser scanning (ALS), stated values commonly refer to a global areal density: The total number of points normalized to the horizontal area they cover (Vosselman and Maas, 2014). This metric is easy to calculate and suitable to describe an average density across a dataset, which is useful for comparability. Alternatively, one can refer to the point spacing, which is more directly related to the acquisition parameters: The pulse and scan frequency of the laser scanner, the scan angle, and the altitude of the survey. Both point density and spacing are used in this thesis, and details of their computation as a function of the survey parameters are described in section 2.2.1.

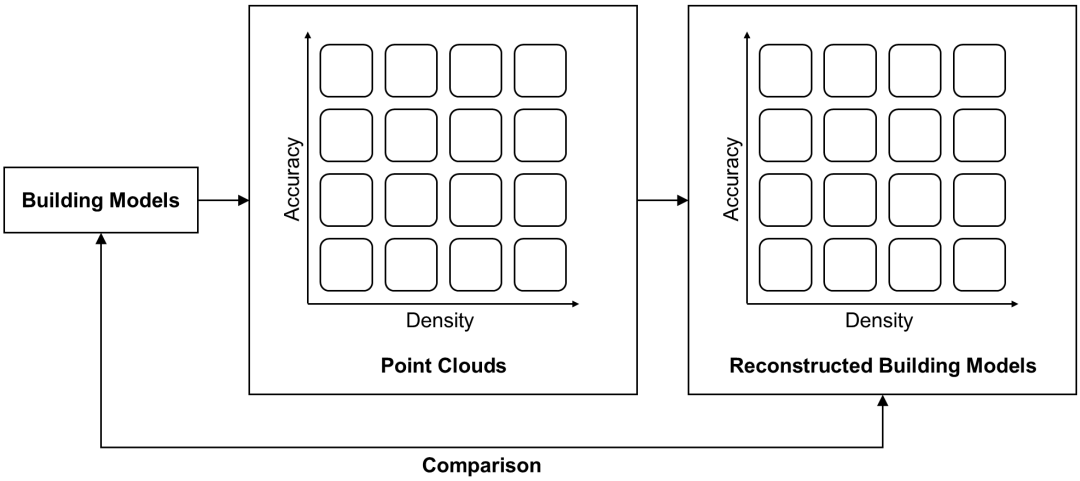


Figure 2.1 The foundational idea underlying the methodology: A dataset of existing building models is converted into multiple point clouds that differ along the two dimensions of density and accuracy. Based on each of these, a set of new building models is reconstructed such that there is one output model for each input model. This enables a quantitative evaluation of the reconstructed building models in comparison to the original ones.

Point Cloud Accuracy

As opposed to the density, the point cloud *accuracy* is a more ambiguous term because it is often used in a very general sense. A clearer understanding can be reached by approaching it from the perspective of errors in the point cloud: It may contain systematic and random errors, which are commonly understood to relate to the metrological terms accuracy and precision, respectively (Kim et al., 2022). According to Toth and Koppanyi (2017), the main sources of errors in LiDAR point clouds from airborne laser scanning (ALS) are the Global Positioning System (GPS) and the Inertial Measurement Unit (IMU), which identify the position and orientation, as well as the laser scanner itself. They note that most systematic errors can be removed in the process of aligning adjacent point cloud strips, which is a step commonly required in ALS as large areas can only be surveyed in multiple, parallel trajectories. Habib (2017) note that the relevant sources of random error can be subdivided into position noise, angular noise, and range noise. Position noise stems from the error in the GPS, whereas angular noise can be a result of deviations in the attitude determination by the IMU or by the mechanically moving mirror in the laser scanner. Range noise refers to the inaccuracy in the laser ranging measurement. While angular noise exerts a stronger effect horizontally, range noise primarily introduces a vertical error component. Both are dependent on the current scan angle, and angular noise additionally varies with altitude.

The USGS Lidar Base Specifications (Heidemann, 2018) recommend to follow the *ASPRS Positional Accuracy Standards for Digital Geospatial Data, Edition 2, Version 2* (2024), which demand the separate specification of the horizontal and vertical error components of geospatial data in terms of the root mean square error (RMSE). They assume that the errors follow a normal distribution and require the absence or removal of any systematic errors.

In the light of these references, it was decided to focus on the simulation of random error for the purpose of this investigation. This is achieved by adding a normally distributed, random quantity to the position of each point, distinguishing between horizontal and vertical error. Further details of the implementation are provided in section 2.2.2. Within the scope of this thesis, any usage of the term *accuracy* with respect to the point clouds analyzed in the experiment shall therefore refer to the level of random positional error of the points.

3D Building Reconstruction Quality

The *quality* of the reconstructed 3D building models is the main dependent variable of this experiment that manipulates point density and accuracy. Yet, there is no single metric that encompasses all quality

aspects of building models. On the contrary: Many ways in which their quality can be assessed are conceivable depending, for instance, on the requirements of the use case. Certain metrics rely only on the reconstructed building model, such as the validity of its geometry, and they are particularly suitable if no ground-truth data is available (Dukai et al., 2021). For the purpose of this project, however, the most meaningful metrics measure the difference or distance between each of the reconstructed models and the input models. Among these, a distinction can be made between metrics that are a direct measure of the geometric distance between two corresponding models, and metrics that evaluate the difference in certain derived quantities of both models. The selected metrics are presented in section 2.2.4 and further implications are discussed in section 4.3.1.

2.2 Data Pipeline

2.2.1 Simulation of Airborne Laser Scanning

Simulation Software

Airborne laser scanning (ALS) was simulated using the software HELIOS++, short for Heidelberg LiDAR Operations Simulator (Winiwarter et al., 2022). It enables a comprehensive and physically accurate simulation of LiDAR surveys. A HELIOS++ survey is defined by several parts: A 3D scene, a laser scanner and its physical properties, a platform to which the scanner is mounted, and, in the case of a moving platform, its trajectory (see figure 2.2). To execute simulations, HELIOS++ offers a command line interface as well as Python bindings in the pyhelios library.

Various pre-configured scanners that implement properties of real LiDAR devices from manufacturers such as Riegl or Leica are provided with the software. However, each property can be adjusted as required. The platform can be either stationary or mobile, in which case the trajectory in terms of individual legs can be defined along with the survey. Accordingly, it is possible to simulate terrestrial laser scanning (TLS), mobile laser scanning (MLS), and airborne laser scanning (ALS). The scene can be assembled from files in four different 3D data formats: Wavefront OBJ mesh files, GeoTIFF raster files, VOX voxel models, and XYZ point clouds. Internally, raster files are triangulated and XYZ point clouds are converted into voxels with a user-defined size. Surface properties like reflectivity can be defined in material files identical to those defined by the Wavefront OBJ specifications. Additionally, voxel models allow a simulation of translucency.

The configuration of an ALS simulation is stored in XML files. Typically, this comprises one file that describes the scene by defining its parts, and another file that describes the survey, scanner, and platform settings, as well as the platform trajectory, if applicable. The pre-configured scanners and platforms are available in separate XML files and can be referenced by an ID tag. In summary, this allows an automated setup of multiple simulation definitions with different survey parameters to generate point clouds with varying properties from the same 3D scene.

The presented methodology employs HELIOS++ to generate point clouds with varying density. Its capabilities to simulate error are limited: There is an accuracy setting for the simulated laser detector, which is interpreted as the standard deviation of the ranging measurement, yet it only enables addition of noise in the ranging direction. While there is a setting for platform noise, its application results merely in a random shifting of the platform along its trajectory, which displaces the ranging measurement, i.e., the point, on the intersected object accordingly, but the point is still located exactly on the object's surface, such that no random error is added to the point coordinates. For this reason, point cloud random error was simulated in a subsequent step, which is described in section 2.2.2. Accordingly, all HELIOS++ settings related to the simulation of error were disabled.

Scene and Survey Parameters

The selected method for 3D building reconstruction (see section 2.2.3) requires as input a classified point cloud in which building and ground points must be present. To achieve this, the input scene must at least contain a representation of the terrain and the buildings. Correspondingly, for all experiments the scene

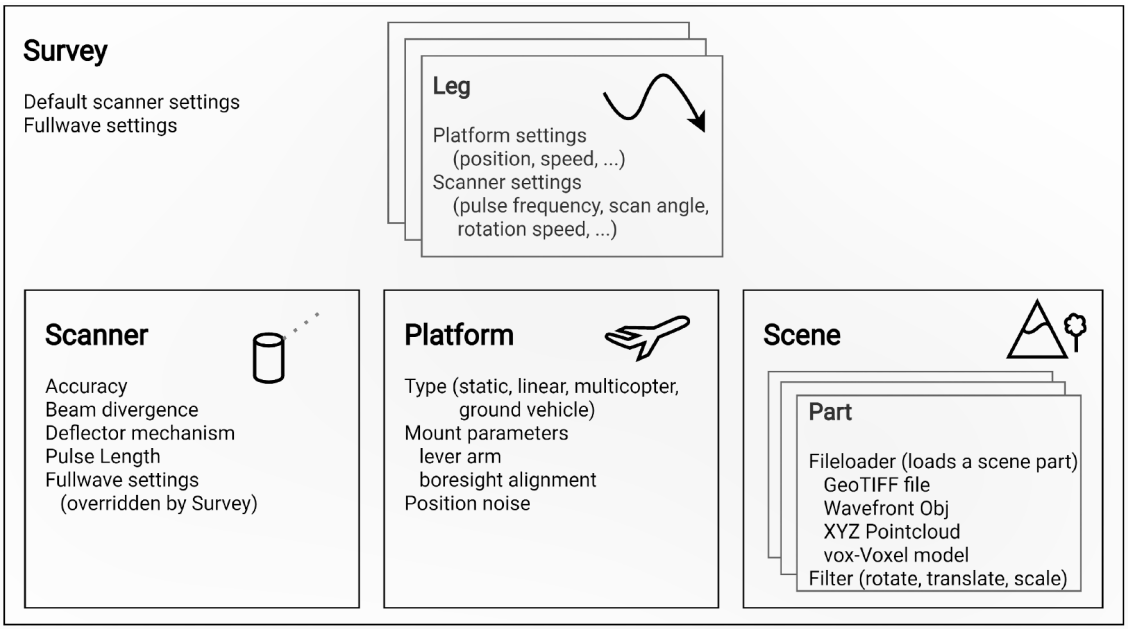


Figure 2.2 The components of a simulated airborne laser scanning survey in HELIOS++: Scanner, platform, and scene. Legs define a moving platform's trajectory and enable location-dependent scanner settings. (From Winiwarter et al., 2022; *CreativeCommons CC BY 4.0* 2024)

was assembled from a digital terrain model (DTM) and 3D mesh models of the input building models. Material definitions within the scene XML file as well as within the material library file accompanying a Wavefront OBJ mesh can be used to assign classes to the DTM and individual parts of the mesh, which HELIOS++ uses to classify the output point cloud.

The pre-configured laser scanner definition implementing the properties of the model Riegl VQ-1560i was used in the experiments. Because this is a dual-channel scanner with each channel rotated by 14ř against the movement direction, the configuration was adapted to feature only one channel that is not rotated. This was done obtain parallel, equally spaced lines of points with the aim of eliminating any influence of an irregular point pattern, as would otherwise be generated due to the two rotated channels.

Further parameters of the platform and the scanner were used to control for the point density: HELIOS++ allows setting the platform's altitude and velocity for each section of its trajectory, as well as the scan angle of the scanner. Following a linear flight path, this yields a swath or strip of point measurements with a width of

$$w_s = 2\ h\ \tan\left(\frac{\theta}{2}\right), \tag{2.1}$$

where w_s is the swath width, h the altitude, and θ the scan angle (Vosselman and Maas, 2014). To cover a larger area, the trajectory of the platform is defined such that multiple swaths with a certain overlap area are surveyed.

From two further scanner properties, the pulse frequency f_p and the scan frequency f_s, the number of scan lines N can be computed as

$$N = \frac{f_p}{f_s}. \tag{2.2}$$

Together with the swath width w_s, the velocity v, and the altitude h, this enables the calculation of the theoretical point spacing on the ground in along-track (s_l) and across-track (s_c) direction (Baltsavias, 1999):

$$s_l = \frac{v}{f_s} \tag{2.3}$$

$$s_c = \frac{w_s}{N} = 2\ h\ \tan\left(\frac{\theta}{2}\right)\ \frac{f_s}{f_p} \tag{2.4}$$

Finally, the theoretical point density is obtained from the along-track and across-track point spacing as

$$d = \frac{1}{s_l\ s_c} \tag{2.5}$$

2.2.2 Point Cloud Processing

Swath Merging

The simulated ALS survey yields multiple parallel point cloud swaths that must be merged into a single point cloud for the 3D building reconstruction. Since adjacent swaths overlap to a degree, a simple merging of the point data would essentially double the point density in the overlapping areas, which runs counter to the notion of assessing a fixed point density's impact on the reconstruction outcome. To avoid this, an approach to decide which points from these areas to keep must be established.

A simple decision criterion is the distance of each point to the location along the trajectory from which it was measured: At each location in the overlapping area, the ranging measurements that are closer to the ranging origin are kept, while the others are discarded. In the case of the error-free ALS simulation, the corresponding decision boundary is the vertical plane at the center of the trajectories.

However, there are areas within the overlap and beyond this boundary that are only visible from the further trajectory due to the scene geometry: They lie in the shade of the LiDAR device on the closer trajectory. This results in gaps in the respective point swath at these locations. To control for the influence of such gaps (which could be investigated separately), points from the other swath that fill them may be kept. Therefore, a second decision criterion was established that enables identification and removal of duplicate points in the overlap while retaining complementing points.

To this end, for each individual point cloud swath a mean nearest-neighbor distance was computed (from each point's distance to its k nearest neighbors, to account for irregularities). Then, for adjacent swaths with overlap, for each point in one cloud the distance to the closest point in the other cloud was identified. Finally, points in areas beyond the decision boundary are kept if their distance from the closest point in the other cloud is larger than the mean nearest-neighbor distance of that cloud, and discarded otherwise.

Introduction of Random Error

Besides point density, the accuracy of the point cloud is the second independent variable whose influence on the reconstruction outcome is to be explored. In the context of this project, the accuracy of a point cloud is equated with the level of random positional error of the points. The reasons for this choice are outlined in section 2.1.2.

After point clouds with varying densities were generated from the 3D scene by simulating ALS surveys, their accuracy is thus manipulated by introducing random error to the point positions. Accuracy specifications of geospatial data in general and point clouds in particular commonly distinguish between horizontal and vertical accuracy, and they refer to the root mean square error (RMSE) of the positional information, which is assumed to be normally distributed and free of systematic error components (*ASPRS Positional Accuracy Standards for Digital Geospatial Data, Edition 2, Version 2* 2024).

Random error in the point locations can therefore be simulated by adding a normally distributed quantity to each of the three coordinates. For the vertical coordinate z, the standard deviation equals the vertical $RMSE_V$. The standard deviation of the horizontal coordinates x and y corresponds to $RMSE_X$ and $RMSE_Y$. Since the horizontal $RMSE_H$ is defined as

$$RMSE_H = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\Delta h_i^2}, \tag{2.6}$$

where the horizontal error Δh_i of observation i of all observations N is the distance of the observed location from the true location

$$\Delta h_i = \sqrt{\Delta x_i^2 + \Delta y_i^2}, \tag{2.7}$$

it follows that

$$RMSE_H = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\Delta x_i^2 + \frac{1}{N}\sum_{i=1}^{N}\Delta y_i^2} = \sqrt{RMSE_X^2 + RMSE_Y^2}. \tag{2.8}$$

Because it is assumed that the error in x and y is equal, their respective RMSE is obtained as

$$RMSE_X = RMSE_Y = \frac{1}{\sqrt{2}}\, RMSE_H. \tag{2.9}$$

2.2.3 Reconstruction of 3D Building Models

3D Reconstruction Software

Geoflow is an open-source software for 3D building reconstruction from point clouds that was developed at the 3D Geoinformation Group at Delft University of Technology and in collaboration with the company 3DGI. It was used for the reconstruction of approximately 10 million building models covering the entire Netherlands for the Dutch 3D cadastre 3DBAG (Peters et al., 2022). The building models are updated when new data become available.

As input, Geoflow requires footprints of all buildings that should be reconstructed and a classified point cloud that distinguishes between a building and a ground class. The buildings are reconstructed at three levels of detail (LOD), namely 1.2, 1.3, and 2.2, as described by the extended LOD specifications suggested by Biljecki et al. (2016): LOD 1.2 and 1.3 both model all roofs as flat by inferring the building height, but LOD 1.3 distinguishes between heights of different building parts, whereas the simpler LOD 1.2 assumes a single height for the entire building. LOD 2.2 models the actual roof geometry, yet without roof overhangs. To comply with the CityGML standard for storage and exchange of semantic building models (Kolbe, 2009; Kolbe et al., 2021), the building models are modeled by boundary representation, and all bounding surfaces carry an attribute that identifies them as roof, wall, or footprint. In the default configuration, the output building models are saved in three 3D data formats: As CityJSON (Ledoux et al., 2019), as GeoPackage layers (Yutzler and Daisey, 2024), and as Wavefront OBJ. The former two store semantic information as attributes and the latter distinguishes roof and non-roof surfaces by material definitions.

The reconstruction method makes two fundamental assumptions about the building geometry: That buildings are piecewise planar objects, and that all walls are vertical (Arroyo Ohori et al., 2024). The latter implies a 2.5D approach to reconstruction, where a roof structure is identified according to which the building footprint is subdivided and each segment extruded to the corresponding height. Figure 2.3 visualizes the individual steps Geoflow takes to reconstruct a 3D building model from the footprint and the classified input point cloud (1): A region-growing algorithm identifies roof segments in the point cloud (2) and their bounding and intersection lines (3). After projecting these lines to the ground, a regularization step has the purpose of removing duplicate lines that occur at intersections of roof planes, where both the bounding lines of the planes and the intersection lines are present. To this end, the lines are clustered first by orientation and then by distance, which enables computing an average line for each cluster (4). In a second optimization step, the complexity of the resulting subdivision is reduced by grouping faces that belong to the same, original roof segment. The final lines defining these optimized faces are used to partition the footprint (5). Thereafter, the individual parts are extruded to the height of their corresponding roof segments (6).

The principle configuration of the reconstruction workflow is stored in a JSON file, which represents a flowchart that can be visualized and edited using the graphical user interface of Geoflow. Reconstruction parameters that are defined as global variables in the flowchart can be manipulated for each execution

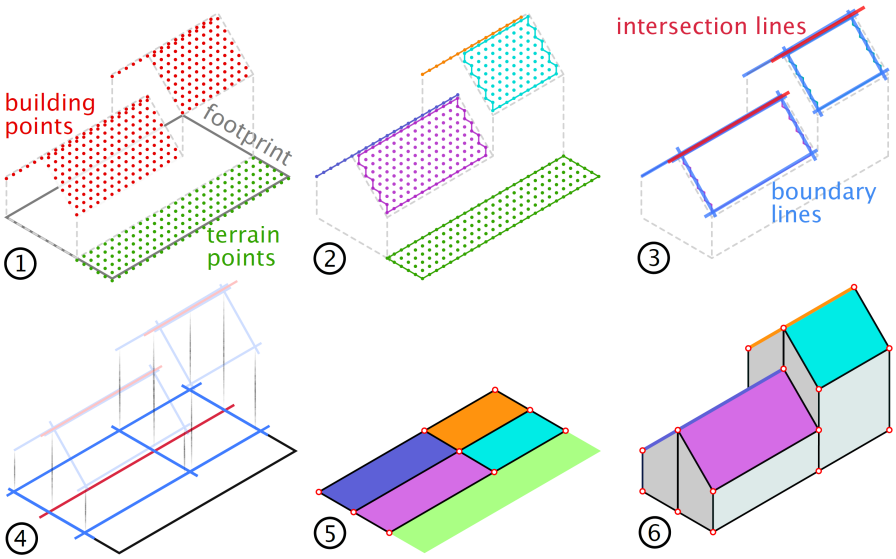


Figure 2.3 The steps to 3D building reconstruction taken by the software Geoflow. (From Peters et al., 2022; *CreativeCommons CC BY-NC-ND 4.0* 2024)

by altering the values within the JSON flowchart, by providing values in a configuration TOML file, or by passing them as command line arguments. Each takes precedence over the previous at runtime. This enables the use of the identical reconstruction process while tuning reconstruction parameters to different input data.

2.2.4 Evaluation of the Reconstructed Models

Approach to Evaluation of the Model Quality

An introduction the term *quality* as it is used in this thesis was given in section 2.1.2. In accordance with these statements, multiple evaluation metrics are used to accommodate the ambiguity in the definition of 3D building model quality. The quality of the reconstructed models is evaluated primarily in terms of distance metrics with respect to the input models, which is possible because the experiment is designed as an end-to-end simulation. In addition, several derived metrics of the reconstructed and reference building models are computed and compared. The former group includes the Hausdorff distance (Hausdorff, 1914) and the Jaccard index (Jaccard, 1912), also known as intersection over union (IOU). Examples of derived quantities that are used in the latter group are building model height, surface area, volume, and complexity (the number of faces of a 3D model). This section gives an overview of the metrics used for evaluation.

Hausdorff Distance

The Hausdorff distance is a metric from set theory (Hausdorff, 1914). In this geometric application, the unidirectional Hausdorff distance d_H is defined as the largest of all minimum distances from every point a on the surface A of one model to any point b on the surface B of the other model:

$$d_H(A, B) = \max_{a \epsilon A} \min_{b \epsilon B} \|a - b\| \tag{2.10}$$

This distance is not symmetric: Generally, $d_H(A, B) \neq d_H(B, A)$. The bidirectional, symmetric Hausdorff distance D_H is therefore defined as the maximum of the two:

$$D_H(A, B) = \max\left(d_H(A, B),\ d_H(B, A)\right) \tag{2.11}$$

In practical terms, it can be obtained by considering for every point on the surface of one model its distance to the closest point on the surface of the other. The Hausdorff distance is the largest of all these as evaluated for both models.

This project uses an implementation from the Python library PyMeshLab (Muntoni and Cignoni, 2021) that first discretizes the problem by randomly sampling a customizable number of points on the vertices, edges, and faces of one model and then finds the shortest distance to the surface of the other model. The evaluation is based on the input and reconstruction output files in Wavefront OBJ format. In both cases, this is initially a single file containing all building models. The individual models are defined by Wavefront OBJ object groups of faces, which are initiated by a line starting with the letter "o" followed by the name of the group. Because PyMeshLab is unable to subdivide a loaded mesh into individual meshes according to this grouping, each Wavefront OBJ file is split up into one file per building before computing the Hausdorff distances.

Root Mean Square Minimum Distance

A slight variation of the Hausdorff distance, the root mean square minimum distance (RMS minimum distance) does not report the single largest of the minimum distances, but the square root of the mean of all square minimum distances:

$$d_{RMS\,min}(A,B) = \sqrt{\frac{1}{|A|}\sum_{a\epsilon A}\left(\min_{b\epsilon B}\|a-b\|\right)^2} \tag{2.12}$$

Like the Hausdorff distance, this is not a symmetric measure. If the number of samples taken from both surfaces A and B is equal, the bidirectional, symmetric RMS minimum distance can be obtained by taking the square root of the average of both mean square minimum distances:

$$D_{RMS\,min}(A,B) = \sqrt{\frac{d^2_{RMS\,min}(A,B) + d^2_{RMS\,min}(B,A)}{2}} \tag{2.13}$$

Intersection over Union

The Jaccard index (Jaccard, 1912) or intersection over union (IOU) is defined as the ratio of the size of the intersection to the size of the union of two sets A and B:

$$IOU(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2.14}$$

It is a measure of the similarity of both sets that takes values between zero if they are disjunct, and one if they are equal. In geometry, it can be applied to solids A and B with the magnitude $|\cdot|$ referring to the volume of a solid. In this project, IOU is used to evaluate the similarity of each reconstructed output building model to the corresponding input building model. The computation is performed with a custom workspace in the software FME®, which proved to be the only software capable of the required volume and volumetric intersection computations. Attempts to implement these using PostGIS and its functions *CG_Volume* and *CG_Intersection*, which rely on the corresponding functions of CGAL (the Computational Geometry Algorithms Library) in the background, failed due to its stricter enforcement of a planarity threshold with respect to the faces. An implementation using the CGAL Python bindings was considered as well, but the provided functionality subset does not include the required functions.

Difference Metrics of Derived Quantities

Finally, the quality of the reconstructed building models in terms of their similarity to the original models can be measured as the difference in properties derived of the 3D models. To this end, the difference in height, surface area, volume, and complexity of the models was evaluated. The height attribute is computed by

the reconstruction software and therefore present in both input and output data. Geoflow provides four different values as attributes in the output CityJSON files: The minimum, 50th percentile, 70th percentile, and maximum elevation of all points used in the reconstruction of the LOD 2.2 roof geometry. The 70th percentile serves as reference height for the default 3D models in LOD 1.2. Surface area and volume of all models are computed using the software FME®. The complexity of a model is determined as its number of faces after triangulation.

2.2.5 Pipeline Implementation

To be able to conduct this experiment conveniently and flexibly, the sequence of steps detailed in the previous sections was encapsulated in a modular, adaptive, and reusable data processing pipeline. The flowchart in figure 2.4 outlines its structure. It is implemented in Python and takes a single JSON file as input for the configuration of all individual parts. It manages the preparation of the configuration files for HELIOS++, the execution of the simulated ALS survey, the merging of the point clouds and the optional addition of random error, the optimization of the reconstruction parameters with respect to the point cloud properties, the 3D building reconstruction, and the evaluation of the reconstructed building models, providing a single output dataset containing the results in all metrics for all building models across all scenarios as well as the corresponding input parameters.

Central to the implementation is an Experiment class that has one or more instances of a Scenario class, where each Scenario comprises all processing steps related to the associated point cloud with its distinct properties. The optimization of the reconstruction parameters (see section 2.3) also makes use of the Experiment and Scenario classes, but in this case, each Scenario instance uses the same point cloud but alters the reconstruction parameters and evaluates the influence on the reconstructed models. It can therefore be considered an experiment within each scenario of the experiment, where the reconstruction parameters instead of the point cloud properties constitute the dependent variables that influence the quality metrics. Figures A.1 and A.2 in appendix A.1 provide UML diagrams of all relevant classes and their relationships. The source code is available on GitHub at https://github.com/florifal/city-to-scan-to-city (Faltermeier, 2024).

The pipeline makes use of various Python libraries and separate software: The ALS survey simulations are run with the HELIOS++ Python bindings (Winiwarter et al., 2022). The subsequent point cloud processing is done using PDAL (Butler et al., 2024) and SciPy (Virtanen et al., 2020). The software for 3D building reconstruction Geoflow (Peters et al., 2022) is executed as subprocess. The Bayesian optimization of the reconstruction parameters is performed using the Python library Bayesian Optimization (Nogueira, 2024). The evaluation of surface area, volume, and IOU of the building models is solved with the software FME®, for which Safe Software Inc. generously provided a license. The Hausdorff distance and RMS minimum distance between building models was computed using the library PyMeshLab (Muntoni and Cignoni, 2021). Various other geospatial and tabular data analysis steps are done with GeoPandas (Bossche et al., 2024) and Pandas (McKinney, 2010; The Pandas Development Team, 2024). Statistical visualization relies on Seaborn (Waskom, 2021), which is based on Matplotlib (Hunter, 2007), and 3D visualization is done using PyVista (Sullivan and Kaszynski, 2019).

2.3 Optimization of the Reconstruction Parameters

2.3.1 The Case for an Optimization

The 3D building reconstruction software Geoflow (see section 2.2.3) exposes many parameters for the adjustment of its individual processing steps. This enables an optimization of the reconstruction process and its results with respect to the properties of the input point cloud. In this case, this concerns the point density and the amount of random error present. The key steps that can be fine-tuned in this way include the detection of the roof planes, their alpha shape, and their bounding and intersection lines, the line

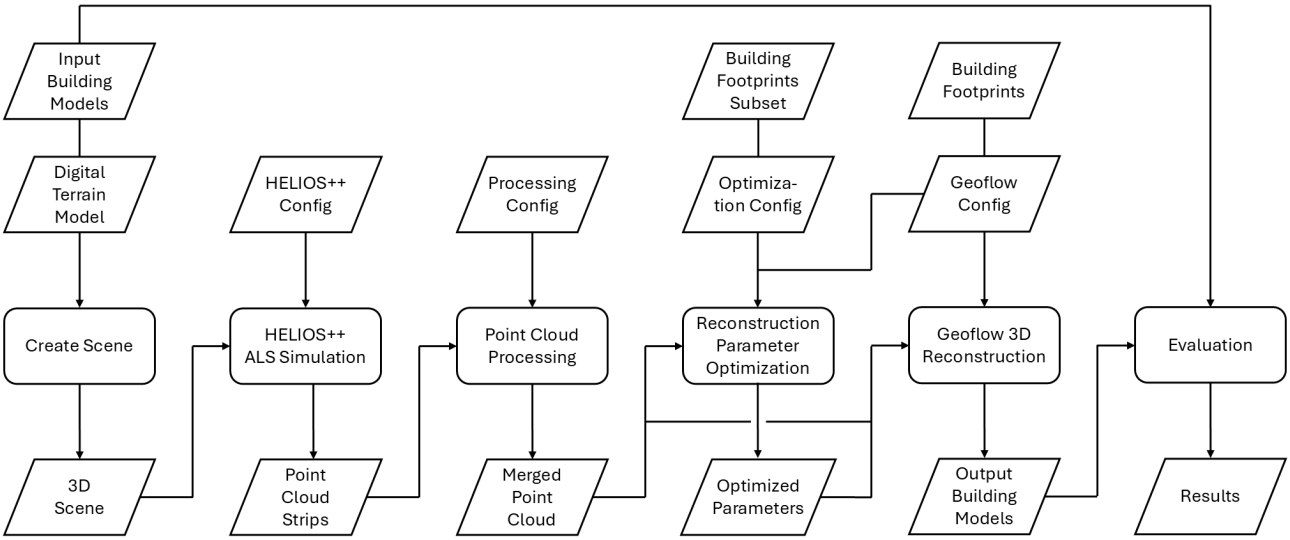


Figure 2.4 A flowchart of the data pipeline outlining the steps, input data, configurations, and output data involved.

regularization, and the optimization of the initial roof structure's complexity. Table 2.1 lists a subset of key parameters and explains their meaning.

The following argument substantiates the decision to conduct such an optimization. On the one hand, there is no way to guarantee that the degree to which a set of optimized parameter values is, in fact, unambiguously optimal with respect to the point cloud's properties. This depends on which properties and qualities in the output building models are considered optimal, which may vary depending on factors such as the use case, the desired balance along the continuum between preserving most information and eliminating most error present in the input, and other constraints. One can argue, therefore, that an optimization of the parameters poses the risk of introducing a bias to the reconstruction. This would, to some extent, impede the comparability between the output building models from different scenarios.

On the other hand, however, using a single, constant set of parameter values for the reconstruction of all scenarios that are not optimized for their individual properties does not imply the absence of a bias. On the contrary: These parameters would intrinsically yield optimal results for a certain set of input properties and the corresponding scenario, and lead to an inferior reconstruction quality with increasing distance from this location of the optimum. It was therefore determined that, while perhaps not ideal, an optimization step is not only sensible but necessary, and that its implications regarding the possible introduction of a slight, albeit controllable, bias would pose the lesser evil.

2.3.2 Optimization Method

Considerations in Selecting an Optimization Method

To optimize multiple reconstruction parameters for each of a large number of point cloud scenarios with different properties, an efficient optimization method is required. It must be capable of identifying local optima of a target function across a high-dimensional parameter space, i.e., within given ranges for each of the parameters. The goal is the optimization of the reconstruction quality, which must therefore be quantified by the target metric.

Several approaches were considered. Two methods that are commonly used for the optimization of hyperparameters of neural networks, a related task, are grid search and random search (Bergstra and Bengio, 2012). A grid search probes the parameter space at regular intervals, whereas a random search does so at random locations. For the problem in hand, however, they were discarded due to their relative inefficiency and ineffectiveness, respectively: Grid search uses much of its computational budget on locations that are unlikely to deliver good results. Random search may be more efficient, but the fact that it is

Step	Parameter	Description	Default	Range	Unit
Plane Detection	plane_epsilon	Maximum distance between plane and candidate points	0.2	[0.1, 2]	m
Plane Detection	plane_k	Number of nearest neighbor points for region growing algorithm	15	[10, 30]	
Plane Detection	plane_min_points	Minimum number of points to constitute a plane	15	[10, 30]	
Plane Detection	plane_normal_angle	Maximum angle between plane and point normal (dot product)	0.75		
Plane Detection	plane_normal_k	Number of nearest neighbor points for normal estimation	5		
Alpha Shape	alpha_threshold	Distance for alpha shape computation of detected planes	0.25	[0.1, 1]	m
Line Detection	line_epsilon	Maximum distance between line and candidate points	0.4	[0.1, 1]	m
Line Regularization	line_reg_distance	Maximum distance up to which lines are merged if approximately parallel	0.5	[0.1, 1]	m
Line Regularization	line_reg_extension	Distance by which regularized lines are extended	1.5	[1, 3]	m
Roof Optimization	optim_data_term	Weight for data term in graph-cut optimization of the roof partition	7	[5, 10]	

Table 2.1 A subset of parameters of the reconstruction software Geoflow for the configuration of individual reconstruction steps. Since there is no official documentation of the parameters, their descriptions were paraphrased from configuration files on GitHub by Peters and Dukai (2024a,b). The default values are taken from the first of these, according to which they are optimized for a point density of $8\,\mathrm{pt\,m^{-2}}$. If ranges are given, they were used to optimize the corresponding parameters.

unguided leaves unused potential. Neither make use of the information obtained about the target function from its previous evaluations.

Bayesian optimization is a probabilistic method that is particularly useful for the optimization of non-differentiable black-box functions that are costly to evaluate (Shahriari et al., 2016; Wang et al., 2023). Based on prior evaluations, it creates and updates a surrogate model of the function and derives an acquisition function that indicates promising locations in the parameter space, which are preferentially probed in subsequent iterations. Further, it balances this exploitation of prior knowledge about the target function with the exploration of uncertain regions. The algorithm is initialized with a number of random iterations, which is followed by the guided optimization.

These properties make Bayesian optimization a suitable approach for the optimization of the reconstruction parameters: The objective function is the quality of the 3D reconstruction of building models with a point cloud as input variable and multiple parameters influencing the outcome. A single evaluation requires a 3D reconstruction step followed by an evaluation of the results with respect to the input building models, which takes several minutes even with a small number of buildings and a quality metric that is simple to evaluate. Finally, a high-dimensional parameter space, unknown interactions between the parameters, and the black-box nature of the reconstruction process justify the selection of this method.

Target Metric for Optimization

The optimization was conducted for LOD 2.2. As discussed in section 2.1.2, there is no unambiguous definition of the reconstruction quality and the choice of a metric to measure it must depend on the question or application. For the optimization of the reconstruction process, the overarching goal is to achieve geometric similarity to the scanned input buildings, which are the input 3D building models in the present study. Therefore, measures of geometric distance such as the Hausdorff distance, RMS minimum distance, or IOU are adequate candidates.

The calculation of the 3D IOU using FME® requires the computation of the volume of both compared building models, the solid formed by their 3D intersection, and its volume. This makes it both computationally costly and, in practice, frequently unreliable in the case of invalid geometries with respect to the requirement of a watertight solid. The Hausdorff and RMS minimum distances both showed good correlation with the 3D IOU in test runs of the optimization and are significantly faster and more reliable to compute. For these reasons, IOU was discarded and, from the remaining two, the RMS minimum distance was selected as target metric due to its fractionally closer correlation to IOU and the fact that it captures more information about the global distance distribution between the two models instead of only quantifying its upper bound.

Strategies to Improve the Optimization Efficiency

Further strategies are employed to further reduce the amount of time required for the optimization: First, the parameters are optimized based on a subset of all buildings. While there may be ways to choose a subset that is representative of all buildings in terms of certain metrics, which would again have to be considered carefully, in this case a random subset is selected at first and then adjusted manually to a slight degree to include a wider variety of building geometries.

Second, to reduce the amount of optimization iterations required for each scenario, it was deemed practicable to reuse the optimizer states and thereby to include the prior knowledge contained about the target function from the adjacent, most similar scenarios: If available, the target function results from a previously optimized scenario with the closest point density and identical error level, and with the closest error level and identical point density are used to initialize the optimization of the next scenario. To kindle this optimization cascade, a single scenario is optimized independently with a sufficiently large number of initial random and following guided iterations. Subsequent scenarios reuse optimizer states from adjacent scenarios and are optimized with a smaller number of iterations to tune the optimizer to the slightly altered properties of the input point cloud that affects the outcome of the target function.

Third, a timeout for the reconstruction had to be introduced to keep the optimization viable for high-density scenarios. If the reconstruction with a particular set of parameters that are probed during an individual optimization iteration exceeds this timeout, it is aborted and a penalty value of the target metric is reported to the optimizer. The penalty value for the RMS minimum distance is based on observational experience of values typically encountered. Because this approach causes the optimizer to avoid probing locations in the parameter space that are in close proximity to the penalized location, it unfortunately introduces a certain bias to the optimization. Further implications thereof are analyzed in the results and discussion.

2.3.3 Parameters and Ranges for Optimization

The effect of many of the reconstruction software's parameters on the reconstruction process and result are not documented. Table 2.1 shows a subset of parameters for which a description is given in configuration files in the software's repository (Peters and Dukai, 2024a,b). Further, it lists default values that, according to the first of these files, are optimized for a point density of $8\,\mathrm{pt\,m^{-2}}$. In response to a request, the developers generously shared another set of values optimized for a point density of $3\,\mathrm{pt\,m^{-2}}$. A small number of other files were available that demonstrated further values of these parameters, which allowed a limited deduction of valid ranges. The selection of the parameters, for which an optimization was expected to benefit the reconstruction, and their ranges was therefore based on the best knowledge available.

Among the parameters listed in table 2.1, eight had different values between some of the available parameter sets, indicating a dependence on the input data, and therefore appeared promising for an optimization. For these, the selected value ranges are given in the table. Since the parameters plane_k and plane_min_points always had identical values, this was set as a constraint in the optimization. The remaining two parameters' values did not change across all available configuration files and were therefore deemed to be of lesser importance with respect to the input point cloud and excluded from the optimization.

2.4 Experiment Design

2.4.1 Input Data Selection

Source of the 3D Building Models and the Digital Terrain Model

For many cities, regions, and even countries, 3D city and building models are publicly available today (Wysocki et al., 2022). Among the largest datasets is the 3D cadastre of the Netherlands (3DBAG) with about 10 million building models and coverage of the entire country. The building models were reconstructed from ALS point clouds using the reconstruction software Geoflow in a collaboration between the 3D Geoinformation Group of the Delft University of Technology and the company 3DGI (Peters et al., 2022), and the data is openly available (*Copyright and License of the 3DBAG* 2024; *CreativeCommons CC BY 4.0* 2024). The building models are provided in three levels of detail (LOD 1.2, 1.3, and 2.2) and three 3D data formats: CityJSON (Ledoux et al., 2019), GeoPackage (Yutzler and Daisey, 2024), and Wavefront OBJ. Since this project uses the same software for the 3D reconstruction, this corresponds to its specifications as outlined in section 2.2.3.

The 3DBAG was selected as input data source for several reasons: First, the fact that it comes in three levels of detail would allow for a multi-LOD comparison and evaluation. Further, the provided range of different data formats makes the dataset very versatile in terms of processing, evaluation, and visualization without the need for further conversion. Many applications and libraries rely on particular formats: For instance, all libraries that were considered for the computation of the Hausdorff distance require 3D models in a mesh format such as Wavefront OBJ. CityJSON, on the other hand, contains all semantic information that were passed through the reconstruction from the input building footprints or that was generated during the reconstruction process. Finally, the GeoPackage format is useful for analysis and visualization using established GIS software and libraries.

Moreover, the 3DBAG contains cadastral information about the buildings in addition to their 3D geometries. The regular 2D cadastre of the Netherlands (BAG) provides further attributes that are not inherited by the 3DBAG but can be joined based on the common building identifier. Taken together, this would allow an evaluation of the reconstruction results that distinguishes between buildings with different attributes.

A key requirement for the evaluation of the 3D reconstruction quality is an in-principle correspondence between certain constraints the reconstruction method imposes on the building models. In particular, this concerns vertical boundaries of the building whose location is determined based on model assumptions. In the case of Geoflow and the 3DBAG, the lowest point of the building is determined as the fifth percentile of the elevation of all points classified as ground within four meters of the building footprint. By using input data source that was generated with the same reconstruction software as used in this experiment, it can be ensured that deviations between the building models' geometries that stem from such constraints can be minimized, allowing for a better comparability.

The corresponding digital terrain model (DTM) that is used to compose the 3D scene for the ALS simulation (see section 2.2.1) was obtained from the Actueel Hoogtebestand Nederland (AHN) of the Netherlands, which is available as public domain (*CreativeCommons CC0 1.0* 2024).

Considerations in Choosing a Geographical Source Region

The 3DBAG dataset is subdivided in tiles of different sizes that can be downloaded individually (see figure 2.5). The partition follows a quadtree structure and was guided by the requirement to distribute the processing workload evenly during the reconstruction, such that each tile contains 3500 buildings or less (Peters et al., 2022). Since the data processing pipeline used to conduct the experiment is built to work with a contiguous area with a number of building models, one or several adjacent of these tiles had to be selected. Non-adjacent tiles could, in theory, be processed by conducting multiple experiments in multiple passes of the pipeline whose results could be evaluated jointly.

There were two conflicting goals with respect to the selection of a 3DBAG tile or tiles and its corresponding geographical source region. The first was to obtain a heterogeneous set of building models, both in

Gebruiksdoel (Dutch)	Building Use (English)
Woonfunctie	Residential
Overige gebruiksfunctie	Miscellaneous
Industriefunctie	Industrial
Logiesfunctie	Lodging
Winkelfunctie	Retail
Kantoorfunctie	Office
Bijeenkomstfunctie	Gatherings
Gezondheidszorgfunctie	Healthcare
Onderwijsfunctie	Education
Sportfunctie	Sports
Celfunctie	Prison

Table 2.2 The building use types in the Dutch cadastre (BAG) and their English translation.

terms of their geometric properties, such as size and roof shape, and their semantic attributes that are of interest for a differentiated evaluation. Simultaneously, this had to be achieved within a limited area that would ensure the feasibility of the experiment with respect to the computational capacities. Ultimately, this entails a trade-off between dataset size and represented variability.

While no quantitative investigation of the geometric variability was made, a qualitative exploration of the 3DBAG indicated the suitability of central areas of larger cities in the Netherlands, such as Amsterdam, Rotterdam, Den Hague, or Utrecht, which feature a wide variety of different building geometries due to a mix of historical and modern architectures and of residential, commercial, and mixed purposes. Geometries typical of large, industrial buildings are underrepresented, however.

Among the semantic attributes from the Dutch cadastre BAG, the building use type was of particular interest. The BAG distinguishes eleven distinct use types whose Dutch names and approximate translations to English for the purpose of this thesis are listed in table 2.2. On the lowest cadastral level, these use types are assigned to individual addresses, where multiple use types are possible. They are then aggregated to the level of the corresponding buildings, which are consequently assigned the set of use types present in all their registered addresses. Figure 2.6 shows the distribution of building use types across the entire BAG dataset. It is computed such that multiple use types of a building are weighted by the frequency in which they occur across all of the building's addresses, resulting in a building-level use type frequency distribution, and each building contributes equally to the overall distribution.

To be able to make meaningful statements about the impact of the point cloud properties on the reconstruction results while distinguishing between building use types, each type must be represented in the input dataset with a minimum frequency. To this end, the building use information from the BAG was aggregated to the level of the 3DBAG tiles. The variance of their frequency distribution was computed as a decision criterion: A lower variance in the frequency of all building use types indicates a more even distribution, increasing the likelihood that most use types are represented in sufficient frequency. This enabled the identification of tiles which satisfy this criterion, and the hypothesis that central areas of larger cities would, again, prove to be the most suitable could be confirmed.

The final decision for the input data region was made based on these qualitative and quantitative considerations, additionally taking into account the total number of buildings in the selected areas. The inner city of Utrecht, the smallest of the four cities mentioned above, was found to be the best fit across all requirements. The first selection of six 3DBAG tiles in Utrecht with approximately 4000 buildings turned out to exceed the computational capacity, which had previously been tested with only around a hundred buildings, so the scope had to be limited to one of the tiles that contains approximately 1100 buildings on a ninth of the area (the downloaded dataset contains 1079 buildings, whereas the number of BAG footprints in the tile amounts to 1112). The map in Figure 2.7 demonstrates the 3DBAG tiles in the city center of Utrecht and provides information about the building use distribution in each. It also shows the tile selected as input dataset with the number 10/492/594. Figure 2.8 visualizes the 3D building models in this dataset.

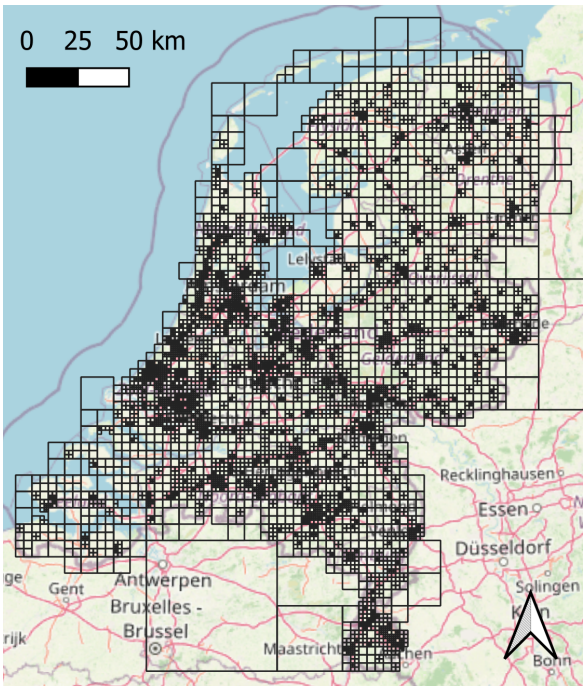


Figure 2.5 The tiles into which the 3DBAG dataset is subdivided for download. (Background map © OpenStreetMap)

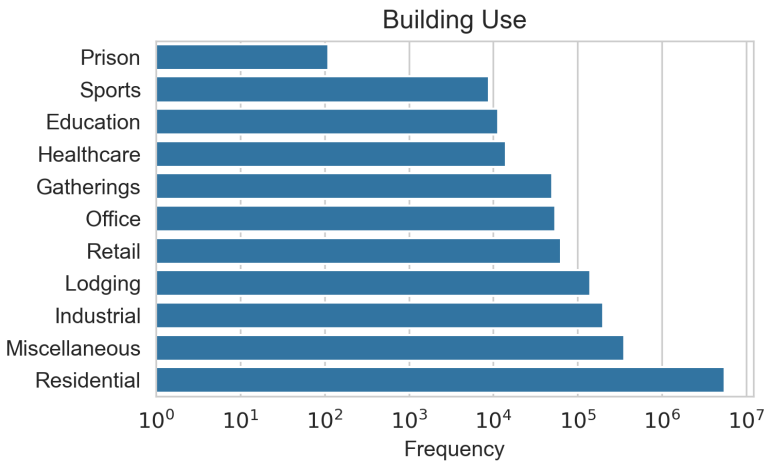


Figure 2.6 Distribution of building use types in the Netherlands according to the Dutch cadastre (BAG).

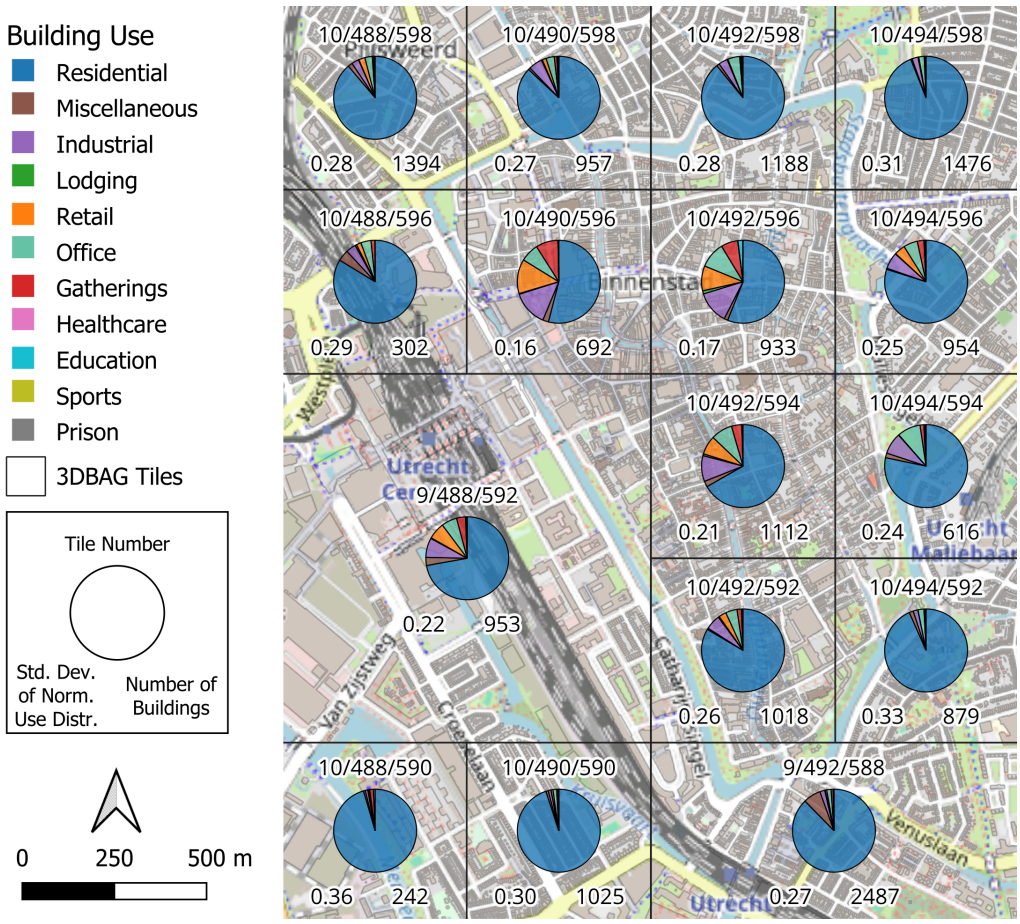


Figure 2.7 A map of the city center of Utrecht demonstrating the local subdivision of the 3DBAG dataset into tiles. The pie charts visualize the building use distribution in each tile as computed from the cadastre BAG. Further given are the tile number, the number of buildings per tile, and the standard deviation of the normalized building use distribution (lower numbers indicate a more even distribution of use types). The building models in tile 10/492/594 serve as input dataset. (Background map © OpenStreetMap)

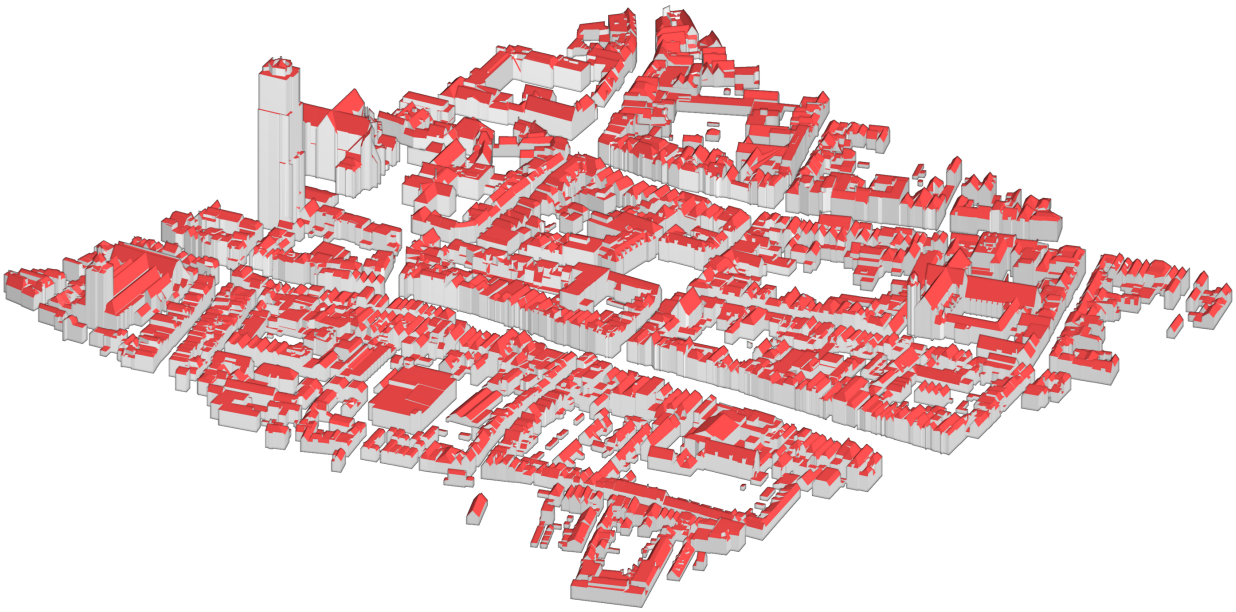


Figure 2.8 A visualization of the input dataset from the Dutch 3D cadastre (3DBAG) showing the building models in the city center of Utrecht that are located in the 3DBAG tile 10/492/594.

Level	Point Density [pt m^{-2}]	Point Spacing [m]
1	2	0.71
2	2.83	0.59
3	4	0.5
4	5.66	0.42
5	8	0.35
6	11.31	0.30
7	16	0.25
8	22.63	0.21
9	32	0.18
10	45.25	0.15
11	64	0.125
12	90.51	0.11
13	128	0.09

Table 2.3 The point cloud density levels and corresponding point spacing as set in the scenarios of the experiment.

Level [%]	$RMSE_V$ [m]	$RMSE_H$ [m]	$RMSE_{X,Y}$ [m]
0 %	0	0	0
10 %	0.03	0.1	0.07
20 %	0.06	0.2	0.14
30 %	0.09	0.3	0.21
40 %	0.12	0.4	0.28
50 %	0.15	0.5	0.35
60 %	0.18	0.6	0.42
70 %	0.21	0.7	0.49
80 %	0.24	0.8	0.57
90 %	0.27	0.9	0.64
100 %	0.3	1	0.71

Table 2.4 The levels of random error of the point clouds as set in the scenarios of the experiment: Vertical error $RMSE_V$, horizontal error $RMSE_H$, and corresponding errors in the x and y coordinates.

2.4.2 Scenarios

Point Cloud Density and Error Levels

The point density levels span a range from 2 up to 128 pt m^{-2}, which covers resolutions of older point cloud datasets from surveys in past decades as well as high-resolution point clouds measured by unmanned aerial vehicles with modern laser scanners flying at low altitudes. The intervals between the point density levels are not linear but increase exponentially. This choice was made for two reasons: First, it ensures a denser sampling of lower point densities that remain to be more relevant in the practice of large-scale 3D building reconstruction. Second, the computational effort increases disproportionately with higher point densities, which makes the high-density scenarios very time-consuming to process and evaluate. Therefore, the point density levels d were set following an exponential relationship:

$$d = 2^{\frac{n}{2}},\ n \epsilon [2, 14] \tag{2.15}$$

This results in a total of thirteen density levels (see table 2.3). These values for the point density are theoretical and should be considered as target values. The density obtained in the surveyed point cloud, which is computed as the number of points projected to the horizontal plane per unit area is always higher because of the additional points measured on sloped and vertical surfaces.

The levels of random error in the point clouds are set at intervals of 10 % from the minimum to the maximum error level, which amounts to a total of eleven error levels (see table 2.4). The minimum error level is 0 m, corresponding to a point cloud free of noise. The maximum error was selected as 0.3 m vertically, and 1 m horizontally. The goal here was again to cover a range of values that can be found in point cloud datasets generated with the latest technology as well as in older datasets (Vosselman and Maas, 2014). As detailed in section 2.2.2, the vertical root mean square error $RMSE_V$ translates immediately into the corresponding error in the z coordinate, whereas the root mean square errors in the x and y coordinates are obtained from the horizontal $RMSE_H$ by dividing by $\sqrt{2}$.

Thus the total number of scenarios, calculated as eleven error levels for each of the thirteen density levels, amounts to 143.

2.4.3 Pipeline Configuration

Airborne Laser Scanning Simulation Settings

The appropriate settings for the ALS survey were derived from the target density levels using the relationships described in section 2.2.1: The corresponding point spacing, equal in across-track and along-track direction, is calculated from the target density. Then, the required pulse and scan frequencies of the laser scanner can be derived as a function of the point spacing and the flight parameters. These were set to a velocity of 60 m s^{-1} at an altitude of 500 m. Due to the absence of elevation variations in the local terrain, no adjustments to the flying altitude were necessary to warrant a constant point density as targeted.

The scan angle was defined as 20°, which results in a swath width of 182 m. HELIOS++ refers to the unilateral (half) scan angle, corresponding to $\frac{\theta}{2}$ in equation (2.1). The distance between the flight trajectories of adjacent swaths was set to half of the swath width, such that the area between two trajectories is surveyed from both sides, leading to a 50 % overlap. As described in section 2.2.2, the approach to merging the point clouds ensures that this does not result in a doubling of the target point density. This swath configuration provides for a more complete point cloud, because areas that are in the LiDAR shadow from the point of view of one of the trajectories are in many cases visible from the other. In effect, it helps to avoid gaps in the point cloud, which is relevant for the reconstruction outcome in roof areas.

This decision may be controversial because such gaps are present regularly in point clouds from real surveys and therefore their impact on the reconstruction results should not be neglected. However, it would impede the ability to distinguish between the effect of point density alone and the additional effect of gaps in the point cloud on the reconstruction outcome. Minimizing the occurrence of gaps in the point clouds helps isolating the effect of the point density. The exploration of the influence of such gaps on the 3D building reconstruction could be investigated in future work.

Reconstruction Optimization Settings

The following settings were applied to implement the strategy to optimize the reconstruction parameters, which is described in section 2.3.2. For the optimization, a subset of 100 buildings corresponding to approximately 10 % was taken from the selected input dataset. The scenario with density level 6 (11.31 pt m^{-2}) and the error level of 0 % was optimized independently with 40 random and 160 guided iterations. All other scenarios reused optimizer states from adjacent scenarios and were optimized with 5 random and 45 guided iterations. The reconstruction timeout and the penalty value for the RMS minimum distance were set to 10 minutes and 5 m, respectively.

3 Results and Analysis

3.1 Point Clouds

3.1.1 Filtering and Merging of the Point Cloud Strips

Figure 3.1 demonstrates the filtering step performed before merging adjacent point cloud swaths. It ensures that the point density is not doubled in the regions where the swaths overlap while retaining points unique to either point cloud by a comparison of within-cloud and between-cloud nearest neighbor distance. It is apparent that most of the points beyond the center line between two trajectories that remain after filtering originate from interactions of the laser beam with vertical surfaces, which do not contribute to the 2.5D reconstruction process (see section 2.2.3). At closer inspection, however, there are numerous locations at which roof parts that were only visible from one of the two trajectories complement the other point cloud. This ensures a higher degree of completeness in the surveyed roof area and thereby helps to isolate the effect of point density variations from the effect of differing degrees of completeness or proportion of gaps in the coverage of building roofs. The reasons for the decision to take this approach in combination with a 50 % overlap between each pair of swaths are detailed in section 2.4.3.

The result after merging the filtered point clouds is depicted in figure 3.2, where the points are colored by their source point cloud. One can observe a particularly evident example where the filtering method completes the neighboring point clouds coverage of roof areas: Due to the large tower at the top left edge of the scene, significant parts of the smaller building at its foot to the bottom right were invisible from the trajectory that delivered the points colored yellow. It was, however, visible from the flight path that surveyed the green points, and due to the filtering method these points fill in the information gap in the adjacent cloud.

3.1.2 Point Density and Random Error Variations

Figures 3.3 and 3.4 exemplify the generated variations in point density and random error levels. It is evident that the maximum error level of 1 m horizontally and 0.3 m vertically, which corresponds to the RMSE in either direction and, due to the simulated absence of any systematic error, is interpreted as the standard deviation of a normally distributed positional error that is added to the points, results in a significant amount of displacement.

Table 3.1 provides the point densities of the final point clouds and compares them to the target densities on which the settings of the ALS survey parameters were based (see table 2.3 in section 2.4.2). The values calculated from the point clouds are areal densities that disregard the vertical distribution of the points. On average, the resulting point density is about 30 % higher than targeted, which is due to the fact that the ALS survey parameters are calculated from the target point spacing under the assumption of a flat terrain. The share and angle of sloped or vertical surfaces in the surveyed area determine the increase in the amount of points measured. Nevertheless, on horizontal or only slightly sloped surfaces the point spacing and density do correspond closely to the target values, since the apparent density increase is mostly a result of the calculation method. For this reason, the following text and figures will continue referring to the target point density.

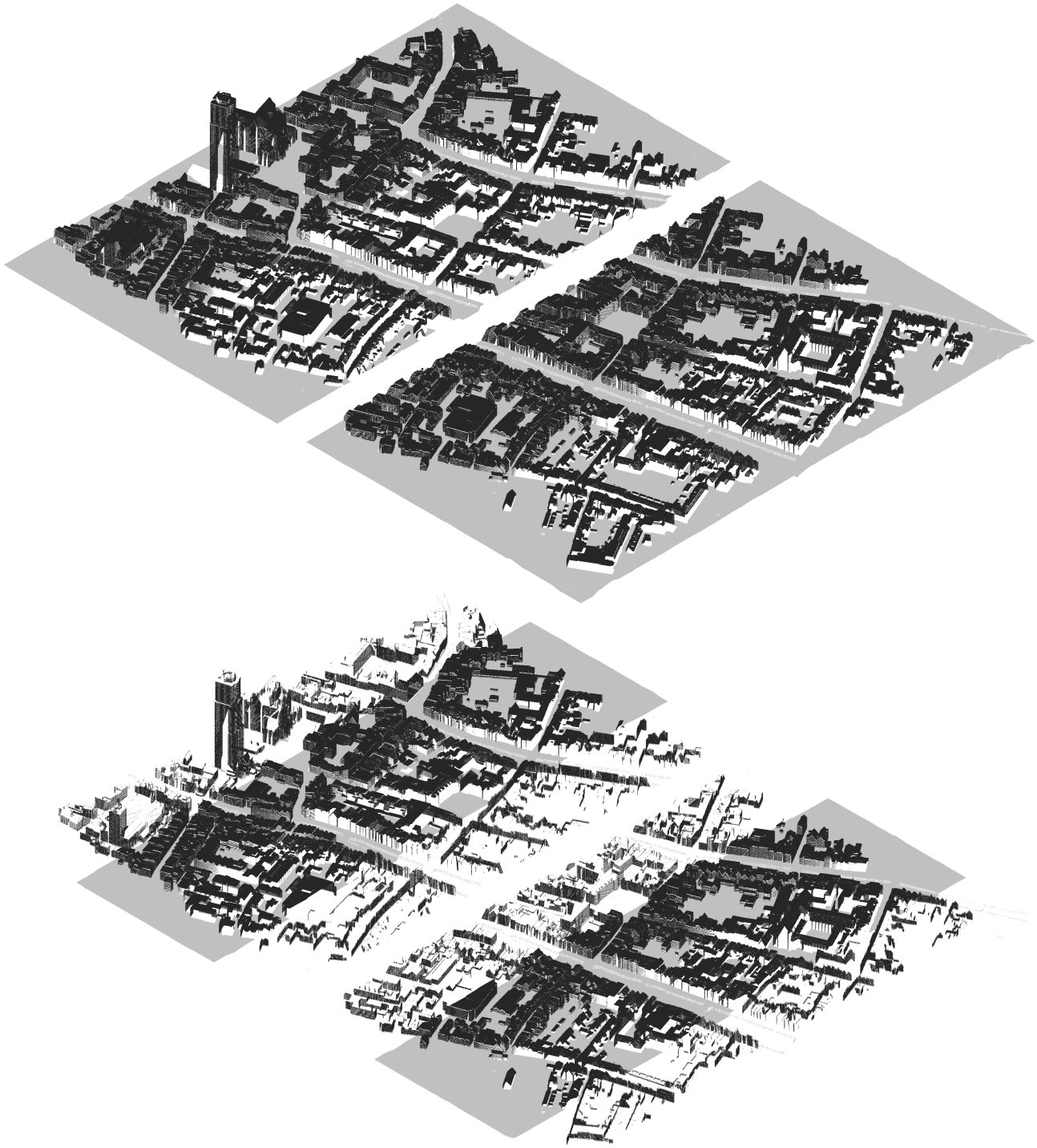


Figure 3.1 Illustration of the point cloud filtering step of two adjacent swaths. Top: The point clouds of the two swaths as obtained from the ALS simulation with 50 % overlap in coverage. The trajectory of the laser scanner in each cloud runs parallel to the long side at the center of the short side. Bottom: The same point clouds after the filtering step that ensures a constant point density after merging while retaining unique information from both clouds by discarding duplicate points beyond the center line between the trajectories but keeping points that complement the other cloud. The points are colored by the two classes *building* and *ground*. The point density is 32 pt m^{-2}.

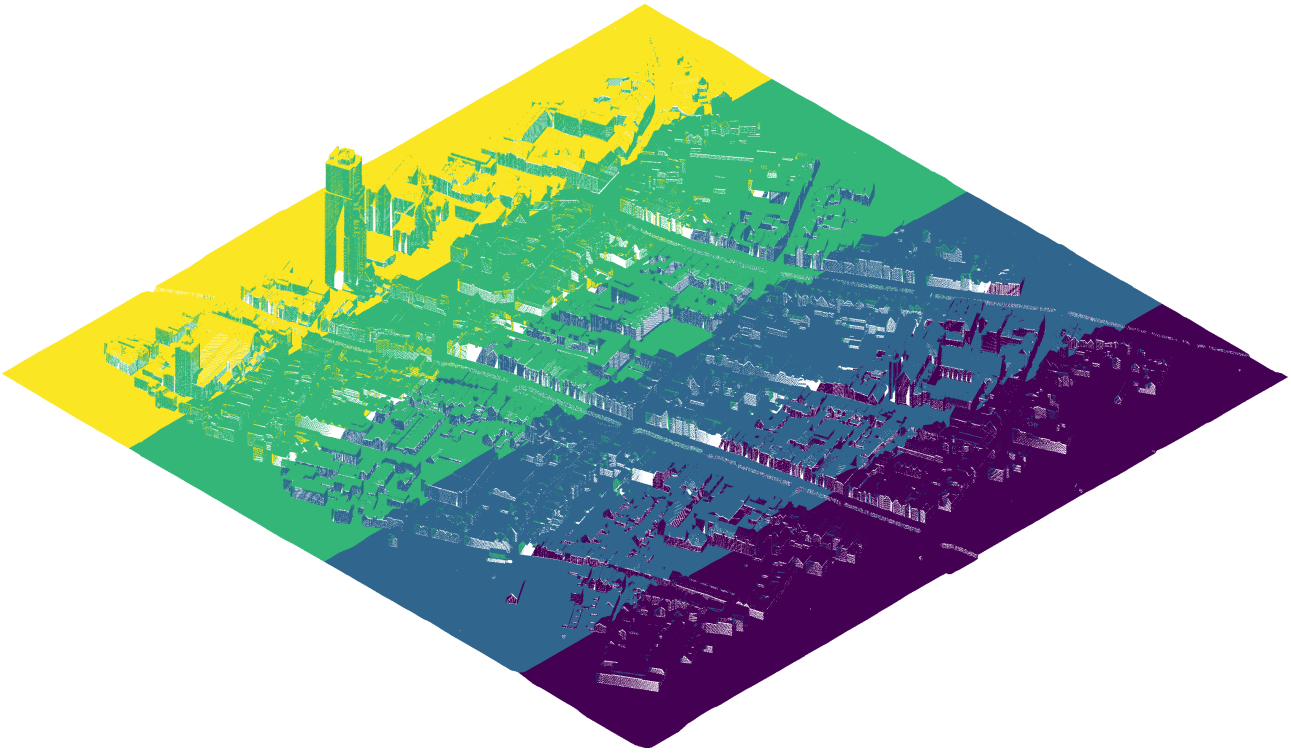


Figure 3.2 The final point cloud from the simulated ALS survey of the input 3D scene after filtering and merging the four individual swaths. The point color indicates the source swath. This point cloud has a density of 32 pt m^{-2}.

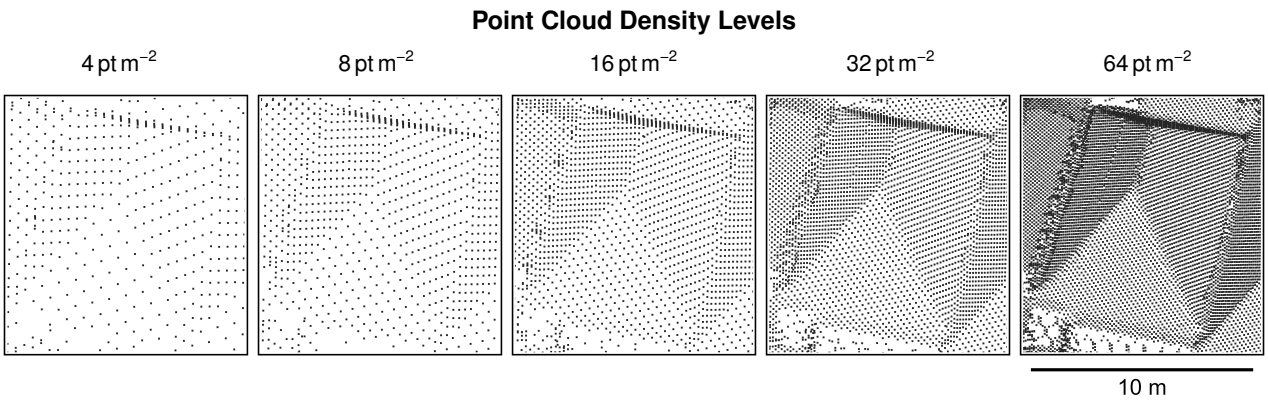


Figure 3.3 Point clouds covering a roof at five different density levels: 4, 8, 16, 32, and 64 pt m^{-2}. These point clouds are free of random error.

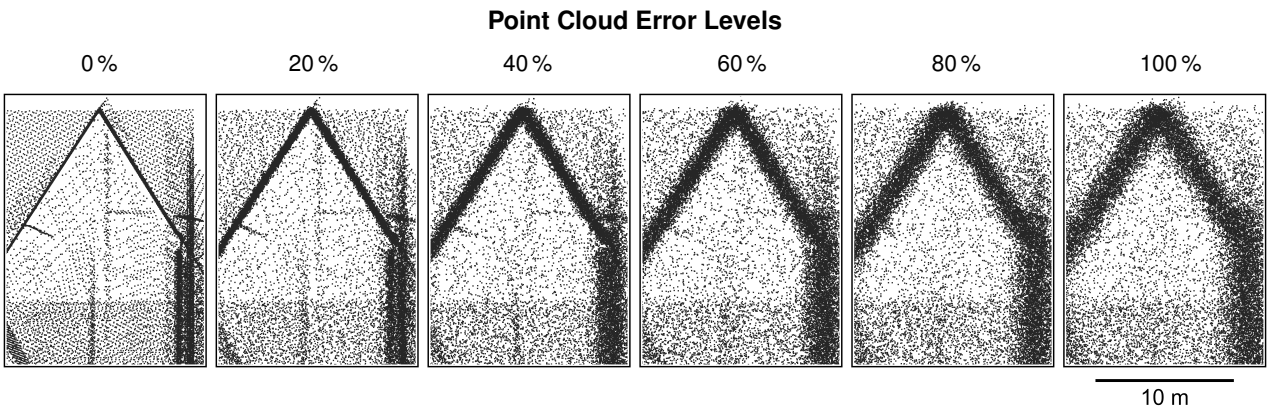


Figure 3.4 Point clouds covering a roof at six levels of random error: 0 %, 20 %, 40 %, 60 %, 80 %, and 100 % of the maximum horizontal and vertical errors, which are 1 m and 0.3 m, respectively. The point density is 16 pt m^{-2}.

Target Density [pt m^{-2}]	Actual Density [pt m^{-2}]	Difference [%]
2	2.58	29.0 %
2.83	3.66	29.5 %
4	5.19	29.8 %
5.66	7.35	30.0 %
8	10.42	30.2 %
11.31	14.77	30.5 %
16	20.93	30.8 %
22.63	29.62	30.9 %
32	41.98	31.2 %
45.25	59.49	31.4 %
64	84.35	31.8 %
90.51	119.54	32.1 %

Table 3.1 Point cloud densities as targeted by means of point spacing and corresponding ALS survey parameters, and as measured in the resulting point clouds.

3.2 Reconstruction Optimization

3.2.1 Optimized Reconstruction Parameter Values

The optimization procedure had the purpose to tune the reconstruction process to the point cloud properties of each scenario by identifying optimal values of the relevant reconstruction parameters. The objective metric serving to measure the goodness of the probed parameter values was the mean RMS minimum distance that compares the reconstructed building models to the input models on which the point clouds are based. It was evaluated as a function of probed sets of values from the seven-dimensional parameter space, where the probing locations were selected using a Bayesian approach.

Despite several strategies that were employed to improve the efficiency of the optimization process (see sections 2.3.2 and 2.4.3), the reconstruction parameters could only be optimized up to a point density of 64 pt m^{-2} due to computational constraints. Beyond this density, too many of the optimization iterations would exceed the timeout imposed on the reconstruction process to obtain a meaningful number of evaluations of the probed sets of parameter values. While the time limit could have been extended and the number of iterations reduced correspondingly to maintain a constant duration of the optimization procedure while ensuring a higher proportion of successful reconstructions and evaluations, any possibly identified, optimized parameter sets that cause such an extended reconstruction duration would have been impractical for the reconstruction of the full dataset of more than 1000 buildings. Therefore, all scenarios with higher point density levels (90.51 and 128 pt m^{-2}) use the reconstruction parameters found for the 64 pt m^{-2} scenario at the corresponding error level.

Figure 3.5 presents for each of the seven parameters the optimal value found in terms of the objective metric across all scenarios. In fact, the optimization outcome reveals meaningful relationships between the point clouds properties and the optimal value of most parameters, into which a visual analysis of the results can grant the most important insights. While there is a degree of stochasticity in the results due to the nature of the optimization method, three groups of relationships can be distinguished.

The first group demonstrates a clear directionality between point density and error level and the most suitable parameter value. It includes three parameters: plane_epsilon and plane_min_points both serve to adjust the plane detection algorithm. On average, plane_epsilon should be chosen lowest for a high-density, low-error scenario, and it should be increased with decreasing density and increasing error level. This finding matches the expectation: Both with high-noise and low-density point clouds, the range within which points likely represent measurements of a single plane (or should at least be assumed to to do, for the sake of being able to detect any planes at all) should be higher. The parameter plane_min_points increases with both point density and error level. Again, this corresponds to an analytic understanding of

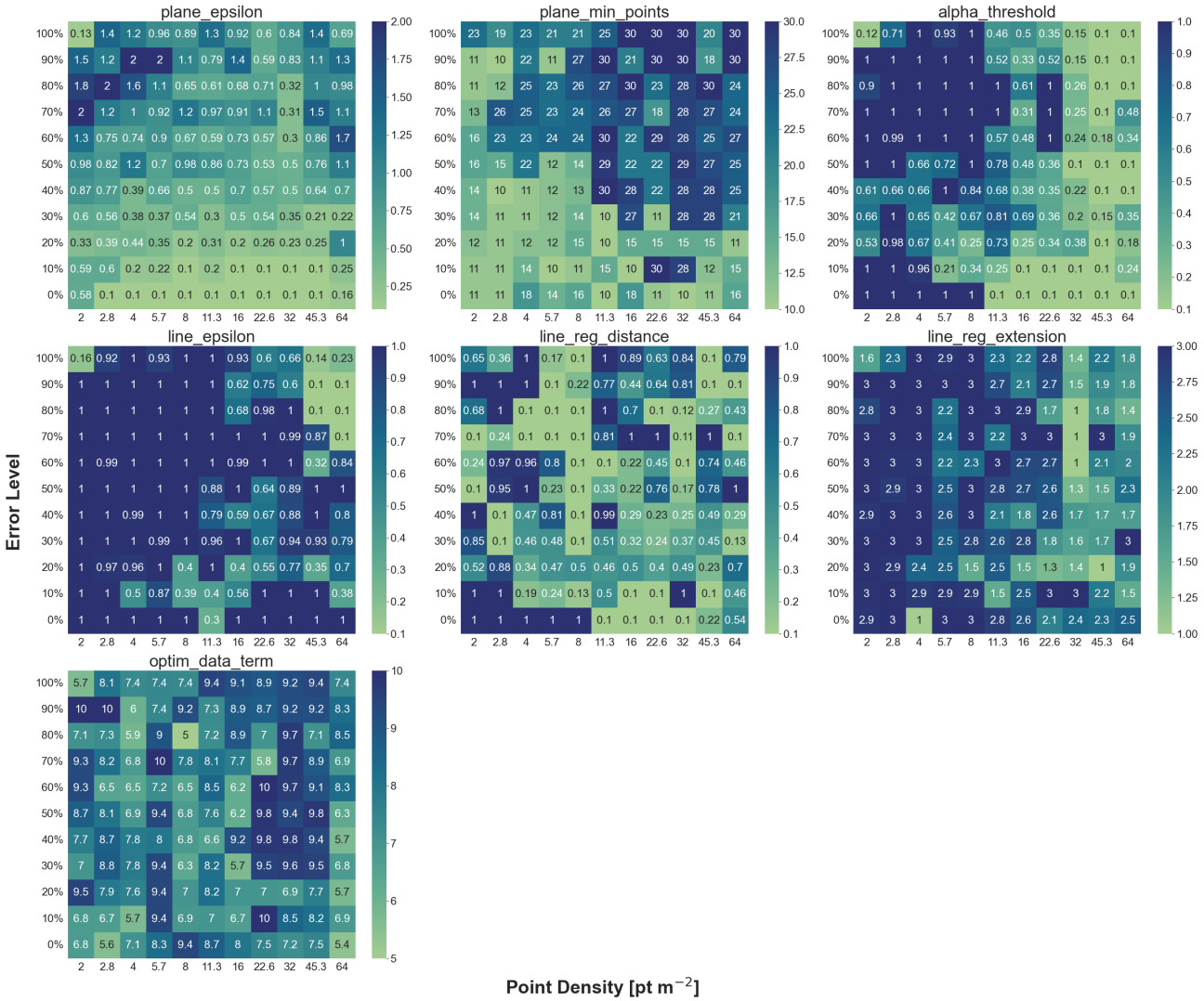


Figure 3.5 The distribution of the optimized values of the seven reconstruction parameters across the point density and error levels.

the parameter's effect: The more points there are in a given area, or the higher their dispersion, the more points should be required to constitute a plane. Finally, alpha_threshold specifies the distance for the alpha shape computation of the detected planes. Like plane_epsilon, it decreases with increasing point density and, to a lesser extent, with decreasing error level, and it can be explained with the same reasoning.

In the second group, the relationship is less clear but still distinguishable. It, too, contains three parameters: The optimized value of line_epsilon is mostly located at the upper end of the permitted parameter range, but lower values appear toward high-density and lower-error scenarios. Albeit less distinctive, this reflects the pattern found for plane_epsilon, which aligns with the similar purpose of the parameters. The parameter line_reg_extension as optimized exhibits a tendency to decrease with increasing point density but no clear relationship to the error level. Since its function is to extend the detected lines to allow better identification of complete roof planes, a stronger reliance on this mechanism in the case of sparse points is consistent with its objective. Lastly, optim_data_term demonstrates only a weak correlation to the point cloud properties, but a slight tendency toward higher values with high-density point clouds appears to be present.

The last group consists only of line_reg_distance, which defines the distance up to which sufficiently parallel lines are assigned to the same cluster for regularization. The results indicate that the optimization was unable to identify any meaningful relationship between the point cloud properties and parameter values that are conducive to the reconstruction quality.

3.2.2 Optimized Quality Metric

The top left plot in figure 3.6 demonstrates the best values of the RMS minimum distance achieved in the optimization of each scenario. In addition to the regular optimization iterations, the default parameter values (see table 2.1) that are tuned for a point density of $8\,\mathrm{pt\,m^{-2}}$ were evaluated for every scenario as a baseline for comparison, based on the same subset of buildings. The top center plot shows the RMS minimum distance obtained in this case. (Note that the vertical scales of the plots are not aligned to emphasize the relations within the individual plots.) As difference plot at the top right illustrates, particularly large improvements in RMS minimum distance were attained for low point densities and high error levels. In most cases of point densities greater or equal to $8\,\mathrm{pt\,m^{-2}}$ and error levels from 0 % to 20 %, the optimization method was unable to identify parameter values that provide a significant improvement in RMS minimum distance.

3.2.3 Effects of the Single-Objective Optimization on Other Metrics

As outlined in the section introducing the quality metrics (see section 2.2.4), the quality of a 3D building model cannot be assessed in one dimension, neither on its own nor in comparison to the building (model) it represents. In practice, the quality definition will depend on various quantitative and qualitative considerations, not least the intended application. For the purpose of this project, a single quality metric for the optimization of the parameters with respect to the objective function was chosen, and the RMS minimum distance was suitable due to its explanatory power at a relatively low computational cost.

For this reason, the optimization results were evaluated with particular attention to possible biases introduced in other quality metrics that would indicate a trade-off between the optimized and the other metric. In most instances, the quality metrics showed good correlation to the optimized RMS minimum distance, which was true in particular for those that likewise quantify geometric similarity: Hausdorff distance and 3D IOU.

However, a significant case of an inverse relationship was discovered in the complexity of the reconstructed models (measured as the number of their constituent faces after triangulation): The minimization of the geometric distance between the output and input models generally favors parameter values that lead to a higher model complexity. This applies in particular to scenarios with high point density and, even more strongly, with high error levels of the input point cloud.

The relationship is demonstrated in the bottom left plot in figure 3.6: Building models reconstructed using the optimized parameters have fewer than 200 faces up to a point density of $4\,\mathrm{pt\,m^{-2}}$ across all error levels. This number keeps increasing only gradually beyond this density for low error levels, but the combination of high point densities and error levels yields building models with a complexity in excess of 1000 faces. Starting at a point density of $32\,\mathrm{pt\,m^{-2}}$, this trend begins to reverse, mostly for high error levels. For the two higher density levels (45.3 and $64\,\mathrm{pt\,m^{-2}}$), this reversal can at least partially be explained by the implemented reconstruction timeout taking effect in many instances, which imposes a de facto limit on a further complexity increase, since the most complex building models take longest to reconstruct. This leaves the trend reversal at $32\,\mathrm{pt\,m^{-2}}$ unexplained, however, as here the timeout did not play a role.

As the bottom center plot reveals, a similar relationship can be observed when using the default parameter values that were tuned for a point density of $8\,\mathrm{pt\,m^{-2}}$: While, in this case, the face numbers remain low for all error levels up to a higher point density, they increase even more sharply and to a higher level for high-density and high-error point clouds while delivering building models that present more poorly in terms of their geometric distance to the input models. Note that the reconstruction timeout was suspended for the evaluation of the default parameter values to obtain a complete picture of the relationship. It would have become effective in some scenarios with point densities above $32\,\mathrm{pt\,m^{-2}}$.

The occurrence of this effect on the model complexity when optimizing for a minimal geometric distance is intuitively comprehensible considering that a finer subdivision of a model's bounding surface into faces allows a closer approximation of the input model, and even more so if the information about the original model's shape is obscured by increasing amounts of random error in the point cloud. While this indeed yields better values of the target metric, it comes at the expense of the reconstructed models' resem-

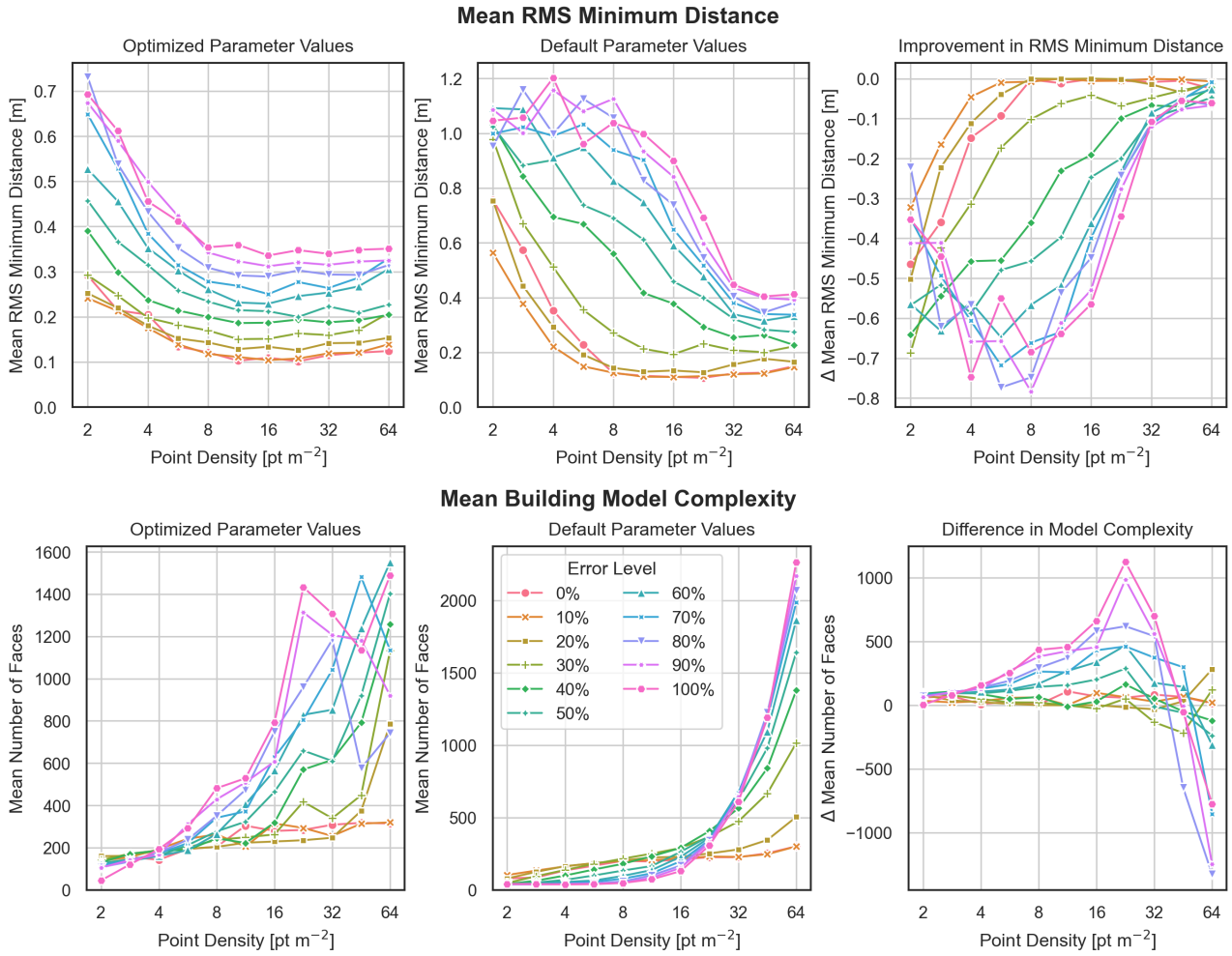


Figure 3.6 Results of the optimization of the reconstruction parameters across all scenarios in terms of two metrics: RMS minimum distance, which was the optimization target, and building model complexity. In all cases, the results are based on the reconstruction of the subset of 100 buildings used for the optimization, and the mean across all buildings is presented. The plots to the left show the results obtained in the best performing optimization iteration for each scenario. For comparison, the center plots show the results from a reconstruction of the same subset using the default reconstruction parameters (see table 2.1). The plots to the right demonstrate the difference between the two. Note that the vertical scales of the plots are not aligned to emphasize the relations in the individual plots.

.

blance of real buildings, which generally consist of a comparatively small set of comparatively large, planar surfaces.

A multi-objective optimization that additionally targets a limited model complexity (or other quality aspects) could help to strike a balance in this trade-off situation, but it is challenging due to the difficulty of harmonizing the units of measurement to obtain a meaningful, combined target metric. Suitable values of scaling parameters could be determined through exploration, but would introduce yet another degree of freedom in the interpretation of the reconstruction quality.

3.2.4 Selection of Parameter Values for the Reconstruction

Two considerations led to the decision not to use the parameter values found to be optimal in terms of the RMS minimum distance. First, the disproportionately complex building models they result in may render any insights regarding the research question meaningless for practical purposes. Second and from a purely technical perspective, the duration and stability of the reconstruction process are also negatively affected by such parameter values which impeded the feasibility of the experiment. The goal was, there-

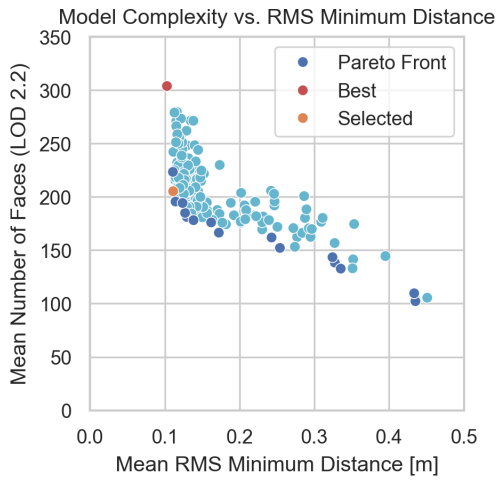


Figure 3.7 The correlation between the optimization target metric (RMS minimum distance) and building model complexity as obtained from the optimization of one scenario (11.3 pt m^{-2}, 0 % error). The Pareto optimal, best (in terms of the target metric), and selected solutions are highlighted. The selected solution is the one within 10 % of the best one that has the lowest complexity.

fore, to select for each scenario a set of parameter values from another optimization iteration that provides a good ratio of the cost in terms of a higher RMS minimum distance to the benefit in terms of a lower model complexity. In other words, a parameter set that performs negligibly worse in the target metric but significantly better in the model complexity.

Figure 3.7 provides an example of the correlation typically found between the two conflicting quantities. It is evident that the last bit of improvement in the geometric distance comes at an excessive deterioration in the model complexity. Conversely, a significant improvement in model complexity is attainable at a small cost in the optimized metric. To limit the introduction of additional randomness in the reconstruction outcome, the approach was to select the parameter values probed in the optimization iteration within 10 % of the best target metric outcome that yielded the least complex models. This choice is by definition Pareto optimal: There can be no other solution that improves one of the metrics without deteriorating the other. Finally, additional randomness can only be introduced to the experiment outcome by the fact that the optimization procedures for different scenarios have approached their respective optimum stochastically, i.e., the amount of evaluations of the target function within this range from the optimum varies. Figure A.3 in appendix A.2 provides the same overview of the selected parameter values as was given in figure 3.5 for the parameter values corresponding to the best target metric outcome.

3.3 3D Building Reconstruction

3.3.1 Reconstruction Results

The reconstruction of the full set of buildings present in the input data was performed using the reconstruction parameters that were selected from the optimization procedure as described in the previous section. The duration of the reconstruction scaled from a few minutes for the lowest point densities to several hours for densities of 64 pt m^{-2} and above. Additionally, in the latter case the reconstruction process became increasingly unstable: It repeatedly stalled at two particular steps in the reconstruction pipeline related to the optimization of the roof partition and had to be restarted in these cases. The standard output of the reconstruction software Geoflow allowed the identification of the buildings responsible for these difficulties, which were in many, but not all cases those with large footprints and complex geometries. The reconstruction of the scenarios with a point density of 90.5 pt m^{-2} could only be finished by removing three of these footprints from the reconstruction input. At the point density of 128 pt m^{-2} the reconstruction succeeded

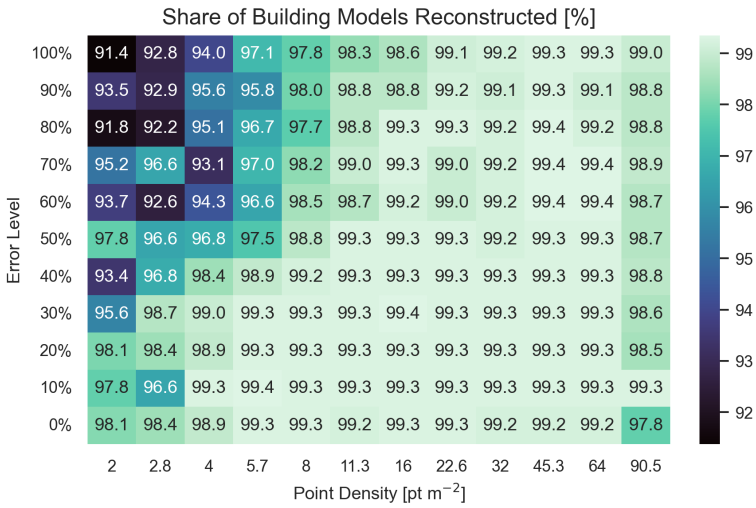


Figure 3.8 The share of the total number of building models present in the input dataset (1079) that was reconstructed successfully in each scenario.

after many attempts for only two scenarios. For this reason, no results can be presented for this density level.

For all other scenarios, figure 3.8 displays the share of the input building models that was reconstructed successfully. Evidently, higher point densities and lower error levels favor a more complete reconstruction output in general, with the exception of 90.5 pt m^{-2} density level, which exhibits a slight decline in the share of building models reconstructed. This is likely due to the increasing instability of the reconstruction method, possibly in conjunction with the choice of reconstruction parameters.

3.3.2 Evaluation of the Reconstructed 3D Building Models

Aggregated Results: Geometric Distance and Model Complexity

Figure 3.9 provides the evaluation results of the reconstructed 3D building models in LODs 1.2, 1.3, and 2.2 in terms of the RMS minimum distance, Hausdorff distance, and model complexity. The latter is displayed as the ratio of the output model face number to that of the input model. In all cases, the mean across all building models is presented. Despite their close correlation, both of the two geometric distance measures are included intentionally: The RMS minimum distance was the target of the reconstruction parameter optimization and can therefore be expected to provide the most unbiased results, whereas the Hausdorff distance is better established as a metric in the literature. Both offer valuable insights to address the research question.

Most findings apply qualitatively to all LODs: An improvement in the geometric similarity between the output and input building models can be clearly observed with increasing point density up to approximately 11.3 or even 16 pt m^{-2}, and with decreasing level of error down to its absence. For both point cloud properties, the improvement is most pronounced in the value range associated with a lower information content (a low density and a high error level) and becomes more gradual toward better values. High levels of random positional error are more detrimental at the lowest point densities. With the point density increasing beyond 16 pt m^{-2}, the geometric distance remains at best constant (LOD 1.2 at low error levels) or deteriorates gradually.

The reconstructed building models have on average a significantly higher number of faces than the input building models in almost all scenarios and LODs. At low point densities, a higher error level yields more simplified building models, whereas building models reconstructed from point clouds with lower error levels possess more details. Only at LOD 2.2 does this cause the reconstructed building models to have fewer faces on average than the input models. This relationship reverses between approximately 4 and 8 pt m^{-2}, beyond which high error levels increasingly result in exceedingly complex building models that have up to

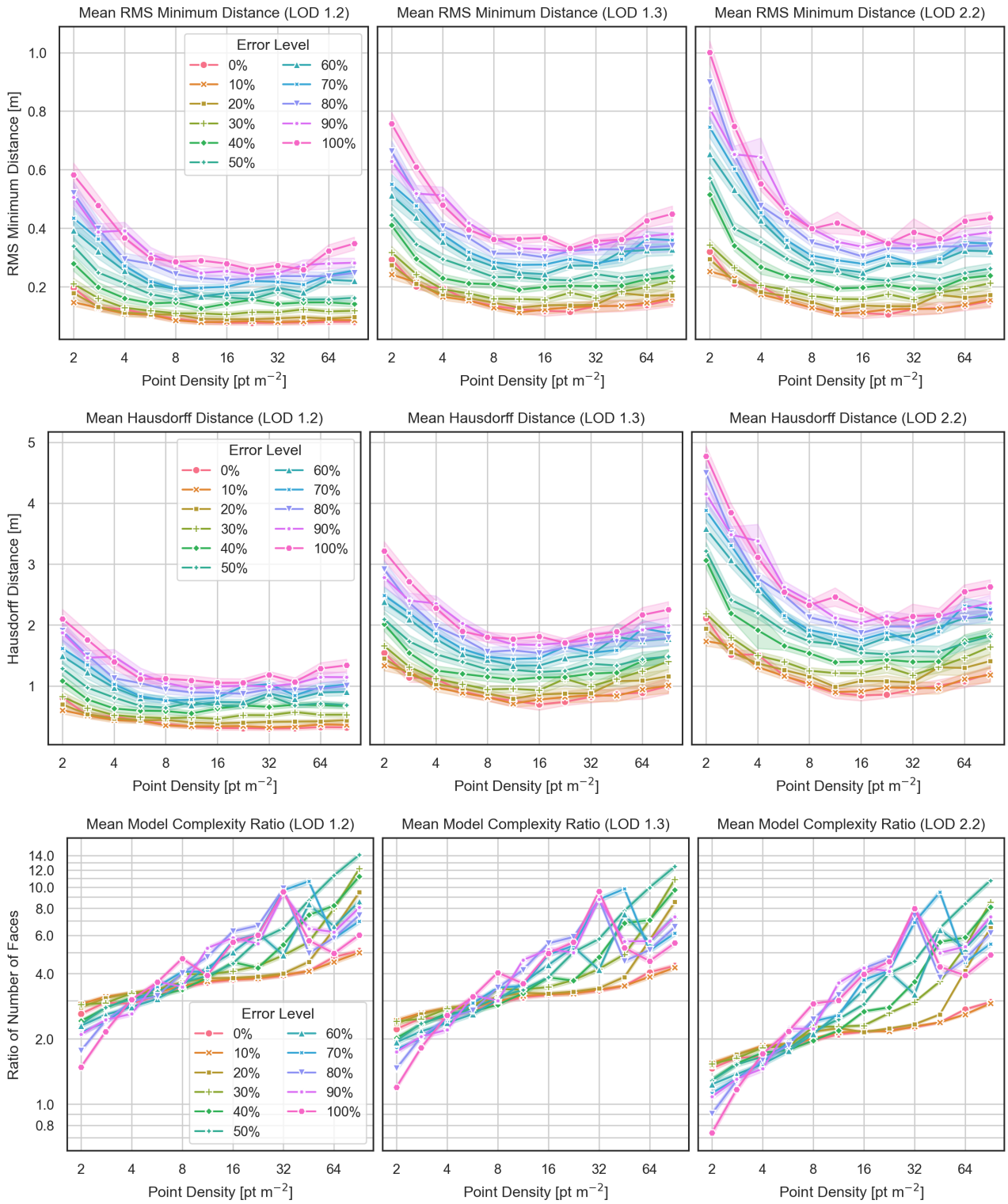


Figure 3.9 Evaluation of the reconstructed 3D building models in three LODs in terms of three metrics: RMS minimum distance, Hausdorff distance, and building model complexity ratio (output to input model face count). In all cases, the mean across all building models is presented. The error bars show 5 % of the standard deviation in each scenario, which gives an indication of the high spread.

an order of magnitude more faces than the input models. For low-error scenarios, the model complexity increases as well with increasing point density, but much more gradually. At point densities of 45.3 pt m^{-2} and higher, the impact of the error level on the model complexity displays a similarly chaotic pattern as was observed in the results of the parameter optimization.

A comparison between the three LODs provides two insights: First, an increasing level of detail clearly poses a greater challenge to the reconstruction process and therefore leads to a higher geometric distance across all evaluated point cloud properties. This is to be expected considering the difference in the information content of the involved models: At LOD 1.2, the reconstruction algorithm must only infer a single height value as opposed to a potentially complex roof structure at LOD 2.2.

Conversely, the relative increase in model complexity is more pronounced at LODs 1.2 and 1.3 than at LOD 2.2. For the most part, this may be a consequence of the reconstruction parameter settings, which were found to be biased for more complex models due to the single-objective optimization for a low geometric distance. Nevertheless, an unnecessarily high model complexity can be considered particularly undesirable at the LODs that should represent building geometries in the most simplified manner.

The geometric distance metrics as well as the model complexity ratio exhibit a degree of stochasticity that can be attributed to at least two factors: First, a probabilistic method was used for the optimization of the reconstruction parameters. This inevitably introduces a certain randomness in the degrees to which the different scenarios are optimized, which is difficult to compare. Second, the large number of executions of the reconstruction software revealed that the reconstruction process itself is not deterministic. Differences can be observed between the building models generated from the same input data in different reconstruction runs. In some cases, this took the form of vertices with extreme outlier coordinates that corrupted the meshes. (These were identified by checking for outliers in each coordinate across all vertices, and the reconstruction was restarted in case of a positive result.) Both factors likely contribute to the observed random variation in the quality metrics.

Aggregated Results: Height Difference

Figure 3.10 visualizes the mean height differences between the reconstructed and the input models. Three different height metrics are compared that all relate to the roof: The minimum, 70th percentile, and maximum elevations of points used to reconstruct the LOD 2.2 roof. Since they refer to the height of the points, not the reconstructed roof itself, they do not, in general, correspond to the respective roof heights: At low error levels, there will be a close correspondence, whereas at higher error levels, the minimum and and maximum point elevations will respectively be smaller and larger than the corresponding roof heights, since they capture the extrema of the point distributions whereas the roof will be based on average measures. As noted in section 2.2.4, the 70th percentile height serves as reference for the default LOD 1.2 models.

At first glance, a systematic difference of approximately –5 m becomes apparent across all three metrics, which points to a corresponding discrepancy between the two input point clouds. It should therefore be reflected in the 3D building models' height coordinates as well, but this is not the case. Moreover, there is no corresponding bias in the other metrics: The Hausdorff distance, at least, should exhibit the same deviation. This indicates that the difference arises in the computation of the height attributes for the output CityJSON file, but its exact source could not be resolved. The identical coordinate reference system is used in the entire workflow (EPSG:7415, Amersfoort / RD New + NAP height).

Disregarding this bias, the height differences remain challenging to interpret. It is useful to approach the analysis by formulating the patterns that are expected to then explain the observed differences. First, changes in point density should only affect the height of the points if the denser sampling of the building surfaces includes higher or lower areas that are not captured by lower point densities. This may indeed imply that the observed minimum height decreases and the observed maximum height increases with increasing point density. Both can be observed in the diagrams, but the minimum height changes significantly across all error levels with increasing point density, while the maximum height remains largely constant at low error levels and only increases with the point density at higher error levels. This will addressed below.

Second, an increasing error level implies an larger displacement of points in the vertical direction, where the height difference corresponds to a normal distribution. Correspondingly and assuming a constant point

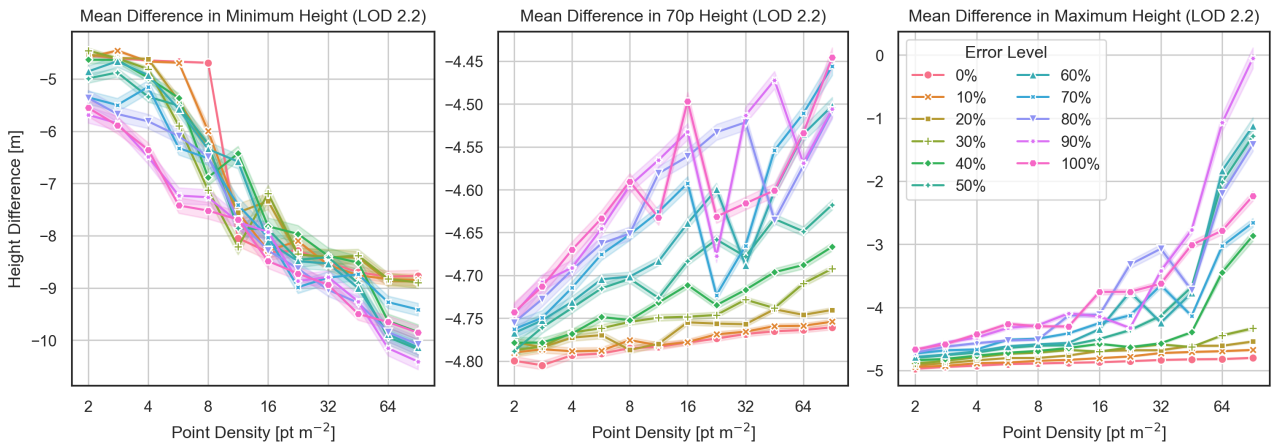


Figure 3.10 Mean difference in three roof height metrics between the reconstructed and the input 3D building models: Minimum, 70th percentile, and maximum elevation of all points used to reconstruct the LOD 2.2 roof. The 70th percentile height is used as reference of the default LOD 1.2 models. The error bars show 5 % of the standard deviation in each scenario, which gives an indication of the high spread.

density, the elevation of the lowest point (minimum height) is expected to decrease and the elevation of the highest point (maximum height) should increase with an increasing error level. Again, these patterns are visible in the diagrams, yet the effect's dependence on the point density once more differs between the minimum and maximum height. However, the maximum difference change of about –5 m to 5 m respectively between minimum and maximum height corresponds well.

Thus, the difficulty lies in understanding the interactions of changes in both point density and error level. Why does an increasing point density affect the minimum height much more than the maximum height at low error levels? Following the logic stated above, it must be due to the denser sampling of the building's surface exerting a stronger influence on the registration of low roof points as opposed to high roof points. Since an increased point density also causes a denser sampling of the building walls, it is possible that the reconstruction algorithm mistakenly associates a larger number of wall points with roof surfaces. This can only affect the minimum height but not the maximum height and may therefore explain the observed difference in the dependence on the point density. Additionally, it helps to understand why the minimum height decrease with point density is linear, whereas maximum height increases non-linearly and more steeply toward higher densities.

It does not explain, however, why the combination of a maximum error (which has a standard deviation of 0.3 m vertically) and the highest point density results in a difference in maximum height of 5 m relative to zero error. Certainly, approximately 32 % of all points are displaced vertically by more than 0.3 m and half of these to a higher elevation, which may include some significant outliers, but not up to 5 m on average. It would be possible that the mean is skewed by few buildings with exceptional internal height variability that only affects these values at dense surface sampling. However, this can be ruled out because, first, outliers have been filtered for the visualization in these diagrams and, second, the median exhibits the same pattern.

Distributions

To provide a different perspective on the aggregated findings in the previous section, examples of the distributions of two metrics across the reconstructed building models are given for a subset of scenarios in figure 3.11: The RMS minimum distance as a measure of geometric similarity at the LODs 1.2 and 2.2, and the building model complexity ratio (output to input model face number) at LOD 2.2. Outliers, defined as values with a modified Z-score greater than 3.5, were excluded from the calculation of the kernel density estimate. All scenarios with point densities of 4, 8, 16, and 32 pt m^{-2} are represented.

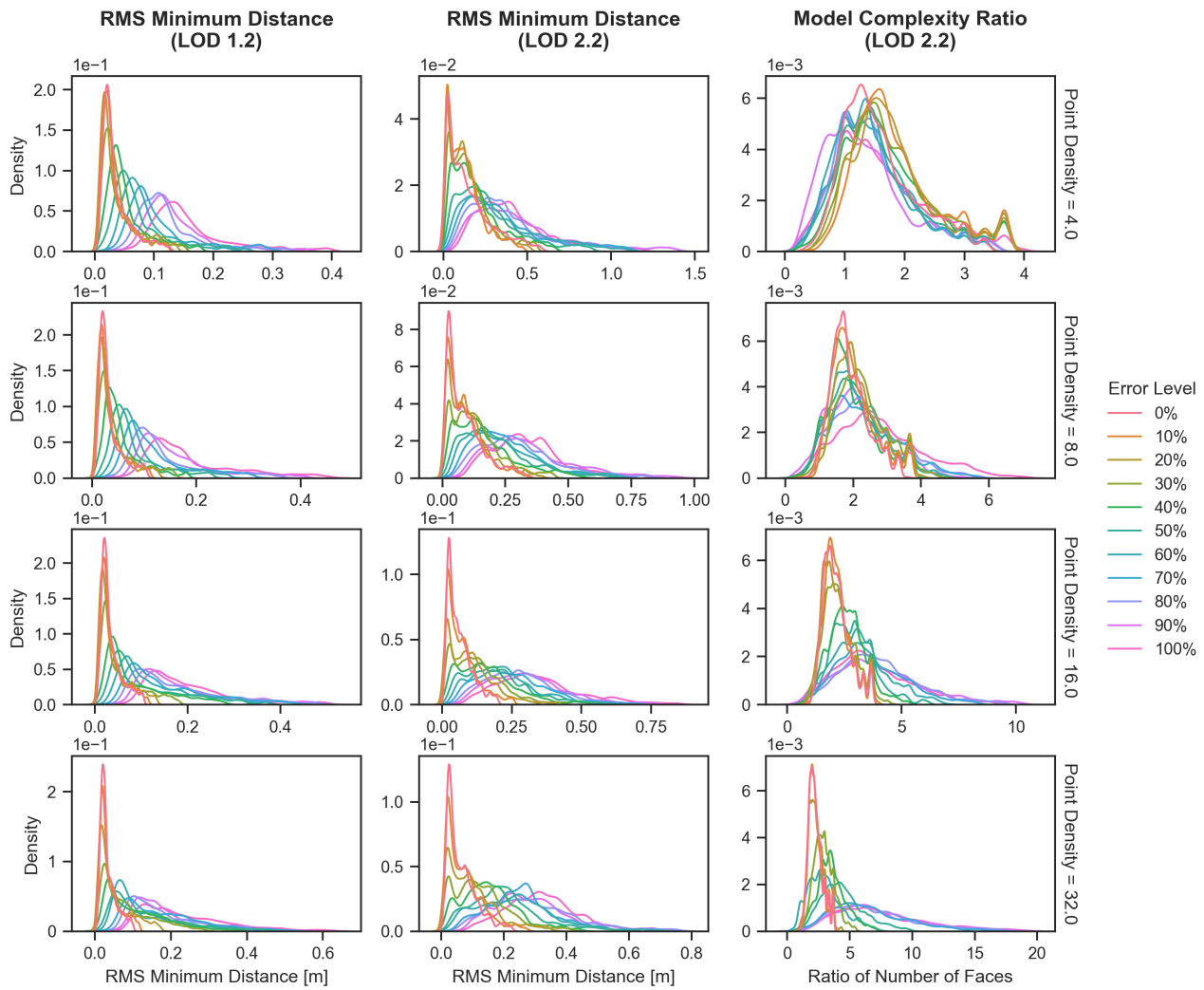


Figure 3.11 Building-level distributions of two quality metrics at two LODs and four point density levels (4, 8, 16, and 32 pt m^{-2}) across all error levels: RMS minimum distance at LODs 1.2 and 2.2, and the building model complexity ratio at LOD 2.2. The labels to the right indicate the respective point densities. The vertical axes represent the kernel density estimate. Note that outliers, defined as values with a modified Z-score greater than 3.5, are not included in the calculation to facilitate the reproduction of the central part of the distributions in each plot. For the same purpose, neither the horizontal nor the vertical axes' scales are aligned.

The qualitative distribution of the RMS minimum distance is fairly similar in all cases: Low error levels result in a high peak close to zero and a long tail while, with increasing error levels, the mode wanders to the right and the distribution becomes wider. Moreover, the value spread contracts with increasing point density at LOD 2.2, whereas it increases slightly at LOD 1.2. Notably, the distributions of lower-error scenarios at LOD 2.2 exhibit a second, lower peak slightly further from zero, and higher error levels show further undulation. The origins of these cannot be traced here, but a relation to building models with different geometric properties appears likely and is demonstrated in the following section.

Corresponding to the insights from the aggregated data, the distributions of the model complexity ratio at LOD 2.2 exhibit a different relationship to each other than those of the RMS minimum distance: At a low point density, the distributions of higher-error scenarios lean to the left of the lower-error distributions, indicating a lower relative number of faces. As the point density increases, the lower-error distributions remain comparatively stationary, whereas those at high error levels widen significantly with their tails extending to higher values.

Model Surface Area, Volume, and Intersection over Union

As described in section 2.2.4, methods to evaluate the surface area, volume, and IOU of the reconstructed building models with respect to the input models were developed using the software FME®. The IOU is a comparative metric by itself, and area and volume difference to the input models can be computed for comparison. Unfortunately, the license for academic purposes provided by Safe Software Inc. expired before the final dataset was available, and an extension was not granted. Therefore, these evaluations could not be performed for the building models reconstructed in this experiment.

However, multiple evaluations of the IOU in previous, smaller-scale experiments showed a close correlation with both Hausdorff and RMS minimum distance. While this suggests that its assessment may have provided only a limited additional benefit, it may nevertheless have delivered valuable insights on this larger scale with a wider variety of building geometries. The derived metrics of surface area and volume, however, do not have such close relatives and the insights they may have yielded therefore cannot be replaced.

3.3.3 Evaluation of the Reconstructed 3D Building Models by Groups

Figure 3.12 visualizes how groups of buildings with different geometric and semantic properties contribute to the distribution of the RMS minimum distance between input and reconstructed building models. Because this was found to differ more strongly between the levels of random error than between point density levels and a small data subset had to be selected to facilitate the visualization, the figure represents building models reconstructed from point clouds at the single density of 11.3 pt m^{-2} and error levels of 0 %, 30 % and 60 %.

Geometric Properties of the Input Building Models

The first row of distributions in figure 3.12 illustrates the influence of the input building models' complexity on the reconstruction quality: Across all presented error levels, an increase in model complexity is accompanied by a deterioration in geometric similarity. This emphasizes that, on average, more complex buildings are more difficult to reconstruct accurately. Moreover, it appears that the face number of the input models can at least partially explain the bimodality of the overall distribution that can be observed at an error level of 30 %: Reconstructed building models that are derived from models with fewer than 25 faces are largely responsible for the first mode, indicating a lesser impact of a moderate point cloud error on simpler geometries. At an error level of 60 %, however, the gap closes, but the correlation holds. Although other point density levels are not represented in figure 3.12, they have been found to exhibit the same qualitative relationship.

A differentiation of the RMS minimum distance distribution by the volume of the input buildings reveals a similar pattern. This is plausible considering that the building volume and the complexity of their models are moderately correlated with a correlation coefficient of 0.69 for buildings with a volume of up to 2000 m^3. A low building volume is, however, less predictive of a low RMS minimum distance than a low complexity of the input model, as far as this can be judged qualitatively from the distributions.

The roof geometry is a property that may have a significant influence on the reconstruction outcome. One way to quantify it is to compute the share of the total roof area that is flat, as opposed to the sloped proportion. This value is provided for each building model in the Dutch 3D cadastre and its influence is visualized in the third row of figure 3.12. The areal proportions of the individual distributions demonstrate that buildings with no flat roof parts constitute the largest group, and that the groups become successively smaller with increasing share of flat roof areas. Buildings with entirely flat roofs are more frequent again. At every depicted error level, the latter group's distribution covers the lowest range of values, which confirms the expectation that flat roofs are easier to identify and reconstruct than sloped ones. Correspondingly, all groups of buildings that have any sloped parts perform worse in the reconstruction than those with entirely flat roofs. Among them, buildings with exclusively sloped roofs exhibit the best values, and buildings with

both flat and sloped roof parts perform worse. These relationships do not vary to a strong degree between the three error levels.

Semantic Properties of the Buildings

While it is unlikely that the age or usage of a building themselves exert an influence on the reconstruction, it is possible that they correlate with other properties that do. For instance, the architectural epoch during which a building was designed may be expressed in the geometry of its roof: The proportion of new buildings with flat roofs has certainly increased during the 20th century. Likewise, buildings that were constructed with a particular purpose in mind may exhibit certain stylistic features: Industrial and some commercial edifices may follow a more functional design pattern, whereas residential buildings may be smaller and more distinctive. Additionally, these aspects certainly differ between urban and rural regions and are influenced by cultural factors. It may therefore be possible to identify a relationship between semantic properties of buildings and the quality of the reconstructed 3D models as a function of the point cloud properties investigated here.

A significant proportion of the buildings in the study area, the inner city of Utrecht, were constructed before the year 1500. The second-largest groups were built between 1800 and 1900, and after 1950, respectively. The youngest group of buildings has its mode at the lowest value and the shortest tail at all error levels, which indicates that the reconstruction delivers the best results for modern architectures. Indeed, buildings in these group have on average about 130 faces and a 40 % share of flat roof areas, whereas the older groups have averages of 200 faces or more and between a 24 % and 32 % share of flat roof areas. Both of these geometric attributes were already identified to affect the quality of the reconstruction. The building groups from before 1500 and between 1800 and 1900 mainly show a distribution with a wider spread toward larger values of the RMS minimum distance. Although this difference is subtle, the pattern can be found across the provided error levels as the geometric similarity deteriorates. The other groups' shares in the dataset are too insignificant to allow meaningful statements.

The building use does not appear to affect the distribution of the RMS minimum distance at any of the three error levels with the data used here. Many buildings in the study area are assigned multiple use types, and in this analysis these buildings contribute to each of the usage groups. This, of course, diminishes any differences between the groups and is therefore a confounding factor in the attempt to identify any influence of the building use. This finding is unfortunate because the input region was selected with the specific aim to maximize the variety of use types present to allow meaningful statements about their influence on the reconstruction quality. The fact that this variety is mainly a result of multiple use type assignments and that this would impede the evaluation as opposed to facilitate it was not anticipated.

3.3.4 Visualization of the Reconstructed 3D Building Models

This section aims to provide a visual comparison both among the reconstructed building models and between them and the original building model from which the point clouds were generated. Because of the large number of scenarios representing different combinations of point cloud density and accuracy (143 in total, of which 132 passed the reconstruction pipeline successfully), a subset of scenarios had to be selected for the visualization. Because the reconstruction quality improves most significantly in the lower range of point density levels, this range is sampled more densely, resulting in the selection of 2, 2.8, 4, 5.7, 8, 16, 32, and 64 pt m^{-2}. The error levels were chosen uniformly from 0 % to 100 % in steps of 20 %. Moreover, a building model with a variety of geometric features was selected for this purpose: It has both historic and modern components, sloped and flat roof parts, and a series of dormers that are well suited to indicate the performance of the reconstruction with respect to finer geometric details. This permits meaningful insights that are representative of the overall results. Figure 3.13 shows the original building model from the 3DBAG, and figures 3.14 and 3.15 provide an overview of the reconstruction results for the selected scenarios.

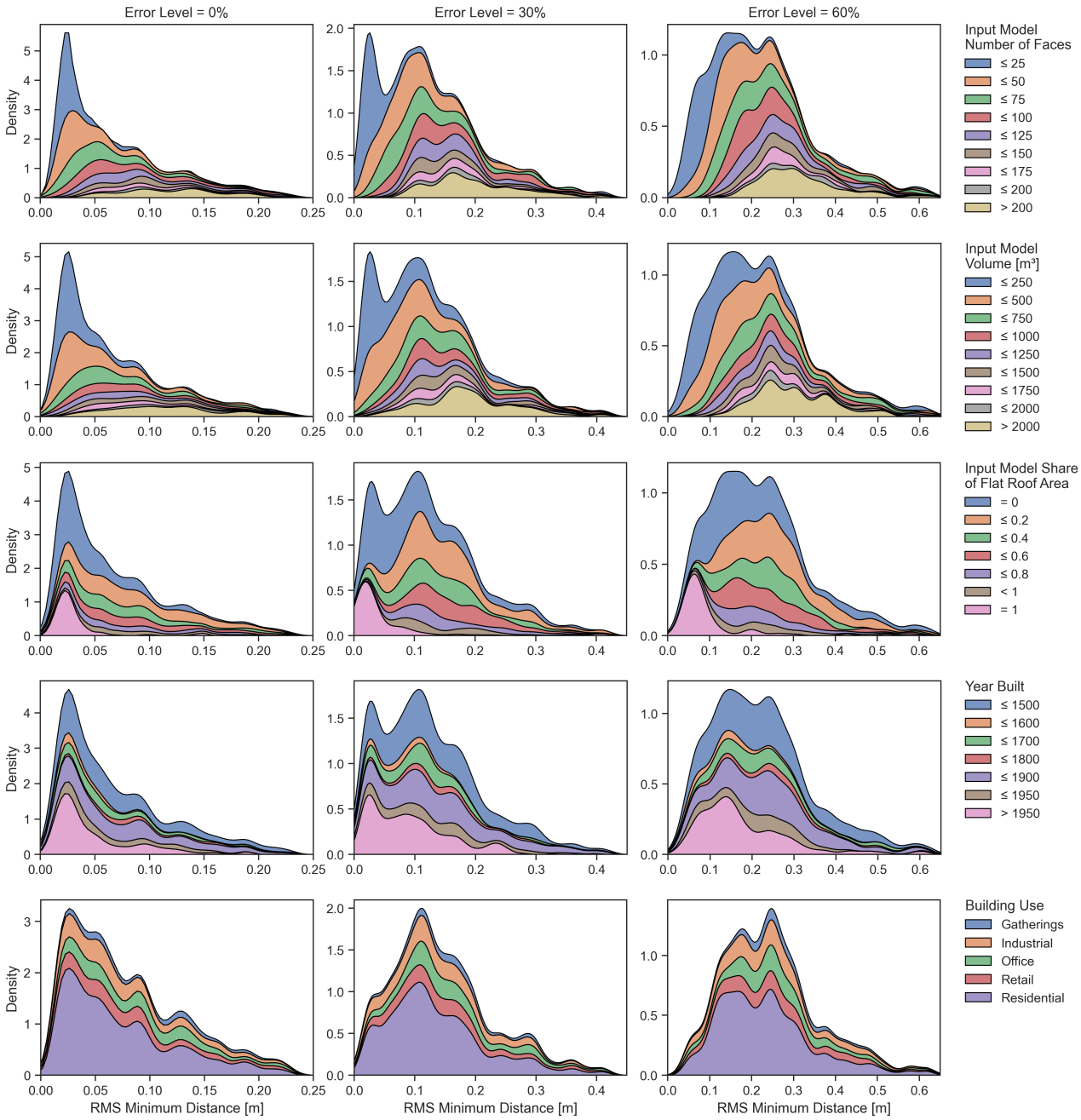


Figure 3.12 Building-level distributions of the RMS minimum distance at 11.3 pt m^{-2} density and an error level of 0 %, 30 %, and 60 %, differentiated by groups of properties: Geometric properties of the scanned LOD 2.2 input building models (number of faces, volume, and ratio of flat to total roof area), and semantic properties from the Dutch cadastre BAG (year built and building use). The legend of numerical properties provides only the upper limit of each bin; its lower limit is the upper limit of the previous bin. Slight variations in the overall shape of the distribution result from taking the sum of the groups' kernel density estimates. Note that the overall shape of the distributions in the bottom row differs from the others because many buildings have multiple use types and therefore contribute to more than one group. As before, outliers, defined as values with a modified Z-score greater than 3.5, are not included in the calculation to facilitate a better reproduction of the central part of the distribution.

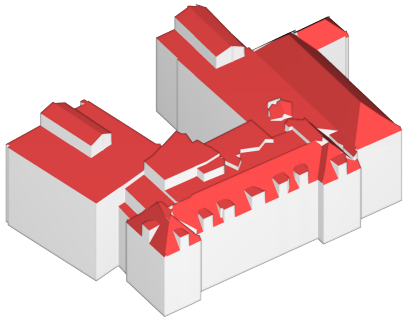


Figure 3.13 The original building model from the Dutch 3D cadastre (3DBAG) of the building in 28 Hamburgerstraat, Utrecht, that served as input for the generation of the point clouds used to reconstruct the building models presented in figures 3.14 and 3.15. Its identifier in the Dutch cadastre (BAG) is 0344100000069160.

Low Point Densities

Even when reconstructed from low point densities up to 4 pt m^{-2}, the building models capture the roof geometry of the input model reasonably well up to an error level of 20 %, even more so starting at 2.8 pt m^{-2}. At higher error levels the results indicate increasing difficulties in the identification of sloped roofs, reflected in a corresponding number of roof parts that are modeled as flat. A possible interpretation of this is that, in these scenarios, the optimization algorithm favored reconstruction parameters resulting in flat roofs because they yield a higher geometric similarity in terms of the RMS minimum distance, as opposed to parameter values that may still be able to reconstruct sloped roofs but do so less reliably. Nonetheless, simplified yet valid roof geometries can be considered successful reconstruction attempts at these low point densities in the presence of moderate error. Only the reconstruction results from point clouds with the highest error levels exhibit significant erroneous artifacts.

The input building's dormers are first identified and modeled individually at a point density of 5.7 pt m^{-2}, with diminishing success up to 40 % error level. Interestingly, they are first captured in traces at 4 pt m^{-2} and 20 % error. Their absence at this density and 0 % error is therefore likely due to a random variation in the optimized parameter values: The minimum threshold of 16 points for the detection of a plane (parameter plane_min_points), which is higher than the values between 10 and 13 points in other scenarios in this range shown in the figure, certainly impedes the identification of fine geometric details such as the dormers. (Note that figure 3.5, where the corresponding value for this scenario is 18, does not show the parameter values used for the reconstruction, as is explained in section 3.2.4.)

A relationship that was found in figure 3.6 and discussed in section 3.3.2 becomes clearly visible in these figures: At the lowest point density, an increasing error level leads to a stronger simplification of the reconstructed models in terms of their number of faces, whereas this pattern reverses with increasing point density, resulting in the reconstruction of highly complex models with fractured geometries at high error levels. This reversal begins to take effect at densities of 4 pt m^{-2} and, more so, 5.7 pt m^{-2}, as is immediately apparent both in the 3D models as well in their face numbers as provided in figure 3.14.

High Point Densities

At point densities of 8 pt m^{-2} and above (see figure 3.15), the reconstructed models at low error levels are very close to the input models, both visually and in terms of their RMS minimum distance. This remains true up to 20 % error level. This degree of accuracy in the representation is accompanied, however, by a significant increase in the models' complexity, which is not inherently beneficial: The building model at 64 pt m^{-2} and 0 % error has 167 % more faces than the input building model but offers no advantage over

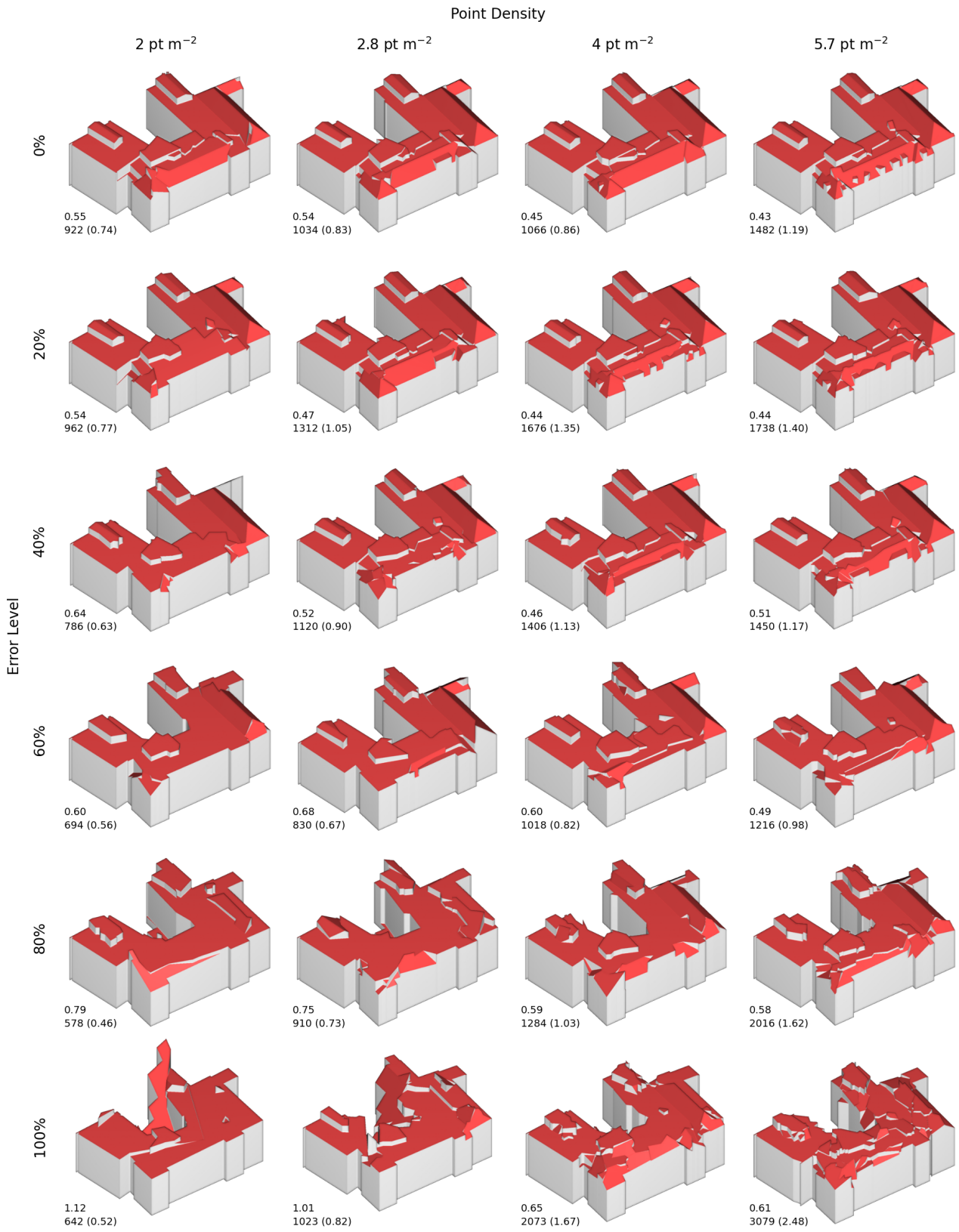


Figure 3.14 3D building models reconstructed from point clouds at 2, 2.8, 4, and 5.7 pt m^{-2} density and error levels between 0 % and 100 % in steps of 20 %. The numbers to the bottom left of each model indicate its RMS minimum distance [m] (top) from the input building model and its face number (bottom), including the ratio to that of the input model in brackets. The corresponding input building model from the 3DBAG is demonstrated in figure 3.13.

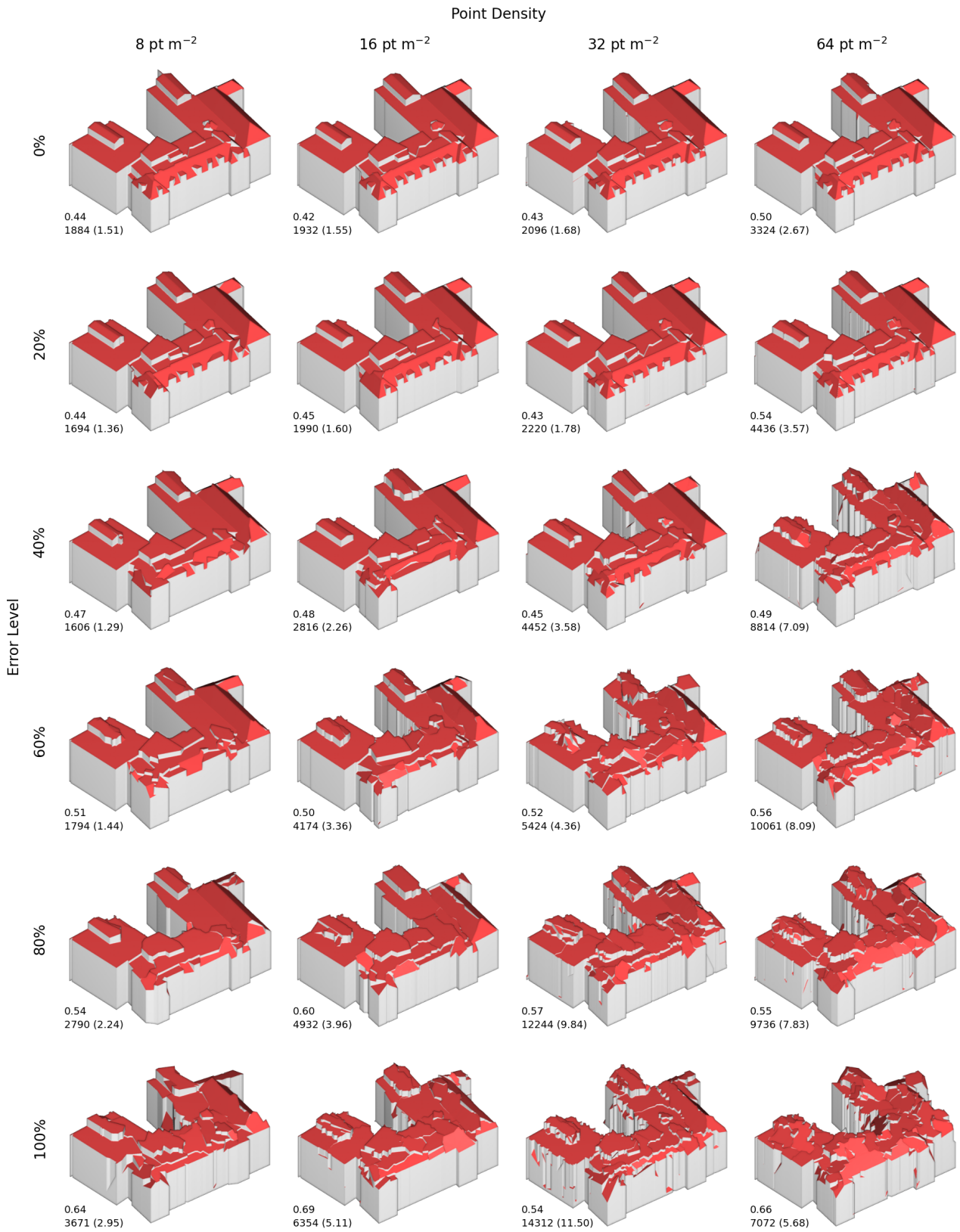


Figure 3.15 3D building models reconstructed from point clouds at 8, 16, 32, and 64 pt m^{-2} density and error levels between 0 % and 100 % in steps of 20 %. The numbers to the bottom left of each model indicate its RMS minimum distance [m] (top) from the input building model and its face number (bottom), including the ratio to that of the input model in brackets. The corresponding input building model from the 3DBAG is demonstrated in figure 3.13.

the 16 pt m^{-2} model with only 55 % more faces, neither in terms of visual nor measured similarity with an RMS minimum distance of 0.5 compared to 0.42 m).

The trend toward highly complex models increases significantly at higher error levels, in addition to the appearance of highly fractured roof geometries. The roofs of the building models at 32 and 64 pt m^{-2} density and with 60 % error level and above bear little resemblance of the input model's roof, but they succeed in maintaining a comparatively low RMS minimum distance. As discussed in section 3.2.3, this effect is likely due to the optimization for the RMS minimum distance as a single target metric, which revealed a trade-off between it and the model complexity. Therefore, a multi-objective optimization approach that includes the model complexity as a balancing factor into the maximization of the geometric similarity may deliver better results.

In summary, the findings from the visual model comparison correspond well to those from the aggregated results in section 3.3.2. The best reconstruction results are obtained between densities of 5.7 and 32 pt m^{-2} and low error levels. Higher densities yield no benefit in terms of the model accuracy but unnecessarily increase their complexity. At lower densities, the reconstructed models lack the fine geometric details but the resulting simplifications still allow for a good representation of the building. Higher error levels lead to even more strongly simplified building models at low densities: They feature mostly flat roofs, making them closer in appearance to models in LOD 1, but they provide a decent quality. At high point densities, however, high error levels result in building models whose roof geometries only distantly resemble those of the input model, and they can be considered largely unusable for practical applications. This may be due to constraints in the optimization approach, and an improved optimization scheme may allow significant improvements.

4 Discussion

4.1 Overview

The discussion is structured as follows: Section 4.2 details challenges and limitations that are specific to the methodology employed in this study and provides opportunities for improvement that arise from these. An interpretation of the findings including a discussion with respect to the research question and hypotheses can be found in section 4.3. General directions for future work that may build upon this study are given in section 4.4.

4.2 Technical and Methodological Limitations and Opportunities

4.2.1 Design of the Data Pipeline: Limitations of Scale and Variety

The data pipeline was designed and developed with the aim to produce results that closely resemble real data because this would ensure their relevance and applicability to applications in practice. For this reason, it was decided to choose an approach that is based on entire datasets of buildings on a contiguous area as opposed to a pipeline that processes and evaluates single buildings: This facilitates a realistic simulation of an ALS survey and results in a large-scale point cloud that serves as input for the reconstruction software in conjunction with the corresponding building footprints, which likewise corresponds to its application in practice. A single pipeline execution from data preparation to reconstruction evaluation thus yields a multi-dimensional dataset encompassing the point clouds' density and error levels, the reconstructed building models, their semantic and geometric properties, and the various evaluation metrics.

Test runs of the developed application with around one hundred buildings in a correspondingly limited area and point densities up to $64\,\mathrm{pt\,m^{-2}}$ performed well. Therefore, a dataset of around 4000 building models from the 3DBAG covering the central city of Utrecht was selected for the experiment initially. The ALS simulation, while time-consuming, was stable and finished as expected. However, the reconstruction software proved to be unstable when deployed at this scale: It would frequently stall during processing. To address this, the smaller dataset of around 1000 buildings that is presented here was selected. It was sufficiently stable to enable obtaining the desired results with the exception of the highest point density level, which again proved too demanding of the reconstruction algorithm. Nevertheless, it required significant attention during the execution with many restarts of the reconstruction process necessary at point densities above $32\,\mathrm{pt\,m^{-2}}$.

Another downside of the implemented solution is the fact that the variety of buildings in terms of their semantic and geometric properties depends on and is constrained by the selected input data region. Since the aim was to investigate a heterogeneous set of buildings, extensive deliberation of the choice of input data was required along with the additional evaluation of the entire 3DBAG dataset to identify suitable data tiles. Despite this effort, the evaluation of the reconstruction results as a function of the point cloud properties and semantic properties of the buildings such as their usage did not quite deliver the desired results. While the input dataset was among the most heterogeneous in the Netherlands with respect to the distribution of building use, the fact that this variety mainly stems from multiple classification of many buildings rendered the differentiation in usage groups meaningless.

Both these problems could be addressed by an implementation that processes single buildings: Though unrealistic, HELIOS++ could be used to perform ALS simulations at this scale, or an alternative ray-casting solution could be used to generate point clouds from individual building models. The reconstruction software could then be executed based on a single footprint and the corresponding, cropped part of the DTM.

On the one hand, this would increase performance significantly. On the other, individual buildings could be selected independent of their location, which would allow to control for geometric and semantic variety, thus permitting a meaningful evaluation differentiating these properties. Possibly, this would mean sacrificing a certain amount of realism of the point cloud data and the application of the reconstruction software, but in view of the results obtained with the current solution this would be tolerable considering the improved ability to increase the scale and control for variety.

4.2.2 Characteristics of the Reconstruction Method

Multiple executions of the reconstruction for individual scenarios revealed that the reconstruction algorithm is not entirely deterministic: Subsequent runs might fail or succeed without any changes made to the input data or the reconstruction parameters. In some cases, the output building models would contain faulty vertices, characterized by a single coordinate with an extreme value. These were identified by a check for outlier values in each of the coordinates, and a repetition of the identical reconstruction cycle commonly fixed this problem. Generally, however, consecutive and identical execution yield slight variations in the coordinates. It could not be determined if this is a consequence of floating point errors or if there are, in fact, steps in the algorithm that rely on random initialization or sampling approaches. Comparing the output of identical reconstruction runs demonstrated that these differences in the output building models also manifest in values of the evaluation metrics to a limited extent. This must be kept in mind when interpreting the results: The variations are sufficiently large to at least partially explain some of the apparently random deviations in the aggregated results, where relationships that would (within boundaries) be expected to be monotonic exhibit fluctuations.

One reason that led to the decision to use input building models that were generated with the same reconstruction method used in the experiment, and vice-versa, was the goal to avoid systematic deviations between the input and output models that could impede the evaluation by introducing a signal that is unrelated to the impact of the point cloud properties. (While a systematic bias could be accounted for easily in theory, it was expected that in practice it might have further implications or perhaps prove to have random components after all, which would complicate its handling.) Despite this precaution, some of the reconstructed building models differ from the corresponding input models in the elevation of their ground plane. This limits the utility of some of the evaluation metrics like the Hausdorff and RMS minimum distances or the IOU because in such cases the optimum value is not achievable even if the reconstruction of the roof geometry is without fault. For the definition of the ground plane elevation, the reconstruction software takes into account all points classified as ground within 4 m of the building and computes their 5th percentile. Two factors may therefore influence a deviation from the input model in this respect: First, the used DTM, although at a high resolution of 0.5 m raster cell size, can result in different measurements than the real ALS survey that collected the point data for the original reconstruction. Second, the addition of random positional error to the points for the purpose of simulating different accuracy levels also affects the vertical distribution of the ground points.

4.2.3 Evaluation

The evaluation is primarily based on a comparison of the complete input and output building models. As described in section 2.2.3, the reconstruction software Geoflow takes a 2.5D approach that identifies individual roof planes, subdivides the building footprint accordingly, and subsequently extrudes the footprint vertices to the determined elevations in the roof structure. For this reason, a more specific approach to evaluate the reconstruction quality could focus only on the roof geometry of the output building model, since the horizontal extent of the building, including the position of the walls, is largely determined by the building footprint, which is provided as input to the reconstruction. The fact that the roof planes are semantically labeled as such in the CityJSON output data would facilitate this level of specificity in the evaluation. Quality metrics computed in this way would be more sensitive to changes in the parts of the reconstruction that are actually influenced by the reconstruction algorithm as opposed to being determined by the input data. This would also help to address the problem stated in the previous section of systematic differences

in the elevation of the building models' ground plane, as it does not affect the reconstructed roof geometry. It must be noted, however, that the reconstruction pipeline is configured to detect deviations between the input footprint and point cloud where points located within the footprint are classified as ground, and to crop the footprint of the output 3D model correspondingly, which does exert an influence on an evaluation of the entire model.

As presented in section 3.3.1 about the reconstruction results, not all input building models were reconstructed successfully in some scenarios. While the proportion is greater than 99 % in most scenarios, it decreases to just above 90 % at low point densities and high error levels. Arguably, it would therefore have been appropriate to evaluate only those building models that were reconstructed successfully in every scenario to avoid impairing the comparability between the scenarios. Nevertheless, the evaluation was consciously based on all building models reconstructed in each scenario to utilize the entire available information content for this purpose, and possible but likely minor discrepancies between the individual results were accepted to this end.

4.2.4 Optimization of the Reconstruction Parameters

The rationale underlying the decision to adapt the reconstruction parameters for each individual point cloud scenario by means of an automated, Bayesian optimization algorithm is detailed in section 2.3.1. The optimization step proved to deliver surprisingly convincing results: Overall, the best parameter values identified by the probabilistically guided optimizer align well with how the parameters would be set manually based on a mechanistic understanding of their effects. Nevertheless, there are several shortcomings regarding both its implementation and its application that leave room for improvement.

Even before the first random iterations of the optimization, there is prior knowledge about the effects of the individual parameters that could be leveraged to improve the guidance toward promising locations in the parameter space. While the optimizer is capable of obtaining this knowledge and thus identifying these locations over many iterations, introducing it at the outset would help to improve efficiency or potentially, with equal time invested, effectiveness of the optimization. Because Bayesian optimization is a method utilized frequently to tune the hyperparameters for the training of artificial neural networks, its improvement is an active field of research. Approaches to integrate prior knowledge about the parameters have been proposed by Swersky et al. (2013), Swersky (2017), Souza et al. (2021), and Wang et al. (2024). A separate, promising strategy to improve the performance of Bayesian optimization is to additionally utilize information obtained about gradients of the objective function (Wu et al., 2017).

One of the efficiency strategies employed here (see section 2.3.2) was to avoid the necessity to optimize the parameters for each scenario from scratch by initializing the optimizer with the results from scenarios with the most similar point cloud properties. This is possible because it can be assumed that the contour of the objective function shifts only gradually with small changes in point density and positional error. In this way, the number of optimization iterations could be reduced significantly as the optimizer adapts to the altered input data. The pivotal question is, of course, to what extent this initialization with prior knowledge from adjacent scenarios possibly impedes the identification of an optimal set of parameter values. The results of a single complete, independent optimization of a scenario compared to those of with an preloaded optimizer state indicated that the standalone process, while more time-consuming, indeed leads to better results. While further exploration would be necessary to confirm this, better optimization results might therefore be achieved by optimizing the reconstruction parameters for each scenario independently, albeit at the cost of performance.

A significant obstacle to a better optimization outcome may have been the selected parameter ranges. They were defined after multiple initial exploratory runs on a test dataset, but at least two pieces of evidence point to the fact that they may be too narrow: First, in a significant number of cases the ideal parameter values identified by the optimizer are located at the range limits, indicating that an exploration of the parameter space beyond these limits may be worthwhile according to the acquisition function. Second and with a mechanistic understanding of the parameters, it appears reasonable to assume that, if the ideal minimum number of points to detect a plane is 11 at $2\,\mathrm{pt\,m^{-2}}$ density as found by the optimizer, a 32-fold density increase to $64\,\mathrm{pt\,m^{-2}}$ should most likely result in an ideal number of points higher than 16 or even

30 as found for high error levels. Especially in the latter case, a higher value for this parameter may have helped to avoid the complex and fractured roof geometries by including a larger number of points to detect the average location of a plane obscured in the noise. To some extent, however, this problem was, in fact, anticipated and accepted as a trade-off for two reasons: To prevent the optimizer from choosing values that seemed unpractical in the light of the limited documentation available about the reconstruction parameters of Geoflow, the reconstruction software, and to not unnecessarily slow down the optimization process by requiring the exploration of a larger parameter space.

A hindrance to the comparability between reconstruction results using optimized parameters from different scenarios is the multimodal nature of the objective function that was defined to evaluate to the RMS minimum distance as the single target metric: In test runs, various combinations of parameter values proved to deliver the same result in terms of this metric but may result in reconstructed building models with quite different properties. Fortunately, this behavior appears to have been mitigated by the strategy to initialize optimizers with the results from the most similar, adjacent scenarios. On the other hand, this results in the possibility that the successive optimization was guided along a single path of optima in the objective function where multiple others may be existing.

As described in section 3.2.3, it was found that the optimization for the RMS minimum distance as the single target metric introduced a bias toward highly complex building models. While these support the minimization of the geometric distance by facilitating a closer fit of the reconstructed model to the input model, in some cases this comes at the expense of the resemblance of real buildings' relative geometric simplicity: They consist mainly of large, planar surfaces as opposed to the fractured geometries with many small faces as reconstructed in the high-density, high-error scenarios. At high point densities and lower error levels, however, the increased model complexity does, at least visually, not compromise the integrity of the representation. Nonetheless, a multi-objective optimization capable of targeting the model's likeness to architectural geometries in addition to its mere geometric distance from the input model could result in a significant improvement.

4.3 Interpretation and Implications of the Results

4.3.1 Facets of Quality and Challenges in Their Quantification

As has already been alluded to several times previously, the criteria that determine the quality of a 3D building model may differ between properties of the input data and its acquisition method, limitations in the reconstruction process, requirements regarding the intended application, and other factors. For this reason, expressing the quality in terms of one or multiple quantities is not straightforward and the informative value depends on the context. A meaningful evaluation of this elusive property as the dependent variable in an experiment therefore posed a conundrum, which was addressed here by utilizing multiple metrics that each quantify different aspects of the term.

On the other hand, there is arguably a subtle semantic difference between the quality of 3D building reconstruction and the quality of 3D building models as its outcome. The former could be understood in a narrower sense as a measure of proximity between the modeled object and the model, whereas the latter must be interpreted with respect to their intended application. In this sense, the evaluation methods selected in this study mainly address the former task.

While the quality metrics were selected with deliberation, numerous studies have reflected on and explored further approaches for the quality assessment of 3D building models. Sargent et al. (2007) and Akca et al. (2010) emphasize the importance to consider the requirements by users and applications of the 3D building models in their creation and evaluation. To this end, Akca et al. (2010) present a method to evaluate the quality of 3D building models in comparison to a reference point cloud or a reference model of higher quality by using the least squares 3D surface matching (LS3D) method (Akca, 2010), which computes the Euclidean distance between the input and the reference data for co-registration: A matching and alignment step using translation, rotation, and scaling identifies the accuracy of the reference system. Any further mismatch is interpreted in terms of the positional accuracy of the building models. This positional

accuracy metric in terms of the Euclidean distance is similar in nature to the geometric distance metrics used in this study, but requires a one-to-one correspondence between locations in the compared datasets that is established in the co-registration step.

Dukai et al. (2021) present the methods used for the quality assessment of the building models of the Dutch 3D cadastre, which contains this information as metadata. They recognize the challenge of evaluating the quality of real-world 3D building models for which no ground-truth data is available. To overcome it, they evaluate properties of the input point cloud, the time difference between the building footprint delineation and the ALS survey, the goodness of fit between the reconstructed model and the point cloud, the validity of the model geometry, and the complexity of the roof in terms of the number of sloped and horizontal roof planes. While these methods are particularly useful in the absence of a ground truth, the validity of the reconstructed models is an aspect of data quality that would also be an interesting metric for the present experiment.

4.3.2 Research Question and Hypotheses

Overall, the observed impact of the point cloud properties on the reconstruction quality confirms the expectations: Generally speaking, an increasing density and accuracy of the investigated point clouds (the latter defined as a decrease in random positional error of the points) is associated with an improved quality of the reconstructed 3D building models in terms of their geometric distance from the input models. This was found to be valid across all error levels and up to a point density of about 16 pt m^{-2}, beyond which no further improvement in quality could be observed. Within these limitations, this confirms the first hypothesis.

The second hypothesis can be accepted as well: On average, each subsequent increment in point density and accuracy delivered a smaller quality improvement than the previous, confirming the expectation that a high reconstruction quality requires a disproportionately high point cloud quality. As before, this applies to all investigated error levels and up to 16 pt m^{-2} density. However, the hypothesis does not capture the entire nature of the found relationship: The observation that an increase in point density beyond a certain point results in a deterioration in reconstruction quality was not anticipated. Possible reasons for this are outlined in the following section.

There are multiple constraints to these findings. First, they apply within the scope of the quality metrics used in this project, with a focus on measures of geometric distance or similarity to the input models. As discussed in the previous section, the aspects that must be considered to assess the quality of a 3D building model depend on various factors. While this study provides multiple indicators of quality, it cannot be exhaustive in its assessment.

Second and related, the investigated relationship between point cloud and reconstruction quality becomes less clear if the model complexity is used as a benchmark. Because it was not controlled for in the optimization of the reconstruction parameters, it introduced a bias toward building models with a large number of faces. At low error levels, the effects are limited and may to some extend even be desired: Using the additional information available in denser point clouds to reconstruct finer details in building geometries will inevitably increase the models' complexity to some extent. Indeed, at low error levels the observed complexity increase with increasing point density is moderate. However, if both density and error levels are high, the resulting building models can exhibit an extreme number of faces and, as a consequence, have fractured geometries and provide little practical value.

Finally, the third hypothesis cannot be accepted or rejected with certainty within the scope of this project because it relates primarily to the practice of point cloud acquisition for 3D building modeling and, therefore, only indirectly to the relationship investigated here. However, the obtained insights may allow practitioners to make more well-informed decisions with respect to both the point cloud properties required to achieve a certain quality in the 3D reconstruction and the improvements they can expect from investing in denser and more accurate point data. By these means it may indeed be possible to translate the results presented here into a more configuration of the acquisition method that is more specific to the purpose of 3D building reconstruction.

4.3.3 Reconstruction Quality as a Function of Point Cloud Properties

A conspicuous feature of the found relationship between point cloud density and reconstruction quality is the stagnation and even reversal of the initially improving trend beyond a density of about 16 pt m^{-2} that can be observed in the Hausdorff and the RMS minimum distance. This contradicts the expectation that the higher information content of a denser point cloud should translate into a corresponding quality increase in the reconstructed building models. One question this raises is whether the very gradual deterioration is merely an artifact of the metric: Does it fail its purpose to accurately capture the quality of the building models reconstructed from such high densities? Yet its comparative value holds, and as such its indication is valid. A clearer answer to these questions can only be found by consulting additional metrics and even performing a qualitative assessment of the reconstructed models.

However, there are explanations of the observed pattern that relate to the methodology. A likely reason for the absence of a further quality improvement at higher point densities may lie in the degree to which the optimization method was capable of identifying suitable reconstruction parameter values. It was already discussed that the identified settings are in many cases, in particular at high densities, located at the limits of the explored parameter ranges, and that wider ranges may have enabled better results in these cases. Possibly, a larger number of optimization iterations or an independent optimization without reusing the results from adjacent scenarios would as well be beneficial to determine parameter values that are more conducive to a full utilization of the information content of dense point clouds. Even if these adjustments were to be implemented and found to deliver the expected results, however, there is no definite way to determine if the optimized parameters for the various scenarios are, in fact, optimized to the same degree. Generally, it is an indicator of a comparable optimization level if the optimization converges on a value, representing a local optimum in the objective function.

Lastly, there is the possibility that the design of the reconstruction algorithm itself is less suitable for highly dense point clouds, implying that no location in the parameter space could possibly improve the results further. For this reason, section 4.4.2 recommends the investigation of further reconstruction methods and their susceptibility to different point clouds properties.

4.4 Opportunities for Future Work

4.4.1 Point Cloud Properties

This study focuses on two central properties of point clouds that relate to their quality: Density and accuracy. However, other properties exist that may as well affect the outcome of 3D building reconstruction and could therefore be worthwhile to explore. For instance, point clouds from real laser scanning surveys commonly exhibit data gaps, which occur in areas that are occluded by building parts or other objects and are therefore not visible from the point of view of the scanner (Oude Elberink and Vosselman, 2011; Ostrowski et al., 2018). This can result in flawed building models, particularly in the case of data-driven reconstruction methods which do not make use of geometric primitives but rely solely on the information present in the point cloud (Jung and Sohn, 2017). Because of their practical relevance, the influence of such data gaps on the reconstructed building models would therefore be worthwhile to explore. In the framework of the processing pipeline used here, they could be simulated by adding objects that cause occlusions to the 3D scene or by locally removing points from the complete point clouds. Moreover, their impact in dependence of the type of reconstruction method could be investigated by using both model-driven and data-driven methods. An evaluation by groups of buildings with different sizes could yield insights as to how strongly smaller buildings are affected in particular, since they may be occluded more frequently.

Oude Elberink and Vosselman (2011) note that local variations in point density may also exert an influence on the reconstruction: The slope of surfaces, their distance from the scanner, and its reflective properties can cause variations in the point density. To quantify such variations, it would be important not to rely on the common, two-dimensional measure that effectively projects the points onto the horizontal plane. While it is commonly used and works well to describe average point densities across larger areas for comparison to other datasets, any local variations thus observed are merely a function of the slope of

the local surfaces: Vertical walls, for instance, will result in a local 2D density that is significantly higher, but does not reflect the actual point density distribution on the 3D surface. Instead, localized volumetric or areal density metrics would be suitable.

A further property that may affect the reconstruction quality is the classification of the points (Ostrowski et al., 2018). In the present simulation, the classification by the two classes *ground* and *building* is without fault, but in practical applications the algorithm may fail to classify all points correctly. Potentially, the effect of such a misclassification in boundary regions between adjacent classes would equate to a local thinning of the points used for reconstruction (or an addition of points where no building exists). This may turn out to have an impact similar to local density variations, but would have to be investigated separately to come to a certain conclusion.

4.4.2 Reconstruction Methods

The present study utilizes a reconstruction methods that makes light assumptions about building geometries (that they are piecewise planar and have vertical walls, i.e., are 2.5D) and therefore leans toward the data-driven end of the spectrum of reconstruction methods. It would be particularly interesting to conduct a similar investigation using reconstruction methods from different classes, as it can be expected that the impact of point cloud properties on the building models differs significantly between them. One interesting candidate could be City3D (Huang et al., 2022), which is based on PolyFit (Nan and Wonka, 2017) and openly available. It works by first identifying a set of candidate planes from the point cloud from which a subset is then selected such that the surfaces constituted by their intersection is a 2-manifold without boundary. For this reason, City3D generates an actual 3D representation of buildings (as opposed to the 2.5D models investigated here) that can model more complex building geometries, provided the information is present in the point cloud and correct. This method can therefore be characterized as yet more data-driven than Geoflow, keeping only the reasonable assumption of piecewise planarity.

The 3D building reconstruction method developed by Kada and McKinley (2009) prioritizes regular and valid roof geometries and therefore relies on a set of parameterized roof primitives for this purpose. It is an established and proven reconstruction algorithm that was used to create the 3D city models of Berlin and Cologne, Germany. As such, it could be a good candidate to serve in an investigation of how a more model-driven reconstruction approach is affected by different point cloud properties.

Finally, there is the emerging class of reconstruction methods that are based on deep neural networks (Buyukdemircioglu et al., 2022). This field of research is very dynamic and numerous experimental architectures have been proposed in the literature. A method by Kada (2022) uses continuous convolutions for the segmentation of point clouds to generate simple models with four roof segments. Liu et al. (2024) adapt the autoregressive, generative model PolyGen (Nash et al., 2020), which can process voxel and image data among other input types to generate 3D meshes, for the task of 3D building reconstruction from LiDAR point clouds. Their method generates vertices and identifies suitable combinations to form faces of building models. While they report good results based on various quality metrics, the reconstructed models are not generally watertight. Chen et al. (2024) employ a graph neural network to reconstruct watertight building models, whereas Chen et al. (2022) utilize deep learning to learn an implicit surface representation in an intermediate step of a method that likewise yields watertight models.

This is but a subset of the methods that have been explored. Some of the suggestions are closer in their approach to traditional reconstruction methods, like the one by Kada (2022), whereas others rely on capabilities that are inherent to deep neural networks. Future work could establish the robustness of such deep-learning-based reconstruction methods to differences in the quality of point clouds.

4.4.3 Data Sources

To explore the feasibility and benefit of the presented methodology, this study evaluated a single dataset of 3D building models from the city center of Utrecht. Although the location of the input dataset was selected with great care to maximize the variety and heterogeneity of the contained building models, this can naturally only succeed to a limited degree on such a small area. To be able to generalize the findings

from these project, it would be valuable to conduct similar experiments using datasets of building models from other cities, rural regions, and countries, to achieve insights with respect to a larger geometric and semantic diversity. Many municipalities and even entire countries today make their 3D city and building models available publicly (Wysocki et al., 2022), which enables further research in this direction.

5 Conclusions

The presented research sheds light on the quantitative relationship between two defining point cloud properties, density and accuracy, and the quality of point-cloud-based 3D building reconstruction. To this end, an analysis of the terms density, accuracy, and quality establishes their semantic connotations in a wider context and their meaning within the scope of this work. Based on this theoretical foundation, an end-to-end pipeline was developed to conduct an experiment that involves the simulation of airborne laser scanning based on existing 3D building models, a 3D building reconstruction step from the resulting point cloud using an established method, and a comprehensive quality evaluation of the reconstructed building models. It enables the analysis of an arbitrary number of scenarios, where each corresponds to a particular combination of point cloud properties.

If the reconstruction quality is assessed primarily by the geometric distance between the reconstructed and reference 3D building models, the results show a clear, directional relationship that serves to answer the research question and largely confirms the hypotheses. Both the point cloud density and accuracy measurably impact the quality of the reconstructed building models. Increasing either or both of them improves reconstruction quality, albeit with diminishing returns. At a high accuracy, even a low density can deliver accurate, if less detailed, building models. Moreover, improving the accuracy is always beneficial, whereas the point density exhibits an upper threshold beyond which further increases yield no additional benefit. This finding is discussed with respect to the limitations of the methodology: The observed relationship may be contingent on the selected approach to optimize the reconstruction parameters, the reconstruction method itself, and the metrics used for evaluation.

The analysis of further properties of the reconstructed models, namely model complexity and height, exhibit different patterns. In particular, the model complexity rises with increasing point density, which reflects the representation of more details in the building models but likely also a bias introduced by the reconstruction parameters. The effect of the point cloud accuracy on the model complexity depends on the density: At low densities, a decreasing accuracy leads to more simplified models, reflecting the challenge posed by the point clouds diminishing information content. At high densities, the effect is reversed and a decreasing accuracy yields disproportionately complex models, which can be interpreted in part as an erroneous interpretation of the random error as utilizable information, again pointing to the dependency on the reconstruction parameters.

The large and varied dataset of building models used for the experiment permits an analysis of the results that distinguishes between groups of buildings by their geometric and semantic properties. It is found that both the complexity and the volume of the input building model are predictive of the reconstruction quality. The reconstruction of input building models with a low complexity is less susceptible to a decreasing accuracy than that of more complex models. The share of an input model's roof that is flat also determines the reconstruction outcome, showing that buildings with entirely flat roofs are easiest to reconstruct accurately, followed by buildings whose roof parts are all sloped. Thus, mixed roofs pose a particular challenge. Any effect of semantic attributes on the reconstruction outcome must be exerted by means of a correlation with geometric properties. In fact, buildings constructed after 1950 are reconstructed at a higher quality than older buildings, possibly reflecting a shift in architectural styles toward simpler shapes.

The reconstruction method used in this study can be counted among the more data-driven approaches. To enable more generally applicable statements about the impact of point clouds properties on 3D reconstruction quality, it would be valuable to investigate reconstruction methods that implement different paradigms. In particular, future research could consider an even more data-driven method capable of representing horizontal variations in the building walls, a model-driven method that is based on a predefined set of parameterized roof shapes, and potentially a representative of the novel class of reconstruction methods based on deep learning, provided that it uses an approach that is inherently different from tra-

ditional reconstruction methods. With respect to the distinction between data-driven and model-driven paradigms, a third property of point clouds that is relevant in practice and whose impact on the reconstruction should therefore be understood well is the presence of data gaps, which therefore can also be a starting point for future research. Finally, while a large dataset of building models was used here, better insights can likely be obtained by extending the investigation using data from other regions or countries to better understand how geometric variety influences the relationship.

Despite these limitations, this study provides useful insights into the quantitative dependency of 3D building reconstruction quality on point cloud density and accuracy. These findings may help to inform decision-making related to the planning of laser scanning surveys for 3D building reconstruction when quality requirements are known and must be met. From an economic point of view, the findings are helpful to identify which point cloud quality is sufficient for certain applications of 3D building models. For instance, more frequent updates of 3D city models that aim to satisfy an increasing demand for up-to-date digital representations of urban environments may require higher survey frequencies. This calls for an enhanced efficiency in terms of achieved model quality relative to acquisition cost, and the findings presented here can facilitate such an optimization.

References

Akca, D. (2010). "Co-Registration of Surfaces by 3D Least Squares Matching". In: *Photogrammetric Engineering & Remote Sensing* 76.3, pp. 307–318. DOI: 10.14358/PERS.76.3.307.

Akca, D., M. Freeman, I. Sargent, and A. Gruen (2010). "Quality Assessment of 3D Building Data". In: *The Photogrammetric Record* 25.132, pp. 339–355. DOI: 10.1111/j.1477-9730.2010.00598.x.

Alva, P., F. Biljecki, and R. Stouffs (2022). "Use Cases for District-Scale Urban Digital Twins". In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-4-W4-2022, pp. 5–12. DOI: 10.5194/isprs-archives-XLVIII-4-W4-2022-5-2022.

Arroyo Ohori, K., H. Ledoux, and R. Peters (2024). *3D Modelling of the Built Environment*.

ASPRS Positional Accuracy Standards for Digital Geospatial Data, Edition 2, Version 2 (2024). Version 2. DOI: 10.14358/ASPRS.PAS.2024.

Baltsavias, E. P. (1999). "Airborne Laser Scanning: Basic Relations and Formulas". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 54.2, pp. 199–214. DOI: 10.1016/S0924-2716(99)00015-5.

Batty, M. (2018). "Digital Twins". In: *Environment and Planning B: Urban Analytics and City Science* 45.5, pp. 817–820. DOI: 10.1177/2399808318796416.

Batty, M., D. Chapman, S. Evans, M. Haklay, S. Kueppers, N. Shiode, A. Smith, and P. M. Torrens (2000). *Visualizing the City: Communicating Urban Design to Planners and Decision-Makers*. Working Paper 26. London, UK: Centre For Advanced Spatial Analysis, University College London, p. 38.

Beil, C. and T. H. Kolbe (2024). "Applications for Semantic 3D Streetspace Models and Their RequirementsA Review and Look at the Road Ahead". In: *ISPRS International Journal of Geo-Information* 13.10 (10), p. 363. DOI: 10.3390/ijgi13100363.

Bergstra, J. and Y. Bengio (2012). "Random Search for Hyper-Parameter Optimization". In: *Journal of Machine Learning Research* 13.10, pp. 281–305.

Biljecki, F. and Y. Dehbi (2019). "Raise the Roof: Towards Generating LOD2 Models Without Aerial Surveys Using Machine Learning". In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-4-W8, pp. 27–34. DOI: 10.5194/isprs-annals-IV-4-W8-27-2019.

Biljecki, F., H. Ledoux, and J. Stoter (2016). "An Improved LOD Specification for 3D Building Models". In: *Computers, Environment and Urban Systems* 59, pp. 25–37. DOI: 10.1016/j.compenvurbsys.2016.04.005.

Biljecki, F., J. Stoter, H. Ledoux, S. Zlatanova, and A. Çöltekin (2015). "Applications of 3D City Models: State of the Art Review". In: *ISPRS International Journal of Geo-Information* 4.4 (4), pp. 2842–2889. DOI: 10.3390/ijgi4042842.

Billen, R., A.-F. Cutting-Decelle, O. Marina, J.-P. de Almeida, M. Caglioni, G. Falquet, T. Leduc, C. Métral, G. Moreau, J. Perret, G. Rabin, R. San Jose, I. Yatskiv, and S. Zlatanova (2014). "3D City Models and Urban Information: Current Issues and Perspectives - European COST Action TU0801". In: *3D City Models and Urban Information: Current Issues and Perspectives - European COST Action TU0801*. 3D City Models and Urban Information: Current Issues and Perspectives - European COST Action TU0801. EDP Sciences, pp. 1–118. DOI: 10.1051/TU0801/201400001.

Bossche, J. V. den, K. Jordahl, M. Fleischmann, M. Richards, J. McBride, J. Wasserman, A. G. Badaracco, A. D. Snow, B. Ward, J. Tratner, J. Gerard, M. Perry, cjqf, G. A. Hjelle, M. Taves, E. ter Hoeven, M. Cochran, R. Bell, rraymondgh, M. Bartos, P. Roggemans, L. Culbertson, G. Caria, N. Y. Tan, N. Eubank, sangarshanan, J. Flavin, S. Rey, and J. Gardiner (2024). *Geopandas/Geopandas: V1.0.0*. Version v1.0.0. Zenodo. DOI: 10.5281/zenodo.12515477.

Butler, H., A. Bell, M. P. Gerlek, chambbj, P. Gadomski, C. Manning, M. oskot, B. Couwenberg, N. Barker, P. Ramsey, J. Dark, N. Chaulet, E. Rouault, K. Mann, C. Foster, G. Villemin, M. Rosen, Grigory, O.

Moore, S. Lewis, K. McKelvey, D. Brookes, K. Evers, M. Dobias, R. Coup, V. Vergara, xantares, Bram, and A. Yonas (2024). *PDAL: The Point Data Abstraction Library*. Version 2.7.1. Zenodo. DOI: 10.5281/zenodo.10884408.

Buyukdemircioglu, M., S. Kocaman, and M. Kada (2022). "Deep Learning for 3D Building Reconstruction: A Review". In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLIII-B2-2022, pp. 359–366. DOI: 10.5194/isprs-archives-XLIII-B2-2022-359-2022.

Chen, S., W. Zhang, N. H. Wong, and M. Ignatius (2020). "Combining CityGML Files and Data-Driven Models for Microclimate Simulations in a Tropical City". In: *Building and Environment* 185, p. 107314. DOI: 10.1016/j.buildenv.2020.107314.

Chen, Z., H. Ledoux, S. Khademi, and L. Nan (2022). "Reconstructing Compact Building Models from Point Clouds Using Deep Implicit Fields". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 194, pp. 58–73. DOI: 10.1016/j.isprsjprs.2022.09.017.

Chen, Z., Y. Shi, L. Nan, Z. Xiong, and X. X. Zhu (2024). "PolyGNN: Polyhedron-Based Graph Neural Network for 3D Building Reconstruction from Point Clouds". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 218, pp. 693–706. DOI: 10.1016/j.isprsjprs.2024.09.031.

Copyright and License of the 3DBAG (2024). URL: https://docs.3dbag.nl/en/copyright/ (visited on 10/05/2024).

CreativeCommons CC BY 4.0 (2024). Deed - Attribution 4.0 International - Creative Commons. URL: https://creativecommons.org/licenses/by/4.0/ (visited on 10/05/2024).

CreativeCommons CC BY-NC-ND 4.0 (2024). Deed - Attribution-NonCommercial-NoDerivatives 4.0 International - Creative Commons. URL: https://creativecommons.org/licenses/by-nc-nd/4.0/ (visited on 10/05/2024).

CreativeCommons CC0 1.0 (2024). Deed - CC0 1.0 Universal - Creative Commons. URL: https://creativecommons.org/publicdomain/zero/1.0/ (visited on 10/05/2024).

Dreek, U., M. Kosmatin Fras, A. Lisec, and D. Grigillo (2020). "The Impact of Point Cloud Density on Building Outline Extraction". In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLIII-B2-2020, pp. 407–413. DOI: 10.5194/isprs-archives-XLIII-B2-2020-407-2020.

Dukai, B., R. Peters, S. Vitalis, J. van Liempt, and J. Stoter (2021). "Quality Assessment of a Nationwide Data Set Containing Automatically Reconstructed 3D Building Models". In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVI-4-W4-2021, pp. 17–24. DOI: 10.5194/isprs-archives-XLVI-4-W4-2021-17-2021.

Faltermeier, F. L. (2024). *City-to-Scan-to-City*. URL: https://github.com/florifal/city-to-scan-to-city.

Ferré-Bigorra, J., M. Casals, and M. Gangolells (2022). "The Adoption of Urban Digital Twins". In: *Cities* 131, p. 103905. DOI: 10.1016/j.cities.2022.103905.

Fischer, A., T. H. Kolbe, F. Lang, A. B. Cremers, W. Förstner, L. Plümer, and V. Steinhage (1998). "Extracting Buildings from Aerial Images Using Hierarchical Aggregation in 2D and 3D". In: *Computer Vision and Image Understanding* 72.2, pp. 185–203. DOI: 10.1006/cviu.1998.0721.

Haala, N., C. Brenner, and K.-H. Anders (1998). "3D Urban GIS from Laser Altimeter and 2D Map Data". In: *International Archives of Photogrammetry and Remote Sensing* 32, pp. 339–346.

Haala, N. and M. Hahn (1995). "Data Fusion for the Detection and Reconstruction of Buildings". In: *Automatic Extraction of Man-Made Objects from Aerial and Space Images*. Ed. by A. Gruen, O. Kuebler, and P. Agouris. Basel: Birkhäuser, pp. 211–220. DOI: 10.1007/978-3-0348-9242-1_20.

Haala, N. and M. Kada (2010). "An Update on Automatic 3D Building Reconstruction". In: *ISPRS Journal of Photogrammetry and Remote Sensing*. ISPRS Centenary Celebration Issue 65.6, pp. 570–580. DOI: 10.1016/j.isprsjprs.2010.09.006.

Habib, A. (2017). "Accuracy, Quality Assurance, and Quality Control of Light Detection and Ranging Mapping". In: *Topographic Laser Ranging and Scanning: Principles and Processing*. Ed. by J. Shan and C. K. Toth. 2nd ed. CRC Press.

Hausdorff, F. (1914). *Grundzüge Der Mengenlehre*. Leipzig: Veit & Comp. 476 pp.

Heidemann, H. K. (2018). *Lidar Base Specifications (Ver. 1.3, February 2018)*. Book 11, Chapter B4. U.S. Geological Survey.

Huang, J., J. Stoter, R. Peters, and L. Nan (2022). "City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds". In: *Remote Sensing* 14.9 (9), p. 2254. DOI: 10.3390/rs14092254.

Hunter, J. D. (2007). "Matplotlib: A 2D Graphics Environment". In: *Computing in Science & Engineering* 9.3, pp. 90–95. DOI: 10.1109/MCSE.2007.55.

Jaccard, P. (1912). "The Distribution of the Flora in the Alpine Zone". In: *New Phytologist* 11.2, pp. 37–50. DOI: 10.1111/j.1469-8137.1912.tb05611.x.

Jung, J. and G. Sohn (2017). "Progressive Modeling of 3D Building Rooftops from Airborne LiDAR and Imagery". In: *Topographic Laser Ranging and Scanning: Principles and Processing*. Ed. by J. Shan and C. K. Toth. 2nd ed. CRC Press.

Kaartinen, H., J. Hyyppa, E. Gülch, G. Vosselman, H. Hyyppä, L. Matikainen, A. D. Hofmann, U. Mäder, A. Persson, U. Söderman, M. Elmqvist, A. Ruiz, M. Dragoja, D. Flamanc, G. Maillet, T. Kersten, J. Carl, R. Hau, E. Wild, L. Frederiksen, J. Holmgaard, and K. Vester (2005). "Accuracy of 3D City Models: EuroSDR Comparison". In: *Proceedings of the ISPRS Workshop Laser Scanning: September 12-14, Enschede, The Netherlands*. ISPRS Workshop Laser Scanning 2005. International Society for Photogrammetry and Remote Sensing (ISPRS), pp. 227–230.

Kada, M. (2022). "3D Reconstruction of Simple Buildings from Point Clouds Using Neural Networks with Continuous Convolutions (Convpoint)". In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-4-W4-2022, pp. 61–66. DOI: 10.5194/isprs-archives-XLVIII-4-W4-2022-61-2022.

Kada, M. and L. McKinley (2009). "3D Building Reconstruction from LiDAR Based on a Cell Decomposition Approach". In: *ISPRS Archives Volume XXXVIII-3/W4, 2009*. CMRT09. Ed. by U. Stilla, F. Rottensteiner, and N. Paparoditis. Paris, France.

Ketzler, B., V. Naserentin, F. Latino, C. Zangelidis, L. Thuvander, and A. Logg (2020). "Digital Twins for Cities: A State of the Art Review". In: *Built Environment* 46.4, pp. 547–573. DOI: 10.2148/benv.46.4.547.

Kim, M., J. Stoker, J. Irwin, J. Danielson, and S. Park (2022). "Absolute Accuracy Assessment of Lidar Point Cloud Using Amorphous Objects". In: *Remote Sensing* 14.19 (19), p. 4767. DOI: 10.3390/rs14194767.

Kolbe, T. H. (2009). "Representing and Exchanging 3D City Models with CityGML". In: *3D Geo-Information Sciences*. Ed. by J. Lee and S. Zlatanova. Berlin, Heidelberg: Springer, pp. 15–31. DOI: 10.1007/978-3-540-87395-2_2.

Kolbe, T. H. and A. Donaubauer (2021). "Semantic 3D City Modeling and BIM". In: *Urban Informatics*. Ed. by W. Shi, M. F. Goodchild, M. Batty, M.-P. Kwan, and A. Zhang. Singapore: Springer, pp. 609–636. DOI: 10.1007/978-981-15-8983-6_34.

Kolbe, T. H., T. Kutzner, C. S. Smyth, C. Nagel, C. Roensdorf, and C. Heazel (2021). *OGC City Geography Markup Language (CityGML) Part 1: Conceptual Model Standard*. Version 3.0.0. DOI: 10.62973/20-010.

Ledoux, H., K. Arroyo Ohori, K. Kumar, B. Dukai, A. Labetski, and S. Vitalis (2019). "CityJSON: A Compact and Easy-to-Use Encoding of the CityGML Data Model". In: *Open Geospatial Data, Software and Standards* 4.1, p. 4. DOI: 10.1186/s40965-019-0064-0.

Lei, B., P. Janssen, J. Stoter, and F. Biljecki (2023). "Challenges of Urban Digital Twins: A Systematic Review and a Delphi Expert Survey". In: *Automation in Construction* 147, p. 104716. DOI: 10.1016/j.autcon.2022.104716.

Liu, Y., A. Obukhov, J. D. Wegner, and K. Schindler (2024). "Point2Building: Reconstructing Buildings from Airborne LiDAR Point Clouds". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 215, pp. 351–368. DOI: 10.1016/j.isprsjprs.2024.07.012.

Maas, H.-G. and G. Vosselman (1999). “Two Algorithms for Extracting Building Models from Raw Laser Altimetry Data”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 54.2, pp. 153–163. DOI: 10.1016/S0924-2716(99)00004-0.

Macatulad, E. and F. Biljecki (2024). “Continuing from the Sendai Framework Midterm: Opportunities for Urban Digital Twins in Disaster Risk Management”. In: *International Journal of Disaster Risk Reduction* 102, p. 104310. DOI: 10.1016/j.ijdrr.2024.104310.

McKinney, W. (2010). “Data Structures for Statistical Computing in Python”. In: Python in Science Conference. Austin, Texas, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.

Muntoni, A. and P. Cignoni (2021). *PyMeshLab*. Zenodo. DOI: 10.5281/zenodo.4438750.

Nan, L. and P. Wonka (2017). “PolyFit: Polygonal Surface Reconstruction from Point Clouds”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2372–2380. DOI: 10.1109/ICCV.2017.258.

Nash, C., Y. Ganin, S. M. A. Eslami, and P. Battaglia (2020). “PolyGen: An Autoregressive Generative Model of 3D Meshes”. In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by H. D. III and A. Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 7220–7229.

Nogueira, F. (2024). *Bayesian Optimization: Open Source Constrained Global Optimization Tool for Python*.

Ostrowski, W., M. Pilarska, J. Charyton, and K. Bakua (2018). “Analysis of 3D Building Models Accuracy Based on the Airborne Laser Scanning Point Clouds”. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-2, pp. 797–804. DOI: 10.5194/isprs-archives-XLII-2-797-2018.

Oude Elberink, S. and G. Vosselman (2011). “Quality Analysis on 3D Building Models Reconstructed from Airborne Laser Scanning Data”. In: *ISPRS Journal of Photogrammetry and Remote Sensing*. Quality, Scale and Analysis Aspects of Urban City Models 66.2, pp. 157–165. DOI: 10.1016/j.isprsjprs.2010.09.009.

Paasch, J. M. and J. Paulsson (2023). “Trends in 3D Cadastre a Literature Survey”. In: *Land Use Policy* 131, p. 106716. DOI: 10.1016/j.landusepol.2023.106716.

Padsala, R., S. Valger, T. Santhanavanich, U. Voss, and V. Coors (2024). “Geo-Visualisation of Air-Pollutant Dispersion in Complex Urban Environments Using 3D City Models: Insights from High-Resolution Street Canyon Simulation - Concept and First Results”. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLVIII-4-W11-2024, pp. 89–95. DOI: 10.5194/isprs-archives-XLVIII-4-W11-2024-89-2024.

Pepe, M., D. Costantino, V. S. Alfio, A. G. Restuccia, and N. M. Papalino (2021). “Scan to BIM for the Digital Management and Representation in 3D GIS Environment of Cultural Heritage Site”. In: *Journal of Cultural Heritage* 50, pp. 115–125. DOI: 10.1016/j.culher.2021.05.006.

Peters, R. and B. Dukai (2024a). *Example-Config.Toml*. URL: https://github.com/geoflow3d/gfc-brecon/blob/main/stream/example-config.toml (visited on 10/04/2024).

– (2024b). *Reconstruct.Json*. URL: https://github.com/geoflow3d/gfc-brecon/blob/main/batch/reconstruct.json (visited on 10/04/2024).

Peters, R., B. Dukai, S. Vitalis, J. van Liempt, and J. Stoter (2022). “Automated 3D Reconstruction of LoD2 and LoD1 Models for All 10 Million Buildings of the Netherlands”. In: *Photogrammetric Engineering & Remote Sensing* 88.3, pp. 165–170. DOI: 10.14358/PERS.21-00032R2.

Rebolj, D., Z. Puko, N. . Babi, M. Bizjak, and D. Mongus (2017). “Point Cloud Quality Requirements for Scan-Vs-BIM Based Automated Construction Progress Monitoring”. In: *Automation in Construction* 84, pp. 323–334. DOI: 10.1016/j.autcon.2017.09.021.

Rottensteiner, F., G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf (2012). “The ISPRS Benchmark on Urban Object Classification and 3d Building Reconstruction”. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* I-3, pp. 293–298. DOI: 10.5194/isprsannals-I-3-293-2012.

Rottensteiner, F., G. Sohn, M. Gerke, J. D. Wegner, U. Breitkopf, and J. Jung (2014). “Results of the ISPRS Benchmark on Urban Object Detection and 3D Building Reconstruction”. In: *ISPRS Journal of*

Photogrammetry and Remote Sensing 93, pp. 256–271. DOI: 10.1016/j.isprsjprs.2013.10.004.

Sargent, I., J. Harding, and M. Freeman (2007). "Data Quality in 3D: Gauging Quality Measures from Users Requirements". In: *ISPRS Archives Volume XXXVI-2/C43, 2007*. 5th International Symposium on Spatial Data Quality 2007. Ed. by A. Stein. Enschede, The Netherlands.

Schrotter, G. and C. Hürzeler (2020). "The Digital Twin of the City of Zurich for Urban Planning". In: *PFG Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 88.1, pp. 99–112. DOI: 10.1007/s41064-020-00092-2.

Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas (2016). "Taking the Human Out of the Loop: A Review of Bayesian Optimization". In: *Proceedings of the IEEE* 104.1, pp. 148–175. DOI: 10.1109/JPROC.2015.2494218.

Souza, A., L. Nardi, L. B. Oliveira, K. Olukotun, M. Lindauer, and F. Hutter (2021). "Bayesian Optimization with a Prior for the Optimum". In: *Machine Learning and Knowledge Discovery in Databases. Research Track*. Ed. by N. Oliver, F. Pérez-Cruz, S. Kramer, J. Read, and J. A. Lozano. Cham: Springer International Publishing, pp. 265–296. DOI: 10.1007/978-3-030-86523-8_17.

Sullivan, C. B. and A. A. Kaszynski (2019). "PyVista: 3D Plotting and Mesh Analysis through a Streamlined Interface for the Visualization Toolkit (VTK)". In: *Journal of Open Source Software* 4.37, p. 1450. DOI: 10.21105/joss.01450.

Swersky, K., J. Snoek, and R. P. Adams (2013). "Multi-Task Bayesian Optimization". In: *Advances in Neural Information Processing Systems*. Ed. by C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger. Vol. 26. Curran Associates, Inc.

Swersky, K. J. (2017). "Improving Bayesian Optimization for Machine Learning Using Expert Priors". Thesis.

Tarsha-Kurdi, F., T. Landes, P. Grussenmeyer, and M. Koehl (2007). "Model-Driven and Data-Driven Approaches Using LiDAR Data: Analysis and Comparison". In: ISPRS Workshop, Photogrammetric Image Analysis (PIA07), p. 87.

The Pandas Development Team (2024). *Pandas-Dev/Pandas: Pandas*. Version v2.2.2. Zenodo. DOI: 10.5281/zenodo.10957263.

Tomko, M. and S. Winter (2019). "Beyond Digital Twins A Commentary". In: *Environment and Planning B: Urban Analytics and City Science* 46.2, pp. 395–399. DOI: 10.1177/2399808318816992.

Toth, C. K. and Z. Koppanyi (2017). "Strip Adjustment". In: *Topographic Laser Ranging and Scanning: Principles and Processing*. Ed. by J. Shan and C. K. Toth. 2nd ed. CRC Press.

Tzachor, A., S. Sabri, C. E. Richards, A. Rajabifard, and M. Acuto (2022). "Potential and Limitations of Digital Twins to Achieve the Sustainable Development Goals". In: *Nature Sustainability* 5.10, pp. 822–829. DOI: 10.1038/s41893-022-00923-7.

Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, . Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, and P. van Mulbregt (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17.3, pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

Vosselman, G. and H. Maas (2014). *Airborne and Terrestrial Laser Scanning*. Dunbeath: Whittles Publishing.

Wang, R., J. Peethambaran, and D. Chen (2018). "LiDAR Point Clouds to 3-D Urban Models: A Review". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11.2, pp. 606–627. DOI: 10.1109/JSTARS.2017.2781132.

Wang, X., Y. Jin, S. Schmitt, and M. Olhofer (2023). "Recent Advances in Bayesian Optimization". In: *ACM Comput. Surv.* 55 (13s), 287:1–287:36. DOI: 10.1145/3582078.

Wang, Z., G. E. Dahl, K. Swersky, C. Lee, Z. Nado, J. Gilmer, J. Snoek, and Z. Ghahramani (2024). "Pre-Trained Gaussian Processes for Bayesian Optimization". In: *Journal of Machine Learning Research* 25.212, pp. 1–83.

Waskom, M. L. (2021). “Seaborn: Statistical Data Visualization”. In: *Journal of Open Source Software* 6.60, p. 3021. DOI: 10.21105/joss.03021.

Willenborg, B., M. Sindram, and T. H. Kolbe (2018). “Applications of 3D City Models for a Better Understanding of the Built Environment”. In: *Trends in Spatial Analysis and Modelling: Decision-Support and Planning Strategies*. Ed. by M. Behnisch and G. Meinel. Cham: Springer International Publishing, pp. 167–191. DOI: 10.1007/978-3-319-52522-8_9.

Winiwarter, L., A. M. Esmorís Pena, H. Weiser, K. Anders, J. Martínez Sánchez, M. Searle, and B. Höfle (2022). “Virtual Laser Scanning with HELIOS++: A Novel Take on Ray Tracing-Based Simulation of Topographic Full-Waveform 3D Laser Scanning”. In: *Remote Sensing of Environment* 269, p. 112772. DOI: 10.1016/j.rse.2021.112772.

Wu, J., M. Poloczek, A. G. Wilson, and P. Frazier (2017). “Bayesian Optimization with Gradients”. In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc.

Wysocki, O., B. Schwab, and B. Willenborg (2022). *OloOcki/Awesome-Citygml: Release*. Version ver1.0. Technical University of Munich. DOI: 10.5281/ZENODO.5899096.

Xu, H., C. C. Wang, X. Shen, and S. Zlatanova (2021). “3D Tree Reconstruction in Support of Urban Microclimate Simulation: A Comprehensive Literature Review”. In: *Buildings* 11.9 (9), p. 417. DOI: 10.3390/buildings11090417.

Xu, Y. and U. Stilla (2021). “Toward Building and Civil Infrastructure Reconstruction From Point Clouds: A Review on Data and Key Techniques”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, pp. 2857–2885. DOI: 10.1109/JSTARS.2021.3060568.

Ye, X., J. Du, Y. Han, G. Newman, D. Retchless, L. Zou, Y. Ham, and Z. Cai (2023). “Developing Human-Centered Urban Digital Twins for Community Infrastructure Resilience: A Research Agenda”. In: *Journal of Planning Literature* 38.2, pp. 187–199. DOI: 10.1177/08854122221137861.

Yutzler, J. and P. Daisey (2024). *OGC GeoPackage Encoding Standard*. Version 1.4.0.

Appendices

A Figures

A.1 UML Class Diagrams of the Data Pipeline

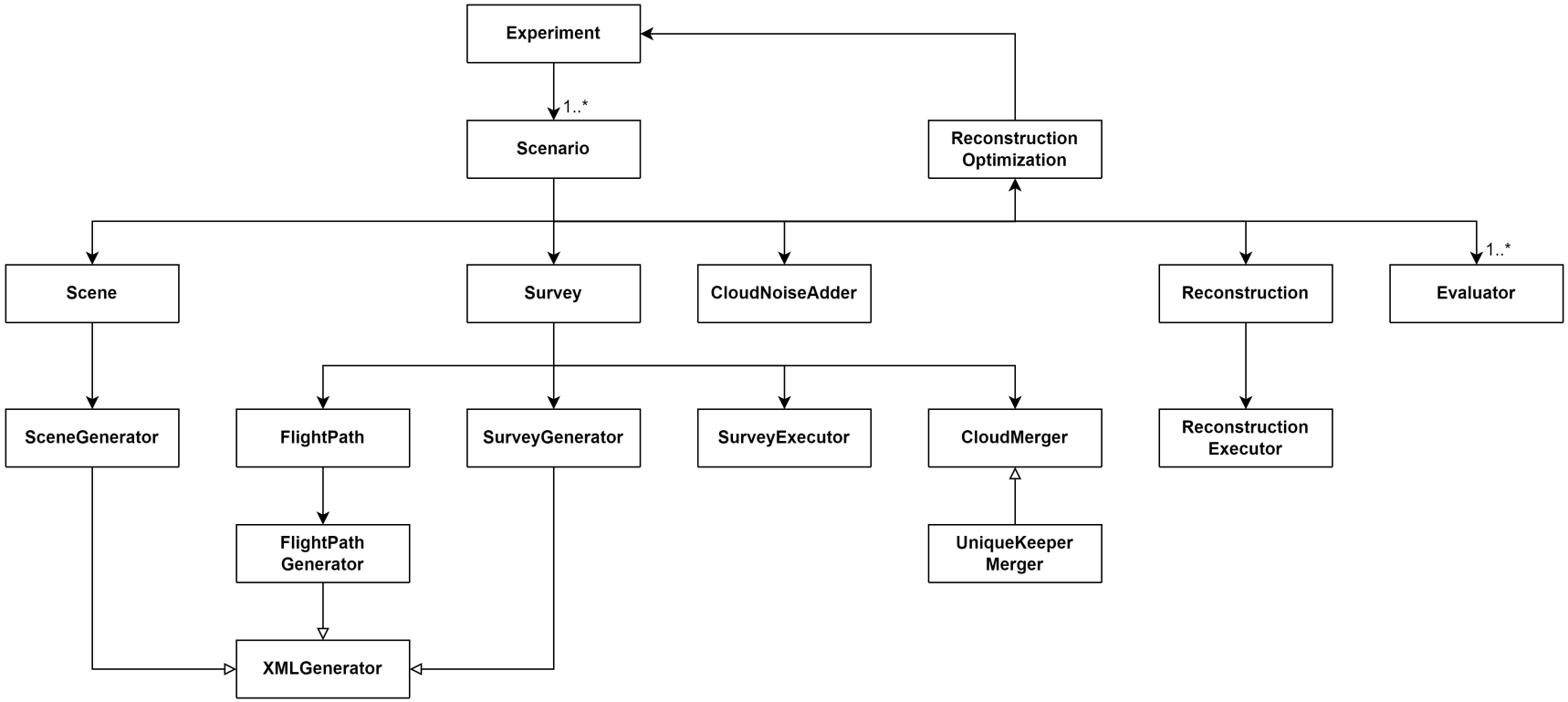


Figure A.1 Simplified UML diagram of the data pipeline classes and their relationships. Unless specified otherwise, multiplicities are 1. A UML diagram of the classes relating to the Evaluator class is provided in figure A.2.

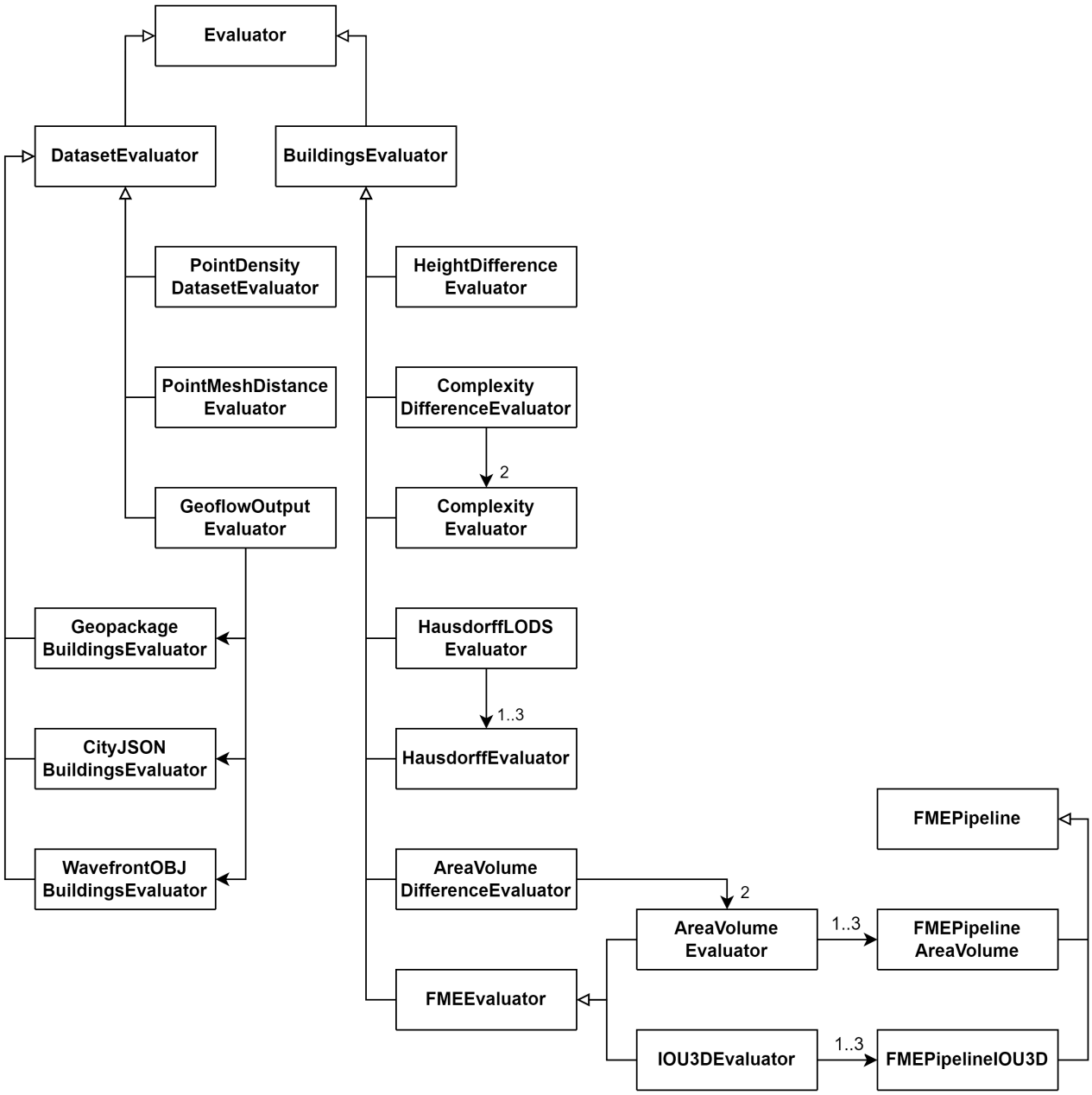


Figure A.2 Simplified UML diagram of the classes developed for the evaluation of the experiment and their relationships. Unless specified otherwise, multiplicities are 1. The class Evaluator links this diagram to the one in figure A.1.

A.2 Reconstruction Parameters

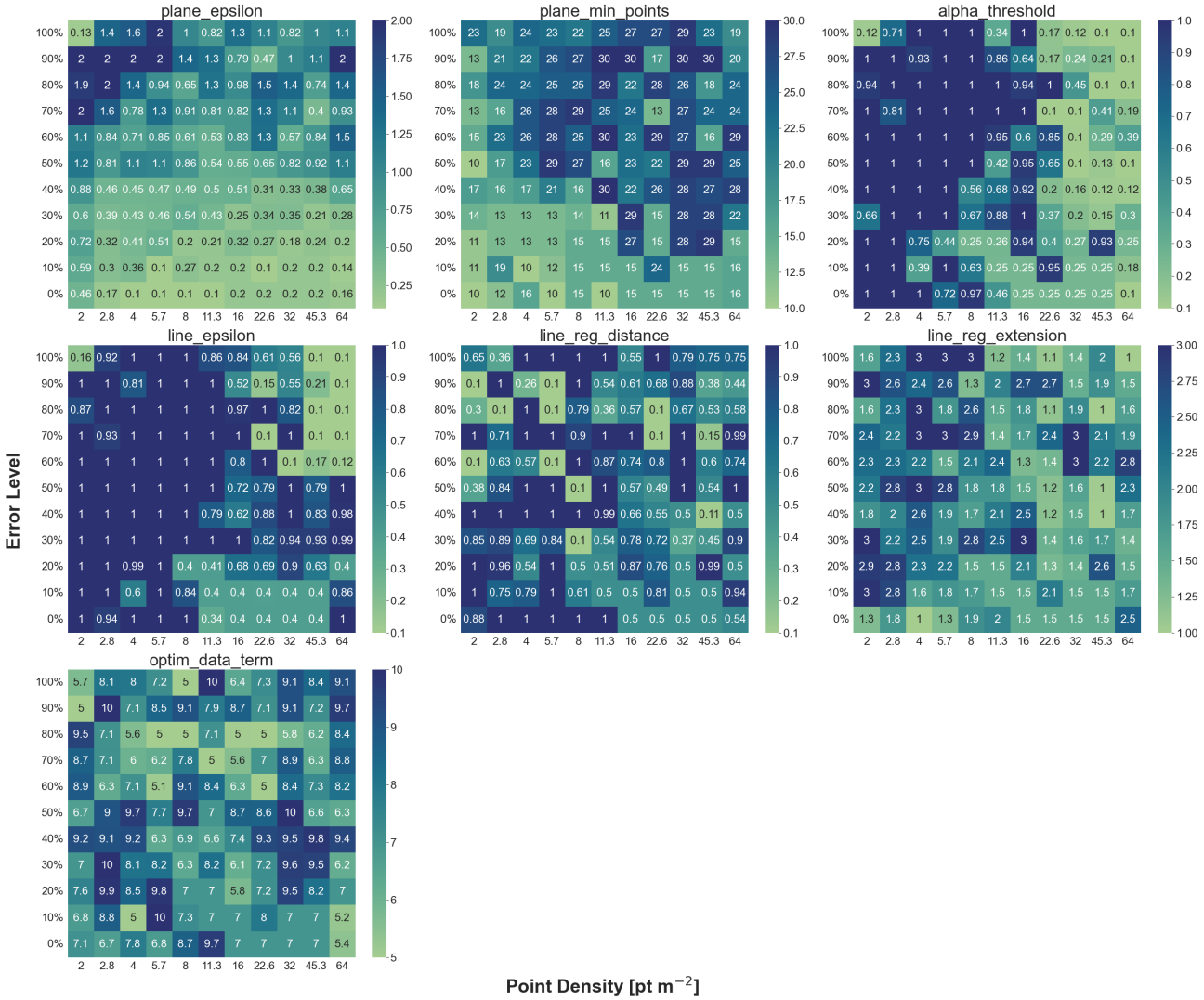


Figure A.3 The distribution of the seven reconstruction parameters' values from the optimization that were selected for the experiment, across the point density and error levels. For each scenario, they are taken from the optimization iteration that delivered the lowest mean model complexity among all iterations that achieved a RMS minimum distance within 10 % of the best one.