remote sensing

MDPI

Article

Evaluating the Quality of Semantic Segmented 3D Point Clouds

Eike Barnefske *, Harald Sternberg

Department of Hydrography and Geodesy, HafenCity University Hamburg, Henning-Voscherau-Platz 1, 20457 Hamburg, Germany; harald.sternberg@hcu-hamburg.de
* Correspondence: eike.barnefske@hcu-hamburg.de

Abstract: Recently, 3D point clouds have become a quasi-standard for digitization. Point cloud processing remains a challenge due to the complex and unstructured nature of point clouds. Currently, most automatic point cloud segmentation methods are data-based and gain knowledge from manually segmented ground truth (GT) point clouds. The creation of GT point clouds by capturing data with an optical sensor and then performing a manual or semi-automatic segmentation is a less studied research field. Usually, GT point clouds are semantically segmented only once and considered to be free of semantic errors. In this work, it is shown that this assumption has no overall validity if the reality is to be represented by a semantic point cloud. Our quality model has been developed to describe and evaluate semantic GT point clouds and their manual creation processes. It is applied on our dataset and publicly available point cloud datasets. Furthermore, we believe that this quality model contributes to the objective evaluation and comparability of data-based segmentation algorithms.

Keywords: 3D point cloud; quality model; annotation tools; datasets; evaluation metric; evaluation parameter

Citation: Barnefske, E.; Sternberg, H. Evaluating the Quality of Semantic Segmented 3D Point Clouds. *Remote Sens.* **2022**, *14*, 446. https://doi.org/10.3390/rs14030446

Academic Editor: Sander Oude Elberink

Received: 20 December 2021
Accepted: 13 January 2022
Published: 18 January 2022

1. Introduction

A major research topic in geodesy is to digitize activities in construction [1–3], in building maintenance [4,5] and in navigation [6,7]. For the digitization of these tasks, digital building parts and furnishing objects must be formed and processed. Digital models of real-world buildings (digital twins) are needed to make complex and large semantic data interpretable for humans and machines [8]. The creation of digital twins is often based on 3D point clouds, which are efficiently captured with depth imaging cameras or light imaging, detection and ranging (LIDAR) systems. The 3D point cloud without any semantic features can already be considered a model, since humans can use their knowledge to interpret semantic point groups as single objects. These semantic point groups are, e.g., the objects and scanning artifacts, as shown in Figure 1.

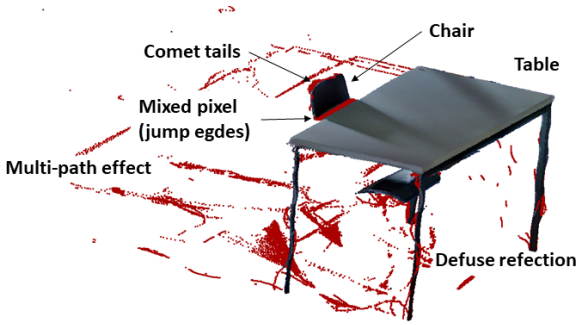

Figure 1. Examples of objects (chair and table) and scanning artifacts in a point cloud. Common scanning artifacts are: comet tails, mixed pixels on edges (jump edges), multi-path effects and defused reflections.

For the digital processing of point clouds, semantic information has to be given to the point cloud to form semantic segments. The initial semantic segmentation is always performed by humans. For this purpose, different tools can be used to form segments as efficiently, reliably, precisely and correctly as possible and to assign the correct semantic label. The efficiency, reliability, precision and correctness of semantic segmentation are characteristics that describe the quality of a semantic point cloud. These characteristics build the quality model, which describes how well the creation of the semantic point cloud works. Evaluation metrics now become parameters of the quality model, which describe the point cloud characteristics. A comparison of different segmentations is possible with the quality parameter. Method comparisons are common in automatic semantic segmentation [9–12], which typically uses machine learning (ML) and artificial intelligence (AI). For method comparisons, point cloud benchmarks are used [13,14]. Semantic point cloud benchmarks are point clouds for which a semantic ground truth (GT) is given. It is assumed that the GT point clouds are free of semantic and geometric errors. However, unfortunately, in most cases, a complete evaluation of the manually or semi-automatically created semantic point cloud benchmarks is not performed. The characteristics of a semantic point cloud that can be evaluated vary strongly among the published point clouds. In some works, the semantic accuracy of a point cloud is evaluated completely [13] or by spot checks [14,15]. Other works evaluate only the completeness and correctness of a building model [16]. Even if some characteristics of the point cloud can be evaluated, then a comparison of the evaluation metric is often not possible, since no uniform metrics are defined. For example, intersection over union (IoU), F1-score, overall accuracy, recall, precision and many others are used to validate the accuracy. The variety problem of the evaluation metric for the case of object detection in images is well known and a tool to translate the evaluation metrics for compression was developed [17].

To the best of our knowledge, a holistic quality model in which availability, integrity and accuracy are represented does not exist for semantic point clouds. Such a quality model has the potential to make the investigation of existing and upcoming GT point cloud datasets comparable. Deviation from reality, the availability of information and applicability to a certain purpose can be determined with our quality model for indoor point clouds.

Fundamental for the development of the quality model is the definition of the semantic segmentation, as well as its separation into detection and classification (Section 2.1). The capture methods of 3D point clouds for indoor applications (Section 2.2), the existing point cloud datasets (Section 2.3), as well as the tools for manual and semi-automatic semantic segmentations (Section 2.4) determine the characteristics needed in the quality model. The development of the quality model is derived from a process description (Section 3.1), a class definition (Section 3.2) and a data model (Section 3.3). The quality characteristics and parameters are defined and discussed in Section 3.4. The descriptive and evaluative use of the quality model is presented and discussed based on different point clouds in Sections 4.1 and 4.2. Finally, Section 5 summarizes the main conclusions and gives an outlook for further development and possible use of the quality model.

2. State of the Art

The surfaces of real objects are often represented as 3D point clouds after digitization. These 3D point clouds are an unsorted list of coordinates with additional (spectral) information. This representation is particularly well suited for measuring systems that use high-frequency scanning of object surfaces. Very efficient storage of single points or point groups (lines or arrays) is thus possible. This has caused the point cloud to become a quasi-standard for 3D object representations. The point cloud represents very efficiently, accurately and with a high resolution the geometry of scenes and objects. Unfortunately, with point clouds, the separation of individual objects is not possible right away. Thus, it is a necessary next processing step to derive information or models from point clouds.

Current research on the separation of point clouds is mainly applied to autonomous operating systems, building modeling and computer vision (CV) tasks. Autonomous operating systems include autonomously driving cars, where information for obstacle avoidance, route planning and sign recognition has to be generated from the 3D point clouds [18,19]. CV and building modeling aim to enrich the point cloud with semantic information. The enriched point clouds are the basis for decision making and the creation of semantic models. If the point clouds represent complex scenes in which individual objects appear several times, then instancing is often the goal. Applications include the modeling of digital twins or the creation of city models, as well as the direct creation of simple building models based on point clouds and prior knowledge [20–22].

Different types of acquisition systems, segmentation tools and semantic point cloud datasets are available, forming the basis for the development of automatic point cloud separation methods. The application of these sets the quality of a semantic point cloud. A large amount of semantic training and benchmark point clouds are available.

2.1. Classification, Object Detection and Segmentation

The definition of classification, object detection and segmentation is not clear in the literature, and these terms vary by research and application field. Different terms are used for the same separation task, or the meaning of the terms may be ambiguous. Some reviews [23,24] distinguish between classification, object detection and segmentation. Other researchers [25] use segmentation as an all-encompassing term for various categorization methods. To avoid misunderstandings, classification, object detection as well as semantic and instance segmentation are briefly defined below for this work.

Classification: Classification is the assignment of a class feature (label) to one object. This can be a single point, a point cloud, a segment of a point cloud or another geometry type. Usually, semantic labels or IDs are assigned. The classification in the following is understood as the assignment of one semantic label to one point cloud segment.

Object detection: In object detection, specific objects are defined based on geometric or spectral features in the point clouds. The individual object and not the entire point cloud is of interest, so that large parts of the point cloud are not evaluated in detail. Several objects in a point cloud can be detected and a unique identifier is obtained. Object detection is often used in conjunction with tracking objects in applications with multiple sub-point clouds. The objects are usually roughly described in terms of geometric size, position and orientation using bounding boxes. In other cases, it is not the objects as a whole that are of interest, but only certain surfaces or shapes [26]. These are searched for in the point clouds (shape detection).

Semantic segmentation: The semantic segmentation has the goal of extending the features of the points by semantic labels. Semantic labels are semantic classes that usually describe real-world objects. The difference for the classification is that the segments are formed in this process step and a label is set for all points of the segment. A semantic segment can consist of several geometrically independent segments. For example, a point can belong to the class *table*; complementarily, it can belong to the subclass *table leg*. Moreover, the results of the classification of each point can form a new segment.

Instance segmentation: An instance segment describes the geometric shape of one object. Instances in a point cloud can be distinguished by a unique identifier. An instance is usually enriched with semantic information. Points of the same semantic segment describe different objects. For example, if two tables are in one point cloud, then both carry the same semantic label. In order to distinguish the tables, instances must be created. Each table is an instance, which usually consists of a geometrically connected point cloud segment.

The creation of a digital twin goes beyond this idea. For modeling a digital twin, new parametrized objects have to be formed that describe the point cloud content by generalizations such as a simple geometry.

2.2. Captured and Synthetic Point Clouds

Almost any semantic 3D point cloud is derived from a synthetic surface model or is captured by contactless sensors. An overview of the methods is given in Figure 2.

Synthetic 3D point clouds are mostly generated from large collections of online model databases, such as [27]. These point clouds are generated efficiently by transforming a surface model into a regular or random point cloud. These points lie on the surface of the previous model or have synthetic noise added. Synthetic 3D point clouds usually represent only a single object or a small group of objects. Usually, they are used for algorithm development or prototype testing [28,29].

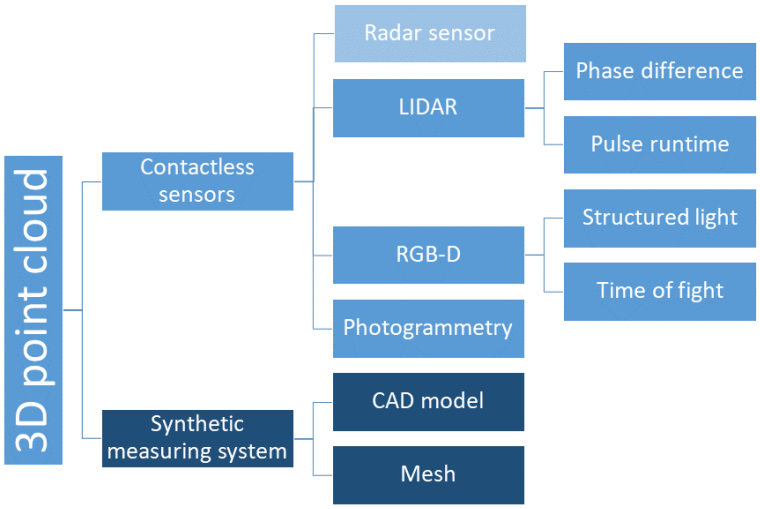

Figure 2. Capturing systems and basic data for the creation of 3D point clouds.

Any acquisition technique for capturing reality has a certain resolution, precision and correctness, which can be found in the resulting point cloud. These point cloud characteristics often depend on the surface of the object, the acquisition distance, the environmental conditions and the measurement sensors.

Optical sensors are the most widely used method for mapping reality. Optical sensors use light of different spectral bands to create a 3D point cloud of real environments with photogrammetric methods, as described in [30]. In particular, depth imaging cameras and LIDAR systems have been widely used in the last decade to create point cloud datasets [23–25]. The reasons are user friendliness, mainly moderate acquisition and evaluation costs [31,32] and the efficient capturing of larger areas. In addition to optical sensors, radar is sometimes used to create point clouds [33,34].

Depth imaging cameras consist of one or more cameras for different spectral ranges and an active emitter. Different principles for determining the image depths are used. For example, the *Matterport Pro 3D Camera* and the *Microsoft Kinect V1* use structured light (SL) [35] and the *Microsoft Kinect V2* uses the time of flight (ToF) method [36].

With the SL cameras, a monochrome near-infrared (NIR) image is captured in addition to a true-color image (red, green, blue (RGB)). The scene to be captured is illuminated by a projector with a known NIR pattern. The pattern consists of various bright and dark dots that are distributed in a non-correlating manner. The projected pattern is distorted by the geometry of the object. The depth is determined in several steps and for each pixel. First, the horizontal displacement of the dot pattern is determined based on the object distance. Based on the distortion, the depth of the respective pixel is then calculated in the next step using the equation for stereo triangulation [37]. For this purpose, the distortion in the unit of pixels, the base length (distance projector–camera) and the focal length in pixels are used. For each pixel, the distortion is determined using a local, e.g., 9×9 pixel area, which is compared with a set of reference images for different depths. The comparison is performed using cross-correlation. An interpolation is performed between the highest

correlation values to increase up to sub-pixel resolution [36,38]. For further information on the SL method using the *Microsoft Kinect V1* as an example, the reader is referred to [39].

Investigations of the *Microsoft Kinect V1* show the precision expressed by the standard deviation (SD) of 1 mm at 800 mm distance and of 11 mm at a distance of 3000 mm [32]. According to [31], the correctness (offset to the target geometry) is up to 40 mm for a captured distance of 1600 mm (within a typical working range of 400 to 4000 mm). Effects such as flying pixels (erroneous point measurement in a gap), color-dependent accuracy changes and multipath overlaps at edges do not or only occur at a very low level [31]. Moreover, for the *Matterport Pro 3D Camera*, which was used for online available training datasets by [40,41], the correctness, precision and resolution have been investigated in different studies. Here, a distance-dependent correctness of up to 80 mm for the furthest capturing distance was also determined. After a scaling factor is eliminated, a precision of better than 10 mm SD can be determined for the entire working range [35]. A LIDAR point cloud was used as a reference for the mentioned study. The resolution of the *Matterport Pro 3D Camera* is 5 (horizontal) and 10 (vertical) points per degree [42].

The ToF technique is based on measuring the travel time of a signal from an emitter to reflect at an object's surface and back to a receiver [30]. Pulse modulation (PM) and continuous-wave (CW) amplitude modulation are the most common ToF methods. In most depth imaging cameras, such as the *Microsoft Kinect V2*, CW amplitude modulation is used. In CW amplitude modulation, the object to be captured is continuously illuminated with NIR light, whose amplitude changes periodically. Because the signal needs a certain time between sensor and object, a phase shift occurs between the transmitted and received signal. This phase shift is proportional to the signal propagation time. If this time is multiplied with the known speed of light, the double distance between object and sensor system can be determined. The phase difference is determined for several modulated frequencies by correlating the received signal with the emitted reference frequencies. As long as the maximum distance is smaller than 2π of the frequency, a distance can be determined as unique [36].

The precision of the *Microsoft Kinect V2*, as with the *Microsoft Kinect V1*, depends on the acquisition distance and varies between 1 and 3 mm SD for the typical working range of 800 to 3000 mm [31,32]. Recent depth imaging cameras, such as the *Microsoft Azure Kinect*, have a precision of less than 1 mm for the same working range (static recording). Ref. [31] observed a constant offset of -18 mm for the whole working range of the *Microsoft Kinect V2*. Systematic erroneous measurements, such as flying pixels, color-dependent accuracy changes of up to 4 mm, multipath-effects at edges of up to 30 mm and a high dependence of distance measurements on temperature changes, are the disadvantages of this measurement principle [31,36,43]. These effects can be considered or eliminated in a later semantic segmentation.

LIDAR systems are used for static and kinematic recordings of scenes. LIDAR systems emit a laser beam, which is projected onto a rotating mirror. Through the rotation, the beam is shifted by a certain increment. For each increment, the vertical and horizontal directions as well as the distance to the surface are registered. Together with the intensity value, and eventually with further spectral values, the 3D point cloud is created. For the distance measurements, there is the phase difference (PD) method, which can be used to realize a higher measuring frequency, and the PM method, which is less object surface-sensitive [30]. PM LIDAR systems are preferred for kinematic scanning on mobile platforms. Kinematic laser scanning usually involves measuring individual profiles, which are assembled as an entire point cloud using navigation data or algorithms, as in [44]. Mobile LIDAR systems are mainly used for outdoor applications and on robots. Medium-range LIDAR systems such as *Velodyne HDL-64E* are often used for creating datasets in research projects with a precision of 20 mm [45]. High-end mobile mapping systems (MMS), such as the *Riegl VMY-1* [46], allow the surveying of large-scale areas with a point accuracy of 15 mm at 50 m distance and a precision of 10 mm. MMS such as the *Nav Vis M6* are used in many studies [47].

The current state of the technology for indoor surveys includes terrestrial LIDAR systems (TLS), such as the *Leica RTC 360*, *Z+F-Imager 5016* or *Faro Fokus X 3D 330*. These systems predominantly use the PD method and are used for distances shorter than 100 m. Laboratory and field investigations show that, with these measuring systems, 3D point clouds with precision of less than 1 mm and correctness of less than 2 mm in the near field of up to 20 m can be reached [48]. However, these values refer to optimal study circumstances such as matt or homogeneous surfaces. In practice, it has been shown for all LIDAR systems that the accuracy of the point clouds varies and scanning artifacts occur. Typical scanning artifacts are comet tails, mixed pixels on edges and multi-path effects on highly reflective surfaces, as shown in Figure 1. Other influencing variables, such as the measurement object, the setup and the environment, as well as the condition of the measurement systems [49], must be taken into account for the determination of the quality of a captured point cloud [50–52]. The resolution, the approximated accuracy, the acquisition method and the working range are crucial parameters that must be known or estimated for the later semantic segmentation of a point cloud.

2.3. 3D Point Cloud Datasets

In various reviews [23–25] and in web databases (e.g., https://paperswithcode.com/datasets accessed on 30 November 2021 and https://www.semanticscholar.org/ on 30 November 2021) on point cloud datasets and methods for point cloud processing, an overview of more than 100 publicly available point cloud datasets is given. These contributions summarize information on application areas, applied sensors, environmental circumstances or file formats. The main goal of these publications is to provide benchmarks for arithmetic evaluations. A semantic segmentation is not available for all existing datasets. A selection of semantic 3D point clouds is examined in more detail. The focus will be on the initial human segmentation and its evaluation. Not all datasets could be documented in the same level of detail.

The datasets in Table 1 were derived from synthetic surface models. All show one object of one known class. In some datasets, the object models are subdivided so that they can be used for semantic and instance segmentation. Since the point clouds are derived from synthetic models, the geometry can be considered free of scanning artifacts. However, errors can still occur during annotation and alignment.

Table 1. Synthetic datasets with year of publication, data source, separation method (classification (Cls), semantic segmentation (SSeg) and instance segmentation (ISeg)), number of models, number of classes and environment.

Dataset	Year	Data Source	Separation Method	No. of Models	No. of Classes	Environment
ShapeNet [27]	2015	Trimble 3D Wareh., Yobi3D	Cls, SSeg	$> 220,000$	3135	In-/Outdoor
ModelNet [53]	2015	Trimble 3D Wareh., Yobi3D	Cls, ISeg	151,128	660	In-/Outdoor
Shape2Motion [26]	2019	ShapeNet, Trimble 3D Wareh.	ISeg Cls, SSeg	2440	45	In-/Outdoor

An evaluation metric for classifications is introduced by the *ShapeNet* dataset, which describes how accurate or unique a classification is. Human annotators classify a semantic model until the classification accuracy varies by less than 2% [27]. The *ModelNet* dataset consists of 3D CAD models taken from web databases. The annotation is performed using *Amerzone Mechanical Turk* (AMT). The annotators classify different models using a web-based tool. For this, a model and a label are proposed. The annotators improve the correctness of a label for a displayed model by yes-or-no questions. An evaluation is conducted by the dataset designers for the ten most popular categories [53]. In the

Shape2Motion dataset, a semantic segmentation of movable parts, such as wheels or car doors, and their properties is performed. An evaluation of the classification is carried out by simulating the motion directly after the segmentation and classification [26].

Complex point cloud simulation tools, such as the *HELIOS++* [54] or *Gazebo* together with the *Robotics Operation System* [55], have reached a high level of development. These tools can be used to create point clouds from surface and CAD models that contain the characteristics of specific sensors and system configurations.

Indoor datasets are commonly captured with depth imaging cameras. Some of the most popular datasets are summarized in Table 2. For a large number of datasets, depth imaging cameras are used in combination with an initial measurement unit (IMU). Together with the poses from the IMU and the images, a Simultaneous Localization and Mapping (SLAM) procedure is used to compute a multi-dimensional representation of the captured scene. The semantic annotation occurs either in images, videos, meshes or in 3D point clouds.

Table 2. Indoor datasets recorded by depth cameras with year of publication, sensor, sensor method, separation method (classification (Cls), object detection (ObjD) and semantic segmentation (SSeg)), surface area and number of classes.

Dataset	Year	Sensor	Sensor Method	Separation Method	Surface area Points	No. of Classes
SceneNN [56]	2016	Kinect v2	ToF	Cls, SSeg	7078 m² *1,450,748*	19
S3DIS [40]	2016	Matterport	SL	Cls, SSeg	6020 m²	12
ScanNet [57]	2017	Occipial (iPad)	SL	ObjD, SSeg	78,595 m²	17
Matterport3D [41]	2017	Matterport	SL	Cls, SSeg	219,399 m²	40
ScanObjectNN [58]	2019	SceneNN, ScanNet	ToF, SL	Cls, SSeg	*2.971.648*	15

The *Stanford Large-Scale 3D Indoor Spaces* (S3DIS) dataset is semantically segmented as a 3D point cloud using the software *Cloud Compare* (CC) [59]. For the *SceneNN*, *ScanNet* and *Matterport3D* datasets, a mesh is the segmentation base. All annotations are performed with custom tools. The *SceneNN* dataset is first automatically segmented coarsely and then finely. The graph-based segmentation algorithm of [60] is adapted and the segmentation is afterwards improved by the operator by separating, merging and re-forming the segments. The semantic annotation is performed by users attaching labels to the segments [56,61]. The semantic segmentation of the *ScanNet* dataset is performed by automatic pre-segmentation and a subsequent fine segmentation with classification using tools on AMT. In addition to semantic segmentation with meshes, CAD models are fitted into a mesh and are available as a different data format [57]. The *Matterport3D* dataset is semantically segmented in two stages and verified by ten experts. In the first stage, floor plans are derived using planes projected onto the mesh. In the second stage, the meshes of individual rooms resp. regions are segmented according to classes and instances using *ScanNet's* tool [41]. For the *ScanObjectNN* dataset, the *SceneNN* and *ScanNet* meshes are the basis. A selection from this dataset is used and improved. Segments are rebuilt and categories are harmonized. A 3D point cloud with 1024 points is calculated out of each mesh.

The verification of depth image datasets is mainly performed by experts or the authors [41,58]. Alternatively, the same dataset is semantically segmented by different people to identify error annotations [56]. No information is available about the validation of the *S3DIS* dataset [40].

A selection of recent semantic 3D point clouds generated with LIDAR systems is summarized in Table 3. These datasets will be used later in the quality model. Most 3D point clouds from LIDAR systems are for outdoor scenes and are captured with multi-sensor systems (MSS). With MSS, the capturing of larger areas is more efficient than with TLS. The geometric accuracy of a few centimeters, which is necessary for the majority of

applications in geodesy and civil engineering, is maintained. In addition to the LIDAR measurements, many MSS capture RGB images from the scanned scene to colorize the point cloud. Furthermore, these images can be used for semantic segmentation.

The GT semantic segmentation of the datasets *Paris-Lille 3D*, *Semantic3D*, *MLS1 TUM City Campus* (MSL1 TUM CC), *Toronto3D* and *Complex Scene Point Cloud* (CSPC) is conducted completely or in parts with *CC*. For these datasets, the 3D point cloud format is the basis for data processing. This is also the case for the *SemanticKITTI* dataset, which is semantically segmented using a custom offline tool [13]. The *Building Indoor Point Cloud* (BIPC) dataset uses the *LabelMe* tool [62] for the segmentation and classification of 2D images. The 2D semantic segments are projected into 3D space after annotation. Any incorrect annotations in the point cloud are corrected using another 3D tool [63]. Another method to semantically segment 3D point clouds is to fit geometries, such as planes or boxes, into the point cloud. This is applied to parts of the dataset *Semantic3D* [15]. All points within a certain distance from the geometry are selected. The resulting segment is assigned to a class.

Table 3. LIDAR-recorded datasets with year of publication, sensor, sensor method, separation method (semantic segmentation (SSeg) and instance segmentation (ISeg)), number of points, number of classes and environment.

Dataset	Year	Sensor	Sensor Method	Separation Method	No. of Points	No. of Classes	Environment
Paris Lille 3D [64]	2018	Velod. HDL-32E	MMS car	SSeg	1,431 M	50	Outdoor
Semantic3D [15]	2017	Unknown TLS	TLS	SSeg	4 B	8	Outdoor
SemanticKITTI [13]	2019	Velod. HDL-64E	MMS car	SSeg	4.5 B	28	Outdoor
MSL1 TUM CC [14]	2020	Velod. HDL-64E	MMS car	SSeg, ISeg	1.7M	8	Outdoor
Toronto3D [65]	2020	Teled. Opt. Mev.	MMS car	SSeg	78.3 M	8	Outdoor
CSPC-Dataset [66]	2020	Velod. VLP-16	MMS backp.	SSeg	68.3 M	6	Outdoor
BIPC-Dataset [63]	2021	Velod. VLP-16	MMS backp.	SSeg	-	30	Indoor

Closely related to the semantic segmentation is its evaluation. The *Semantic3D* dataset is evaluated by class comparisons in the overlapping areas of the neighboring point clouds. For this purpose, all points in the neighborhood of an adjacent point cloud are selected from a given point with a search radius of 50 mm. The classes of the selected points are compared with the class of the initial point [15]. The *SemanticKITTI* and the *CSPC* datasets are evaluated and improved by experts in a second processing step [13,66]. For the *BIPC* dataset, the segments created in 2D are evaluated on the 3D point cloud [63]. Statistical evaluation of semantic accuracy for all datasets is not documented. No information on the verification of semantic segmentation is available for the *Paris-Lille 3D*, *MLS1 TUM CC* and *Toronto3D* datasets.

Based on the datasets from the last six years, it can be concluded that more and more LIDAR systems are being used. Mainly LIDAR datasets of outdoor areas are created, because of the larger range and the higher resolution of these systems. For indoors, depth imaging cameras are still commonly used. Since many of these data come from the CV domain, surface models or voxels are additional output formats, along with point clouds and images. It can be seen that the datasets are not necessarily larger in terms of classes and points, but the annotation is more specialized and improved compared to early datasets. Earlier datasets are evaluated with new tools and optimized for specific tasks. The manual annotation can be still identified as a bottleneck.

2.4. Point Cloud Annotation Tools

Many annotation services and tools are used for autonomous driving or driver assistance. For this application, a few outdoor classes need to be (roughly) annotated. An overview and comparison of 33 annotation tools for this area of application is presented in

[67]. These annotation tools mainly use simple geometries, such as bounding boxes, plane and lines, to form instances and semantic classes. The very efficient and coarse semantic segmentation of large datasets is possible with these (semi-automatic) methods.

A number of commercial label services, such as *Playment* (https://playment.io/ accessed on 20 November 2021), *scale.ai* (https://scale.com/ accessed on 20 November 2021) and *basic.ai* (https://www.basic.ai/ accessed on 20 November 2021), have also extended their services towards 3D point clouds. The disadvantage of these services is that they cannot be used for projects with confidential data. For applications where confidentiality and accurate semantic segmentation are relevant, offline tools can be used. Some of these tools are highly specialized for certain fields of application, so that only certain data can be imported or annotated according to predefined classes or rules [13,68].

The tools for annotation are diverse in terms of user interaction. In this context, annotation tools use virtual reality visualization [69]. Other tools use segmentation in 2D or 3D space [62,68], as well as fully manual and semi-automatic segmentation [70]. A selection of tools is briefly presented in Table 4. The tools are distinguished by the functionality of semantic and instance segmentation. In addition, they are categorized according to the central functions for the segmentation.

Table 4. Selection of annotation tools (x = present). Distinguished for instance and for semantic segmentation. Segmentation is performed in 2D or 3D with free-hand tools, automatically or with bounding boxes or geometries.

Tool	Instance	Semantic	Free Hand	Automatic	Bounding Box
Recap [71]		x	3D		3D
CloudCompare [59]	x	x	3D		3D
SemanticKITTI [13]	x	x	3D		
PCCT [72]		x		2D	

Recap [71] is a commercial software used to segment and classify 3D point clouds. Single or multiple registered point clouds are visualized in one project as one 3D point cloud. By rotations, displacements and zooming via mouse buttons, any perspective can be selected. In each *Recap* project, individual classes can be created, according to which a point cloud is semantically classified. For each class, an individual file is exported, which carries the class name. For the classification, point cloud segments are formed by free-hand selection, wrapping with simple geometries or fitting of layers.

CloudCompare [59] is one of the most commonly used open-source tools for point cloud processing and analysis, which can be used to segment and classify point clouds. Different methods to create annotations are offered as plugins. Moreover, *Semantic3D* and *MSL1 TUM CC* use the available functions in the main program for efficient semantic segmentation. The point cloud is displayed in 3D and navigation with mouse buttons is possible. The workflow uses the geometric and spectral point features to segment the point cloud in stages. After pre-segmentation, point cloud segments are further subdivided by free-hand selection. Individual segments can be combined into one semantic class, which is exported as a single file.

The **SemanticKITTI** annotation tool [13] was initially developed for the classification of kinematic 3D point clouds from a *Velodyne* LIDAR system. In addition to the point clouds, navigation and synchronization data are required for this tool. The processing of all *Velodyne* raw data is performed by this tool. Individual scans are registered at the beginning, resulting in a continuous acquisition sequence. For segmentation and classification, the point cloud is divided into 100×100 m tiles. The segmentation is carried out with a free-hand lasso and a point marking tool. Predefined or custom classes can be used. The original point cloud files are not modified with this tool. For each point, a label file is created that contains the semantic information and instances of each point.

The **point cloud classification tool** (PCCT) [72] is a tool for the semantic segmentation of (primarily) static panoramic scans. Point clouds are projected into 2D space for classi-

fication. This is achieved by cutting the point cloud horizontally or vertically into slices. Alternatively, vertical cylinders in different distances are used as a projection plan. The segmentation is performed in the 2D plane using a pixel-based regional growing method [60]. Via a browser application, users can assign one of 20 predefined indoor and outdoor classes to the displayed segments. The PCCT is multi-user-capable. For each semantic class, a point cloud is provided.

The presented tools show the range of functions for the segmentation of the point clouds. As more flexibility is given to the user for segmentation, more details of the segments can be formed. Tools such as the *PCCT* form the segments according to fixed rules. Here, different results can only be achieved by the classification of different users. With all other tools, classification and segmentation performance are not separable.

3. Quality Model for Semantic Point Clouds

Many published datasets and tools are indicated as high-quality. This statement is true for the application for which these datasets are intended. The term "high-end" may refer to the quality of the acquisition, the reconstruction of a mesh, the semantic segmentation or any other aspect. However, in the rarest cases, all possible aspects are of high quality. In order to describe the quality of the datasets, the first step is to define the main quality characteristics. Unfortunately, a quality model cannot be created for all conceivable cases. This would be too complex and no longer understandable, and the focus should therefore be on one aspect per quality model. This aspect will be the focus of semantic segmentation for our quality model. The measurement methods, datasets and annotation tools described in Section 2 are the basis for the quality model's development.

One approach to describe the quality is to use the ISO 9000:2015 (3.6.2) [73] and DIN 55350:2020 [74]. Here, quality is defined as the "degree to which a set of inherent characteristics of an object fulfills requirements" [73]. The point cloud and the segmentation processes are the subjects of investigation, whose quality characteristics should be fulfilled to a certain degree. The characteristics can be expressed by quality parameters. Thus, the quality is a simple comparison of the actual and required quality parameter values of an object. ISO 9000:2015 also defines quality for the process of creation, so that process characteristics are required as well. Besides the process (segmentation) characteristic, there are characteristics that are affected by previous steps, such as capturing, and this influences the final object characteristics (semantic point cloud). This interaction is shown in Figure 3. The prior characteristics are derived from the acquisition method and must meet a minimum standard. Only if the minimum standard is fulfilled can the actual processing step be performed. The unprocessed point cloud must have a minimum resolution and fulfill a certain geometric *level of accuracy* (LoA). A suitable scheme to define the geometric LoA is provided by DIN18710:2010 [75]. The prior and the process characteristics influence the new object's characteristics.

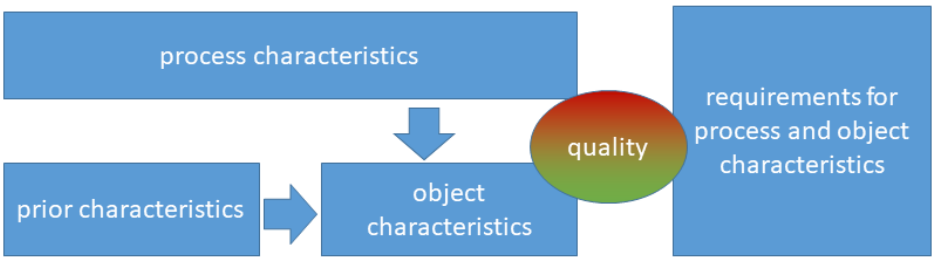

Figure 3. Interaction of the different characteristic types and the requirement to determine quality.

The way in which a semantic segmentation is performed can be expressed in the object's quality parameters. For example, erroneous points should be determined by the semantic knowledge of the annotator and should be assigned to an appropriate class. Thus, additional knowledge is introduced into the application. The quality of this knowledge is

an example of a process-dependent influence on the quality. Aspects such as how a process is carried out and how it is evaluated must be expressed by characteristics. For the semantic segmentation, the degree of correctness and repetition accuracy can be determined if either a reference is available (correctness) or the process is performed n times independently (repetition accuracy). For the repetition accuracy, the number of repetitions and the type of process must be defined. This is an additional aspect that should be covered by a quality model. For the creation of a quality model that takes into account the above-mentioned aspects, the following basic requirements are necessary:

- An application must be defined;
- A semantic segmentation process must be described;
- An abstract model of the semantic must be created;
- A data model must be created;
- Measured or synthetic point clouds must be available;
- Characteristics and parameters must be defined;
- Target values for the quality parameters must be defined.

These seven constraints set the framework for the development of the quality model. The applied case is the creation of a semantic segmented point cloud for the modeling of indoor components and furniture. One application for which such a scenario is necessary is the creation of a Building Information Model (BIM) from a point cloud (Scan-to-BIM). The semantic segmentation of a point cloud for this is a complex and an increasingly demanding application in geodesy and civil engineering [76]. The differentiation of clutter or scanning artifacts from filigree classes, such as tables or chairs, is a problem that cannot be adequately solved by current automatic processes.

3.1. Classification Process

A process description outlines the individual steps that are to be implemented. Thereby, goals (tasks), data, definitions, tools and framework conditions are addressed. Point clouds belong to the group of geodata, so that a process description based on the model for geodata [77] is chosen. The point cloud semantic segmentation process is shown in Figure 4. It consists of the object in the real world, two models (c and d), the data (b) and an action statement describing the interaction.

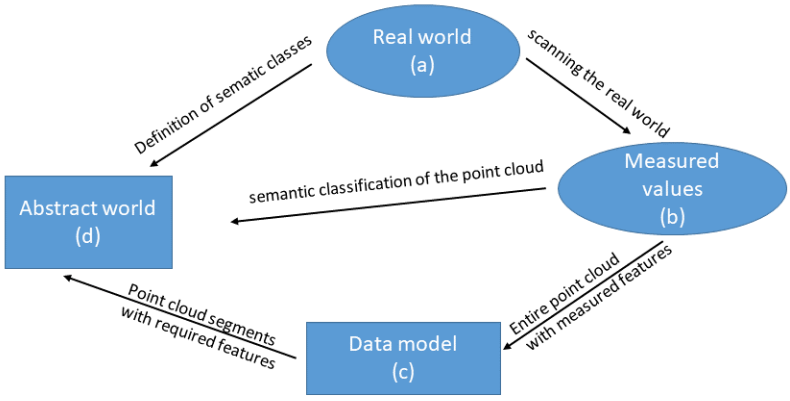

Figure 4. Process for semantic point cloud segmentation.

Semantic segmentation is an extension of point cloud features that can be described as a process. An abstract model (d) of the reality defines semantic classes, which describe which objects are represented and in which level of detail. The abstract model is always a generalization of the real world (a), which is captured by measurement methods as measured values (b). The measured values are the unclassified point clouds or individual points and consist of the geometric and spectral features. The data model (c) defines the file format in which the measured values are available and into which format they

are transformed for the abstract model (by semantic segmentation). The data model for semantic point cloud segmentation specifies that any point has a new semantic feature and each semantic class is a segment.

3.2. Abstract Model

The abstract model for a semantic segmentation describes which semantic classes are represented and defines the class content. The definition of the abstract model should correspond to the application for which the point cloud is used. When defining the classes, two variants are used. Variant 1 is to determine exactly those classes that are needed. Variant 2 is a hierarchical class definition (CD). For this, super-classes are formed stepwise, so that object parts can be distinguished. Variant 1 leads to very small semantic CD, such as that required for autonomous driving [78]. Variant 2 leads to a CD with more than 50 classes [64]. A very small number of classes has the advantage that the semantic segmentation can be performed faster and the classes can be more precisely defined. A distinction between trees and traffic signs is easy in point clouds. If it is necessary to distinguish between beech and oak trees, the definition is much more complex. It should be done in multiple steps and with additional training of human or algorithmic annotators. In such a case, the definition of the abstract model should be structured hierarchically, as shown in Figure 5. A simple structure in two stages is applied to the *SemanticKITTI* dataset. There, the first hierarchical level contains the class soil, which is distinguished in the second level by roads, sidewalks, parking lots and other surfaces [13]. This structure makes the semantic segmentation more explicit and simpler.

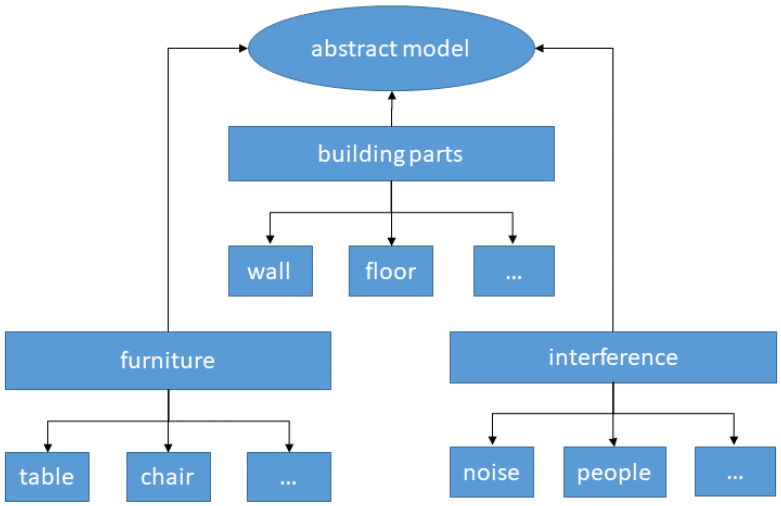

Figure 5. Hierarchical abstract model for the definition of semantic classes. The application is the classification of indoor point clouds.

The optimal abstract model contains all possible classes. This is not possible due to the wide range of applications for point clouds. In Figure 5, a two-level model is shown, where only classes (or objects) are considered that are in a building. It would be too specific for modeling an entire building, since no external objects are included. In turn, for modeling parts of a building, this model is too general, because, in such a model, furniture and disturbances are not included. The advantage of a detailed model is that classes that are not needed are simply ignored. This favors a general model. A possible way to build a universal model would be to refer to the linguistic model *WordNet* [79], as it is already the basis for *ShapeNet* [27] as well as others. All nouns are attributed to the word entity. Starting from the entity, top-level nouns are formed, which can be distinguished in any direction. An application-independent hierarchic abstract model can thus be built. *WordNet* also has the advantage that cross-connections between hierarchical classes are possible and it has a directing effect for the creation of specific abstract models.

For building models from point clouds, an orientation to existing standards, such as the Industry Foundation Classes (IFC) [80], as well as national [81,82] and international [83] guidelines for the Level of Development (LoD), would be possible and helpful. Unfortunately, these standards and guidelines do not yet offer an exact and detailed description of what a semantic class has to look like, but they regulate in which level which contents have to be presented. Moreover, it is still necessary to create an explicit CD. This is the basis for the work of the annotators. The following points should be considered when defining the abstract model:

- A general semantic model should provide the structure for the abstract model;
- The classes of the abstract model should be structured hierarchically, so that, in one definition level, only a number of around five classes exists. The next lower-definition level should contain only points of one higher-level class;
- For each level, all points are classified;
- The classification is an iterative process;
- The level of detail is mainly based on the application and the existing technical functions of the tool.

In addition to class names and class structures, the content of the classes must be defined and represented in such a way that it is understood by the annotator without doubt. The following points for the content definition should be considered:

- The semantic definition must be written in the language of the annotator in order to avoid linguistic misunderstandings, such as translation errors.
- Objects must be described unambiguously by describing their shape, size or color. It is to be considered that objects of the same semantic class are represented differently in the point clouds. If objects appear in different designs, then this is to be described adequately. A definition of objects can be created, as described in [84].
- Special and unknown objects are to be illustrated by examples, so that the idea of the annotator is identical with what is being represented.
- A definition consists of a written and a figurative description.
- Topological relations should be represented to facilitate the decision in case of difficult-to-recognize object appearances. For example, furniture could be defined as standing on the floor or walls running perpendicular to the floor.
- Geometric boundaries should be clearly defined, as this is the only way to achieve the required geometric accuracy. Using the class *door* as an example, the following definition is possible: *A door ends at the frame, at the seal or at the wall. Erroneous points should be separated completely from the objects.*

The abstract model setup must be communicated to the annotators in a suitable format (training) and checked on a regular basis. In [13], this is implemented by training the annotators on the data previously and providing feedback on the performance. This is possible since each point is semantically segmented by at least two annotators. Feedback to annotators can be given directly by moving or highlighting the segment in the tool [26]. In addition, videos, teaching tools and abstract models are common and useful [67].

3.3. Data Model

The data model is defined differently in the literature. Occasionally, the *abstract model* is also called the *data model*. In the context of this paper, the definition by [77] is applied, who sets the data model equal to the *physical model*. The data model defines which file format is to be used for the measured and the semantic point cloud. In addition, it is clarified how the objects are organized in this file format and which attributes an object can have.

The data model consists of two layers. One is the unclassified point cloud and the other is the classified point cloud. For the purely semantic segmentation of single point clouds, a very simple data model can be chosen. It provides the point cloud as an unsorted list of points with their geometric and spectral attributes. For the unclassified point cloud,

these are typically 3D positions (x, y and z coordinates), color values as RGB values and reflected intensity as values. If more attributes are needed for the point feature description, many applications use a database. Besides the structure and file format, the data type of the feature has to be defined. This is usually done in the file format description. A data model for the semantic segmentation must be able to represent semantic features in addition to geometric and spectral features.

3.4. Quality Model

The quality model describes the characteristics and suitability of a semantic segmented point cloud for a certain application. Process- and object-specific quality characteristics from the quality domains of reliability, integrity (usefulness) and accuracy are used as the basis for the evaluation [85]. The quality characteristics are chosen in such a way that they are applicable for manual semantic segmentations, if these are performed according to the process in Section 3.1. The three quality areas are described by seven quality characteristics (Figure 6). Each quality characteristic is expressed by quality parameters.

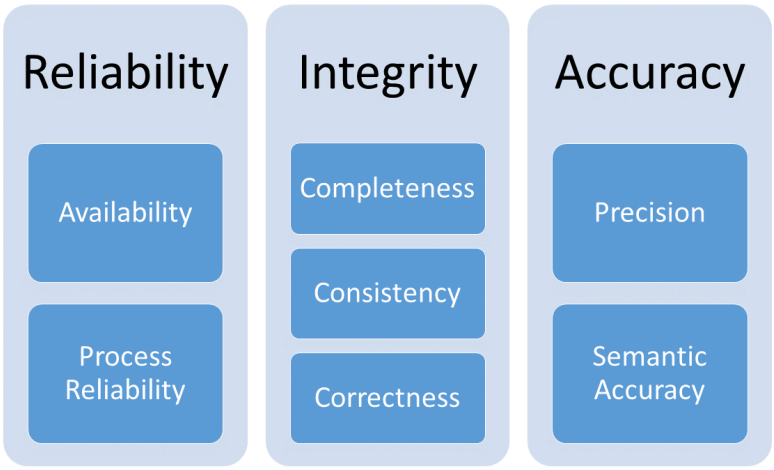

Figure 6. Quality model for 3D point clouds. Concept idea inspired by [85].

The model shown in Figure 6, which is further described below, has its roots in the idea of [85]. It is accepted in many disciplines and is used for various applications. To the best of our knowledge, this model idea has not yet been applied to the semantic segmentation of 3D point clouds. Our central contribution in terms of the quality model is the compilation and selection of characteristics and parameters to make a semantic segmentation of a point cloud describable and evaluable. The development of the quality model has the goal of questioning and improving the quality of the datasets. Only with all-round high-quality datasets is it possible to develop reliable and accurate algorithms and tools [86]. In addition, the practical use of point clouds for reality capturing should be considered by the model characteristics.

3.4.1. Quality Characteristics

Quality characteristics are selected characteristics from the total of all characteristics that a semantic point cloud has. This characteristic selection relates to the requirements of a certain application in which the object is to be used [74]. When applying a quality model to a semantically segmented point cloud, the quality of the segmentation and classification, as well as the quality of the raw point cloud, is primarily expressed by the quality characteristics.

The **availability** describes which data and information are available to the annotator or the algorithm prior to the task. Parameters for this characteristic express which information is known about the point cloud. This relates to point clouds, task definitions and processes. A quality parameter expresses whether the information is available in a specific form and

in the required quantity. It is the basis of all further characteristics and must be fulfilled in order to carry out a semantic segmentation and its evaluation.

The **process reliability** describes how the process was carried out. This characteristic can be determined by performing the semantic segmentation several times. After each segmentation, accuracy parameters are determined, which can be used as terminating criteria, as in [27]. Other variants require that a certain number of iterations must be fulfilled in order to determine a quality parameter. This variant is preferred for the description of the process quality, since it maps the variance of the metric and parameter values. Using this, the achievable performance can be determined by a process setting. Due to the complexity of these tasks, the average repetition factor to determine this parameter is usually very small, as shown in [13] factor 2, in [57] factor 2.3 and in [68] factor 4.

Completeness gives the degree to which the necessary information, determinations and execution of the work steps for the classification are present.

Consistency is the degree to which the measured values match the data model. Here, it is necessary to check whether the point cloud features are present and whether they take the corresponding range of values.

For the semantic segmentation, **correctness**, **precision**, and **semantic accuracy** are different characteristics that have different underlying causes and different effects on the usability of the point cloud. Moreover, these characteristics are often defined and summarized in different ways. For example, if only the performance of a semantic segmentation method is to be considered, correctness and precision are often combined with accuracy. The accuracy is commonly given when ML and AI algorithms are used. In many semantic segmentation applications, this defined characteristic is expressed by the parameter IoU, also known as the Jaccard index, or the F1 score, also known as Dice's index. These parameters are the weighted averages of both characteristics. It is advantageous to apply one combined characteristic of accuracy and its meaningful parameter, e.g., IoU, for better comparison. Other applications use more than one parameter to describe the different perspectives of accuracy for a more distinguished and cause-oriented view.

For the analysis of the semantic segmentation process, two types of errors are possible: a point is erroneously assigned to a class to which it does not belong or a true point of this class is not recognized as member of it. These two errors are known as first- and second-type errors from statistical tests [87]. The segmentation **precision** can be considered an error of the first type. This error specifies how well an annotator or an algorithm can distinguish classes—for example, how accurately class boundaries can be drawn. The segmentation **correctness** can also be considered a second-type error. This error describes how well a class can be recognized, e.g., how unique the point features are. Thus, the best features are used to obtain a class of homogenous points. This type of error can be of importance depending on the analysis in question. For example, it may be less critical if not all points of a large class (such as floor) are detected during semantic segmentation, as long as these points are not classified or assigned to a class (e.g., scanning artifacts) that is not further used. More problematic are additional points (e.g., from scanning artifacts) that are assigned to the class floor, because the point cloud represents incorrect semantics.

Thus far, correctness and precision based on the number of points describe the quality of a semantic point cloud. However, these characteristics do not give any information about the geometry of the semantic classes and its geometric size changes due to errors. In order to be able to evaluate the geometric aspect as well, the characteristics of correctness and precision have to be extended. Geometric correctness can be determined if a (dense) reference point cloud or surface model is available. The correctness can be determined for each individual point. This information can no longer be evaluated for several hundred thousand points. The correctness of a point cloud can be determined by the mean, average deviation or standard deviation of all points in a segment. In general, correctness is the degree to which the abstract model matches the achieved semantic segmentation result. This can be divided into user-dependent and software-dependent correctness. The user-dependent correctness is based on the understanding of the CD and the usage of the

software by the user. The software-dependent correctness refers to errors in the software (e.g., incorrect parameters or programming). However, a separation is only possible if the semantic segmentation is carried out several times under controllable conditions.

The quality characteristic of precision is described by the parameters for the semantic and geometric precision. The term "precision" should be defined clearly, because there are different definitions in use. In the geodetic context, precision is often understood as repeatability [87]. The deviation of the results of an experiment to its mean value after n repetitions is determined. For the semantic segmentation process, this definition would lead to the determination of how much the individual segmentations deviate from each other. This shall not be the main subject of the investigation, since a deviation to the mean of several segmentations usually has no relevance for a practical application. Nevertheless, it makes sense to repeat a segmentation and to calculate a joint point cloud from these repetitions in order to increase the reliability, as mentioned above. Usually, deviation from a reference point cloud is required. This can be expressed by the ratio of true points to all points assigned to a class [88]. This term describes how much of the segmentation is "correct" and is commonly used in ML. Mostly, the inverse proportion is of major importance for the development of an application, because this describes what does not work yet [89]. This proportion is then the subject of analysis. In addition to the use of the number of points, it is advantageous for 3D models and point clouds to also use the areal ratios as well as geometric parameters.

The **semantic accuracy** describes how well the semantic label fits to a semantic point cloud segment. The difficulty is in defining what is semantically correct, which attributes are described and which depth of description and distinction must be applied. For the definition of what is semantically correct, no universally valid definition can be found. An attempt to standardize this problem was discussed in Section 3.2. For the type of attribute description, the IFC standard [80] can be used. This is designed for the development and not for the documentation. This can be explained using the example with the tables. The table itself forms a semantic class. These classes can be differentiated during the next stage into a frame and table top. As far as we know, there is no standardized scheme for this definition, so that an individual CD as shown in Appendix A must be developed and applied.

3.4.2. Quality Parameters

The seven quality characteristics used for semantic segmentation can be described by quality parameters. These parameters describe the property that an object has for a certain characteristics. For instance, these parameters are the presence of a certain data format as a qualitative parameter or the *number of points* (NoP) as a quantitative parameter. This will be demonstrated in an example in Section 4.1. The evaluation of point clouds by the quality model will be covered in Section 4.2. For the evaluation, the quality parameters must be determined and threshold values must be set. Furthermore, the parameters for the semantic segmentation can be distinguished into parameters with object relation (O), concerning the point cloud, and process relation (P), such as the time required for an action or the use of a certain CD. All parameters for the semantic segmentation task are briefly explained and shown in Tables 5–11. The parameters are numbered in the text and refer to the corresponding table entry with P#.# for a clear understanding.

Quality parameters for characteristic **availability** describe which information must be available about the process and the point cloud for a description and an evaluation (Table 5). These parameters are the abstract model expressed by the *CD* (P1.1), the size of the point cloud expressed by the *NoP* (P1.2) and the *area size* (P1.3), as well as the object features (e.g., x-, y-, z-coordinates) before (P1.4) and after (P1.5) the semantic segmentation. Furthermore, the *file format output* (P1.6) and *use restrictions* (P1.7) must be investigated. The *use restrictions* refer to the question of whether a dataset can be used for an application or processing step. Further restrictions are that certain datasets may not be used for training.

The parameter P1.7 ensures an objective evaluation of the datasets. Thus, it is considered that any dataset has a certain bias, which is learned by ML algorithms [86].

Table 5. Parameters for availability.

P. no.	Parameter Name	Unit	Range	P/O
P1	**Availability**			
P1.1	CD exists		yes/no	P
P1.2	Number of points		>0	O
P1.3	Area size	m^2	>0	O
P1.4	Object charac. in		yes/no	O
P1.5	Object charac. out.		yes/no	O
P1.6	File format out		e.g., pts	O
P1.7	Use restriction		yes/no	O

Table 6. Parameters for process reliability.

P. no.	Parameter Name	Unit	Range	P/O
P2	**Reliability of Process**			
P2.1	Number of segmentations		>1	P
P2.2	Average time required	%	0–100	P

The parameters *number of segmentations* (NoS) (P2.1) and *average time required* (ATR) (P2.2) describe the **reliability of the process** (Table 6). If a point cloud is independently semantically segmented more than once, the reliability can be measured. The more frequently a process is carried out, the more reliable are the correctness and accuracy. This is the theoretical assumption. The parameter NoS describes how often a segmentation was performed with a certain method. It is the basis for the calculation of other parameters and can also be used as a quality measure. The ATR can be used to compare different semantic segmentation methods. The ATR is calculated for each method. The average time of all annotators with any method is of interest. The maximum segmentation time of all methods is the value Δt_{max}. The ATR is calculated from Equation (1), where i stands for the respective segmentation. Δt_i is therefore the time needed for the segmentation i. Moreover, the user-dependent segmentation time can be analyzed if all segmentations performed with a certain tool are compared. The parameter ATR describes the process and allows the planning of the working time.

$$ATR = \frac{\sum_{i=1}^{i_{max}} \| \frac{\Delta t_i * 100}{\Delta t_{max}} \|}{i} \quad (1)$$

Table 7. Parameters for completeness.

P. no.	Parameter Name	Unit	Range	P/O
P3	**Completeness**			
P3.1	Semantic segmentation rate	%	0–100	O
P3.2	Number of classes		>0	O

The **completeness** of a semantically segmented point cloud (Table 7) is described by the *semantic segmentation rate* (SSR) (P3.1) and *number of classes* (NoC) (P3.2). The parameter SSR describes how many points have been assigned to any class. The SSR is the quotient of the number of classified points (P_{cls}) and all points (P_{all}) (Equation (2)).

$$SSR = \frac{P_{cls}}{P_{all}} * 100 \quad (2)$$

A point cloud that is only segmented in parts often occurs in the application phase. The semantically segmented parts of the point cloud are used for training or for the evaluation

of an algorithm. The rest of the data are then semantically segmented using the automatic method. The parameter NoC describes how many classes are available for a certain dataset.

Table 8. Parameters for consistency.

P. no.	Parameter Name	Unit	Range	P/O
P4	**Consistency**			
P4.1	Geometric Consistency (GC) of x, y, z	m	≥0	O
P4.2	Spectral Consistency of RGB (SCRGB)		0–255	O
P4.3	Spectral Consistency of I (SCI)		0–255	O
P4.4	Class equality		0–1	O

The **consistency** of the data (Table 8) is determined by the units and the scaling ranges of the object features (P4.1 to P4.3). Each object parameter directly relates to a quality parameter. The determination can be achieved automatically or taken from the data (e.g., using a text editor). Furthermore, the consistency is described by the measure of the *class equality* (CE) (P4.4). This is calculated from the target value of a balanced class distribution (C_{target}). All classes should be represented by the same amount of points, so that, later, an ML procedure has optimal learning conditions. However, this requirement is never given with real datasets, because classes such as walls and floors are overrepresented by points. The proportion of points of a class in relation to the total NoP is expressed by a ratio in the value range 0–1. The actual distributions are then calculated (C_{act}). The differences between the target and actual values for each class are determined. The sum of the absolute differences divided by two is a measure of balance (Equation (3)), where 0 represents a balanced ratio and 1 an unbalanced ratio.

$$CE = \frac{\sum_{i=1}^{k} \|(C_{target} - C_{act})\|}{2} \tag{3}$$

Table 9. Parameters for correctness.

P. no.	Parameter	Unit	Range	P/O
P5	**Correctness**			
P5.1	Recall of points *class x*	%	0–100	O
P5.2	Recall of area *class x*	%	0–100	O

The **correctness** (Table 9) of the semantic segmentation can be described by the parameter *recall of points* (RP) (P5.1). The PR is the rate between correctly assigned true positive (TP) points and the NoP in the abstract model for a certain class (TP and false negative (FN) points) (Figure 7). It is expressed by Equation (4). This parameter depends on the size differences of the class in the abstract model. If the classes differ greatly, as can be evaluated by the parameter CE, a comparison of different classes may lose significance. For a small set, even a few FN points can significantly lower the parameter. This problem is discussed in [89] and described by a new parameter for informativeness. For applications in the context of point clouds, this parameter is unsuitable due to the irregular distribution of the points.

$$RP = \frac{TP}{TP + FN} \tag{4}$$

$$RA = \frac{TP_{area}}{TP_{area} + FN_{area}} \tag{5}$$

To avoid the point cloud density problem, the representation in the form of areas can be used. Here, the areas are calculated for the point cloud segments. Instead of the NoP, the TP area size can be inserted into Equation (4). The result is the *recall of area* (RA) in Equation (5). The correctness is now described by the area that is covered by TP points

divided by the area of all reference points of this class. As an intermediate step to calculate these parameters, the areas that are correctly and incorrectly assigned are calculated. In the case of incorrect assignments, the distinction between FN and FP areas is of interest. The parameter RA expresses the influence of FN surfaces. The influence of the false positive (FP) areas is described in the following, among others, by the *precision of area* (PA). The FN and FP points are visualized in Figure 8. This visualization allows an analysis of the semantic segmentation, e.g., the assignment of scanning artifacts to a class or the occurrence of classification gaps can be determined.

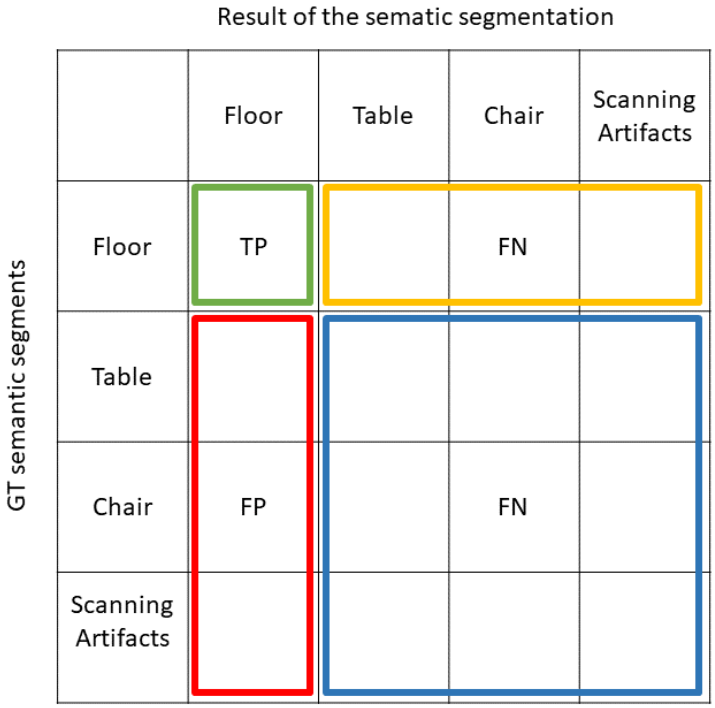

Figure 7. Schematic representation of the confusion matrix for the floor class with entries for TP, FN, FP and true negative (TN) points.

Table 10. Parameters for precision.

P. no.	Parameter	Unit	Range	P/O
P6	**Precision**			
P6.1	Precision *class x*	%	0–100	O
P6.2	Precision area *class x*	%	0–100	O
P6.3	MD of FP pts. *class x*	mm	≥0	O
P6.4	SD of FP pts. *class x*	mm	≥0	O

The precision is expressed by the *precision of points* (PP) (P6.1) and the PA (P6.2). The PP is the ratio of TP points of a class to all points assigned by the segmentation of this class (Equation (6)). The assigned points could also be expressed as the sum of the TP and the FN points (Figure 7).

$$PP = \frac{TP}{TP + FP} \tag{6}$$

$$PA = \frac{TP_{area}}{TP_{area} + FP_{area}} \tag{7}$$

The consideration of the characteristic precision based on areas that are spanned by the point cloud segments can be advantageous when using the point cloud as a model. For

a geometric expression, Equation (7) can be used to determine PA. The visualization of the FP points is given in Figure 8, which is a good starting point for the analysis process.

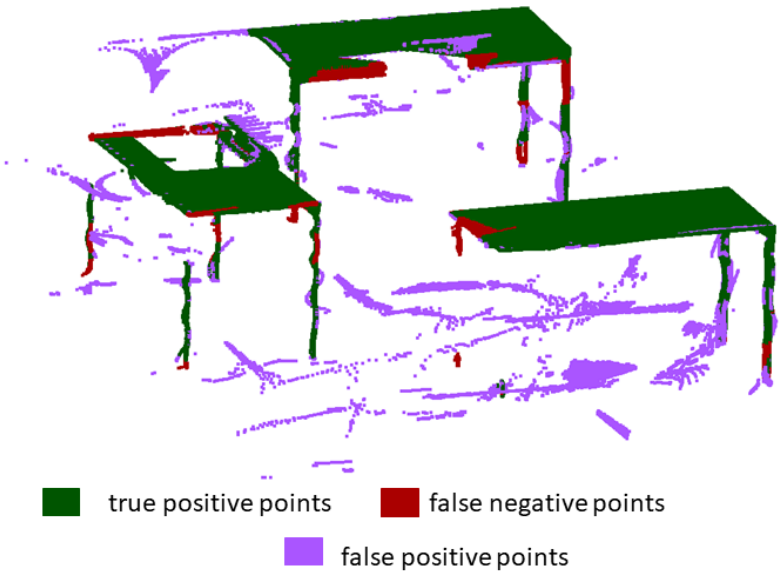

Figure 8. Segmented point cloud of the class table colored by TP, FP and FN points.

The geometric part of the precision can also be described by the parameters *maximum deviation* (MD) *of FP points* (P6.3) and *SD of FP points* (P6.4). The *MD of FP* and *SD of FP points* rely on the FP points of the semantic segmentation. They are the points that change the geometry of the semantic class, as shown in Figure 9. For this consideration, only classes with semantic objects are considered, since, normally, the goal of semantic segmentation is to extract objects and to remove scanning artifacts. The geometric deviation of the point cloud segment is of major importance for creating a model. If the point cloud is used to create a mesh, then the MD, which is the enlargement of the class segment, is decisive. This is expressed by the furthest FP point. For modeling on the basis of point clouds or the representation of the recorded objects by symbols, as is the case at the LoD 100 for a BIM application [90], the parameter *SD of FP points* is more meaningful.

The **semantic accuracy** (Table 11) is described by parameters that can be expressed by yes-or-no questions. Documentation of the process and visual inspections can be used to determine the *CD applied* parameter (P7.1) and whether it is structured hierarchically (P7.2). The parameter *CD applied* can be answered with *yes* if the CD is used and at least one class is segmented. The parameter *Hierarchical CD* can be confirmed if the used CD has several levels (at least two) and so different semantic detailing levels are available. The query whose class was finally used is expressed by the parameter P7.3. If the class is present and semantically correct, the parameter is answered with *yes*.

Table 11. Parameters for semantic accuracy.

P no.	Parameter	Unit	Range	P/O
P7	**Semantic Accuracy**			
P7.1	CD applied		yes/no	P
P7.2	Hierarchical CD		yes/no	O
P7.3	*class x* used		yes/no	O

3.4.3. Descriptive and Evaluative Function

A quality model such as the one above can have two functions. One is descriptive and the other is evaluative, as described by ISO 9000 (2015) [73].

For the **descriptive use**, the aim is to display and analyze how individual parameters (defined as significant by the model) vary when influences change. Different settings, tools or work processes for a semantic segmentation can be compared. Quality parameters are not transformed into another representation or range for this purpose. The main influencing characteristics for the development of a semantic segmentation process are considered and this is one main application of the quality model. More precisely, the influence of the initial (manual) segmentation of a point cloud is investigated. Thus, the model also provides the basis for describing an automatic (e.g., ML-based) semantic segmentation process, as considered in many works, such as [91–94].

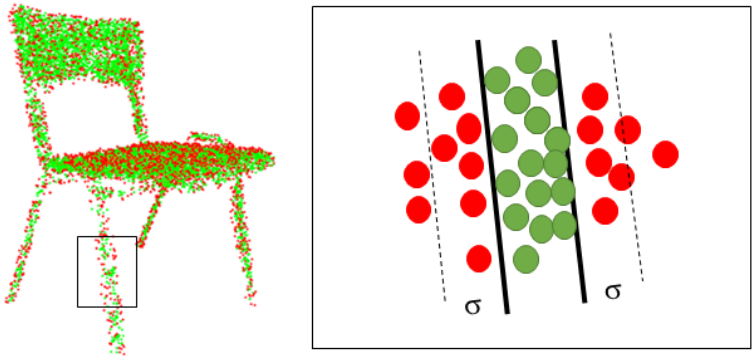

Figure 9. Calculation of the SD of FP points σ on the example of a chair. Green TP points are within the object boundaries. The red FP points were added to the chair class but actually belong to another class.

For the **evaluative use**, the suitability of a point cloud for an application should be assessed. It should be derived from the parameters whether a point cloud in combination with the segmentation method is suitable for a certain application or not. For this purpose, the calculated parameters of the quality model are crucial. An example application would be to use a semantic point cloud to determine the wall surface area, to calculate the renovation costs, based on the as-built wall surface area. For this task, correct semantic segmentation is crucial. The point cloud should be evaluated by applying a quality model in advance. The quality of the individual parameters must be defined by limit or target values. These values are derived from the application. The evaluation steps are defined according to the scheme shown in Figure 10.

After the limit or target values have been defined, they are compared with the determined actual values. This adjustment can be represented in an automatic procedure by one Boolean value. In the simplest overall evaluation method, all parameters must be true for sufficient quality. The weighting of the parameters for special cases prevents excessively rigorous filtering. The central issue is the limit or target values, which are not always known and have to be estimated based on experience.

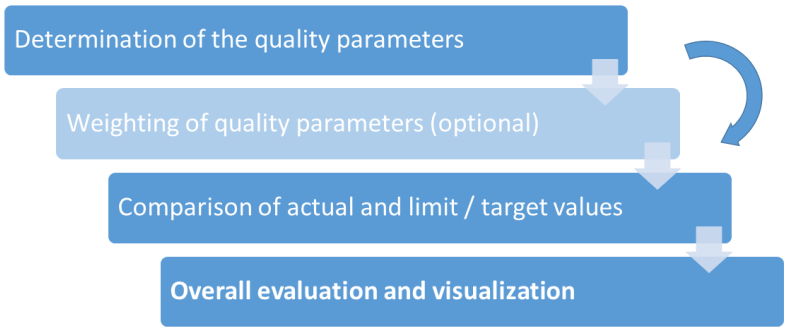

Figure 10. Evaluation of the suitability of a point cloud with the quality model.

4. Applying the Quality Model

The benefit of the quality model as a basis for describing and evaluating the properties of a semantic 3D point cloud will now be explained by some examples. The performance of the quality model is shown on the basis of two of our own indoor point clouds and other publicly available point cloud datasets. Our own point clouds are shown in Figure 11 and were semantically segmented independently, multiple times, using two different semantic segmentation tools. The quality of the point cloud and the semantic segmentation process are described by the quality parameters. The evaluation performance of the quality model is considered for our own and the publicly available datasets. The applications of interest are the analysis of:

- Semantic point cloud as a model;
- Semantic point cloud as a modeling basis;
- Semantic point cloud as training data.

Target values are defined in each case. The geometric, semantic and formal characteristics of the point cloud are processed and used for an application. However, these point cloud characteristics have a degree of uncertainty if the semantic point cloud was created by capturing a real object and performing a semantic segmentation. The possible errors and the quantitative uncertainty of the sensors are described in Section 2.2. It can be stated that the usually resulting effects of currently available and used sensors do not significantly affect the indoor modeling applications. Our own point clouds were recorded with the *Z+F Imager 5016* using a resolution of 6 mm at 10 m. The quality was set to *high* to reduce the noise while still having a moderate (in practice useful) recording time of 6 minutes [95]. The geometric correctness of this point cloud on a flat surface can be estimated as 2 mm to 3 mm according to the investigation of [48], using the *DVW-test-field*-method according to [96]. This accuracy varies due to the different surface shapes and other object properties. In addition, scanning artifacts occur, as shown in Figure 1 and described in Section 2.2. The focus is now on the semantic segmentation, where errors are caused by tool settings and the annotator.

Figure 11. Points to be examined without semantic segmentation. Objects of the chair, table and floor classes, as well as scanning artifacts, are shown.

The point clouds in Figure 11 are very challenging for semantic segmentation. A CD was developed and applied in order to investigate segmentation problems. This CD consists of five classes and is partly hierarchically structured. The classes of the first level are floor, furniture and scanning artifacts. In the second level, the furniture class

is divided into table and chair. Tables and chairs are two object classes that are spatially and geometrically similar, which makes segmentation difficult. The points of these two classes also have similar spectral properties. Finally, the object surfaces are highly reflective and the geometric shape is susceptible to the occurrence of scanning artifacts. The floor class was integrated to simulate scenic segmentation with foreground and background objects. The separation of scanning artifacts is a complex task, even for humans, where subjective decisions must be made and learned. The test point cloud does not represent any real particular task, but is intended to demonstrate achievable performance on challenging cases. The point clouds show recordings of a laboratory (*Lab*) and a seminar room (*Room*), which were automatically segmented with the *PCCT* using the spectral parameters color and intensity. The point clouds were processed by up to nine different annotators. These are the test point clouds *Lab RGB*, *Lab I* as well as *Room RGB* and *Room I*. Furthermore, the point clouds *Lab* and *Room* were processed with *Recap*, in which the annotators determine the segments by themselves. These are the datasets *Lab R* and *Room R*.

4.1. Quality Model to Describe Semantic Point Clouds

The description of a semantic point cloud and a segmentation process is always based on a selection of characteristics, with the goal of being able to answer a specific research or practical question. The research question for the following consideration is:

What influence do the segmentation tool and different annotations have on the quality of the semantic segmented point cloud?

The motivation for this question is to develop an efficient, effective and traceable segmentation process. Different experimental settings and development stages shall be described, so that their influences on the process can be analyzed. This should also result in more convenient point clouds for models and training data, as well as improved process and algorithm understanding. All characteristics of the model are described in detail in the following.

4.1.1. Reliability Characteristics

The reliability of a point cloud can be described mainly by formal information or metadata, as listed in Table 12. The creation and use of a CD, which regulates which objects will be segmented and classified, is of primary importance. A comparison of semantic segmentation is only possible if the CD is kept constant. The parameter *CD exists* must be available to utilize all other semantic-based descriptions. The accuracy of the implementation of the CD is described by the parameters of semantic accuracy in Section 4.1.3. For the test point clouds, a CD exists, which describes the semantic classes of floor, furniture, chair and table, and scanning artifacts.

The size of the point cloud is another formal parameter, which is described by the NoP and the surface area. The NoP that can be processed by segmentation tools varies widely. Sometimes, the point cloud is automatically reduced to a maximum NoP. This filtering changes the point cloud structure and, depending on the application, can result in unwanted effects, such as the loss of surface details. The *Lab* and *Room* point clouds consist of 2.7 and 14.5 million points. The surface area of the objects covered by points is 51 m^2 and 61 m^2 for the *Lab* and the *Room* point clouds, respectively. Based on these two parameters, an additional useful parameter, the average point cloud density, can be calculated. The average point cloud density can be used as the resolution of the point cloud. This varies with the distance to the recording device, and this shows that the parameters of the quality model are chosen to be fundamental, so that optional parameter extensions are possible.

The segmentation tools require certain point cloud features to enable processing. Spectral features are often used to perform an automatic segmentation or to color the point for better visual differentiation. Most point clouds have geometric features (coordinates) and spectral features for color and intensity. In addition to these features, normals (N) are calculated to create perspective images or orient the single point within their neighborhood, as done with the *PCCT*. These features can enrich the point cloud after the semantic seg-

mentation. The exported feature can change during the semantic segmentation. The point clouds in the example are only extended by the feature semantic class. This is expressed by exporting each class as a single *pts* file. Closely related to the feature parameters of the point cloud is the file format that is available for import and export for software. The *pts* format is supported by all tools being used. This file format corresponds to the data model of Section 3.3. The used data model states that all segments should be available as an individual file. If the data model requires that the semantics of the point cloud have to be included in one file, then a different export file format must be used. This file format must have one additional space for the semantic label. The point clouds *Lab* and *Room* are currently not licensed and are only used internally, so there is no restriction on usage (P1.7). This means that the use of the datasets cannot be traced.

Table 12. Calculated and determined values for the quality parameters of availability and reliability of process. Object parameters with * are calculated in the segmentation software.

P. no.	Parameter Name	Lab RGB	Lab I	Lab R	Room RGB	Room I	Room R
P1	**Availability**						
P1.1	CD exists	yes	yes	yes	yes	yes	yes
P1.2	NoP	2,790,352 points			14,526,242 points		
P1.3	Area size	51 m^2			61 m^2		
P1.4	Object char. in.	x, y, z, I, R, G, B, xN*, yN*, zN*					
P1.5	Object char. out.	x, y, z, I, R, G, B, Class					
P1.6	File format out.	pts/csv	pts/csv	pts	pts/csv	pts /csv	pts
P1.7	Use restriction	no	no	no	no	no	no
P2	**Reliability of Process**						
P2.1	NoS	7	7	9	8	8	8
P2.2	ATR	13%	13%	55%	38%	45%	49%

In addition to point cloud metadata, metadata about the process are also represented by the process reliability, as shown in Table 12. Reliability can be determined if a process is performed independently multiple times. It can be determined by observing which parameters change systematically and which are random. According to the research question, two influences should be analyzed. On the one hand, the influence of different users is considered, and on the other hand, that of different tools is assessed. The repeat accuracy of different users is investigated in Section 4.1.4. At this point, the focus is on the two different tools. For a statistical consideration, the number of seven to nine annotations per tool is too small. However, a qualitative or comparative description of the influences of the tools in the form of a tendency is possible despite the small number of samples. For this purpose, the following values are not based on the annotations of individual annotators, but on a joint point cloud with all annotations. For the determination of the parameters of the datasets *Lab RGB* and *Lab I*, seven different annotations were performed; for the *Lab R* dataset, nine annotations were performed, and for the datasets *Room RGB*, *Room I* and *Room R*, eight annotations were performed.

The ATR is calculated based on the longest time for semantic segmentation for each point cloud. The maximum time is 120 minutes for the point cloud *Lab* and 194 minutes for the point cloud *Room*. For both point clouds, the semantic segmentation with *Recap* takes the longest. The ATR values in Table 12 show that the *PCCT* provides an average of only 13% of the maximum time for small point clouds such as *Lab*. With *Recap*, the ATR is 55% for the *Lab* point cloud. For the larger dataset, it can be seen that the *PCCT* can be used to work faster on average, but the differences in time decrease with increasing point cloud size.

The parameters of availability and process reliability are the basis on which to describe further parameters that have a more practical meaning for the investigated question. Thus far, it is described how a process can be carried out with the selected data and resources, how reliable this process and the other quality parameters are, as well as how efficient the tools and its usage are in comparison to others.

4.1.2. Integrity Characteristics

The integrity of the semantic point cloud is described by the parameters of the characteristics completeness, consistency and correctness, which are shown in Table 13. Completeness refers to the point cloud and its individual points. More precisely, it indicates how many points are still present after processing with a segmentation tool. After processing with *Recap*, the NoP was significantly reduced. The segmented point cloud still consists of 74% of the original points for the *Lab R* dataset and 41% for the dataset *Room R*. This point reduction is due to the tool. In other applications, this may arise from the task description—for example, if only *x*% of the point cloud is to be semantically segmented manually and the rest automatically. In addition to object completeness, semantic completeness can be determined. All classes described in the CD should exist in the semantic point cloud. This parameter is important for large and hierarchical CDs, when all levels are not or not yet classified. With respect to the segmentation tool, it must be possible to select or include the necessary classes. With *PCCT* and *Recap*, all five classes can be named and set with respect to the application. The point clouds are classified for all classes, but, for the following consideration, only the most detailed level is used. The furniture class is a super-class of the sub-classes table and chair. A super-class exists automatically if all sub-classes are present.

Section 4.1.1 describes which characteristics must be present for the point cloud. The presence of the characteristic is the necessary condition to evaluate whether the point clouds can be used. This is usually only possible if the point cloud features are consistent, which is the case for all six datasets, as shown in Table 13 by P4.1 to P4.3. The spectral features are scaled to the value range of 0 to 255 and the geometric features are given in meters.

A point cloud should have an equal amount of points for each class if it will be used as training data. The CE takes values of 0.65 (*Lab*) and 0.62 (*Room*), indicating that the class distribution is unequal (0.0 means equally distributed). The point cloud *Lab* consists of 90% of the floor class. The remaining 10% of points comprise chairs (3%), tables (6%) and scanning artifacts (1%). The distribution of the point cloud *Room* is comparable (Table 13).

Table 13. Calculated and determined values for the quality parameter of integrity.

P. no.	Parameter Name	Lab RGB	Lab I	Lab R	Room RGB	Room I	Room R
P3	**Completeness**						
P3.1	SSR	1.000	1.000	0.739	1.000	1.000	0.409
P3.2	NoC	5	5	5	5	5	5
P4	**Consistency**						
P4.1	GC x, y, z		10.34, 8.56, 1.00 m			8.07, 6.11, 0.82 m	
P4.2	SCRGB		0–255			0–255	
P4.3	SCI		0–255			0–255	
P4.4	CE	0.65	0.65	0.65	0.62	0.62	0.62
P5	**Correctness**						
P5.1	RP *floor*	99.9%	99.9%	100.0%	99.8%	99.9%	100.0%
P5.1	RP *chair*	96.1%	95.7%	99.2%	81.6%	66.0%	99.7%
P5.1	RP *table*	89.6%	89.6%	99.8%	94.5%	87.8%	99.7%
P5.1	RP *scan. artif.*	27.1%	27.6%	69.6%	47.1%	35.6%	77.2%
P5.2	RA *floor*	100.0%	100.0%	100.0%	99.8%	99.8%	100.0%
P5.2	RA *chair*	97.7%	97.1%	99.5%	97.7%	90.4%	99.7%
P5.2	RA *table*	96.8%	96.1%	99.8%	96.7%	95.9%	99.6%

The characteristic correctness can be determined if it is possible to describe what is true. This description can be made for a semantic point cloud by a semantically enriched geometry. The geometry either describes the target state from planning data or is captured and processed by a higher degree of correctness. This is the case if the point cloud was captured with a more accurate measurement system and a more accurate semantic segmentation method. For most furnished indoor scenes, no highly accurate geometric planning data are available. In this work, a measurement and segmentation method is used

that is significantly more accurate than the method under investigation. The method used for the creation of the semantic GT point cloud is based on simultaneous acquisition and semantic segmentation with the line scanning system *Leica T-Scan5*. The *Leica T-Scan5* is used in conjunction with the *Leica Lasertracker AT 960*. Based on the technical manufacturer specifications [97], the geometric accuracy (GA_P) of predominantly flat surfaces can be determined according to Equation (8).

$$GA_P = 80\,\mu m + 3\,\mu m * d\,m\,(SD\,of\,2\sigma) \quad (8)$$

A maximum distance (d) between the laser tracker and the *Leica T-Scan5* of 10 m can be assumed. The maximum GA_P is therefore 0.11 mm. This can be set equal to the geometric correctness for the following consideration. The semantic is obtained by scanning the real objects individually with the *Leica T-Scan5* and assigning a semantic class during the measurement. Errors can occur due to the assignment of an incorrect class. This was minimized by intensive checks in the field and during data preparation (four-eyes principle). The original point clouds acquired with the *Leica T-Scan5* are further compressed and harmonized so that the maximum point density is less than 1 point/mm^2. The GT point clouds are considered free of semantic errors and contain only semantic objects. Scanning artifacts are not included in the GT point cloud. The point cloud in Figure 12 is the reference model for determining the correctness and the precision of the point cloud to be analyzed.

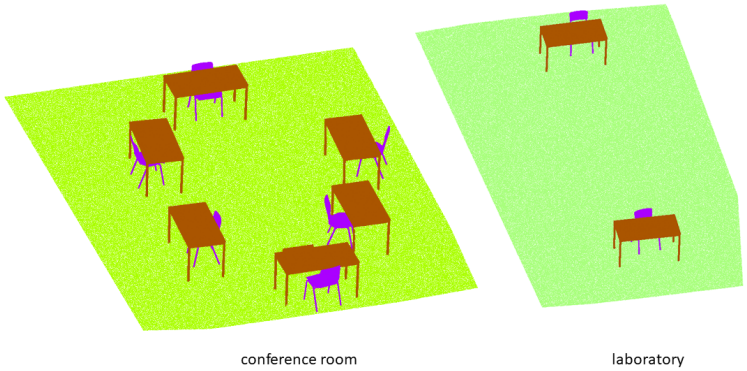

Figure 12. GT point cloud for determination and verification of correctness and precision parameters.

The determination of the correctness parameters can be performed if the GT and the analyzed point clouds are in the same coordinate system. Both point clouds are transformed via discrete target points into a local room coordinate system. Residuals of up to 7 mm (*Lab*) and 5 mm (*Room*) occur as a result of this transformation of the analyzed point cloud. The residuals are considered to denote uncertainty when comparing the point clouds to determine the quality parameter for correctness and precision.

The class segments of the GT point clouds are geometrically compared with those to be analyzed. For this comparison, the following rules apply:

- If the point distance between both point clouds is less than a threshold, then a point in the point cloud under investigation has been correctly semantically segmented. These points are TP points.
- If a segmented point in the investigated point cloud is closer to a segment of another class, then it is an FP point of the selected class.
- The FP points are also FN points of the other classes. By comparing the GT point cloud segments of the other classes with the sub-point cloud of the investigated point cloud, the FN points can be determined.

The resulting confusion matrix of TP, FP and FN points (Figure 7) provides the basis for determining the parameters RP and RA. These parameters express how correct a semantic segmentation is—RP by the ratio of the TP points to all points of the semantic target class and RA by the ratio of the TP area to the total area of a semantic target class. The areas are

calculated via a triangular meshing with the *Ball Pivoting Algorithm* by [98]. Depending on the application, either the RP or RA is more appropriate. RP is more meaningful for applications in which the individual points are important. This is the case if ML applications have to be validated. In these applications, it should be checked how well a task is solved with a dataset. For the use of a point cloud as a model or as a basis for modeling with parametrized geometries, the RA is more suitable.

All correctness parameters in Table 13 refer to a joint point cloud, which was calculated from all segmented point clouds of each dataset. A small program based on *Open3D* functions [99] was used for this purpose. The class membership of each point in the joint point cloud is based on the majority of the classifications within a dataset. The performance of individual annotations can be found in Tables A6–A8 in Appendix C.

The RP varies between 27.1% and 100.0%. The floor class is best recognized, with 99.9% to 100.0%. The chair and table classes were determined differently depending on the tools. For the *Lab R* and *Room R* datasets, the RP is higher than 99.1%. The RP of the *PCCT* datasets varies for the smaller semantic objects between 66.0% and 96.1%. It can be seen that semantic segmentation is better for the smaller point cloud *Lab* (PR higher than 89.6%) than for the larger *Room* point cloud (RP higher than 66.0%). The scanning artifacts are predominantly not detected in the segmentation with the *PCCT* (PR less than 47.1%). Moreover, with *Recap*, these classes are determined poorly, with a PR of only 69.6% and 77.2%, respectively (Table 13).

The RA is determined only for the object classes, since scanning artifacts are not useful for the visualization of an area. For all datasets, this parameter is higher than 90.3%. For the floor class, it is even higher than 99.7%. Since this parameter is based on the same data as the RP, similar behavior can be expected. However, differences occur due to the different point densities. For example, the RA is higher than the RP for all *PCCT* datasets, since this tool is used for areal segmentation and small groups of points (e.g., at class boundaries) are more often assigned to an incorrect class. This can be observed, e.g., in dataset *Room I*, with an RP for the chair class of 66.0% and with 87.8% for the table class. Here, the RA is 90.4% for the chair class and 95.9% for the table class (Table 13). The differences between RP and RA are smaller or do not occur for *Recap*, because the segments can be formed more finely and individually.

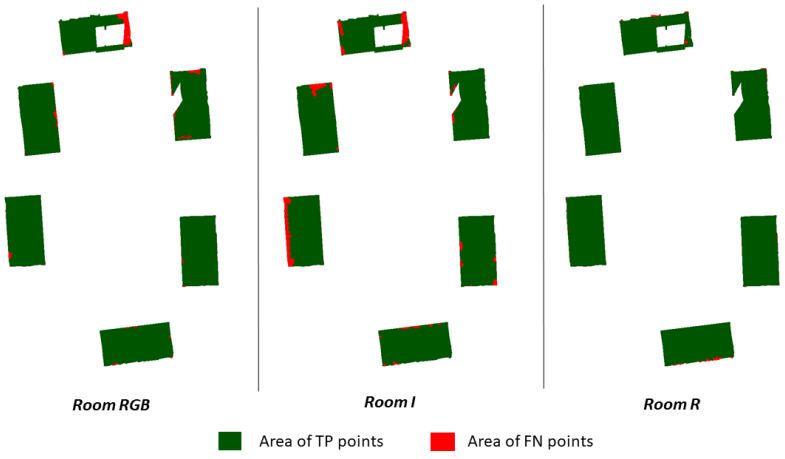

Figure 13. TP and FN areas of dataset *Room* at different semantic segmentations for the table class.

The analysis of the areas can also be useful, if the inverse RA, the area of FN points, is considered. This indicates which areas are not assigned to the correct class. These are holes or missing parts in the segmented point cloud. By visualizing these areas, it is possible to identify certain problematic sections for which the applied tools do not allow correct class assignment. The problematic sections are colored red in Figure 13.

4.1.3. Accuracy Characteristics

The accuracy in the quality model is expressed by the quality characteristics of precision and semantic accuracy. Precision is described by two ratio parameters. Additionally, MD and SD are determined from the FP points. The geometric accuracy is described for the handling of the semantic definitions and in terms of implementation per class (Table 14).

The PP of the floor class is higher than 99.5% for all datasets, so that all segmentation methods work equally well for this class. Based on the PA, the maximum incorrect area can also be determined with 0.5% of the object areas. The use of points or areas leads to no measurable differences.

In contrast, the semantic augmentations applied for the chair and table classes show varying precision. The semantic segments with *Recap* for chair and table consist of more than 95.4% of TP points and 95.6% of the TP area. Thus, only 0.23 m^2 of the table area and 0.17 m^2 of the chair area is falsely semantically segmented. The proportion of object class points in the scanning artifacts class is very small, with 4.6% and 2%. In points, this corresponds to approximately 125,000 and 290,000, respectively. The MD from the GT geometry is up to 92 mm. The *SD of FP points* is less than 39 mm.

The floor was determined by a plan fit; tables and chairs were segmented free-hand. It can be observed that more precise work can be achieved via free-hand segmentation. For the table and chair classes, the *SD of FP points* varies between 9 mm and 16 mm (Table 14). The *PCCT* segmentation of the two point clouds is less precise. The PP for chairs varies between 67.3% and 78.5%. In terms of surfaces, the PA varies between 90.4% and 91.7%. For the table class, the PP varies between 92.1% and 93.6%. The table class has a wider range for PA, from 87.7% to 97.6%. The PA of the chair class is considerably higher than the PP. For the table class, a higher PP can be determined for the dataset *Lab*. For dataset *Room*, the PA is lower and can be expressed as the area. For the table class, up to 1118 mm^2, and for the chair class, up to 380 mm^2 are incorrectly segmented. The proportion of object points in the scanning artifacts class is very high, which is expressed by the PP, which ranges from 30.8% to 61.3% for the *PCCT* datasets. This agrees with the observations on the RP for the object classes in Section 4.1.2.

An influence due to the segmentation with *RGB* or *I* values cannot be observed. However, it can be seen that the smallest object class, chair, is less precise for the dataset *PCCT I*. A comparison of the values in Table 14 shows that the PP and the PA are influenced by the size and content of a point cloud. As an example, this can be observed by the table class for the datasets *Lab RGB* and *Room RGB*. For *Lab RGB*, the PA is higher than the PP. For the dataset *Room RGB*, the PA is 5.5% lower than the PP. This occurs for the *PCCT* datasets, because scanning artifacts are present behind the objects. The scanning artifacts are often assigned to object classes at the front. This can be seen in Figure 14.

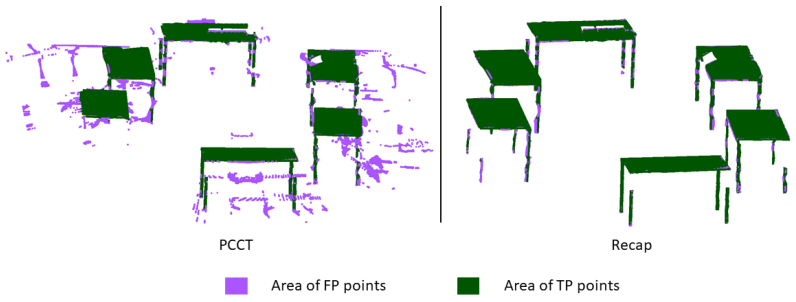

Figure 14. TP and FP points for the table class. The point cloud was semantically segmented with *PCCT* (**left**) and *Recap* (**right**).

The more scanning artifacts are spread, the more the PA of the object is affected. This can observed with the parameter *MD of FP points*. The *MD of FP points* values are shorter for the *Lab RGB* than for the *Room RGB* dataset. This is also true for the *SD of FP points*, which is larger than 600 mm for the *Room RGB* dataset. The geometry of the class segments

becomes describable via the *SD and MD of FP points*. Based on the parameters PP and PA, it could be assumed that all point clouds are well-suited as a model. However, this is not the case due to the high deviations caused by scanning artifacts and overlapping segments in the *PCCT* datasets. The *SD of FP points* can describe, without visualization or human interpretation of the point cloud, that a point cloud is suitable or not as a model. A point cloud can be advantageous as a basis for modeling if the *MD of FP points* is large and the *SD of FP points* is small. These observations indicate isolated outliers. With SD and MD *of FP points* in combination, the quality of semantic point cloud segments can also be described geometrically.

The semantic accuracy can only be determined if the dataset-specific CD was used, since there is no general one. For these descriptions, the CD in Appendix A is applied. For all *Lab* and *Room* datasets, the CD was applied during all annotations. For other datasets, the respective CD of the dataset must be used. For our examples, the used CD is hierarchical, as can be seen in Table 14. In the CD, there are two semantic levels, which are filled out. Moreover, the class segments according to the CD are present or can be created by merging sub-classes into one super-class. Proof of the correct semantic class can be obtained by comparison with a reference or a visual inspection, as in Figure 8. For the example datasets, the furniture class cannot be seen directly, but it can be formed from the table and chair classes. One analytical strategy may be looking only at the most detailed classes. Since the furniture class is not directly present in our datasets, this class is not examinable and the parameter P7.3 is set to *no* in Table 14.

Table 14. Calculated and determined values for precision and semantic accuracy.

P. no.	Parameter Name	Lab RGB	Lab I	Lab R	Room RGB	Room I	Room R
P6	**Precision**						
P6.1	PP *floor*	99.7%	99.8%	99.8%	99.6%	99.6%	99.9%
P6.1	PP *chair*	78.5%	78.2%	95.8%	77.9%	67.3%	95.5%
P6.1	PP *table*	92.1%	93.1%	98.2%	93.2%	93.6%	97.5%
P6.1	PP *scan. artif.*	53.4%	52.6%	95.4%	61.3%	30.8%	98.0%
P6.2	PA *floor*	99.8%	100.0%	100.0%	99.5%	99.5%	99.5%
P6.2	PA *chair*	91.7%	91.6%	96.7%	90.4%	90.8%	95.7%
P6.2	PA *table*	96.8%	97.6%	98.8%	87.7%	87.6%	97.3%
		mm	mm	mm	mm	mm	mm
P6.3	MD FP pts *floor*	249	666	83	131	884	92
P6.3	MD FP pts *chair*	1278	1244	48	1699	1485	55
P6.3	MD FP pts *table*	1237	1558	53	1906	1967	53
		mm	mm	mm	mm	mm	mm
P6.4	SD FP pts. *floor*	42	87	38	61	161	24
P6.4	SD FP pts. *chair*	151	152	9	646	591	16
P6.4	SD FP pts. *table*	207	214	14	279	443	14
P7	**Semantic Accuracy**						
P7.1	CD applied	yes	yes	yes	yes	yes	yes
P7.2	Hier. CD	yes	yes	yes	yes	yes	yes
P7.3	*floor* used	yes	yes	yes	yes	yes	yes
P7.3	*furniture* used	no	no	no	no	no	no
P7.3	*chair* used	yes	yes	yes	yes	yes	yes
P7.3	*table* used	yes	yes	yes	yes	yes	yes
P7.3	*scan. artif.* used	yes	yes	yes	yes	yes	yes

4.1.4. Descriptive Use for Multiple Annotations

Regarding the research question, the individual annotation performance is also of interest. To investigate this aspect, 47 independent segmentations from nine different annotators are used for two point clouds. The metadata of the point cloud do not change due to

the individual annotations, so only eight parameters describe the annotation differences. These are ATR, RP, RA, PP, PA, *MD of FP points*, *SD of FP points* and *'class' used*.

The processing time is determined in relation to the maximum time required and is presented for each annotation in Table A2 in Appendix B. An analysis of the individual segmentations shows that the largest differences in processing time occur for *Recap*. The fastest annotation has been performed with only 20% of the maximum duration for *Lab* resp. with 15% for *Room* by *PCCT*. For the semantic segmentations with the *PCCT*, the segmentation duration varies by 5% for *Lab I*, by 10% for *Lab RGB*, by 31% for *Room I* and by 39% for *Room RGB*. It can be seen from the values in Table A2 that the processing time is more consistent with *PCCT* than with *Recap* for different annotators. The longest semantic segmentation with PCCT was 38% faster than with Recap. Thus, *PCCT* has the advantage of a shorter processing time and better planning capability for tasks.

Further differences for the individual annotations can be found for the characteristics of correctness and precision. The parameters RP and PP are calculated for each segmentation (Tables A3–A8 in the Appendix C). Based on the small variation in all values for RP and PP for the floor class, it can be concluded that this class can be segmented very reliably, correctly and precisely using a geometry fit.

For the table and chair classes, the individual results are different. The correctness and the precision vary strongly. For *Recap*, the minimum RP is 49.6% and the maximum is 99.8%. The lowest PP value is 87.5% and varies up to 14%. It can be seen that the reliability for chairs and tables decreases, because different annotations reach different accuracies. The class with the lowest correctness is the scanning artifacts class (RP of max. 80.1%). The worst annotation for scanning artifacts with *Recap* contains only 42.3% TP points. Similar trends can be seen for the datasets processed with *PCCT*, but these are even lower in terms of precision and correctness.

To investigate whether the different results in a multiple segmentation occur by random or whether there is a systematic effect, we tested whether the set of the RP and the PP per class is *normally* or *t* distributed. The hypothesis is that the RP or the PP is *normally* distributed around an expected (average) value per class; thus, the annotation performance would then also be *normally* distributed. Random differences would be describable in this way. The *Kolmogorov–Smirnov test* [100] was used to test this hypothesis.

It was found that, for most classes of the *Room* datasets, the RP and PP are *normally* distributed. For the smaller *Lab* dataset, no *normal* distribution could be observed. The hypothesis can therefore not be confirmed. A possible reason for the different distributions could be that the larger point cloud has more random segmentation errors than the smaller point cloud, which is reflected in the parameters. In the small point cloud, the operator is more focused and the assignments are less ambiguous. This observation is supported by the fact that the RP and PP of the *Lab* datasets are predominantly higher.

The parameters *MD of FP points*, *SD of FP points*, PA and RA behave in a similar way to RP and PP, so these will not be discussed further. The parameter *class used* must be tested before joining to avoid gross errors in the joined point cloud. This can be tested during the joining by allowing only certain classes and excluding segmented point clouds that contain other classes.

4.1.5. Summary of the Descriptive Use

The description from Sections 4.1.1–4.1.3 focuses on comparing the tools and how they perform differently for smaller and larger point clouds. The basis of the investigation for each tool was a joined point cloud, which is free of individual segmentation patterns. It can be concluded that, with the quality model, semantic point clouds can be described for a comparison. Without further knowledge about the point cloud or the segmentation tool, an analysis of the point cloud can be performed based on 23 parameters. The quality model is holistic and does not only refer to parameters for correctness and precision, such as in [13,57]. *Recap* is more suitable than the PCCT for the outlined applications. Nonetheless, with the appropriate settings for the automatic segmentation, the PCCT is

more efficient. The separation of objects and scanning artifacts has proven to be the main problem. Based on the analysis process, and in connection with the developed tools, it is possible to investigate other segmentation tools.

The second part of the research question was discussed in Section 4.1.4. It can be noted that the processing time and the achieved accuracy are user-specific. There is no common relationship between long processing time and higher accuracy. However, it can be observed that, with *Recap*, a longer processing time leads to more accurate results in most cases. With *PCCT*, the processing time is, on average, 42% shorter. The influence of the user is noticeably large when using *Recap*. This can be seen in Table A4. For the same point cloud and tool, differences of up to 18% (PP) for object classes occur. This observation confirms the hypothesis that multiple processing is necessary, in order to allow a realistic evaluation of the quality of the point cloud.

4.2. Quality Model to Evaluate Semantic Point Clouds

The description of the semantic points from the previous Section 4.1 is the basis for an evaluation of a semantic point cloud. Due to the large number of semantic point clouds available on the web, it is difficult to obtain an overview of which point cloud is suitable for which application. The quality model, with its parameters, provides a framework for the comparison and selection of datasets. The parameters can be used to evaluate the characteristics of the point cloud in terms of metadata, geometry and semantics. Thus, the point clouds that do not meet the important criteria of an application can be excluded. In the following, the quality parameters for almost all datasets from Section 2.3 were researched. The research results were summarized in an Excel database. A threshold set and query functions were added. For the used thresholds, it can be queried whether they are met, not met or unknown. For the example semantic point cloud as a model, the query result is shown in Figure A1 of Appendix D. The public datasets in the database are extended by the datasets of the point clouds *Lab* and *Room*. For our own datasets, it is ensured that all parameters are known.

The collection of datasets shows that most of the metadata for the point clouds are sufficiently documented or can be determined from the datasets. Thereby, implicit parameters are derived. For example, a class definition is present, even if this is not written down, and can only logically be derived by an application or from the point cloud itself. In addition, it is concluded that at least one semantic segmentation took place. Therefore, the parameter NoS is assumed to be 1, if nothing else is found. Parameters concerning the size of the dataset, the file format and the data model can usually be taken from publications, web documentation or directly from the dataset.

The correctness and precision parameters are unknown for all external datasets. This is a central weakness of existing practice in dealing with datasets provided as training data or for modeling. This work tackles the problem by providing the quality model. The model should attempt to encourage the evaluation of published datasets (at least in part) for geo-semantic accuracy. This kind of evaluation is standard for automatic semantic segmentations in almost all publications. Since most automatic ML methods learn from human-annotated datasets that are not evaluated, these methods "learn" possible errors in the data. Thus, learning is done with a GT dataset, which is not always a true representation of reality. It is only the reality as seen (most of the time) by one annotator. In the end, only a relative evaluation of ML procedures is possible with currently available datasets.

The use of the Excel database does not aim to determine exactly one dataset for which all parameters are fulfilled. It should rather be an aid with which a selection can be made. Not all parameters are always relevant for all applications and can therefore be disregarded. A possible use of the quality model is now presented for the example application from above.

4.2.1. Point Cloud as Model

The semantic point cloud as a model is usually useful for an application for which the capturing sensor properties are well known and the point cloud has to be semantically segmented at least once. The data model and the abstract model have to be known. The parameters of the quality characteristics of availability, process reliability, completeness, consistency and semantic accuracy must be fulfilled. The flowing example is a visualization of the floor, table and chair classes in CC.

The parameters CE and ATR have no meaning in this example, since no comparison of procedures is queried with regard to duration or training. The quality characteristics of correctness and precision are determined by one or more semantic segmentations, which always contain uncertainties. Thus, these parameters should never be set as 100%. Holes in the point cloud lower the correctness. Depending on how these holes occur and what additional information is available, the correctness can play a minor role. For the visualization, it is important that as few FP points as possible are present in the classes. This means that the precision must be high. The parameters RA and PA are favored over RP and PP in this application, since non-uniform density can be expected. Based on these considerations, we chose to set the thresholds for parameter RA to 70% and for the parameter PA to 80%. The scanning artifacts class is not considered, because it contains no object information. In addition, if PA is satisfied, the thresholds for *SD of FP points* and *MD of FP points* must be set to low values. Here, we suggest 50 mm as the threshold for *SD of FP points* and 100 mm as the threshold for *SD of FP points*. These limits vary from application to application. For a visual analysis of an indoor scene, our suggestions are sufficient to recognize objects such as tables and chairs.

The semantic conditions are fulfilled for the datasets *SceneNN*, *S3DSP* and *ScanNet* and our own datasets, *Lab* and *Room*. The *ScanNet* dataset is not available as a point cloud and therefore does not correspond to the research question. *SceneNN* and *S3DSP* are available in the appropriate file format for CC and have the necessary features (x, y, z-coordinates and semantic label). For an exclusive visualization, no restrictions of use are present. Subject to the unknown parameters, the datasets can be used for the example task.

For our own datasets, the semantic and all relevant formal constraints are satisfied (Section 4.1). The RA parameter for correctness and the PA parameter are satisfied for all classes as well, but the datasets processed with the *PCCT* do not meet the thresholds for *SD of FP points* and *MD of FP points*. The *PCCT* dataset cannot be used for the visualization. The *Recap* point clouds for *Lab* and *Room* meet the specifications and can be used. This is shown in Figure 15 for the point cloud *Room R*.

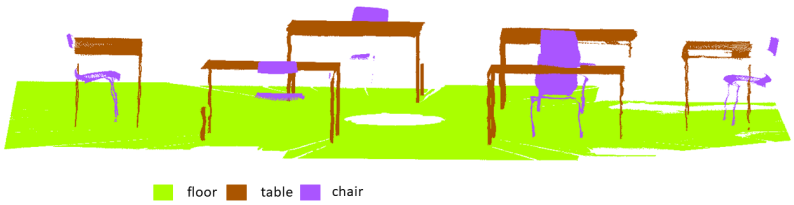

Figure 15. Semantic point cloud consisting of the floor, table and chair classes for visualization of a real room.

4.2.2. Point Cloud as a Basis for Modeling

A similar procedure as in Section 4.2.1 can be followed when using a semantic point cloud as the basis for modeling. The semantic parameters must also be fulfilled to model the needed classes. The semantic and geometric characteristics of the point clouds must be available. The file formats must be compatible with the modeling software. Most point clouds are available in open or open-source file formats that can be loaded with most modeling software, such as *PointCab* (https://pointcab-software.com/en/ accessed on 15 December 2021). However, this is not always the case, as popular modeling programs, such as *Autodesk Revit* (https://www.autodesk.de/products/revit/ accessed on 15 December

2021), only support their proprietary file formats. A software that supports open or open-source file formats should be used.

Due to the chosen semantic and formal target values, only the datasets *SceneNN* and *S3DSP*, as well as *Lab* and *Room*, can be considered for the example. Since no information is available for *SceneNN* and *S3DSP* regarding correctness and precision, these parameters cannot be evaluated. For our own datasets, *Lab* and *Room*, there are parameter values available, which are used to evaluate correctness and precision. As before, correctness is less important than precision as holes and incomplete edges can be closed or completed associatively when modeling with parametric geometry objects. In modeling by triangulation, holes can be closed up to a certain size. Thus, the threshold for correctness can be lowered to, e.g., RA 60% and complete modeling can still be achieved. The precision has higher relevance, because objects are mostly enlarged. The threshold for PA should remain at 80%. The thresholds for *SD* and *MD of the FP points* now have additional relevance as before. Distant single points are usually excluded automatically by the knowledge of the modeler, so this parameter can be very large (e.g., 2000 mm). More important is the *SD of the FP points*, which should remain at 50 mm. The choice of the threshold must also be customized for the task in question. For modeling objects using a model catalog or in LoD 100 or LoD 200 BIM applications, the proposed thresholds are sufficient. Due to the chosen threshold, only the two *Recap* datasets are available.

4.2.3. Point Cloud as Training Data

The third example is to use point clouds as information carriers to train data-based algorithms. For this purpose, the scanning artifacts class is necessary, in addition to the object classes from above. For many semantic indoor datasets, the scanning artifacts class or a comparable class for disturbances/noise is not included. For most outdoor datasets, not all indoor object classes are available.

Only our own datasets are considered in the following. The parameter NoP must be fulfilled, so that enough data for training and evaluation are available. A dataset with only 2 million points is too small for training. The training algorithm parameters will likely lead to unreliable and inaccurate results for other unknown point clouds. The target value for the NoP is set to 5 million points and at least three independent semantic segmentations are considered necessary to verify the knowledge in the data, even if no GT data of a higher accuracy level are available. The parameter ATR has the function of identifying, in the case of a large number of operations, the operators that work particularly fast. For example, these workers could be favored over the slower ones for further work.

The correctness and the precision for this work are equally important, because the method to be trained should learn the optimal handling of the data. Here, the points are the relevant input variables, which is why the RP and PP are used. The suggested thresholds are 75% for scanning artifacts and 80% for objects. It is expected that objects are segmented more distinctly and an interpretation of the scanning artifacts is more difficult. The other geometric parameters should not exceed the limits for the applications described above, but they are of minor importance for this application.

4.2.4. Summary of the Evaluated Use

The three example applications show how a semantic point cloud can be evaluated with the quality model and how it can be decided whether the quality of a dataset is sufficient. It should should be emphasized that, with the parameters, an objective evaluation is possible, even if the relevance of the individual parameters is different in the respective application. The presented applications and used thresholds are only examples, based on our experience.

5. Conclusions and Outlook

Semantic 3D point clouds play a crucial role in the context of the digitization of working environments. A representation of reality as a detailed point cloud or in the

form of a derived model is a fundamental component in many planning and management processes for buildings. Bringing semantic information into a geometric model is the next major step towards the automation of planning and decision making. Integrating the semantics of objects as additional information into a point cloud is a necessary and challenging task that must be solved. The semantics of the point cloud must be describable in terms of resolution, correctness and precision. This requires additional metadata about the point cloud and the previous processes. The requirements of an application must be compared with the actual characteristics and it must be tested whether the requirements are fulfilled.

The quality characteristics of a point cloud can be described by a quality model. For the holistic description of a semantic point cloud, a model based on seven characteristics was deemed to be suitable, offering the user the possibility to describe, compare and evaluate their own as well as third-party point clouds. In order to describe the quality of semantic point clouds with a manageable number of parameters, a quality model was created and tested in this work. The choice of parameters was based on the underlying process, as well as on the abstract model and the data model.

The holistic quality model for semantic point clouds focused on the characteristics of semantic segmentation; the characteristics of geometric creation must also be taken into account. Crucial for the semantic segmentation are the accuracy and reliability with which a point cloud was split into semantic segments. In particular, the human influences on the GT point clouds are usually not considered. The initial semantic knowledge in a GT point cloud is always given by a human. The quality of the knowledge is a variable quantity. It depends on the motivation, training, perception and carefulness of the annotator. One way to keep these individual influences low is to use multiple independent annotators and a unique CD, and to train the annotators well. The use of different segmentation tools, as well as the degree of individualization, have a measurable impact on the final point cloud. The more individualization a tool allows, the better a single semantic segmentation can be. However, this has the disadvantage that the segmentation performance can vary.

The created quality model allows the comparison of publicly available semantic point cloud datasets. The analysis of a selection of publicly available point clouds has shown that, in particular, parameters for the GT correctness and GT precision are usually not provided and therefore a comparison is not possible. This is a central weakness, which has to be addressed in the current practice so that realistic semantics can be represented in a point cloud. Our quality model contributes to the improvement of GT point clouds.

In future, a distinction of the general model is necessary and an adaptation to data-based algorithms is to be recommended. The current quality model is only designed for indoor applications due to the complexity of the semantic environment and must be adapted for outdoor applications. It is conceivable that the data-based algorithms can be understood even better if the characteristics of the input data (point cloud) are described. Based on the determined characteristics of the input data and algorithm response, an objective performance comparison can be achieved.

Author Contributions: Conceptualization, E.B.; methodology, E.B.; software, E.B.; validation, E.B.; formal analysis, E.B.; investigation, E.B.; resources, E.B.; data curation, E.B.; writing—original draft preparation, E.B.; writing—review and editing, E.B. and H.S.; visualization, E.B.; supervision, H.S.; project administration, E.B.; funding acquisition, E.B.and H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The Excel database with 18 point clouds can be found at https://github.com/eb17/Quality-check-of-point-cloud-data-sets (accessed on 15 December 2021). Based on the database, the datasets can be selected according to usability. The developed quality model is implemented in this Excel table.

Acknowledgments: Many thanks to the annotators: Clemens Semmelroth, Stefanie Stand, Annette Scheider, Günter Eppinger, Sarah Lange, Friedrike Köpke, Mona Lütjens and Cigdem Askar. Special

thanks to Stefanie for the support during the recording, to Clemens for the support during the evaluation and to Annette for proofreading.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Class Definition

The CD for the examples is shown in Table A1. A level is assigned to each class. A super-class is always fully divided into sub-classes. The class description is kept brief and provides relations between the classes.

Table A1. Class definition.

Level	Class	Definition
L0	furniture	*Furniture* includes objects that have contact with the floor and stand in the room. Objects that do not belong to the class *chair* or *table* cannot be *furniture*. The class can be further subdivided.
L1	table	The *table* class consists of all the points that describe/contain the table legs, the lower frame of the table, the *table* top and the adjustable feet.
L1	chair	The class *chair* consists of all the points that describe the seat, backrest, tubular frame and rubber feet.
L0	floor	The *floor* class consists of all points describing the flat floor and small edges and floor inlets (maintenance flaps). The *floor* can be considered a plane with a deviation of 50 mm.
L0	scanning artifacts	The *scanning artifacts* class consists of all points that describe objects lying on the ground—for example, cables. Furthermore, this includes all points that are caused by measurement errors (phantom points), reflection of the objects and gap closures due to the evaluation software. Multiple reflections can also occur.

Appendix B. Time Required for Semantic Segmentation

Table A2 shows the actual processing time in relation to the maximum processing time. The maximum time required in minutes for each point cloud is equal to 100%. The percentages can only be compared within one point cloud. The data are only valid for the comparison of the example described in Section 4.1.4. For other investigations, the ordinary times in minutes must be used.

Table A2. Actual processing time in relation to the maximum processing time.

	Lab			*Room*		
No.	**RGB**	**I**	**R**	**RGB**	**I**	**R**
1	12.5%	12.5%	41.7%	41.2%	46.4%	49.0%
2	14.2%	11.7%	64.2%	27.8%	33.5%	34.0%
3	12.5%	12.5%	100%	-	-	-
4	10.0%	10.8%	94.2%	32.5%	42.3%	100.0%
5	9.2%	-	35.8%	33.5%	51.5%	46.4%
6	19.2%	-	37.5%	23.2%	30.9%	30.9%
7	-	12.5%	75.0%	61.9%	61.9%	61,9%
8	-	15,8%	25.0%	46.1%	46.4%	15.5%
9	10.8%	13.3%	20.8%	41.2%	43.8%	51.5%
Avg.	13.0%	12.7%	54.8%	38.0%	44.6%	48.6%

Appendix C. Correctness and Precision for Multiple Annotations

Tables A3–A5 show the parameter PP for all classes of the CD from Appendix A. Tables A6–A8 contain the parameter RP of all annotations.

Table A3. Precision for the datasets *Lab* and *Room*, semantically segmented by *Recap*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	99.8%	95.9%	98.7%	91.7%	99.9%	96.4%	97.6%	89.7%
2	99.6%	91.7%	96.0%	97.8%	77.0%	99.8%	94.0%	98.5%
3	100.0%	85.9%	97.3%	17.7%	-	-	-	-
4	99.7%	97.9%	99.5%	88.2%	99.9%	95.6%	97.4%	87.5%
5	100.0%	93.4%	99.4%	47.4%	99.8%	94.1%	96.9%	77.5%
6	99.9%	97.0%	85.7%	81.1%	99.8%	94.3%	97.6%	33.9%
7	100.0%	96.6%	99.0%	74.3%	99.2%	94.7%	96.2%	82.7%
8	99.8%	99.2%	99.9%	76.4%	99.9%	92.7%	96.7%	90.4%
9	99.5%	99.8%	99.9%	77.6%	99.5%	90.9%	97.0%	95.0%

Table A4. Precision for the datasets *Lab* and *Room*, semantically segmented by *PCCT RGB*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	99.7%	93.9%	86.2%	40.5%	99.6%	77.9%	93.7%	51.3%
2	99.7%	88.5%	86.9%	41.7%	99.6%	79.1%	92.8%	43.7%
3	99.7%	83.3%	77.1%	29.9%	-	-	-	-
4	99.7%	94.6%	86.6%	41.0%	99.7%	80.9%	93.4%	61.4%
5	99.7%	91.6%	83.2%	47.4%	99.7%	82.7%	95.2%	41.7%
6	99.7%	84.2%	95.2%	24.0%	99.7%	77.3%	88.9%	43.4%
7	-	-	-	-	99.3%	74.0%	93.5%	48.5%
8	-	-	-	-	99.7%	78.4%	93.9%	56.0%
9	99.7%	93.9%	80.8%	38.1%	99.5%	80.4%	93.6%	49.9%

Table A5. Precision for the datasets *Lab* and *Room*, semantically segmented by *PCCT I*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	99.7%	93.8%	86.8%	38.5%	99.7%	73.9%	95.3%	26.7%
2	99.7%	93.8%	85.3%	39.8%	99.4%	82.1%	89.8%	42.0%
3	99.7%	81.9%	82.2%	21.7%	-	-	-	-
4	99.7%	93.6%	89.0%	37.8%	99.7%	77.7%	90.3%	33.3%
5	-	-	-	-	99.7%	82.7%	95.2%	41.7%
6	-	-	-	-	99.6%	76.8%	91.5%	30.1%
7	99.8%	77.0%	75.8%	37.7%	99.3%	70.7%	91.0%	20.6%
8	99.7%	94.2%	87.5%	38.0%	99.7%	77.3%	92.6%	35.5%
9	99.7%	83.5%	90.9%	35.5%	99.2%	75.7%	92.7%	30.8%

Table A6. Recall for the datasets *Lab* and *Room*, semantically segmented by *Recap*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	100.0%	98.9%	99.8%	62.0%	100.0%	98.0%	99.2%	75.9%
2	100.0%	99.9%	99.8%	46.4%	100.0%	97.6%	96.5%	76.2%
3	99.5 %	90.7%	94.0%	80.1%	-	-	-	-
4	100.0%	98.3%	99.7%	67.0%	100.0%	99.0%	99.4%	74.0%
5	99.6%	98.5%	99.0%	69.8%	99.9%	99.1%	97.1%	66.7%
6	99.9%	70.4%	99.8%	74.8%	100.0%	49.6%	98.7%	71.1%
7	99.8%	97.8%	99.8%	79.1%	100.0%	97.9%	98.3%	42.3%
8	100.0%	96.4%	99.2%	77.1%	99.9%	99.3%	99.6%	64.4%
9	100.0%	96.0%	99.1%	57.2%	100.0%	99.4%	99.5%	51.1%

Table A7. Recall for the datasets *Lab* and *Room*, semantically segmented by *PCCT RGB*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	99.9%	80.5%	96.0%	26.3%	99.8%	83.0%	88.6%	54.4%
2	99.9%	84.6%	95.4%	19.3%	99.8%	68.2%	92.7%	46.6%
3	99.9%	83.6%	95.6%	20.4%	-	-	-	-
4	99.9%	84.8%	95.6%	22.2%	99.8%	80.4%	94.2%	55.9%
5	99.8%	84.6%	95.2%	18.1%	99.8%	50.8%	91.7%	64.0%
6	99.7%	93.5%	87.3%	32.6%	99.8%	69.7%	96.2%	29.2%
7	-	-	-	-	99.8%	68.2%	91.9%	44.2%
8	-	-	-	-	99.8%	87.1%	91.5%	52.7%
9	99.9%	84.8%	95.0%	15.2%	99.8%	77.2%	91.2%	49.4%

Table A8. Recall for the datasets *Lab* and *Room*, semantically segmented by *PCCT I*.

	Lab				*Room*			
No.	**Floor**	**Chair**	**Table**	**Scan. Artif.**	**Floor**	**Chair**	**Table**	**Scan. Artif.**
1	99.9%	83.8%	96.1%	22.0%	99.8%	46.2%	88.5%	54.7%
2	99.9%	83.2%	96.6%	20.0%	99.8%	57.1%	90.6%	45.3%
3	99.8%	85.1%	90.0%	29.6%	-	-	-	-
4	99.9%	84.3%	95.8%	23.6%	99.8%	80.4%	86.4%	39.0%
5	-	-	-	-	99.8%	74.7%	90.6%	52.0%
6	-	-	-	-	99.8%	65.4%	87.7%	38.0%
7	99.0%	84.0%	95.5%	15.1%	99.8%	56.4%	85.8%	26.5%
8	99.9%	83.7%	96.2%	21.6%	99.8%	81.4%	85.8%	43.5%
9	99.9%	87.6%	95.5%	24.3%	99.8%	61.4%	86.2%	37.6%

Appendix D. Point cloud dataset comparison

Results

t = parameter is true
nt = parameter is not true
- = not enough information available

		Room RGB (PCCT)	Room I (PCCT)	Room R (Recap)	Lab RGB (PCCT)	Lab I (PCCT)	Lab R (Recap)	Paris-Lille 3D	Semantic3D	SemanticKITTI	MLS1 TUM City Campus	Toronto3D	CSPC-Database	BIPC-Database**	SceneNN	S3DSP	ScanNet**	Matterport3D**	ScanObjectNN
P1.1	Class definition exists	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
P1.2	Number of points	t	t	t	t	t	t	t	t	t	t	t	-	t	nt	-	-	t	
P1.3	Area size	t	t	t	t	t	t	-	-	-	t	-	-	-	t	t	t	t	t
P1.4	Object characteristics in	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P1.5	Object characteristics out	t	t	t	t	t	t	t	t	t	-	t	t	-	t	t	t	t	t
P1.6	File format output	t	t	t	t	t	t	nt	t	nt	nt	nt	-	nt	nt	-	nt	t	-
P1.7	Use restriction	t	t	t	t	t	t	t	t	nt	nt	t	nt	nt	nt	nt	nt	nt	nt
P2.1	Number of segmentation	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
P2.2	Average time required	t	t	t	t	t	t	t	t	-	t	t	t	t	t	t	t	t	t
P3.1	Semantic segmentation rate	t	t	nt	t	t	nt	t	t	t	t	t	t	t	t	t	t	t	t
P3.2	Number of classes	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
P4.1	Geometric consistency of geometry (x)	t	t	t	t	t	t	t	-	t	-	t	-	-	-	-	-	-	-
P4.1	Geometric consistency of geometry (y)	t	t	t	t	t	t	t	-	-	-	t	-	-	-	-	-	-	-
P4.1	Geometric consistency of geometry (z)	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P4.2	Spectral consistency of spectral RGB	t	t	t	t	t	t	t	t	t	t	t	-	-	-	-	-	-	-
P4.3	Spectral consistency of spectral I	t	t	t	t	t	t	t	t	t	t	t	-	-	-	-	-	-	-
P4.4	Class equality	t	t	t	t	t	t	-	-	-	-	t	t	-	-	-	-	-	-
P5.1	Recall points floor*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.1	Recall points chair*	t	nt	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.1	Recall points table*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.1	Recall points scan artifacts*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.2	Recall area floor*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.2	Recall area chair*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P5.2	Recall area table*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.1	Precision points floor*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.1	Precision points chair*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.1	Precision points table*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.1	Precision points scan artifacts*	t	nt	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.2	Presision area floor*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.2	Presision area chair*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.2	Presision area table*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.3	Max. derivation FP points floor*	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.3	Max. derivation FP points chair*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.3	Max. derivation FP points table*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.4	Std. derivation FP points floor*	nt	nt	t	t	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.4	Std. derivation FP points chair*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P6.4	Std. derivation FP points table*	nt	nt	t	nt	nt	t	-	-	-	-	-	-	-	-	-	-	-	-
P7.1	Class definition applied	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t	t
P7.2	Hierarchical class definition	t	t	t	t	t	t	t	nt	t	nt	nt	t	t	t	nt	nt	nt	nt
P7.3	Floor used*	t	t	t	t	t	t	t	nt	nt	nt	nt	nt	nt	t	t	t	t	nt
P7.3	Chair used*	t	t	t	t	t	t	nt	nt	nt	nt	nt	nt	nt	t	t	t	t	t
P7.3	Table used*	t	t	t	t	t	t	nt	nt	nt	nt	nt	nt	nt	t	t	t	t	t
P7.3	Scan artifacts used*	t	t	t	t	t	t	nt	t	t	t	nt	t	nt	nt	nt	nt	t	nt
P7.3	Furniture used*	nt	nt	nt	nt	nt	nt	nt	nt	nt	nt	nt	nt	nt	t	nt	nt	nt	nt

* Parameters to be adjusted depending on the number of classes.
** Data is mesh or image set

Figure A1. Point cloud dataset comparison. Example: Point cloud as model.

References

1. Balangé, L.; Zhang, L.; Schwieger, V. First Step Towards the Technical Quality Concept for Integrative Computational Design and Construction. In *Springer Proceedings in Earth and Environmental Sciences*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 118–127. https://doi.org/10.1007/978-3-030-51953-7_10.
2. Frangez, V.; Salido-Monzú, D.; Wieser, A. Depth-Camera-Based In-line Evaluation of Surface Geometry and Material Classification For Robotic Spraying. In Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC), Kitakyushu, Japan, 27–28 October 2020; International Association for Automation and Robotics in Construction (IAARC): Berlin, Germany, 2020. https://doi.org/10.22260/isarc2020/0097.
3. Placzek, G.; Brohmann, L.; Mawas, K.; Schwerdtner, P.; Hack, N.; Maboudi, M.; Gerke, M. A Lean-based Production Approach for Shotcrete 3D Printed Concrete Components. In Proceedings of the 38th International Symposium on Automation and Robotics in Construction (ISARC), Dubai, United Arab Emirates, 2–5 November 2021; International Association for Automation and Robotics in Construction (IAARC): Berlin, Germany, 2021. https://doi.org/10.22260/isarc2021/0110.
4. Westphal, T.; Herrmann, E.M. (Eds). *Building Information Modeling I Management Band 2*; Detail Business Information GmbH: München, Germany, 2018. https://doi.org/10.11129/9783955534073.
5. Hellweg, N.; Schuldt, C.; Shoushtari, H.; Sternberg, H. Potenziale für Anwendungsfälle des Facility Managements von Gebäuden durch die Nutzung von Bauwerksinformationsmodellen als Datengrundlage für Location-Based Services im 5G-Netz. In *21. Internationale Geodätische Woche Obergurgl 2021*; Wichmann Herbert: Berlin/Offenbach, Germany 2021.

6. Willemsen, T. *Fusionsalgorithmus zur Autonomen Positionsschätzung im Gebäude, Basierend auf MEMS-Inertialsensoren im Smartphone. Phdthesis*; HafenCity Universität Hamburg: Hamburg, Germany, 2016.
7. Schuldt, C.; Shoushtari, H.; Hellweg, N.; Sternberg, H. L5IN: Overview of an Indoor Navigation Pilot Project. *Remote Sens.* **2021**, *13*, 624. https://doi.org/10.3390/rs13040624.
8. Grieves, M.; Vickers, J. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In *Transdisciplinary Perspectives on Complex Systems*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 85–113. https://doi.org/10.1007/978-3-319-38756-7_4.
9. Maturana, D.; Scherer, S. VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ, Hamburg, Germany, 28 September–2 October 2015; IEEE: New York, NY, USA, 2015, pp. 922–928. https://doi.org/10.1109/iros.2015.7353481.
10. Hackel, T.; Wegner, J.D.W.; Schindler, K. Fast Semantic Segmentation of 3D Point Clouds with Strongly Varying Densit. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *3*, pp. 177–184. https://doi.org/10.5194/isprsannals-iii-3-177-2016.
11. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep Learning on Point Sets for 3d Classification and Segmentation. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 77–85. https://doi.org/10.1109/cvpr.2017.16.
12. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. Advances in Neural Information Processing Systems, 2017; pp. 5099–5108. Available online: https://arxiv.org/abs/1706.02413 (accessed on 15 December 2021).
13. Behley, J.; Garbade, M.; Milioto, A.; Quenzel, J.; Behnke, S.; Stachniss, C.; Gall, J. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October to 2 November 2019.
14. Zhu, J.; Gehrung, J.; Huang, R.; Borgmann, B.; Sun, Z.; Hoegner, L.; Hebel, M.; Xu, Y.; Stilla, U. TUM-MLS-2016: An Annotated Mobile LiDAR Dataset of the TUM City Campus for Semantic Point Cloud Interpretation in Urban Areas. *Remote Sens.* **2020**, *12*, 1875. https://doi.org/10.3390/rs12111875.
15. Hackel, T.; Savinov, N.; Ladicky, L.; Wegner, J.D.; Schindler, K.; Pollefeys, M. Semantic3d.net: A New Large-scale Point Cloud Classification Benchmark. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *IV-1-W1*, 91–98. https://doi.org/10.5194/isprs-annals-iv-1-w1-91-2017.
16. Khoshelham, K.; Vilariño, L.D.; Peter, M.; Kang, Z.; Acharya, D. The ISPRS Benchmark on Indoor Modelling. *ISPRS- Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *XLII-2/W7*, pp. 367–372. https://doi.org/10.5194/isprs-archives-xlii-2-w7-367-2017.
17. Padilla, R.; Passos, W.L.; Dias, T.L.B.; Netto, S.L.; da Silva, E.A.B. A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics* **2021**, *10*, 279. https://doi.org/10.3390/electronics10030279.
18. Rangesh, A.; Trivedi, M.M. No Blind Spots: Full-Surround Multi-Object Tracking for Autonomous Vehicles using Cameras and LiDARs. *IEEE Trans. Intell. Veh.* **2018**, *4*, pp. 588–599. https://doi.org/10.1109/tiv.2019.2938110.
19. Liu, X.; Qi, C.R.; Guibas, L.J. FlowNet3D: Learning Scene Flow in 3D Point Clouds. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: New York, NY, USA, 2019. https://doi.org/10.1109/cvpr.2019.00062.
20. Wang, C.; Hou, S.; Wen, C.; Gong, Z.; Li, Q.; Sun, X.; Li, J. Semantic Line Framework-based Indoor Building Modeling Using Backpacked Laser Scanning Point Cloud. *ISPRS J. Photogramm. Remote Sens.* **2018**, *143*, 150–166. https://doi.org/10.1016/j.isprsjprs.2018.03.025.
21. Volk, R.; Luu, T.H.; Mueller-Roemer, J.S.; Sevilmis, N.; Schultmann, F. Deconstruction Project Planning of Existing Buildings Based on Automated Acquisition and Reconstruction of Building Information. *Autom. Constr.* **2018**, *91*, 226–245. https://doi.org/10.1016/j.autcon.2018.03.017.
22. Wang, C.; Dai, Y.; Elsheimy, N.; Wen, C.; Retscher, G.; Kang, Z.; Lingua, A. ISPRS Benchmark on Multisensory Indoor Mapping and Positioning. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *V-5-2020*, 117–123. https://doi.org/10.5194/isprs-annals-v-5-2020-117-2020.
23. Bello, S.A.; Yu, S.; Wang, C. Review: Deep Learning on 3d Point Clouds. *Remote Sens.* **2020**, *12*, 1729, https://doi.org/10.3390/rs12111729.
24. Liu, W.; Sun, J.; Li, W.; Hu, T.; Wang, P. Deep Learning on Point Clouds and Its Application: A Survey. *Sensors* **2019**, *19*, 4188. doi:10.3390/s19194188.
25. Xie, Y.; Tian, J.; Zhu, X.X. Linking Points With Labels in 3D: A Review of Point Cloud Semantic Segmentation. *IEEE Geosci. Remote Sens. Mag.* **2020**, *8*, 38–59. https://doi.org/10.1109/mgrs.2019.2937630.
26. Wang, X.; Zhou, B.; Shi, Y.; Chen, X.; Zhao, Q.; Xu, K. Shape2Motion: Joint Analysis of Motion Parts and Attributes from 3D Shapes. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: New York, NY, USA, 2019, https://doi.org/10.1109/cvpr.2019.00908.
27. Chang, A.X.; Funkhouser, T.; Guibas, L.; Hanrahan, P.; Huang, Q.; Li, Z.; Savarese, S.; Savva, M.; Song, S.; Su, H.; et al. ShapeNet: An Information-Rich 3D Model Repository. *arXiv* 2015, arXiv:1512.03012.
28. Omata, K.; Furuya, T.; Ohbuchi, R. Annotating 3D Models and their Parts via Deep Feature Embedding. In Proceedings of the 2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Shanghai, China, 8–12 July 2019; IEEE: New York, NY, USA, 2019. https://doi.org/10.1109/icmew.2019.00090.

29. Mo, K.; Guerrero, P.; Yi, L.; Su, H.; Wonka, P.; Mitra, N.; Guibas, L.J. StructureNet: Hierarchical Graph Networks for 3D Shape Generation. *ACM Trans. Graph.* **2019**, *38*, 1–19. https://doi.org/10.1145/3355089.3356527.
30. Luhmann, T.; Robson, S.; Kyle, S.; Boehm, J. *Close-Range Photogrammetry and 3D Imaging*; De Gruyter: Berlin, Germany, 2013. https://doi.org/10.1515/9783110302783.
31. Wasenmüller, O.; Stricker, D. Comparison of Kinect V1 and V2 Depth Images in Terms of Accuracy and Precision. In *Computer Vision—ACCV 2016 Workshops*; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 34–45. https://doi.org/10.1007/978-3-319-54427-4_3.
32. Tölgyessy, M.; Dekan, M.; Chovanec, Ľ.; Hubinský, P. Evaluation of the Azure Kinect and Its Comparison to Kinect V1 and Kinect V2. *Sensors* **2021**, *21*, 413. https://doi.org/10.3390/s21020413.
33. Schumann, O.; Hahn, M.; Dickmann, J.; Wohler, C. Semantic Segmentation on Radar Point Clouds. In Proceedings of the 2018 21st International Conference on Information Fusion FUSION, Cambridge, UK, 10–13 July 2018; IEEE: New York, NY, USA, 2018. https://doi.org/10.23919/icif.2018.8455344.
34. Qian, K.; He, Z.; Zhang, X. 3D Point Cloud Generation with Millimeter-Wave Radar. *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.* **2020**, *4*, 1–23. https://doi.org/10.1145/3432221.
35. Shults, R.; Levin, E.; Habibi, R.; Shenoy, S.; Honcheruk, O.; Hart, T.; An, Z. Capability of Matterport 3D Camera for Industria Archaeolog Sites Inventory. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-2/W11*, 1059–1064. https://doi.org/10.5194/isprs-archives-xlii-2-w11-1059-2019.
36. Sarbolandi, H.; Lefloch, D.; Kolb, A. Kinect Range Sensing: Structured-Light versus Time-of-Flight Kinect. *Comput. Vis. Image Underst.* **2015**, *139*, 1–20. https://doi.org/10.1016/j.cviu.2015.05.006.
37. Luhmann, T. *Nahbereichsphotogrammetrie Grundlagen-Methoden-Beispiele*; Wichmann: Berlin, Offenbach, 2018.
38. Freedman, B.; Shpunt, A.; Machline, M.; Arieli, Y. Depth Mapping Using Projected Patterns. Patent: US 2008/O2405O2A1, 3 October 2008.
39. Landau, M.J.; Choo, B.Y.; Beling, P.A. Simulating Kinect Infrared and Depth Images. *IEEE Trans. Cybern.* **2016**, *46*, 3018–3031. https://doi.org/10.1109/tcyb.2015.2494877.
40. Armeni, I.; Sener, O.; Zamir, A.R.; Jiang, H.; Brilakis, I.; Fischer, M.; Savarese, S. 3d Semantic Parsing of Large-scale Indoor Spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1534–1543. https://doi.org/10.1109/cvpr.2016.170.
41. Chang, A.; Dai, A.; Funkhouser, T.; Halber, M.; Nießner, M.; Savva, M.; Song, S.; Zeng, A.; Zhang, Y. Matterport3D: Learning from RGB-D Data in Indoor Environments. International Conference on 3D Vision (3DV), Qingdao, China, 10–12 October 2017; IEEE: New York, NY, USA, 2017. https://doi.org/10.1109/3dv.2017.00081.
42. Matterport. Matterport Pro 3D Camera Specifications. Available Online: https://support.matterport.com/s/articledetail?language=en_US&ardId=kA05d000001DX3DCAW (accessed on 23 September 2021).
43. Hansard, M.; Lee, S.; Choi, O.; Horaud, R. *Time-of-Flight Cameras*; Springer: London, UK, 2013. https://doi.org/10.1007/978-1-4471-4658-2.
44. Keller, F. *Entwicklung eines Forschungsorientierten Multi-Sensor-System zum Kinematischen Laserscannings Innerhalb von Gebäuden*. *Phdthesis*; HafenCity Universität Hamburg: Hamburg, Germany, 2015. https://doi.org/978-3844044171.
45. VelodyneLiDAR. Velodyne HDL-32E Data Sheet. Available Online: https://www.mapix.com/wp-content/uploads/2018/07/97-0038_Rev-M_-HDL-32E_Datasheet_Web.pdf (accessed on 24 June 2021).
46. Riegl. RIEGL VZ-400-Data Sheet. Available Online: www.riegl.com/uploads/tx_pxpriegldownloads/10_DataSheet_VZ-400_2017-06-14.pdf (accessed on 24 June 2021).
47. Lovas, T.; Hadzijanisz, K.; Papp, V.; Somogyi, A.J. Indoor Building Survey Assessment. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B1-2020*, pp. 251–257. https://doi.org/10.5194/isprs-archives-xliii-b1-2020-251-2020.
48. Kersten, T.P.; Lindstaedt, M.; Stange, M. Geometrische Genauigkeitsuntersuchungen aktueller terrestrischer Laserscanner im Labor und im Feld. *AVN* **2021**, *2*, 59–67.
49. ISO17123-9. *Optics and Optical Instruments. Field Procedures for Testing Geodetic and Surveying Instruments. Terrestrial Laser Scanners*; British Standards Institution: London, UK, 2018.
50. Kaartinen, H.; Hyyppä, J.; Kukko, A.; Jaakkola, A.; Hyyppä, H. Benchmarking the Performance of Mobile Laser Scanning Systems Using a Permanent Test Field. *Sensors* **2012**, *12*, 12814–12835. https://doi.org/10.3390/s120912814.
51. Wujanz, D.; Burger, M.; Tschirschwitz, F.; Nietzschmann, T.; Neitzel, F.; Kersten, T. Determination of Intensity-Based Stochastic Models for Terrestrial Laser Scanners Utilising 3D-Point Clouds. *Sensors* **2018**, *18*, 2187. https://doi.org/10.3390/s18072187.
52. Neuer, H. Qualitätsbetrachtungen zu TLS-Daten. *Qualitätssicherung geodätischer Mess-und Auswerteverfahren 2019. DVW-Arbeitskreis 3 Messmethoden und Systeme*; Wißner-Verlag: Augsburg, Germany, 2019; Volume 95, pp. 69–89.
53. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3d ShapeNets: A Deep Representation for Volumetric Shapes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; IEEE: New York, NY, USA, 2015. https://doi.org/10.1109/cvpr.2015.7298801.
54. Winiwarter, L.; Pena, A.M.E.; Weiser, H.; Anders, K.; Sánchez, J.M.; Searle, M.; Höfle, B. Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. *Remote Sens. Environ.* **2022**, *269*, 112772. https://doi.org/10.1016/j.rse.2021.112772.

55. Iqbal, J.; Xu, R.; Sun, S.; Li, C. Simulation of an Autonomous Mobile Robot for LiDAR-Based In-Field Phenotyping and Navigation. *Robotics* **2020**, *9*, 46. https://doi.org/10.3390/robotics9020046.
56. Hua, B.S.; Pham, Q.H.; Nguyen, D.T.; Tran, M.K.; Yu, L.F.; Yeung, S.K. SceneNN: A Scene Meshes Dataset with aNNotations. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; IEEE: New York, NY, USA, 2016. https://doi.org/10.1109/3dv.2016.18.
57. Dai, A.; Chang, A.X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. Scannet: Richly-annotated 3d Reconstructions of Indoor Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, US, 21–26 July 2017; IEEE: New York, NY, USA, 2017. https://doi.org/10.1109/cvpr.2017.261.
58. Uy, M.A.; Pham, Q.H.; Hua, B.S.; Nguyen, D.T.; Yeung, S.K. Revisiting Point Cloud Classification: A New Benchmark Dataset and Classification Model on Real-World Data. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; IEEE: New York, NY, USA, 2019. https://doi.org/10.1109/iccv.2019.00167.
59. CloudCompare. 3d Point Cloud and Mesh Processing Software Open-source Project. Version 2.12. Available Online: http://www.cloudcompare.org/ (accessed on 24 June 2021).
60. Felzenszwalb, P.F.; Huttenlocher, D.P. Efficient Graph-based Image Segmentation. *Int. J. Comput. Vis.* **2004**, *59*, 167–181. https://doi.org/10.1023/b:visi.0000022288.19776.77.
61. Nguyen, D.T.; Hua, B.S.; Yu, L.F.; Yeung, S.K. A Robust 3D-2D Interactive Tool for Scene Segmentation and Annotation. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 3005–3018, https://doi.org/10.1109/tvcg.2017.2772238.
62. Wada, K. labelme: Image Polygonal Annotation with Python. 2016. Available online: https://github.com/wkentaro/labelme (accessed on 15 December 2020).
63. Hossain, M.; Ma, T.; Watson, T.; Simmers, B.; Khan, J.; Jacobs, E.; Wang, L. Building Indoor Point Cloud Datasets with Object Annotation for Public Safety. In Proceedings of the 10th International Conference on Smart Cities and Green ICT Systems, Online, 28–30 April 2021; SciTePRESS—Science and Technology Publications: Setubal, Portugal, 2021. https://doi.org/10.5220/0010454400450056.
64. Roynard, X.; Deschaud, J.E.; Goulette, F. Paris-lille-3d: A Large and High-quality Ground-truth Urban Point Cloud Dataset for Automatic Segmentation and Classification. *Int. J. Robot. Res.* **2018**, *37*, 545–557. https://doi.org/10.1177/0278364918767506.
65. Tan, W.; Qin, N.; Ma, L.; Li, Y.; Du, J.; Cai, G.; Yang, K.; Li, J. Toronto-3D: A Large-scale Mobile LiDAR Dataset for Semantic Segmentation of Urban Roadways. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020. https://doi.org/10.1109/cvprw50498.2020.00109.
66. Tong, G.; Li, Y.; Chen, D.; Sun, Q.; Cao, W.; Xiang, G. CSPC-Dataset: New LiDAR Point Cloud Dataset and Benchmark for Large-Scale Scene Semantic Segmentation. *IEEE Access* **2020**, *8*, 87695–87718. https://doi.org/10.1109/access.2020.2992612.
67. Zimmer, W.; Rangesh, A.; Trivedi, M. 3D BAT: A Semi-Automatic, Web-based 3D Annotation Toolbox for Full-Surround, Multi-Modal Data Streams. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; IEEE: New York, NY, USA, 2019. https://doi.org/10.1109/ivs.2019.8814071.
68. Ibrahim, M.; Akhtar, N.; Wise, M.; Mian, A. Annotation Tool and Urban Dataset for 3D Point Cloud Semantic Segmentation. *IEEE Access* **2021**, *9*, 35984–35996. https://doi.org/10.1109/access.2021.3062547.
69. Wirth, F.; Quehl, J.; Ota, J.; Stiller, C. PointAtMe: Efficient 3D Point Cloud Labeling in Virtual Reality. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; IEEE: New York, NY, USA, 2019. https://doi.org/10.1109/ivs.2019.8814115.
70. Monica, R.; Aleotti, J.; Zillich, M.; Vincze, M. Multi-label Point Cloud Annotation by Selection of Sparse Control Points. In Proceedings of the 2017 International Conference on 3D Vision (3DV), Qingdao, China, 10–12 October 2017; IEEE: New York, NY, USA, 2017. https://doi.org/10.1109/3dv.2017.00042.
71. Autodesk-Recap. Youtube Channel. Available Online: http://https://www.youtube.com/user/autodeskrecap/ (accessed on 24 June 2021).
72. Barnefske, E.; Sternberg, H. PCCT: A Point Cloud Classification Tool To Create 3D Training Data To Adjust And Develop 3D ConvNet. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-2/W16*, 35–40. https://doi.org/10.5194/isprs-archives-xlii-2-w16-35-2019.
73. *ISO9000*; Quality Management Systems—Fundamentals and Vocabulary. ISO: Geneva, Switzerland, 2015.
74. *DIN55350*; Concepts for Quality Management and Statistics—Quality Management. DIN: Geneva, Switzerland, 2020.
75. *DIN18710*; Engineering Survey. DIN: Geneva, Switzerland, 2010.
76. Blankenbach, J., Bauaufnahme, Gebäudeerfassung und BIM. In *Ingenieurgeodäsie: Handbuch der Geodäsie, published by Willi Freeden and Reiner Rummel*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 23–53. https://doi.org/10.1007/978-3-662-47188-3_36.
77. Joos, G. Zur Qualität von objektstrukturierten Geodaten. Ph.D. Thesis, Universität der Bundeswehr München, Muenchen, Germany, 2000.
78. Scharwächter, T.; Enzweiler, M.; Franke, U.; Roth, S. Efficient Multi-cue Scene Segmentation. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 435–445. https://doi.org/10.1007/978-3-642-40602-7_46.
79. Miller, G.A.; Beckwith, R.; Fellbaum, C.; Gross, D.; Miller, K.J. Introduction to WordNet: An On-line Lexical Database. *Int. J. Lexicogr.* **1990**; *3*, 235–244. https://doi.org/10.1093/ijl/3.4.235.
80. buildingSMART. Industry Foundation Classes 4.0.2.1. Available Online: https://standards.buildingsmart.org (accessed on 24 June 2021).

81. BIM.Hamburg. *BIM-Leitfaden für die FHH Hamburg*; Technical Report; BIM: Hamburg, Germany, 2019.
82. Kaden, R.; Clemen, C.; Seuß, R.; Blankenbach, J.; Becker, R.; Eichhorn, A.; Donaubauer, A.; Gruber, U. Leitfaden Geodäsie und BIM. Techreport 2.1, DVW e.V. und Runder Tisch GIS e.V. 2020. Available Online: https://dvw.de/images/anhang/2757/leitfaden-geodaesie-und-bim2020onlineversion.pdf (accessed on 15 December 2021).
83. BIM-Forum. Level of Development Specification Part1 & Commentary. 2020. Available Online: https://bimforum.org/lod/ (accessed on 15 December 2021).
84. Günther, M.; Wiemann, T.; Albrecht, S.; Hertzberg, J. Model-based furniture recognition for building semantic object maps. *Artif. Intell.* **2017**, *247*, 336–351. https://doi.org/10.1016/j.artint.2014.12.007.
85. Wiltscho, T. Sichere Information durch infrastrukturgestützte Fahrerassistenzsysteme zur Steigerung der Verkehrssicherheit an Straßenknotenpunkten. Ph.D. Thesis, University Stuttgart, Stuttgart, Germany, 2004.
86. Torralba, A.; Efros, A.A. Unbiased Look at Dataset Bias. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; IEEE: New York, NY, USA, 2011. https://doi.org/10.1109/cvpr.2011.5995347.
87. Niemeier, W. *Ausgleichungsrechnung*, 2nd ed.; De Gruyter: Berlin, Germany, 2008.
88. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; The MIT Press: Cambridge, MA, USA, 2017.
89. Powers, D.M.W. Evaluation: From Precision, Recall and F-measure to Roc, Informedness, Markedness and Correlation. *Int. J. Mach. Learn. Technol.*, **2017**; 2, pp. 37–63
90. Becker, R.; Lublasser, E.; Martens, J.; Wollenberg, R.; Zhang, H.; Brell-Cokcan, S.; Blankenbach, J. *Enabling BIM for Property Management of Existing Buildings Based on Automated As-is Capturing*; Leitfaden Geodasie und BIM: Buehl/Muenchen, 2019. https://doi.org/10.22260/isarc2019/0028.
91. Engelmann, F.; Kontogiannia, T.; Hermans, A.; Leibe, B. Exploring Spatial Context for 3D Semantic Segmentation of Point Clouds. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCV), Venice, Italy, 22–29 October 2017; IEEE: New York, NY, USA, 2017. https://doi.org/10.1109/iccvw.2017.90.
92. Koguciuk, D.; Łukasz Chechliński.; El-Gaaly, T. 3D Object Recognition with Ensemble Learning - A Study of Point Cloud-Based Deep Learning Models. In *Advances in Visual Computing*; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 100–114. https://doi.org/10.1007/978-3-030-33723-0_9.
93. Winiwarter, L.; Mandlburger, G.; Pfeifer, N. *Klassifizierung von 3D ALS Punktwolken mit Neuronalen Netzen*; 20. Internationale Geodätische Woche Obergurgl 2019; Wichmann Herbert: Berlin/Offenbach, Germany, 2019; *20*.
94. Reiterer, A.; Wäschle, K.; Störk, D.; Leydecker, A.; Gitzen, N. Fully Automated Segmentation of 2D and 3D Mobile Mapping Data for Reliable Modeling of Surface Structures Using Deep Learning. *Remote Sens.* **2020**, *12*, 2530. https://doi.org/10.3390/rs12162530.
95. Zoller+Fröhlich-GmbH. *Reaching New Levels, Z+F Imager5016, User Manual, V2.1*; Zoller & Fröhlich GmbH: Wangen im Allgäu, Germany, 2019.
96. Neitzel, F.; Gordon, B.; Wujanz, D. DVW-Merkblatt 7-2014, Verfahren zur Standardisierten Überprüfungvon Terrestrischen Laserscannern (TLS). Technical Report, DVW. Available online: https://dvw.de/veroeffentlichungen/standpunkte/1149-verfahren-zur-standardisierten-ueberpruefung-von-terrestrischen-laserscannern-tls (accessed on 28 October 2021).
97. HexagonMetrology. Product brochure Leica T-Scan TS 50-a. Available online: https://w3.leica-geosystems.com/downloads123/m1/metrology/t-scan/brochures/leica%20t-scan%20brochure_en.pdf (accessed on 24 June 2021).
98. Bernardini, F.; Mittleman, J.; Rushmeier, H.; Silva, C.; Taubin, G. The Ball-pivoting Algorithm for Surface Reconstruction. *IEEE Trans. Vis. Comput. Graph.* **1999**, *5*, 349–359. https://doi.org/10.1109/2945.817351.
99. Zhou, Q.Y.; Park, J.; Koltun, V. Open3D: A Modern Library for 3D Data Processing. *arXiv* **2018**, arXiv:1801.09847v1.
100. Hodges, J.L. The Significance Probability of the Smirnov Two-sample Test. *Ark. Mat.* **1958**, *3*, 469–486.