The Hessigheim 3D (H3D) benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes from UAV LiDAR and Multi-View-Stereo

Kölle, Michael; Laupheimer, Dominik; Schmohl, Stefan; Haala, Norbert; Rottensteiner, Franz; Wegner, Jan Dirk; Ledoux, H.

DOI
10.1016/j.ophoto.2021.100001

Publication date
2021

Document Version
Final published version

Published in
ISPRS Open Journal of Photogrammetry and Remote Sensing

ISPRS Open Journal of Photogrammetry and Remote Sensing 1 (2021) 100001

Contents lists available at ScienceDirect

ISPRS Open Journal of Photogrammetry and Remote Sensing

journal homepage: www.editorialmanager.com/OPHOTO

The Hessigheim 3D (H3D) benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes from UAV LiDAR and Multi-View-Stereo

Michael Kölle [a,*], Dominik Laupheimer [a,*], Stefan Schmohl [a], Norbert Haala [a], Franz Rottensteiner [b], Jan Dirk Wegner [c], Hugo Ledoux [d]

[a] Institute for Photogrammetry, University of Stuttgart, Germany
[b] Institute of Photogrammetry and GeoInformation, Leibniz University Hannover, Germany
[c] EcoVision Lab, University of Zurich & ETH Zurich, Switzerland
[d] Faculty of the Built Environment & Architecture, Delft University of Technology, the Netherlands

ARTICLE INFO

Keywords:
Semantic segmentation
UAV Laser scanning
Multi-View-Stereo
3D point cloud
3D textured mesh
Multi-modality
Multi-temporality

ABSTRACT

Automated semantic segmentation and object detection are of great importance in geospatial data analysis. However, supervised machine learning systems such as convolutional neural networks require large corpora of annotated training data. Especially in the geospatial domain, such datasets are quite scarce. Within this paper, we aim to alleviate this issue by introducing a new annotated 3D dataset that is unique in three ways: i) The dataset consists of both an Unmanned Aerial Vehicle (UAV) laser scanning point cloud and a 3D textured mesh. ii) The point cloud features a mean point density of about 800 pts/m^2 and the oblique imagery used for 3D mesh texturing realizes a ground sampling distance of about 2–3 cm. This enables the identification of fine-grained structures and represents the state of the art in UAV-based mapping. iii) Both data modalities will be published for a total of three epochs allowing applications such as change detection. The dataset depicts the village of Hessigheim (Germany), henceforth referred to as H3D - either represented as 3D point cloud H3D(PC) or 3D mesh H3D(Mesh). It is designed to promote research in the field of 3D data analysis on one hand and to evaluate and rank existing and emerging approaches for semantic segmentation of both data modalities on the other hand. Ultimately, we hope that H3D will become a widely used benchmark dataset in company with the well-established ISPRS Vaihingen 3D Semantic Labeling Challenge benchmark (V3D). The dataset can be downloaded from https://ifpwww.ifp.uni-stuttgart.de/benchmark/hessigheim/default.aspx.

1. Introduction

Supervised Machine Learning (ML), especially embodied by Convolutional Neural Networks (CNNs), has become state of the art for automatic interpretation of various data. However, the applicability and acceptance of such approaches are greatly hindered by the lack of labeled datasets for both training and evaluation (and, consequently, for the verification of their quality). For that purpose, large datasets of labeled 2D imagery were established, for example the *ImageNet* dataset (Deng et al., 2009). As such an extensive annotation process cannot be accomplished by a single person or group, crowdsourcing was employed. Whereas 2D imagery can be very well interpreted by non-experts (i.e., crowdworkers), labeling 3D data is much more demanding. Although first investigations were conducted on employing crowdworkers for 3D data annotation (Dai et al., 2017; Herfort et al., 2018; Walter et al., 2020; Kölle et al., 2020), these approaches typically try to avoid deriving a full pointwise annotation. This is achieved either by working on object level or by focusing only on necessary points by exploiting active learning techniques. However, at least for evaluating ML models for semantic segmentation, full annotations are beneficial, which are typically acquired by experts. In case of outdoor 3D data, existing datasets can be categorized into two different domains (comprehensive literature reviews are given by Griffiths and Boehm (2019) and Xie et al. (2020)): terrestrial data and airborne data.

* Corresponding authors.
E-mail addresses: michael.koelle@ifp.uni-stuttgart.de (M. Kölle), dominik.laupheimer@ifp.uni-stuttgart.de (D. Laupheimer).

https://doi.org/10.1016/j.ophoto.2021.100001
Received 25 February 2021; Received in revised form 26 April 2021; Accepted 21 May 2021
Available online xxxx

i) **Outdoor terrestrial data.** Capturing outdoor terrestrial data has become most popular in the context of autonomous driving. Cars destined for self-driving are equipped with a great variety of different sensors such as cameras, laser scanners, and odometers. Often, only the combination of these sensors allows a comprehensive understanding of the complete scene, which is studied extensively on the basis of the well-known KITTI dataset (Geiger et al., 2012) and the derived SemanticKITTI dataset (Behley et al., 2019) respectively. Although mobile laser scanning point clouds of typical urban scenes are often provided as stand-alone products (Roynard et al., 2018; Munoz et al., 2009; Hackel et al., 2017; Tan et al., 2020), the concurrent availability of LiDAR data and imagery in the form of meshes is often pursued (Riemenschneider et al., 2014). Caesar et al. (2020) provide a unique multi-modal dataset for autonomous driving applications by the combination of both cameras and ranging sensors (i.e., LiDAR & RADAR) for 3D object detection.

ii) **Outdoor airborne data.** Datasets of this category are often referred to as (large-scale) geospatial data and deviate from the previous ones due to a significantly increased distance between target and sensor, which is attached to an airborne platform (mostly small aircraft). So far, publicly available datasets provide labeled point clouds obtained from a single sensor, either a camera (Hu et al., 2020) or a LiDAR sensor (Varney et al., 2020). One prominent example of the latter case is the Vaihingen 3D (V3D) dataset acquired by Cramer (2010), which served as the basis for the ISPRS 3D Semantic Labeling benchmark (Niemeyer et al., 2014). Additionally, some national mapping agencies publish large-scale annotated LiDAR point clouds as open data, which typically realize a rather coarse class catalog and a point density of up to about 40 pts/m^2 (Actueel Hoogtebestand Nederland, 2021; National Land Survey of Finland, 2021). For obtaining higher LiDAR point densities, the carrier of the sensor can be replaced by a helicopter platform allowing the generation of data with point densities of up to about 350 pts/m^2 (Zolanvari et al., 2019). For labeling their data, the authors opt for a broad class catalog (compared to V3D) due to the high point density and the resulting depiction of fine structures. Recently, Can et al. (2020) provided benchmark data composed by the vertices of a purely photogrammetric 3D mesh. They reconstructed the underlying mesh based on high-resolution imagery with a Ground Sampling Distance (GSD) of 1.25 cm.

The Hessigheim 3D (H3D) dataset presented in this paper belongs to the second group but differs from other datasets because it is the first ultra-high resolution, fully annotated 3D dataset acquired from a LiDAR system and cameras integrated on the same Unmanned Aerial Vehicle (UAV) platform. This results in a unique multi-modal scene description by a LiDAR point cloud H3D(PC) and a textured 3D mesh H3D(Mesh). Hence, properties unique for these two acquisition methods can be efficiently combined, which offers new possibilities for high-accuracy georeferencing (Glira et al., 2019) and semantic segmentation (Laupheimer et al., 2020). We consider H3D to be the logical successor of V3D, which was already captured in 2008 and therefore no longer represents the state of the art. As H3D was acquired from a UAV platform by the usage of state-of-the-art sensors, a point density that is about a hundred times higher compared to V3D can be achieved, allowing the expansion of the class catalog of V3D. Furthermore, H3D shall improve the dissemination and acceptance of the mesh representation in photogrammetry and remote sensing. So far, meshes are the state-of-the-art representation for small-scale datasets covering indoor scenes or single objects that are commonly treated by the computer vision community (Kalogerakis et al., 2010; Armeni et al., 2017; Hua et al., 2016; Dai et al., 2017; Shilane et al., 2004). In contrast to unordered point clouds, meshes are graph-based surface structures that provide explicit adjacency information. The surface description enables high-resolution texturing while efficiently storing geometry (which can be interpreted as surface-aware subsampling). A more detailed comparison is given by Laupheimer and Haala (2021). With the help of H3D(Mesh), we want to

Fig. 1. Overview of H3D(PC) and H3D(Mesh) using an orthophoto basemap. H3D(PC) is partitioned into training set (indicated by *class colors*; see Table 1), validation set (indicated by *class colors* and marked by a *yellow box*), and test set (*grey*). H3D(Mesh) is organized in tiles that exceed the annotated point cloud (see Section 2.5). The data splits of H3D(Mesh) are partitioned in accordance with the respective H3D(PC) splits. They are shown in semi-transparent manner (see legend).

i) foster semantic mesh segmentation and ii) evaluate the community's interest in this kind of representation at the same time.

The rest of this paper focuses on presenting H3D as a new benchmark dataset. This includes a detailed presentation of data acquisition, the registration process, and a discussion of the unique characteristics of H3D (Sections 2.1-2.3). Sections 2.4 and 2.5 are dedicated to present the class catalog and the annotation process for H3D(PC) and H3D(Mesh). As we aim at generating a benchmark for semantic segmentation, Section 2.6 describes the general structure of H3D, i.e., the partitioning into disjoint subsets for training, validation, and testing. As labels of the test set are not disclosed to the public, labels are to be predicted by participants (Section 2.7) and transmitted to the authors for evaluation (Section 2.8). We present first results based on two state-of-the-art approaches for semantic segmentation to kick off the benchmark process and to give first baseline results (Section 3) before concluding with a summary in Section 4.

Fig. 2. *Ricopter* platform equipped with *Riegl VUX-1LR* scanner and two oblique *Sony Alpha 6000* cameras used for capturing H3D.

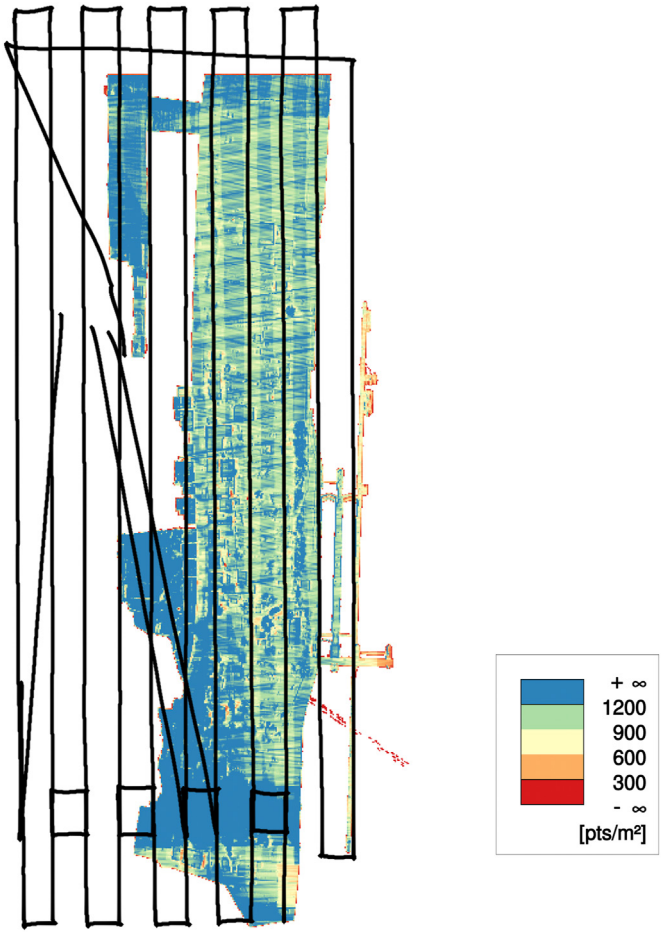


Fig. 3. Achieved point density of H3D(PC). Different point densities are due to diagonal strips for further block stabilization beneficial for adjustment of LiDAR strips. Flight trajectories of LiDAR strips are shown in *black*.

2. The H3D dataset

The H3D dataset was originally captured in a joint project of the University of Stuttgart and the German Federal Institute of Hydrology (BfG) for detecting ground subsidence in the high accuracy range. For this monitoring application, the area of interest, which is the village of Hessigheim, Germany (see Fig. 1), was surveyed at multiple epochs in March 2018, November 2018, and March 2019. Although all those datasets will be made publicly available, this paper only covers the first epoch (i.e., March 2018) as only this dataset has been labeled so far. Concerning later epochs, the data publication shall follow in the course of 2021.

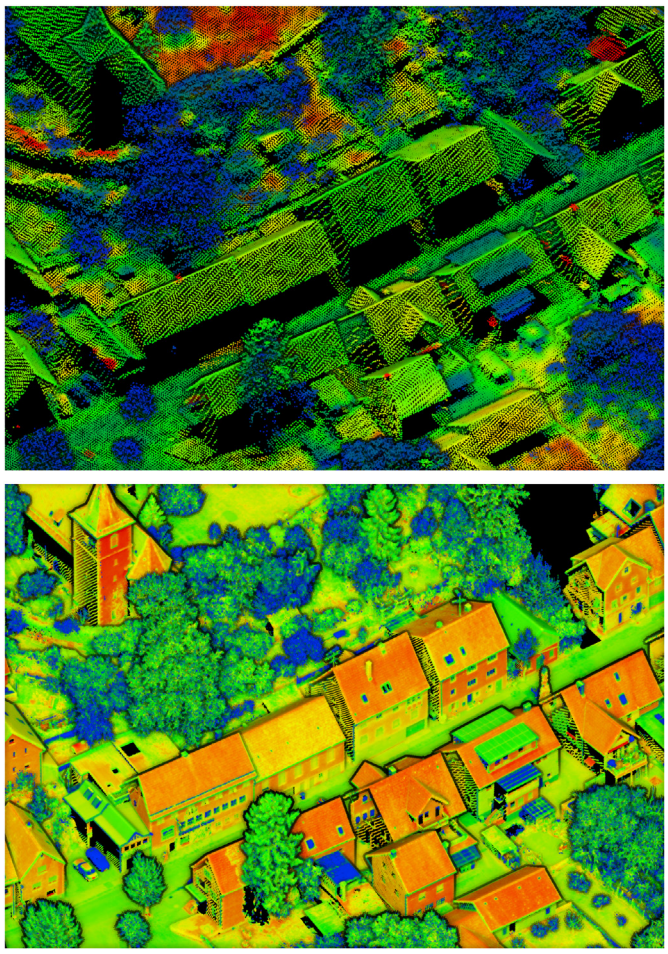


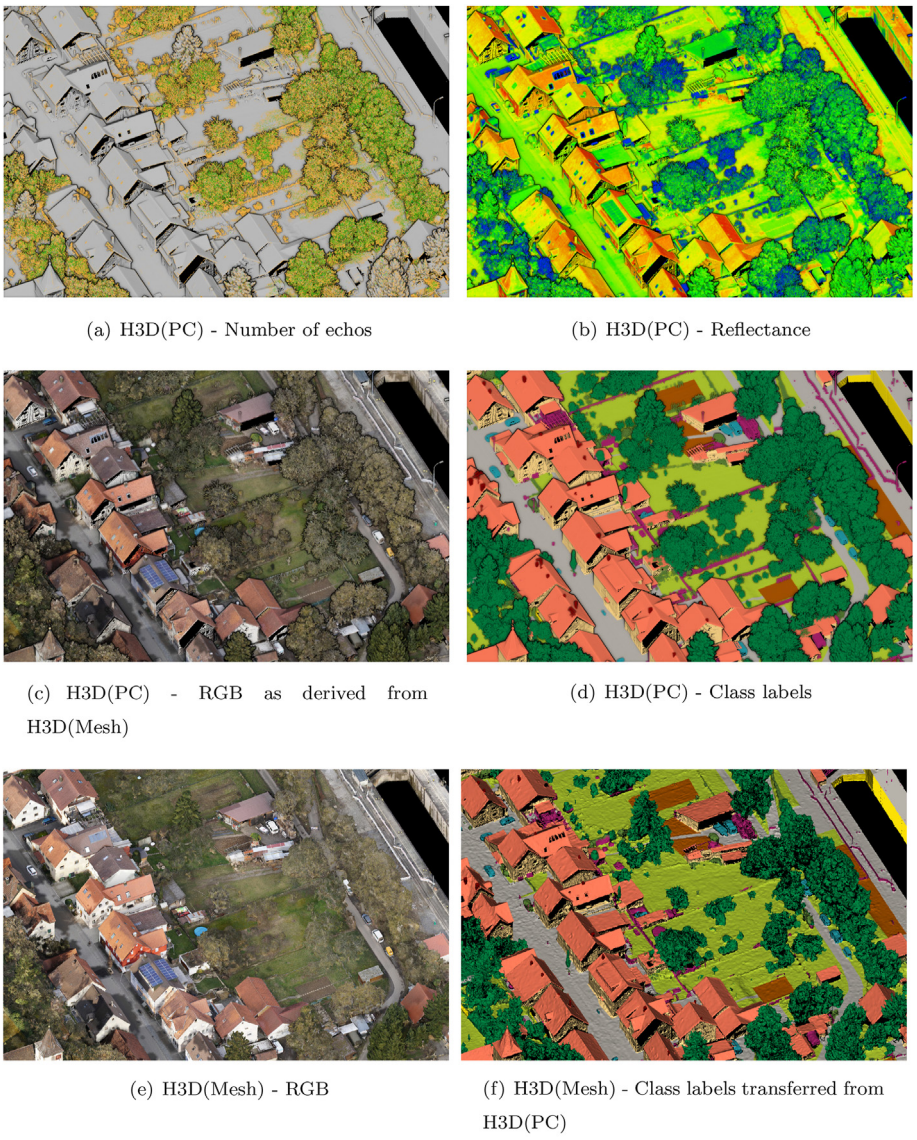
Fig. 4. The same subset of the village of Hessigheim captured by a conventional ALS campaign carried out by the state mapping agency of Baden-Wuerttemberg in 2016 (*top*) and as it is depicted in H3D(PC) (*bottom*).

2.1. Capturing H3D

In all three epochs, our sensor setup was constituted of a *RIEGL VUX-1LR* scanner and two oblique *Sony Alpha 6000* cameras integrated on a *RIEGL Ricopter* platform (see Fig. 2). Considering a height above ground of 50 m, we achieved a laser footprint of less than 3 cm and a GSD of 2–3 cm for the cameras. Using this setup, we obtain two distinct data representations: i) H3D(PC) and ii) H3D(Mesh) (see Section 2.2 and Section 2.3 respectively).

2.2. H3D(PC)

H3D was acquired by a total of 11 longitudinal (i.e., north-south) strips and several diagonal strips (see Fig. 3). Scanner parameters (Pulse Repetition Rate and the mirror's rotation rate) and flying parameters (flying altitude and speed) were set for receiving a point distance of about 5 cm both in and across flight direction. Hence, we obtain about 400 pts/m^2 for one single LiDAR strip and about 800 pts/m^2 for the complete point cloud due to strip overlap. As additional strips were flown for further block stabilization, in some areas significantly higher point densities are achieved (see Fig. 3). Compared to conventional Airborne



(a) H3D(PC) - Number of echos

(b) H3D(PC) - Reflectance

(c) H3D(PC) - RGB as derived from H3D(Mesh)

(d) H3D(PC) - Class labels

(e) H3D(Mesh) - RGB

(f) H3D(Mesh) - Class labels transferred from H3D(PC)

Fig. 5. Available attributes of both modalities of H3D.

Laser Scanning (ALS) flight campaigns applying manned platforms (see Fig. 4 *top*), this high-resolution point cloud allows a more comprehensive 3D scene analysis in comparison to existing datasets. For accurate co-registration of acquired strips with respect to available control planes (Haala et al. (2020)), trajectories were corrected by the bias model offered by the *OPALS* software (Pfeifer et al., 2014). In this context, for each strip a constant offset for each trajectory parameter (ΔX, ΔY, ΔZ, $\Delta\omega$, $\Delta\phi$, $\Delta\kappa$) was estimated.

Apart from the XYZ coordinates of each point, LiDAR inherent features such as the echo number, number of echos, and reflectance were measured. While up to 6 echos were recorded per pulse emitted, the majority of subsequent echos (echo number > 1) are second and third echos (see *yellow* and *green* color in Fig. 5 (a)). Instead of the intensity of received echos, we provide reflectance values[1], which can be interpreted as range corrected intensity. Please note that these values were not corrected for differences in reflectance due to different inclination angles of the laser beam with the illuminated object surface. Reflectance values range from about -30 dB for objects of diffuse reflection properties such as vegetation or asphalt (*dark blue* and *light green* points respectively in Fig. 5 (b)) up to about 20 dB for objects of directed reflection such as roof or façade elements (*red* points in Fig. 5 (b)).

Point cloud colorization was done in a two-step process. We first derived the mesh as outlined in Section 2.3. Afterwards, we extracted an individual RGB tuple for each LiDAR point by nearest neighbor interpolation from the textured mesh. The nearest neighbor interpolation in 3D space (i.e., between mesh and point cloud) is a simple approximation of an occlusion-aware projection of 3D LiDAR points to image space as achieved by the collinearity condition (result is visualized in Fig. 5 (c)).

Additionally, we provide a class label for every point (classes and the annotation will be discussed in Sections 2.4 and 2.5). Both plain *ASCII* files and *las* files are used for data exchange.

2.3. H3D(Mesh)

We generated the 3D mesh with software *SURE* (Rothermel et al., 2012). For the geometric reconstruction of the scene, both LiDAR data and imagery were used to benefit from their complementary properties (Mandlburger et al., 2017). The fusion of both data sources results in a

[1] http://www.riegl.com/uploads/tx_pxpriegldownloads/Whitepaper_LASextrabytes_implementation_in-RIEGLSoftware_2017-12-04.pdf.

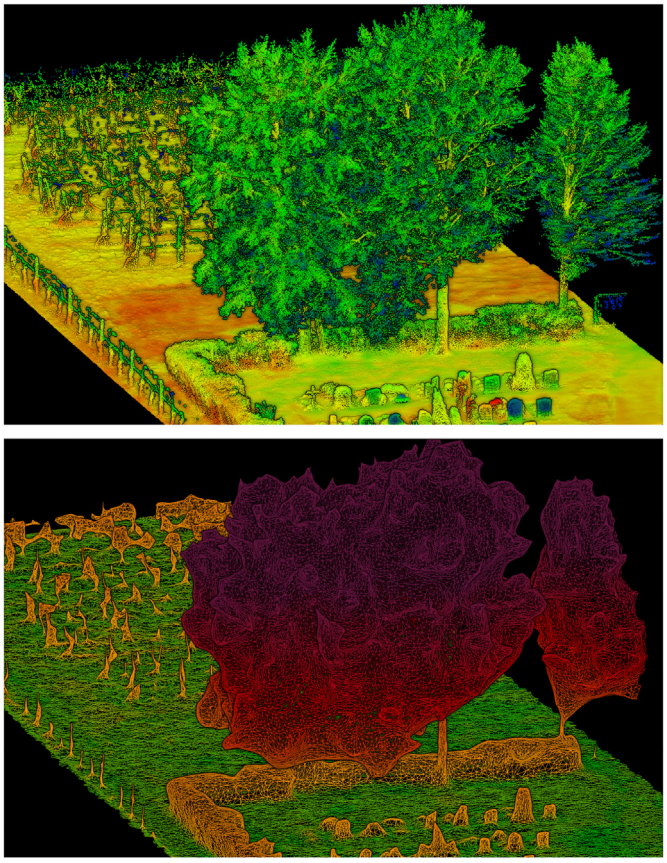


Fig. 6. An exemplary congruent subset of H3D(PC) (*top*, colored according to reflectance) and H3D(Mesh) represented as wireframe (*bottom*, colored according to height above ground). It can be observed that even regions under trees such as ground and hedges can be reconstructed. Generally, fine structures such as vines (e.g., *on the left*) or tombstones of the graveyard (*front center*) are smoothed in the 3D mesh.

more complete mesh compared to a mesh derived from images only. For instance, urban canyons are difficult to reconstruct from imagery (due to required visibility in at least two images) but their reconstruction works smoothly for LiDAR data (where one received echo is already sufficient). This is also the case for regions occluded by semi-transparent objects such as vegetation. Fig. 6 demonstrates that objects underneath tree crowns can be successfully included in the 3D mesh thanks to the concurrently available LiDAR data used for generation of the mesh. Furthermore, the oblique images serve for texturing the generated 3D mesh, which allows a realistic representation of vertical faces (e.g., façades in Fig. 5). The mesh data is provided in a tiled manner. Each tile is given in both textured and labeled mode. For the textured form, each tile consists of i) an *obj* file describing the geometry and referring to (ii) the *mtl* file encoding material properties which links to the (iii) texture atlas that provides textural information (*jpgs*). Their labeled counterparts consist of an *obj* file (containing the same geometry as the respective textured version) and a *mtl* which encodes the class properties (i.e., the color-coding). Therefore, the labeled *obj* files do not require texture atlases since they are pseudo-textured by the class labels. Additionally, we provide Centers of Gravity (CoGs) for each face along with the transferred labels as CoG point cloud. The CoG cloud is available as a plain *ASCII* file enabling simple data handling and data exchange.

2.4. Class catalog

For H3D(PC) and H3D(Mesh), we employ the same fine-grained class catalog, which is based on V3D but is refined due to H3D's higher point density (and due to the purpose of the Hessigheim project, which is monitoring the shipping lock depicted in Fig. 1). This allows to differentiate more details than in V3D. Hence, we added classes *Urban Furniture* (i.e., paved ground), *Soil/Gravel*, *Vertical Surface* (e.g., found at the shipping lock in Fig. 5) and *Chimney* (see Table 1).

Table 1
Class catalog of H3D for epoch March 2018.

class ID	class name
0	Low Vegetation
1	Impervious Surface
2	Vehicle
3	Urban Furniture
4	Roof
5	Façade
6	Shrub
7	Tree
8	Soil/Gravel
9	Vertical Surface
10	Chimney

2.5. Generating ground truth data for H3D

As previously mentioned, the main objective of H3D is to provide labeled multi-temporal and multi-modal datasets for training and evaluation of ML systems for the task of semantic point cloud segmentation. For labeling H3D(PC) (see Section 2.2), we established a manual process carried out by student assistants, resulting in an annotation as depicted in Fig. 5 (d). This classification was generated by extracting point cloud segments with uniform class affiliation (i.e., the point cloud was manually segmented into many small subsets of homogeneous class membership). Segments of each class were afterwards merged to form the semantic segmentation. The whole process was carried out with the help of *CloudCompare* (Girardeau-Montaut, 2021). Quality control was accomplished in a two-stage procedure. First, the student assistants checked each other's labels, and finally, the authors verified the results as last instance. Despite the comprehensive quality check, we are aware of the fact that manual annotations are error-prone and label noise cannot be avoided.

For the 3D mesh, we automatically transfer labels from the manually annotated point cloud by a geometry-driven approach that associates the representation entities, i.e., points and faces (Laupheimer et al., 2020). Therefore, the mesh inherits the class catalog (see Table 1) of the point cloud. In comparison to the point cloud representation, the mesh is more efficient because only a small number of faces is required to represent flat surfaces. For this reason, the number of faces is significantly smaller than the number of LiDAR points (see Table 2). Consequently, several points are commonly linked to the same face. Hence, the per-face label is determined by majority vote of the respective LiDAR points. However, due to structural discrepancies, some faces remain unlabeled because no points can be associated with them (e.g., absence of LiDAR points or geometric reconstruction errors). These faces are marked by the pseudo class label -1. Unlabeled faces cover about 40% of the entire mesh surface. As can be seen from Fig. 1, the majority of the unlabeled area (99.7%) belongs to parts where the mesh tiles exceed the labeled point cloud. This is due to limited availability of class labels of the point cloud (i.e., only a subset of the complete dataset is labeled), so that parts of

Table 2
Class frequencies in the training and validation sets both for H3D(PC) and H3D(Mesh). For H3D(Mesh) we further present the class distribution in terms of covered area. Unlabeled faces are not considered for these statistics.

				Classes									
Modality	Split	total		*Low Veg.*	*I. Surf.*	*Vehicle*	*U. Furn.*	*Roof*	*Façade*	*Shrub*	*Tree*	*Soil*	*V. Surf.*	*Chimney*
H3D(PC)	Train	59445106 Pts	abs.	21375614	10419635	258032	1159205	6279431	1198227	1077141	8086818	8590706	974976	25321
			rel. [%]	35.96	17.53	0.43	1.95	10.56	2.02	1.81	13.60	14.45	1.64	0.04
	Validation	14464248 Pts	abs.	3738743	3212988	183263	455389	3052150	551996	341579	2218551	592510	101243	15836
			rel. [%]	25.85	22.21	1.27	3.15	21.10	3.82	2.36	15.34	4.10	0.70	0.11
H3D(Mesh)	Train	923663 Faces	abs.	1333755	692922	44803	220790	455718	258606	184263	1258053	498072	2143730	5326
			rel. [%]	25.81	13.41	0.87	4.27	8.82	5.01	3.57	24.35	9.64	4.15	0.10
		136836.45m^2	abs.	22146.7	12502.6	660.0	2990.7	7346.3	4123.1	2186.9	16935.5	9754.2	3797.2	59.4
			rel. [%]	26.84	15.15	0.80	3.63	8.90	5.0	2.65	20.53	11.82	4.60	0.07
	Validation	2577554 Faces	abs.	266940	236243	28836	90050	245665	135410	59385	322881	39142	25936	3812
			rel. [%]	18.36	16.24	1.98	6.19	16.89	9.31	4.08	22.20	2.69	1.78	0.26
		37928.05m^2	abs.	4443.6	4203.6	439.9	1276.1	4016.1	2223.7	829.8	4694.4	674.4	472.5	41.1
			rel. [%]	19.06	18.03	1.89	5.47	17.23	9.54	3.60	20.13	2.89	2.03	0.18

Fig. 7. Class distributions of the training set *T* and the validation set *V* both for H3D(PC) and H3D(Mesh). For the mesh, the relative class distribution is given in terms of i) the face count and ii) the area *A* of faces. Unlabeled faces are not considered in this figure.

some mesh tiles remain partly unlabeled. For the overlap of LiDAR and mesh (i.e., the relevant data), 84% of the mesh surface carries an annotation.

2.6. Data splits

The datasets of all epochs are split into a distinct training, validation, and test area (see Fig. 1). The splits are identical in both modalities and chosen in accordance with the mesh tiling. Since H3D is designed as a benchmark, labels of the test set are not disclosed to the participants of the benchmark. Whereas labels of the training and validation sets can be used by participants as desired, we recommend utilizing the pre-defined splits. Detailed statistics of class frequencies in the training and validation sets can be found in Table 2 for both modalities (points vs. faces/CoGs) and are visualized in Fig. 7. For the mesh, we additionally provide the area each class covers, measured by the area of faces assigned to the corresponding class.

In case of the point cloud, the relative number of points for classes covering large areas (such as *Low Vegetation*, *Impervious Surface*, *Tree* and *Roof*) is naturally the highest. Regarding class *Tree*, the number of points is further increased since multiple echos were received for one laser beam. The most underrepresented classes are *Vehicle* and *Chimney*. Although the relative class frequencies for the mesh are similar, we can observe that there are fewer instances of classes corresponding to the ground (*Low Vegetation*, *Impervious Surface* and *Soil/Gravel*), which is due to large face elements used for representing such planar surfaces. On the other hand, the relative number of instances of class *Tree* is increased, because vegetation surfaces are characterized by a high roughness and, thus are typically approximated by a large number of surface elements (i.e., faces.)

2.7. Benchmark challenge

In contrast to V3D, the H3D benchmark challenge is twofold in terms of data representation. For both H3D(PC) and H3D(Mesh), we offer participants to use the training and validation dataset for developing ML approaches for supervised classification and then to apply those to the test dataset. Predicted labels are to be returned to the authors for

Table 3
Baseline results of semantic segmentation for both H3D(PC) and H3D(Mesh). For the mesh, we report the performance metrics weighted by the covered surface.

		F1-scores [%]												
Modality	Method	*Low Veg.*	*I. Surf.*	*Vehicle*	*U. Furn.*	*Roof*	*Façade*	*Shrub*	*Tree*	*Soil*	*V. Surf.*	*Chimney*	mF1 [%]	OA [%]
H3D(PC)	RF	90.36	88.55	66.89	51.55	96.06	78.47	67.25	95.91	47.91	59.73	80.65	74.85	87.43
	SCN	92.31	88.14	63.51	57.17	96.86	83.19	68.59	96.98	44.81	78.20	73.61	76.67	88.42
H3D(Mesh)	RF	89.35	89.06	61.38	57.65	93.29	82.16	69.27	96.09	48.85	62.58	76.43	75.10	86.53
	SCN	89.82	83.66	61.05	52.24	87.09	81.01	59.75	95.06	51.00	56.61	70.22	71.59	83.73

evaluation (see Section 2.8). For the mesh, the predictions are to be returned to the authors as labeled CoG cloud (the respective plain *ASCII* file of CoGs is provided by the authors). The authors will match the predicted per-face labels with the corresponding faces of the *obj* files. Simply put, the CoG cloud is utilized as an efficient link to the underlying mesh structure to keep the memory footprint of submitted data low. The evaluation itself will be done on the mesh (*obj* files).

To avoid overfitting on the benchmark data, the submission protocol allows the upload of one result per author and modality/epoch for the time being. Results of participants will be presented on the official H3D homepage (https://ifpwww.ifp.uni-stuttgart.de/benchmark/hessigheim/results.aspx) in order to reflect the state of the art of semantic segmentation of point clouds and meshes.

2.8. Evaluation metrics

Semantic segmentation results submitted by participants of the benchmark are evaluated by the H3D team by means of the derived confusion matrices. In order to obtain performance metrics for individual classes, the number of True Positives (*TP*), the number of False Positives (*FP*), and the number of False Negatives (*FN*) are determined for each class *c*. These numbers are used for determining Precision (*P*) and Recall (*R*) (Equation (1) and (2). Additionally, we derive a F1-score as the harmonic mean of *P* and *R* for each class (Equation (3)) (Goutte and Gaussier, 2005).

$$P_c = \frac{TP_c}{TP_c + FP_c} \tag{1}$$

$$R_c = \frac{TP_c}{TP_c + FN_c} \tag{2}$$

$$\mathrm{F1}_c = \frac{2 \cdot P_c \cdot R_c}{P_c + R_c} \tag{3}$$

To describe the total performance of a classifier, we combine individual class scores by computing i) the Overall Accuracy ($OA = \sum_c TP_c/N$, with N being the total number of labeled instances) and ii) the mean F1-score (macro-F1). These measures are determined both for H3D(PC) and H3D(Mesh). In case of the latter, the evaluation is based on the covered area of correctly/incorrectly classified faces (as provided as CoG cloud by the participants).

3. Baselines

We initialize the H3D benchmark challenge by providing two baseline solutions for semantic segmentation of H3D. On one hand, we apply a conventional Random Forest (RF) classifier (Breiman, 2001) relying on hand-crafted features (see Section 3.1) and on the other hand, we use a Sparse Convolutional Network (SCN) as end-to-end learning approach (see Section 3.2). In case of H3D(PC), we enrich our baselines with PointNet++ (Qi et al., 2017) and KPConv (Thomas et al., 2019) but restrict ourselves to the discussion of the RF and SCN since they reach top results (results of all approaches are reported on our homepage https://ifpwww.ifp.uni-stuttgart.de/benchmark/hessigheim/results.aspx). Please note that a comprehensive comparison of possible classification approaches is out of scope for this specific paper, but will be gradually created in context of the benchmark challenge. An overview of suitable deep learning approaches for semantic segmentation can be found in Griffiths and Boehm (2019) and Guo et al. (2020).

3.1. Random Forest

For semantic segmentation of the point cloud, we compute geometric features as proposed in Weinmann et al. (2015) and in Chehata et al. (2009) by estimating the structural tensor for a local neighborhood of each point. After extracting the Eigenvalues of that tensor, we can determine the characteristics of the respective point distribution by forming different ratios of Eigenvalues (Weinmann et al., 2015). As computing Eigenvalues and Eigenvectors eventually means to fit a plane to the local point set, we can further enhance our feature vector by taking into account the orientation of these planes. Furthermore, we consider height-based features by determining the height above ground (i.e., above the Digital Terrain Model (DTM) level; DTM is derived by *SCOP++* (Pfeifer et al., 2001)) for each LiDAR point. In addition to purely geometric features, LiDAR-inherent features such as echo ratio and intensity of the received echo are also used for the semantic segmentation. In order to establish a multi-scale approach and to analyze features on different levels of abstraction, we follow the recommendation of Weinmann et al. (2018) and compute each feature for spherical neighborhoods of radii $r = 1, 2, 3$ and 5m. To account for the high density of H3D, we expand the feature vector of each point by additionally deriving all geometric features for radii $r = 0.125$, 0.25, 0.5 and 0.75m.

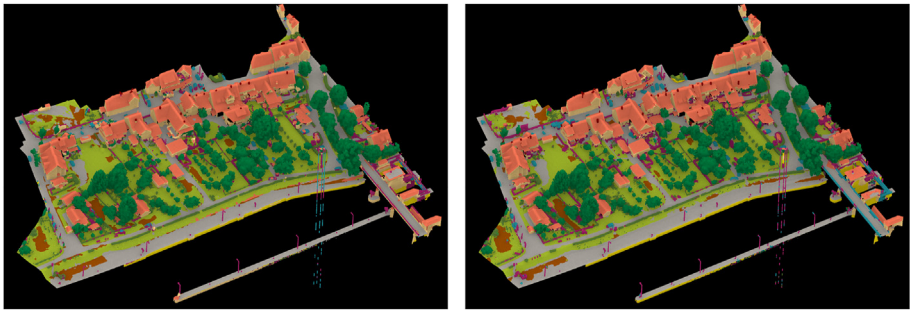
To obtain mesh features, we follow the approach of Tutzauer et al. (2019) and encode each face by its CoG. In this way, we can on one hand compute all aforementioned geometric features for our CoG cloud. On the other hand, we preserve features of the mesh geometry by assigning mesh-inherent features such as face area, face density, and normal orientation to the respective faces. Furthermore, we transfer LiDAR-specific features to the mesh representation by the approach presented in Laupheimer et al. (2020).

For both the point cloud and the mesh, we additionally incorporate radiometric features. For this purpose, RGB tuples are converted to HSV color space and used together with Gaussian smoothed color values for the aforementioned spatial neighborhoods. Since a multitude of HSV tuples is encoded in each face, we additionally calculate the HSV variance for each face.

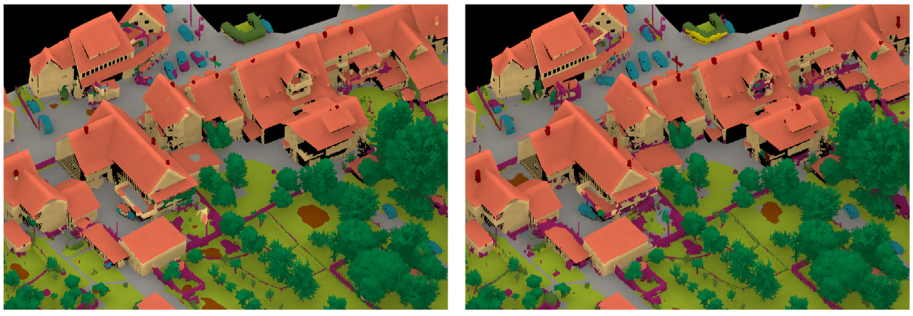
Based on these features, a RF model is trained for H3D(PC) and H3D(Mesh). Prediction results for the test set can be found in Table 3 (see discussion of results in Section 3.3). The RF models are parametrized by 100 binary decision trees with a maximum depth of 18. Niemeyer et al. (2014) have shown that pure pointwise results of the RF classifier can be further refined by a Markov Random Field (MRF). Therefore, we enhance our RF by an a posteriori probability-aware MRF-like smoothing (a posteriori probability is used as unary potential; points within 0.5m radius are considered for the regularization term).

3.2. Sparse Convolutional Network

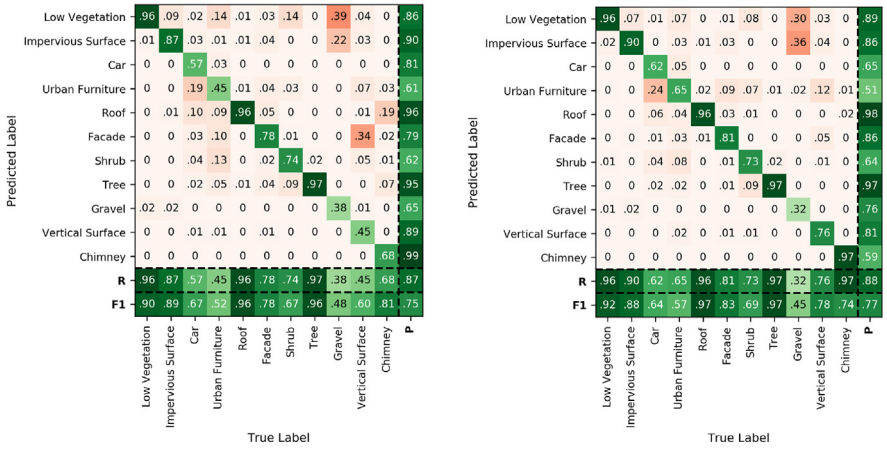
As deep learning has become a de-facto standard in most fields of pattern recognition, we also include a neural network in our baselines. In



(a) Labels predicted by RF (b) Labels predicted by SCN



(c) Labels predicted by RF - close up (d) Labels predicted by SCN - close up



(e) Normalized confusion matrix for the RF classifier

(f) Normalized confusion matrix for the SCN classifier

Fig. 8. Results of semantic segmentation on the test set of H3D(PC) for both our RF (*left*) and our SCN classifier (*right*).

particular, we employ a 3D CNN in U-Net form with submanifold sparse convolutional layers (Graham et al., 2018) to account for the typical spatial distribution of ALS point clouds. The network's general layout and training regime is described in Schmohl and Sörgel (2019). It consists of 3 downsampling levels and 21 convolutional layers in total. The input data (point cloud or mesh) is discretized to sparse 3D voxel grids with voxels of 25 cm side length. We found that smaller voxel sizes did not yield to significantly better accuracies on this dataset. Although additional input features as used for the RF improved accuracy by about 1 percentage point, we opted to report the results with each input point containing just the native point features as a purely learned baseline in contrast to the RF using engineered features. In case of the point cloud, these are the measured point attributes (echo number, number of echos, reflectance & RGB values) as described in Section 2.2. For the mesh, voxels are derived from the CoG point cloud and attributed with the texture information (RGB) only. We tested also a configuration with additional normal information, since this is the most basic native mesh feature besides RGB. However, the achieved performance is slightly worse (−0.6 and −2.34 percentage points for OA and mean F1-score respectively). For evaluation, the inferred voxel labels are transferred back to the enclosed points or faces. Results are also reported in Table 3.

3.3. Discussion of baseline results

Within this chapter, we analyze the performance of our two baseline classifiers (see Section 3.3.1 and Section 3.3.2 respectively) for both H3D(PC) and H3D(Mesh) in order to develop a better understanding of challenges of H3D and to kick-off the benchmark competition.

3.3.1. H3D(PC)

Generally, we can observe from Table 3 that the RF and the SCN classifiers perform similarly well, with the SCN slightly better in terms of

Fig. 9. Regions of distinct classes in H3D(PC) with similar radiometric properties. Issues in automated semantic segmentation based on colors can arise due to small parts of other class affiliation (*1*), misleading colors caused by wetness such as puddles (*2*) and human created structures that have been overgrown and eroded over time (*3* & *4*).

the OA and the mean F1-score. Both baseline solutions (RF & SCN) achieve similar results for the ground classes *Low Vegetation* and *Impervious Surface*. However, performance for class *Soil/Gravel* is rather poor in both cases for there is often confusion with other ground classes (see Fig. 8). Since points of all ground classes incorporate similar geometric properties (e.g., similar normals and smooth surfaces), distinguishing these classes is only possible with the help of radiometric features such as reflectance or color information (see Fig. 5). Whereas this performs well for *Low Vegetation* vs. *Impervious Surface*, segmenting points of *Soil/Gravel* is rather demanding due to similar radiometric properties to *Low Vegetation* (similar to bare soil) and *Impervious Surface* (similar to debris and gravel), which is further amplified by the acquisition during leaf-off season in which even grassland is often dominated by a brownish color. Possible confusions between such ground classes due to the usage of color information are also displayed in Fig. 9.

Similar radiometric and geometric properties are also the reason for the confusion of *Shrub* with *Tree* (greenish color in both cases and rough surfaces). Nevertheless, the extraction of tree points succeeds quite well, probably due to the distinctive multi-echo ability of the employed sensor (see Fig. 5 (a)). Regarding buildings, roofs can be extracted successfully, but the detection of façades seems to be more demanding for both classifiers (see Table 3). This may be caused by the presence of façade furniture (such as balconies with handrails and outdoor furniture differing from smooth façades), which is similar to urban furniture. The confusion matrices in Fig. 8 further indicate that points of class *Urban Furniture* are spread across many other classes. This is due to the great variety of objects belonging to this class, since essentially it serves as quasi-class *Other*. Performance evaluation of classifiers for such fine-grained elements of façade furniture and urban furniture could hardly be evaluated in the past due to the mostly insufficient representation of these fine structures resulting from the poor resolution of former datasets. Therefore this is a unique feature of H3D.

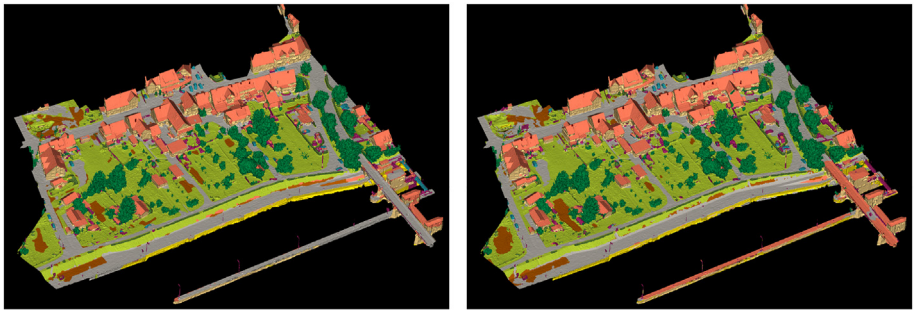
In this context, a significant amount of misclassification can be observed for cars, which are predicted as *Urban Furniture*. This can be explained by a great variety of colors in both classes and some special cases, such as the car-like shapes of covered wood piles often found in gardens (see the results in Fig. 8 (c) and (d) and their associated confusion matrices in (e) and (f)). The classes *Vertical Surface* and *Chimney* newly introduced compared to V3D, seem to be demanding as well. In this context, it is worth mentioning that whereas the RF classifier confuses façades with other vertical surfaces as anticipated, this is not the case for the SCN, where only a small number of vertical surface points are allocated in class *Façade*. This may be due to the larger receptive field of 20m for the lowest SCN filters compared to the maximum neighborhood radius for feature computation of 5m in case of the RF. For chimneys, on the other hand, the RF classifier performs better probably due to the discretization of the SCN approach, so that the RF might capture such small structures more precisely for its pointwise working principle.

To conclude, our baseline solutions indicate, that the depiction of detailed structures and the consequently expanded class catalog of H3D (compared to V3D) poses new challenges for the development of methods for semantic segmentation. Particularly, this applies to entities belonging to newly introduced classes but also for the interaction of those with representatives of common object classes (e.g., *Vertical Surface* vs. *Façade*).

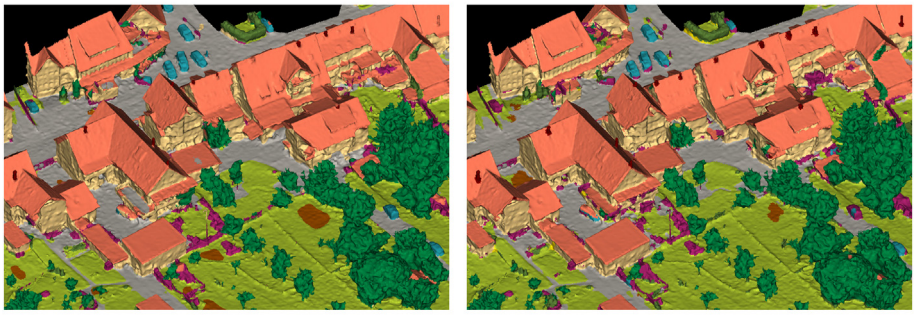
3.3.2. H3D(Mesh)

In addition to results for H3D(PC), Table 3 also reports per-class F1-scores, mean F1-score, and OA for our baseline classifiers RF and SCN on H3D(Mesh). The respective confusion matrices are depicted in Fig. 10 (e) and (f). Due to the non-uniformity of meshes, we evaluate the considered metrics concerning the covered surface for each entity. Each face contributes to the performance metrics depending on its surface area. By these means, a large face has more impact compared to a small face. This contrasts with point cloud evaluation where all points share the same weight. However, we observed that the weighted performance metrics scarcely differ from their unweighted counterparts, which indicates that the majority of large faces is correctly predicted. Moreover, their similarity hints at the constitution of the mesh. The automatically generated mesh differs from an ideal mesh in such sense that the non-uniformity of faces is kept small in order to uphold details in the reconstructed mesh. Loosely speaking, a large number of faces roughly share the same face area. Nonetheless, flat surfaces are represented by few large faces, whereas rough surfaces (e.g., vegetational classes) use many small faces (covering roughly the same area). Therefore, the effect of the surface-driven evaluation is best visible for classes that consist of flat surfaces. For instance, 42% of faces predicted as vertical surfaces are façades in reality (for SCN). Regarding covered area, this makes already 46% of mispredicted façades.

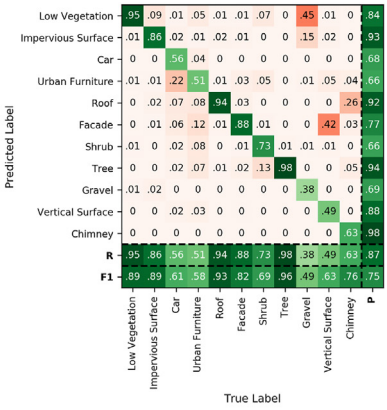
Generally, the feature-driven RF outperforms the data-driven SCN in terms of OA and mean F1-score by 2.8 and 3.5 percentage points respectively. We assume that voxelization and the small set of input features mainly cause the slight inferiority of the SCN. Whereas the RF uses the underlying mesh geometry encoded in the variety of hand-crafted features (see Section 3.1), SCN first voxelizes the textured data and learns features merely on the voxel representation. Furthermore, by utilizing transferred LiDAR features, RF implicitly leverages the fine-grained point cloud geometry that does not suffer from mesh reconstruction errors. Generally, fine-grained structures such as urban furniture and shrubs are demanding objects for meshing algorithms. Hence, the transferred LiDAR features enhance the geometric information and help to correctly classify non-correctly reconstructed objects. For these reasons, we deduce that the end-to-end learning SCN approach cannot fully compete with the feature-driven RF in this case. The results further indicate that feature engineering and multi-modal feature transfer are valid alternatives to state-of-the-art end-to-end learning approaches. Apart from their varying overall performance, the confusion matrices of RF and SCN show similar strengths and weaknesses. Considering per-class F1-scores, we note similar performances for classes *Low Vegetation*, *Vehicle*, *Façade* and *Tree*. Both classifiers struggle with class *Urban Furniture* due to its large intra-class variance (see discussion in Section 3.3.1). For instance, cars are often classified as *Urban Furniture* by mistake. However, due to the utilized superior geometric information, the RF copes better with the variance resulting in an F1-score that is 5.4 percentage points better compared to SCN. For class *Shrub*, the RF is 9.5 percentage points better. In case of the SCN, the majority of predictions of class *Shrub* truly belongs to *Urban Furniture*. As can be seen for the shipping lock in Fig. 10 (b), SCN confuses *Impervious Surface* with *Roof*



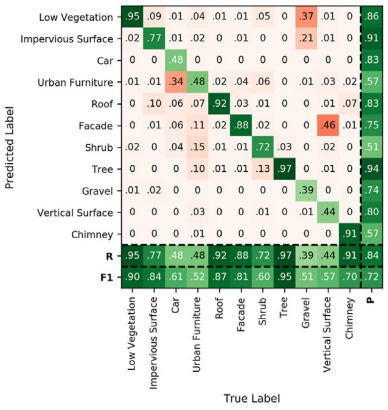
(a) Labels predicted by RF (b) Labels predicted by SCN



(c) Labels predicted by RF - close up (d) Labels predicted by SCN - close up



(e) Normalized confusion matrix for the RF classifier (weighted by face area)



(f) Normalized confusion matrix for the SCN classifier (weighted by face area)

Fig. 10. Results of semantic segmentation on the test set of H3D(Mesh) for both our RF (*left*) and our SCN classifier (*right*).

and hence performs worse than the RF on these classes. The geometric similarity of ground classes in the mesh representation (*Impervious Surface*, *Low Vegetation*, and *Gravel/Soil*) makes it demanding to correctly separate them. Therefore, the color information is decisive for the correct prediction. Both classifiers predict other ground classes for *Gravel/Soil*. *Gravel/Soil* is the only class where SCN outperforms the RF. Most probably, the Gaussian smoothed features cause misprediction of chimneys as *Roof* for RF. In case of SCN, *Chimney* has a high recall but at cost of good precision. On the contrary, RF has a significantly worse recall but very high precision resulting in a better F1-score. The distinction of vertical surfaces and façades is demanding for both classifiers due to their small inter-class variance.

4. Conclusion

In this paper, we presented a new benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes as acquired and derived from UAV LiDAR and Multi-View-Stereo: Hessigheim 3D (H3D). We have introduced the multi-modal data corresponding to the first epoch of H3D (March 2018), comprising H3D(PC) and H3D(Mesh). Follow-on epochs (November 2018 & March 2019) will cover an extended area and offer an even more detailed class catalog. Apart from discussing the data acquisition and processing, we applied different classifiers to H3D(PC) and H3D(Mesh) as a baseline for the benchmark. The results indicate great potential for testing ML approaches on H3D due to its large sets of labeled data. Eventually, we hope H3D to become a second established ISPRS benchmark dataset in company with V3D.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The H3D dataset has been captured in the context of an ongoing research project funded by the German Federal Institute of Hydrology (BfG). We would like to thank the University of Innsbruck for carrying out the flight missions. Our gratitude goes to Markus Englich for providing and maintaining H3D's IT infrastructure. We appreciate the funding of H3D as an ISPRS scientific initiative 2021 and the financial support of EuroSDR. The authors would like to show their gratitude to the State Office for Spatial Information and Land Development Baden-Wuerttemberg for providing the ALS point clouds of the village of Hessigheim. Further thanks go to Weixiao Gao from TU Delft for testing alternate segmentation approaches such as KPConv and PointNet++.

References

Actueel Hoogtebestand Nederland, 2021. Dataset: actueel Hoogtebestand Nederland (AHN3). https://www.pdok.nl/introductie/-/article/actueel-hoogtebestand-nederland-ahn3-, 2.2.21.

Armeni, I., Sax, A., Zamir, A.R., Savarese, S., Feb. 2017. Joint 2D-3d-Semantic Data for Indoor Scene Understanding. ArXiv e-prints.

Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J., 2019. SemanticKITTI: a dataset for semantic scene understanding of LiDAR sequences. In: Proc. Of the IEEE/CVF International Conf. on Computer Vision (ICCV).

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. https://doi.org/10.1023/A:1010933404324.

Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O., 2020. nuscenes: a multimodal dataset for autonomous driving. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Pp. 11618–11628.

Can, G., Mantegazza, D., Abbate, G., Chappuis, S., Giusti, A., 2020. Semantic Segmentation on Swiss3dcities: A Benchmark Study on Aerial Photogrammetric 3d Pointcloud Dataset.

Chehata, N., Guo, L., Mallet, C., 2009. Airborne LiDAR feature selection for urban classification using random forests. XXXVIII-3/W8 ISPRS Archives 207–212.

Cramer, M., may 2010. The DGPF-test on digital airborne camera evaluation – overview and test design, 2010 Photogramm. Fernerkund. GeoInf. (2), 73–82.

Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M., 2017. Scannet: richly-annotated 3d reconstructions of indoor scenes. In: Proc. Computer Vision and Pattern Recognition (CVPR), IEEE.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Li, F.F., 2009. ImageNet: a large-scale hierarchical image database. In: CVPR 2009. Pp. 248–255.

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. Pp. 3354–3361.

Girardeau-Montaut, D., 2021. CloudCompare 3d point cloud and mesh processing software open source project. https://www.danielgm.net/cc/, 2.2.21.

Glira, P., Pfeifer, N., Mandlburger, G., 2019. Hybrid orientation of airborne lidar point clouds and aerial images. IV-2/W5 ISPRS Annl. 567–574. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-2-W5/567/2019/.

Goutte, C., Gaussier, E., 2005. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In: Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 345–359.

Graham, B., Engelcke, M., Maaten, L.v.d., 2018. 3D semantic segmentation with submanifold sparse convolutional networks. In: CVPR 2018, pp. 9224–9232.

Griffiths, D., Boehm, J., 2019. A review on deep learning techniques for 3d sensed data classification. In: Remote Sensing, vol. 11. https://www.mdpi.com/2072-4292/11/12/1499.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2020. Deep learning for 3d point clouds: a survey. IEEE Trans. Pattern Anal. Mach. Intell., 1–1

Haala, N., Kölle, M., Cramer, M., Laupheimer, D., Mandlburger, G., Glira, P., 2020. Hybrid georeferencing, enhancement and classification of ultra-high resolution UAV LiDAR and image point clouds for monitoring applications. V-2-2020 ISPRS Annl. 727–734. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-2-2020/727/2020/.

Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M., 2017. Semantic3d.net: a new large-scale point cloud classification benchmark. IV-1/W1 ISPRS Annal. Photogrammetr. Rem. Sens. Spatial Inform. Sci. 91–98. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-1-W1/91/2017/.

Herfort, B., Höfle, B., Klonner, C., mar 2018. 3D micro-mapping: towards assessing the quality of crowdsourcing to support 3D point cloud analysis. ISPRS J. Photogrammetry Remote Sens. 137, 73–83.

Hu, Q., Yang, B., Khalid, S., Xiao, W., Trigoni, N., Markham, A., 2020. Towards Semantic Segmentation of Urban-Scale 3d Point Clouds: A Dataset, Benchmarks and Challenges. CoRR arXiv:2009, 03137.

Hua, B., Pham, Q., Nguyen, D.T., Tran, M., Yu, L., Yeung, S., 2016. Scenenn: a scene meshes dataset with annotations. In: 2016 Fourth International Conference on 3D Vision (3DV). Pp. 92–101.

Kalogerakis, E., Hertzmann, A., Singh, K., 2010. Learning 3d mesh segmentation and labeling. In: ACM SIGGRAPH 2010 Papers. SIGGRAPH '10. ACM, New York, NY, USA, Pp. 102:1–102:12. https://doi.org/10.1145/1833349.1778839.

Kölle, M., Walter, V., Schmohl, S., Soergel, U., 2020. Hybrid acquisition of high quality training data for semantic segmentation of 3D point clouds using crowd-based active learning. V-2-2020 ISPRS Annl. 501–508. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-2-2020/501/2020/.

Laupheimer, D., Haala, N., 2021. Juggling with Representations: on the Information Transfer between Imagery, Point Clouds, and Meshes for Multi-Modal Semantics.

Laupheimer, D., Shams Eddin, M.H., Haala, N., 2020. On the association of LiDAR point clouds and textured meshes for multi-modal semantic segmentation. V-2-2020 ISPRS Annl. 509–516. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-2-2020/509/2020/.

Mandlburger, G., Wenzel, K., Spitzer, A., Haala, N., Glira, P., Pfeifer, N., 2017. Improved topographic models via concurrent airborne lidar anddense image matching. ISPRS Annal. Photogrammetr. Rem. Sens. Spatial Inform. Sci. 259–266.

Munoz, D., Bagnell, J.A.D., Vandapel, N., Hebert, M., June 2009. Contextual classification with functional max-margin markov networks. In: Proceedings of (CVPR) Computer Vision and Pattern Recognition. Pp. 975 – 982.

National Land Survey of Finland, 2021. File service of open data of national Land survey of Finland. https://tiedostopalvelu.maanmittauslaitos.fi/tp/kartta?lang=en, 2.2.21.

Niemeyer, J., Rottensteiner, F., Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. ISPRS J. 87, 152–165.

Pfeifer, N., Mandlburger, G., Otepka, J., Karel, W., may 2014. OPALS – a framework for airborne laser scanning data analysis. Comput. Environ. Urban Syst. 45, 125–136.

Pfeifer, N., Stadler, P., Briese, C., 2001. Derivation of digital terrain models in the scop++ environment. In: OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Digital Elevation Models.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: deep hierarchical feature learning on point sets in a metric space. In: NIPS 2017. NIPS'17. Curran Associates Inc., USA, pp. 5105–5114. http://dl.acm.org/citation.cfm?id=3295222.3295263.

Riemenschneider, H., Bódis-Szomorú, A., Weissenberg, J., Van Gool, L., 2014. Learning where to classify in multi-view semantic segmentation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (Eds.), Computer Vision – ECCV 2014. Springer International Publishing, Cham, pp. 516–532.

Rothermel, M., Wenzel, K., Fritsch, D., Haala, N., 12 2012. SURE: photogrammetric surface reconstruction from imagery. In: Proceedings LC3D Workshop, vol. 8 (Berlin).

Roynard, X., Deschaud, J.-E., Goulette, F., 2018. Paris-lille-3d: a large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. Int. J. Robot Res. 37 (6), 545–557. https://doi.org/10.1177/0278364918767506.

Schmohl, S., Sörgel, U., 2019. Submanifold sparse convolutional networks for semantic segmentation of large-scale ALS point clouds. IV-2/W5 ISPRS Annl. 77–84. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-2-W5/77/2019/.

Shilane, P., Min, P., Kazhdan, M., Funkhouser, T., 2004. The princeton shape benchmark. In: Shape Modeling Applications, 2004. Proceedings. IEEE, pp. 167–178.

Tan, W., Qin, N., Ma, L., Li, Y., Du, J., Cai, G., Yang, K., Li, J., Jun 2020. Toronto-3d: A Large-Scale Mobile Lidar Dataset for Semantic Segmentation of Urban Roadways, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). https://doi.org/10.1109/CVPRW50498.2020.00109.

Thomas, H., Qi, C.R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L., 2019. Kpconv: flexible and deformable convolution for point clouds. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Pp. 6410–6419.

Tutzauer, P., Laupheimer, D., Haala, N., 2019. Semantic urban mesh enhancement utilizing A hybrid model. ISPRS Annl. IV-2/W7, 175–182. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-2-W7/175/2019/.

Varney, N., Asari, V.K., Graehling, Q., 2020. Dales: a large-scale aerial lidar data set for semantic segmentation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 717–726.

Walter, V., Kölle, M., Yin, Y., 2020. Evaluation and optimisation of crowd-based collection of trees from 3D point clouds. V-4-2020 ISPRS Annl. 49–56. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/V-4-2020/49/2020/.

Weinmann, M., Blomley, R., Weinmann, M., Jutzi, B., 2018. Investigations on the potential of binary and multi-class classification for object extraction from airborne laser scanning point clouds. In: Proceedings of the 38th Annual Meeting of the DGPF. Pp. 408–421.

Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS J. 105, 286–304.

Xie, Y., Tian, J., Zhu, X.X., dec 2020. Linking points with labels in 3d: a review of point cloud semantic segmentation. IEEE Geosci. Rem. Sens. Magaz. 8 (4), 38–59.

Zolanvari, S.M.I., Ruano, S., Rana, A., Cummins, A., da Silva, R.E., Rahbar, M., Smolic, A., 2019. Dublincity: annotated lidar point cloud and its applications. In: BMVC.