Juggling With Representations:
On the Information Transfer Between Imagery,
Point Clouds, and Meshes for Multi-Modal Semantics

Dominik Laupheimer\textsuperscript{a,}\textsuperscript{*}, Norbert Haala\textsuperscript{a}

\textsuperscript{a}Institute for Photogrammetry, University of Stuttgart, Germany

Abstract

The automatic semantic segmentation of the huge amount of acquired remote sensing data has become an important task in the last decade. Images and Point Clouds (PCs) are fundamental data representations, particularly in urban mapping applications. Textured 3D meshes integrate both data representations geometrically by wiring the PC and texturing the surface elements with available imagery. We present a mesh-centered holistic geometry-driven methodology that explicitly integrates entities of imagery, PC and mesh. Due to its integrative character, we choose the mesh as the core representation that also helps to solve the visibility problem for points in imagery. Utilizing the proposed multi-modal fusion as the backbone and considering the established entity relationships, we enable the sharing of information across the modalities imagery, PC and mesh in a two-fold manner: (i) feature transfer and (ii) label transfer. By these means, we achieve to enrich feature vectors to multi-modal feature vectors for each representation. Concurrently, we achieve to label all representations consistently while reducing the manual label effort to a single representation. Consequently, we facilitate to train machine learning algorithms and to semantically segment any of these data representations – both in a multi-modal and single-modal sense. The paper presents the association mechanism and the subsequent information transfer, which we believe are cornerstones for

\textsuperscript{*}Corresponding author

Email address: dominik.laupheimer@ifp.uni-stuttgart.de (Dominik Laupheimer)
multi-modal scene analysis. Furthermore, we discuss the preconditions and limitations of the presented approach in detail. We demonstrate the effectiveness of our methodology on the ISPRS 3D semantic labeling contest (Vaihingen 3D) and a proprietary data set (Hessigheim 3D).

Keywords: Multi-Modality, Data Fusion, 3D Textured Mesh, 3D Point Cloud, Imagery, Ground Truth, Semantic Segmentation

1. Introduction

Over the years, data acquisition has become more redundant, more complete, faster, and denser – spatially and temporally. Sensors such as cameras, LiDAR scanners, and RaDAR sensors guarantee multi-modal capturing of our world. Depending on the application and the desired mapping scale, the respective sensors are mounted on platforms such as satellites, airplanes, Unmanned Airborne Vehicles (UAVs), or autonomous vehicles. In the domain of photogrammetry and remote sensing, particularly for urban mapping, data acquisition via imagery and Airborne Laser Scanning (ALS) is common. Currently, data capture in urban areas at Ground Sampling Distances (GSDs) down to a few centimeters is becoming state of the art. Traditionally, the airplane has been the platform of choice. However, more flexible and lightweight UAVs have grown in popularity in the past decade \cite{Haala et al. 2020}.

Figure 1: Visualization of the proposed multi-modal data fusion by means of the enabled label transfer. The figure depicts the transferred annotations to the mesh (center) and an oblique image (right) as transferred from the respective manually labeled point cloud (a subset of Hessigheim 3D). Faces that cannot be linked to points are shown in textured fashion. Background or non-associated pixels respectively are colored in reddish. Pixels that are linked to an unlabeled face are depicted in black. The label scheme is given in \textbf{Figure 2}.
Imagery is the fundamental photogrammetric data representation providing (multi-)spectral information. Images project 3D real-world objects into 2D image space. By nature of the projection into grid-like pixel space, images suffer from occlusions, distortions, discretization, and the loss of the third dimension. However, with the help of automatic aerial triangulation, the intrinsic defects can be rectified and 3D reconstruction is possible. As a precondition to proper reconstruction, images have to provide unambiguous texture and capture each object point at least twice. Derived 3D products of the Multi-View Stereo (MVS) pipeline are colored Point Clouds (PCs) and/or textured meshes, both mapping the surface of the captured region.

In contrast, due to the polar measurement principle, 3D PCs are the immediate ALS output. In comparison to MVS, LiDAR scanning provides multi-target capability, and hence penetrates semi-transparent objects such as vegetation. Moreover, the polar measurement principle requires only a single measurement to map a 3D point. On the other hand, bare LiDAR points do not carry color/texture information like PCs/meshes as derived from imagery. The accuracy of ALS points depends on the accuracy of the trajectory. On the contrary, the accuracy of MVS points is correlated with the GSD which, theoretically, can be scaled arbitrarily. For a detailed comparison of these two capturing methodologies, we refer to Mandlburger et al. (2017).

Initially, photogrammetry and laser scanning have been competitive systems with individual processing pipelines. However, at present, they are seen as complementary systems whose fusion results in more complete and better products. Nowadays, joint acquisition of photogrammetric and ALS data is state of the art for airborne systems and starts to emerge even for UAV-based systems (Mandlburger et al., 2017; Cramer et al., 2018). Recently, Glira et al. (2019) proposed the hybrid orientation of ALS PCs and aerial imagery, which improves the georeferencing accuracy of ALS data by integrating stabilizing image block geometry into the strip adjustment. As a side product, the hybrid orientation enables a precise co-registration of imagery and LiDAR data.

Concerning the recent hybridization trend, from our point of view, enhancing
3D PCs to textured meshes may replace unstructured PCs as default representation for urban scenes in the future. Intrinsically, meshes facilitate multi-modal data fusion by utilizing LiDAR points and MVS points for the geometric reconstruction while leveraging high-resolution imagery for texturing (hybrid data storage). Therefore, meshes are realistic-looking 3D maps of our real world and are easily understandable—even for non-experts. Besides benefits for visualization, textured meshes have other favorable characteristics. Whereas PCs are unordered sets of points, meshes are graphs consisting of vertices, edges, and faces that provide explicit adjacency information. Meshes are less memory-consuming than PCs since meshing algorithms try to minimize the number of entities while reconstructing the maximum of detail. Before the meshing, PCs will be filtered in such a way that only geometrically relevant points are kept. This embraces noise filtering and filtering of points that can be approximated by the same face (e.g., points on planar surfaces). Furthermore, there will be geometric simplifications based on the desired level of detail and, as the case may be, due to 2.5D mesh geometry. By definition, meshes are surface descriptions that cannot handle multi-target capability like LiDAR PCs. This inevitably leads to a drop in entities to be stored. Moreover, the high-resolution texture information is stored in texture atlases avoiding redundant image content. Therefore, textured meshes provide geometric and textural information in a lightweight fashion. Aside from these structural differences, georeferencing issues of imagery and LiDAR data will cause discrepancies between imagery, PCs, and meshes, too.

Being a hybrid data storage, we believe that the mesh modality is ideally suited to foster multi-modal semantic analysis. To this end, we chose the mesh to be the core of the proposed multi-modal linking and transferring pipeline. The aims and objectives of the paper are postulated in subsection 1.1. In section 4, we describe the entire methodology including the explicit entity linking and subsequent information transfer by deep-diving its building blocks. Subsection 4.1 describes the association of LiDAR data and the mesh in 3D space. Subsection 4.2 outlines the association of imagery and the mesh. Both association
mechanisms operate face-centered. Eventually, the combination of both links 3D points and pixels (cf. subsection 4.3). Subsection 4.4 discusses in detail prerequisites and particular challenges due to the mentioned (structural) discrepancies between the mesh, the PC, and imagery. In section 5, we demonstrate the proper working of the presented association mechanism on two real-world data sets (Gerke et al., 2014; Cramer et al., 2018). Since Ground Truth (GT) is not available across all modalities, a quantitative analysis of the proposed method is difficult. Therefore, we use the proposed label transfer to verify and showcase the proposed methodology. Moreover, we report the best performing parameters for the association mechanism concerning the used imagery and ALS data. We briefly present the used data and the key parameters in section 3. Both data sets provide significantly different resolution and co-registration quality.

1.1. Aims and Objectives

Our key contribution is the explicit linking of pixels, points, and faces to jointly leverage information from available data sources aiming at multi-modal semantics (cf. section 4). Each face will be linked to several points and several pixels. In turn, points and pixels are linked while checking the visibility via the mesh. To the best of our knowledge, there is no other holistic approach that explicitly joins imagery, mesh, and LiDAR data. The explicitly established connections on the entity-level are used to share information across modalities. Depending on the entity relationship, the information is aggregated prior to the transfer. The aggregation of features is achieved by calculating the median; label aggregation is achieved by majority vote (cf. Table 2).

In this study, we seem to focus on the label transfer since the effectiveness of the method can be shown better with labels than with features. Furthermore, annotated GT is of great importance for training supervised classifiers, particularly Deep Learning (DL) approaches. However, GT generation is tedious, time-consuming, and expensive work wherefore real-world GT availability is a rarity. In particular, pixel-wise GT generation is labor-intensive. Generally, there is a lack of GT data sets that jointly provide PCs and oriented imagery.
(and in this way textured meshes). Therefore, available annotations are limited to a single representation and prevent exploiting the potential of multi-modal training. This fact further motivates the necessity for the proposed approach and emphasizes its utility. Our linking and transferring methodology facilitates the consistent labeling of various representations, given a manually annotated representation initially. Hence, our method may help to overcome the imbalance of labeled entities among modalities. For instance, 3D data (PC or mesh) can be projected into various images at a stroke (cf. Figure 10). Therefore, it minimizes the manual effort for GT generation and helps considerably to foster modality-wise training of algorithms.

Our methodology is designed to process real-world data handling structural differences and co-registration issues. To deal with the huge amount of redundant multi-modal data, the association operates in a tiled and parallelized fashion while aiming at a low memory footprint. For the sake of good scientific practice, we critically reflect the preconditions and explore where the proposed approach might be limited (cf. subsection 4.4).

To summarize, the proposed mesh-centered multi-modal entity linking serves as the backbone to share features and labels across entities. Representation-specific features and (manually generated) annotations can be shared at a stroke. Thus, the methodology allows the juggling with modalities and injects great flexibility and versatility. The method enables the generation of multi-modal feature vectors and consistent annotation across modalities. By these means, the proposed association mechanism fosters joint semantic analysis and consequently contributes to the completion of the hybrid processing pipeline.

2. Related Work

The semantic segmentation of 3D data has become a standard task in the domain of photogrammetry and remote sensing. The increasing availability of simultaneously acquired airborne data with different acquisition methods calls for multi-modal fusion and scene analysis (cf. subsections 2.1 and 2.2). Generally,
the semantic analysis deals with various representations such as imagery, voxels, PCs, and meshes. Regardless of modality, state-of-the-art Machine Learning (ML) methods rely on a large amount of GT data. We briefly review available GT of geospatial data in subsection 2.3.

2.1. Multi-Modal Data Fusion

Due to complementary acquisition methods, multi-modal data acquisition has the potential to generate more complete and more detailed mapping products. Thereby, multi-modal products feature improved geometric reconstruction and semantic analysis. However, to the best of our knowledge, multi-modality is kept at a minimum and hence scratches only the surface. For instance, the fusion of imagery and ALS data on the point-level is commonly confined to the colorization of ALS points (Gerke et al., 2014; Cramer et al., 2018). To the best of our information, the explicit fusion of ALS points and MVS points and its contribution to semantics has not yet been investigated. Possible reasons might be structural and georeferencing discrepancies across modalities as discussed in section 1 and the huge memory footprint as caused by redundant multi-modal capturing. Glira et al. (2019) propose a methodology to jointly orientate imagery and ALS data which simplifies the fusion of the derived PCs as a side effect. Recently, there are software solutions that enable data fusion of multi-modal PCs and refine the fusion on the mesh-level. For instance, software SURE by nFrames (Rothermel et al., 2012) produces meshes as generated from LiDAR and MVS dealing with orientation discrepancies of few GSDs.

As outlined in subsection 2.2, the majority of works for semantic interpretation involves only one modality in the narrow sense. In most cases, multimodality is a means to an end that allows abusing annotated data and well-performing classifiers of another modality. To give an example, the well-established and fast semantic segmentation of images is mostly abused as a proxy to 3D scene analysis (Boulch et al., 2017; Lawin et al., 2017; He and Upcroft, 2013; Su et al., 2015; Kalogerakis et al., 2010). Theoretically, any quantity can be projected into image space adding another channel to the image. In practice,
the curse of dimensionality prevents the projection of an arbitrary number of quantities. Peters and Brenner (2019) highlight issues of associating PCs and imagery, particularly time-shifts and occlusions. To by-pass the occlusion problem, they approximate the 3D surface by voxelization of the PC.

Our work differs from existing works since it explicitly aims at a holistic multi-modal data fusion of imagery, PCs and meshes. Thereby, the mesh acts as core modality to solve the occlusion problem. The subsequent information transfer shares features and labels with all modalities. The association of an ALS PC and a challenging 2.5D mesh is already described in Laupheimer et al. (2020a). In the current work, we improve the implementation to cope with 3D meshes with a significantly larger memory footprint than 2.5D meshes. Moreover, we extend the association mechanism to image space (cf. subsection 4.2 and subsection 4.3) and enable information transfer in arbitrary directions (cf. Table 2).

2.2. Semantic Segmentation of 3D Data

DL methods, particularly Convolutional Neural Networks (CNNs), are state of the art for semantic segmentation in image space (Garcia-Garcia et al., 2017; Minaee et al., 2020). Therefore, it seems reasonable to apply well-established DL methods of the image space to PCs. However, the unstructured nature of 3D PCs prevents to apply CNNs directly to them. To overcome the non-Euclidean design, PCs are commonly structured into grid-like 3D or 2D representations by voxelization or multi-view rendering respectively. Several works voxelize the PC and train a supervised classifier. The predicted labels for the voxels will be transferred to all contained points (Hackel et al., 2016; Huang and You, 2016). Voxelization comes along with memory overhead. Therefore, much effort is put into networks that use sparse 3D convolutions (Graham et al., 2018). This approach has been successfully applied to urban PCs (Schmohl and Sorgel, 2019). Detouring via image space, multi-view approaches leverage well-performing semantic image segmentation methods. The per-pixel predictions are back-projected to 3D space (Boulch et al., 2017; Lawin et al., 2017). To give an example, He and Upcroft (2013) segment stereo images semantically, create
the MVS PC, and back-project the 2D semantic segmentation results to the PC. The grid-like proxy enables the use of CNNs but comes along with information loss due to discretization, occlusions, and projection.

The rise of PointNet and its hierarchical successor PointNet++ constitutes a milestone in semantic PC segmentation since they operate directly on unstructured 3D PCs (Qi et al., 2017a,b). Winiwarter et al. (2019) successfully applied PointNet++ to geospatial PCs. The gist of PointNet is to use a symmetric function during encoding to be independent of set permutation. The entire PC is encoded by a global feature vector, which is attached to each encoded per-point feature vector. Operating only on a global scale, PointNet misses local context. Its extension PointNet++ hierarchically applies PointNets to the iteratively subsampled PC and, hence, operates on several scales. This procedure mimics hierarchical feature learning with increased contextual information similar to CNNs in image space. Likewise, Boulch (2019) introduces continuous convolutional kernels that can be applied directly to PCs. Griffiths and Boehm (2019a) review the current state-of-the-art DL architectures for processing 3D data. Xie et al. (2020) review semantic PC segmentation comparing DL and traditional ML approaches. While DL approaches do not require handcrafted features, they rely on a large amount of training data. In contrast, traditional ML depends on handcrafted features and therefore provides better interpretability. Weinmann et al. (2015) calculate and select features based on various vicinities and subsequently perform a semantic segmentation with Random Forest (RF). To avoid noisy predictions, Landrieu et al. (2017) extend the previous pipeline by structured regularization, a graph-based contextual strategy. Likewise, Niemeyer et al. (2014) avoid noisy results utilizing CRF-based methods as statistical context models. Vosselman et al. (2017) first segment the data and subsequently perform the semantic segmentation.

Ahmed et al. (2018) show advances of DL on different 3D data representations. They discuss representation-specific challenges and highlight differences between Euclidean and non-Euclidean data. The emerging field of geometric DL extends basic DL operations to non-Euclidean domains such as graphs and man-
ifolds in order to use topological information (Bronstein et al., 2016). PCs do not provide topological information per se. Therefore, Landrieu and Simonovsky (2017) organize PCs in Superpoint Graphs (SPGs). Ali Khan et al. (2020) transform PCs to an undirected symmetrically weighted graph encoding the spatial neighborhood and apply a Graph Convolutional Network. Chang et al. (2018) propose the Structure-Aware Convolutional Neural Network (SACNN) that uses generalized filters, which aggregate local inputs of different learnable topological structures. By that, SACNNs work with both Euclidean and non-Euclidean data. To summarize, the adaption of (geometric) DL methods contributed to substantial progress in the field of semantic PC segmentation in the last decade.

On the contrary, mesh interpretation has hardly been explored by the community of photogrammetry and remote sensing although recent years show increasing interest in meshed 3D models - particularly, for applications like smart city models (Boussaha et al., 2018). In comparison, meshes are a default data representation in the domain of computer vision. However, that community typically deals with small-scale (indoor) data sets (Kalogerakis et al., 2010). In contrast to photogrammetric meshes, texture is not an inherent characteristic of these meshed models. By analogy to semantic PC segmentation, common approaches for semantic mesh segmentation make a circuit to 2D image space to take advantage of image-based DL methods. Those approaches render 2D views of the 3D scene, learn the segmentation for different views, and finally, back-project the segmented 2D images onto the 3D surface (Su et al., 2015; Kalogerakis et al., 2010). Wu et al. (2015) voxelize the mesh and apply a convolutional deep belief network. Qiao et al. (2019) propose a geometric DL approach that encodes the mesh connectivity using Laplacian spectral analysis and aggregates global information via mesh pooling blocks. MeshCNN mimics traditional CNN convolution and pooling operations (Hanocka et al., 2019). The specialized layers operate on the edges and leverage the intrinsic topological information. Schult et al. (2020) propose the DualConvMesh-Net that combines geodesic and Euclidean convolutions on 3D meshes. Geodesic convolutions utilize the underlying mesh structure and help to separate spatially adjacent but
disconnected surfaces. In contrast, Euclidean convolutions establish connections between nearby disconnected surfaces.

Notwithstanding, semantic segmentation of real-world large-scale meshes is a mostly overlooked topic. Rouhani et al. (2017) gather faces of a MVS mesh into so-called superfacets and train a RF using geometric and photometric features. Tutzauer et al. (2019) utilize a DL approach by training a multi-branch 1D CNN with contextual features and compare the achieved results to a RF. They show that color information is beneficial for semantic mesh segmentation. More precisely, Laupheimer et al. (2020b) attest that per-face color information (i.e. texture) outperforms per-vertex color information (e.g. colored PC) by evaluation of several radiometric feature qualities. However, they also show the inherent limitations of texture due to occlusions, absence of imagery, and the quality of the geometric reconstruction.

2.3. Ground Truth Availability

Garcia-Garcia et al. (2017) and Minaee et al. (2020) review available GT data in image space, 2.5D and 3D space. Annotated imagery often aims at the pure semantic segmentation, wherefore orientation information is not provided (Lambert et al., 2020). The ISPRS 2D Semantic Labeling Contest provides manually annotated orthophotos of Vaihingen and Potsdam (Gerke et al., 2014). Griffiths and Boehm (2019a) list available GT data sets for RGB-D, multi-view, volumetric, and fully end-to-end architecture designs as acquired by various platforms. Xie et al. (2020) review publicly available annotated PCs and discuss their shortcomings.

The computer vision community provides annotated mesh data for indoor scenes (Armeni et al., 2017; Hua et al., 2016; Dai et al., 2017) or for single objects (Shilane et al., 2004). However, to the best of our knowledge, there are no labeled meshed models that cover urban scenes. In contrast, there are many available labeled urban data sets for 3D PCs provided by the community of photogrammetry and remote sensing (Zolanvari et al., 2019; Wichmann et al., 2018; Niemeyer et al., 2014; Hackel et al., 2017).
The rise of data-hungry DL methods demands efficient strategies for GT generation. Synthetically generated GT such as provided by [Griffiths and Boehm (2019b)] boost the generation process per se. However, purely synthetic data is limited by its diversity. [Kölle et al. (2020)] exploit crowdsourcing and active learning to minimize manual labeling effort. [Ramirez et al. (2019)] present a virtual reality tool that gamifies the manual labeling of meshes and PCs.

Our proposed methodology is able to derive consistently labeled meshes and imagery from publicly available annotated real-world PC data and vice versa (provided that the necessary data is available and oriented, cf. section 5). To the best of our knowledge, yet, there is no data set that provides consistently labeled modalities. The proposed labeling tool has the potential to accelerate multi-modal GT generation and consequently multi-modal semantic analysis.

3. Data

To demonstrate the effectiveness of our association mechanism, we utilize the publicly available ISPRS benchmark data set Vaihingen 3D (V3D) and a proprietary data set which will be made publicly available in mid 2021 (Cramer, 2010; Cramer et al., 2018). The original purpose of the proprietary data set aims at the deformation monitoring of the ship lock and its surrounding in Hessigheim, Germany. Thus, challenging water surfaces are part of the acquired data. We refer to this data set as Hessigheim 3D (H3D). Although being already captured in 2008, V3D may still be representative of large-scale country-wide mapping. On the contrary, H3D is an example of small-scale mapping applications with high-resolution imagery and LiDAR data.

In both cases, imagery and ALS data have been acquired from airborne platforms. Whereas V3D data is captured from airplane, H3D data is captured from UAVs. V3D data has been acquired asynchronously. The time-shift between nadir imagery (GSD = 8 cm) and ALS acquisition (4–8 points/m²) is several weeks. H3D provides two sets of oriented imagery: oblique and nadir. Oblique imagery (GSD = 2.5 cm) has been acquired simultaneously along with ALS data.
(400–800 points/m²) from the same UAV. Nadir images (GSD = 3.7 mm) have been acquired from another UAV with a time-shift of several hours to one day. Accordingly, the number of entities and the memory footprint is higher for H3D. Table 1 lists key parameters of both data sets relevant for the underlying study.

| Data Set | Imagery | PC | Mesh |
|----------|---------|----|------|
| V3D 870 m × 700 m | GSDnadir = 8 cm | 4–8 points/m² | 16 tiles |
| | 15 images | 1.2 M points | |
| | @ 14430 px × 9420 px | | |
| H3D 670 m × 255 m | GSDoblique = 2.5 cm | 400–800 points/m² | 94 tiles |
| | 1979 images | | |
| | @ 6000 px × 4000 px | 24.6 M faces | |
| | GSDnadir = 3.7 mm | 138.1 M points | source: MVSoblique + ALS |
| | 524 images | | |
| | @ 11608 px × 8708 px | | |

Table 1: V3D and H3D properties. The project area is given by the mesh tiles that intersect with the labeled LiDAR cloud. The face count is adapted to the overlapping LiDAR cloud.

For both data sets, we generate textured and tiled meshes with SURE 4.0.2 from nFrames. We set tile sizes to 175 m × 175 m (V3D) and 50 m × 50 m (H3D) respectively. The chosen tile sizes empirically showed to be a good compromise between fast tile-wise processing and small tile count. For H3D, we generate a hybrid textured mesh by fusing the simultaneously acquired ALS data and oblique imagery. Utilizing oblique imagery ensures proper texturing of vertical faces such as facades. In contrast, we generate a purely photogrammetric mesh for V3D since the time-shift of imagery and ALS data is roughly one month. For this reason, the geometric and radiometric quality of the H3D mesh outperforms the V3D mesh. However, since the V3D mesh is purely photogrammetric, the relative orientation of imagery and mesh fits perfectly. We determine the shifts between mesh and PC data with the Iterative Closest Point (ICP) algorithm. We do not apply the determined shifts to prove the effectiveness of our association methodology. Moreover, ICP does
not solve the co-registration problem entirely. The V3D ALS data is shifted against the MVS mesh by $\Delta X = 31$ cm, $\Delta Y = -9$ cm, $\Delta Z = 9$ cm. Rephrased, the co-registration of imagery and ALS data differs significantly. The H3D ALS data is shifted against the mesh (as generated from LiDAR and MVS points) by $\Delta X = 8$ mm, $\Delta Y = 10$ mm, $\Delta Z = 6$ mm. The significantly different data sets featuring co-registration issues in 3D space are adequate to showcase the robustness and flexibility of our implementation.

Both data sets carry manual annotations for the LiDAR cloud (Niemeyer et al., 2014; Kölle et al., 2019). The label scheme of H3D is oriented towards the V3D label scheme but is more fine-grained. Furthermore, it has been manually enhanced by class Chimney/Antenna (Laupheimer et al., 2020). Figure 2 shows the union of textured mesh tiles that overlap with the labeled LiDAR cloud for both data sets. The label schemes and respective color-codings are given in the figure caption.

4. Methodology

We aim for a holistic explicit linking of the common data representations in the domain of photogrammetry and remote sensing: imagery, PC, and mesh. The backbone of the proposed association methodology consists of two geometry-driven parts: (a) Point Cloud Mesh Association (PCMA) which links faces and points (cf. subsection 4.1) and (b) Image Mesh Association (ImgMA) which links faces and pixels across images (cf. subsection 4.2). Coupling both association mechanisms yields to (c) Point Cloud Image Association (PCImgA) (cf. subsection 4.3). The PCImgA establishes a connection between points and imagery via the mesh as a mediator. Point visibility is implicitly given through the mesh. Table 2 illustrates the total association mechanism with iconic pictograms.

The established connections between the entities across the distinct representations enable an information transfer that allows features and labels to be shared arbitrarily. Table 2 compactly lists the information transfer operations depending on information type (feature or label) and transfer direction.
Figure 2: Top views of V3D (left) and H3D (right) depicting the annotated LiDAR PC and the respective overlapping mesh tiles in textured fashion. The annotated ALS data is color-coded utilizing the following label schemes.

V3D:
- Power Line (black), Low Vegetation (light green), Impervious Surface (gray), Car (blue), Fence/Hedge (yellow), Roof (red), Facade (white), Shrub (dark green), and Tree (green).

H3D:
- Power Line (black), Low Vegetation (light green), Impervious Surface (gray), Vehicle (blue), Urban Furniture (lilac), Roof (red), Facade (white), Shrub/Hedge (orange), Tree (green), Open Soil/Gravel (brown), Vertical Face (yellow), Chimney/Antenna (magenta).
(a) Point Cloud Mesh Association (PCMA)

| Feature Transfer | Label Transfer |
|------------------|----------------|
| Copy Value       | Copy Value     |
| Mesh $\rightarrow$ PC ($n_{pts}:1$) | PC $\rightarrow$ Mesh ($n_{pts}:1$) |
| Median Aggregation | Majority Vote |

(b) Image Mesh Association (ImgMA)

| Feature Transfer | Label Transfer |
|------------------|----------------|
| Copy Value       | Copy Value     |
| Mesh $\rightarrow$ Img ($n_{pc}:n_{img}$) | Img $\rightarrow$ Mesh ($n_{img}:n_{pc}:1$) |
| Median Aggregation | Majority Vote |

(c) Point Cloud Image Association (PCImgA)

| Feature Transfer | Label Transfer |
|------------------|----------------|
| Median Aggregation | Majority Vote |
| PC $\rightarrow$ Img | Img $\rightarrow$ PC |

Table 2: Overview of the proposed method. For each association mechanism, the transfer operations are given in dependence of the information type (feature or label) and the transfer direction. The pictograms on the right depict the linking of the respective entities. PCImgA provides two association modes: implicit and explicit linking (cf. subsection 4.3). (a), (b) and the implicit version of (c) are face-centered. The relationship of implicit PCImgA is described by $n_{pts}:1:n_{pc}:n_{img}$. The explicit version is pixel-centered (PC $\rightarrow$ Img: $n_{pts}:1:n_{img}$) or point-centered (Img $\rightarrow$ PC: $n_{img}:n_{pc}:1$).
for each part of the entire association mechanism. Concerning the scalability of the proposed multi-modal association approach, we process data tile-wise in a parallelized fashion while keeping the memory footprint low.

4.1. Point Cloud Mesh Association (PCMA)

The PCMA explicitly links faces and points in a face-centered geometry-driven approach. Each face (represented by its Center of Gravity (COG)) is assigned with \( n_{pts} \) points that represent the same surface by following three steps: (i) clipping of the PC to a spherical vicinity of the COG, (ii) filtering of out-of-face points, and (iii) filtering of off-the-face points \( \text{[Figure 3]} \). Out-of-face points are not enclosed by the face borders when projected orthogonally onto the face plane. Off-the-face points do not coincide with the face plane, i.e. they are below or above the face surface. A manually set threshold \( \theta \) decides whether a point coincides with a face or not. Both point types are not mutually exclusive and exist due to the simplification during the meshing, the representation type differences as discussed in section 1 and geometry differences (e.g. in case of 2.5D mesh geometry or due to asynchronous data acquisition).

![Figure 3: Steps (i) - (iii) of the PCMA.](image)

(i) Clipping of the PC (black dots) to the vicinity (blue sphere) of the considered face. Its COG is marked with a black cross. The mesh surface and its vertices are depicted in green.

(ii) Filtering of out-of-face points based on the clipping result (orthogonal view concerning the face surface).

(iii) Filtering of off-the-face points (side view with respect to the face). The face is depicted as a black line. The threshold band is marked in gray.
At first, we roughly reduce the search space for each face in order to accelerate
the association. To this end, we build a kD tree for the PC (tile) and query
the built tree with COGs of all faces. Thereby, we detect all points within dis-
tance $r$ for each face (ball query). The query parameter $r$ is set in dependence
of the manually set association threshold $\theta$ and the maximum distance $t_{\text{max}}$ of
the COG to the respective face vertices. Geometrically, $r$ is set to the length of
the hypotenuse of the triangle as defined by $\theta$ and $t_{\text{max}}$. In simple terms, the
query parameter $r$ is set to the minimum distance that guarantees the manually
set threshold $\theta$ to be effective for the entire face while enclosing the entire face
(cf. Figure 4). Hence, $r$ prevents prefiltering of points by a too small spher-
ical vicinity. The ball query delivers a subset of points, which may contain off-the-face points and out-of-face points.

$$r = \sqrt{t_{\text{max}}^2 + \theta^2}$$

Figure 4: Definition of radius $r$ (blue) for the PC clipping (step (i) of PCMA) shown as side
view with respect to the face (dashed black line). Radius $r$ guarantees to enclose the entire face by enclosing the maximum distance $t_{\text{max}}$ (dashed horizontal orange line) of the COG (black cross) to face vertices. Besides, $r$ avoids to prefilter points by enclosing threshold $\theta$ (dashed vertical orange line) across the entire face. The threshold band is marked in gray.

Second, we filter out-of-face points by neglecting subset points whose or-
thogonal projections on the face plane are not enclosed by the face outline. For
details, we refer the interested reader to [Laupheimer et al. (2020a)]. Visually,
the result of (ii) is the intersection of the spherical subset with radius $r$ and the
infinite triangular prism as defined by the face and its normal vector. We refer
to this as the association prism.

Finally, we filter the remaining off-the-face points. For this purpose, we
calculate the orthogonal distance for each remaining point to the face plane. If
the distance exceeds a chosen association threshold $\theta$, the point is not associated
with the face. Since we have to compensate several discrepancies between PC
and mesh, we use a more sophisticated adaptive thresholding with an arbitrary user-defined number of filter levels \( n_l \).

Each level \( l \) consists of two independent thresholds \( \theta^+_l \) and \( \theta^-_l \) limiting the association prism in the normal direction or the opposite direction respectively. The absolute threshold values increase with ascending level. Starting from level 1, the algorithm tries to associate points with the respective thresholds \( \theta^+_1 \) and \( \theta^-_1 \). If points have been linked, the association stops. Otherwise, the next level \( l \) is activated. This adaptive thresholding accelerates the association process. On the other hand, by nature of our approach, not all points might be associated. Here, our reasoning is to favor near-surface points at the cost of missing to link a few points (fast small margin association).

The association information is stored as a per-point attribute. For each associated point, the respective face index is attached to its attributes. Non-associated points are marked with \(-1\). The stored indices trivialize the transfer of features and labels from the mesh to the PC (Mesh \( \mapsto \) PC in Table 2). In this case, we copy the desired values to the PC at a stroke due to the one-to-many relationship. Reversely, the stored face indices can also be used to transfer features and labels from the PC to the mesh (PC \( \mapsto \) Mesh in Table 2). However, to speed up the transfer, we directly couple the information transfer with the association mechanism. The many-to-one relationship calls for information aggregation. For each face, we derive robust median features as gathered from the PC. Features may embrace sensor-intrinsic and handcrafted features such as pulse characteristics and derived quantities (Eitel et al., 2016). Analogously, majority votes determine the per-face labels as transferred from the associated points. Therefore, the association inherently is a label transfer tool and feature calculation tool (median features). Non-associated faces are marked with \(-1\) and receive zeroed median features (Laupheimer et al., 2020a).

4.2. Image Mesh Association (ImgMA)

The ImgMA explicitly links faces and pixels across various images in a geometry-driven approach. To accelerate the ImgMA we make use of the given
mesh tiling. Each pixel is assigned with the visible face as detected by the following three steps: (I) preselection of visible mesh tiles per image utilizing Minimum Bounding Boxes (MBBs) of the tiles, (II) ray casting per image and tile (image-tile-pair), and (III) fusion of ray casting results per image via depth filtering across tiles. We end up with $n_{px}$ associated pixels across $n_{img}$ images for each face. Figure 5 shows the workflow by means of an oblique example image and two vertically separated tiles.

Figure 5: Steps (I) - (III) of the ImgMA shown in accordance with the information transfer $\text{Mesh} \rightarrow \text{Img}$ (cf. Table 2). The depicted example shows the association of an image (lower right) with two vertically split mesh tiles of H3D. (I): Preselection of visible mesh tiles per image shown schematically in isometric (left) and top view (right). The stretched camera pyramid (green) intersects with some MBBs (blue). Non-intersecting MBBs are marked in reddish. Dashed lines indicate non-visible parts. (II): Ray casting per image-tile-pair. The reddish area shows where ray casting fails due to missing intersections of image rays and considered mesh tiles. Black indicates intersection with unlabeled faces. (III): Final result after fusing ray casting results per image via depth filtering across tiles.

For each image, we first detect visible tiles and perform the subsequent ray casting procedure in a parallelized fashion for the subset of tiles only. We define a tile to be visible if its MBB intersects with the stretched camera pyramid (MBB visibility check). The stretched camera pyramid is defined by the projection center and the projection rays crossing the corner pixels of the respective
image. The lowest of all MBB faces limits the stretched camera pyramid. Since
the MBBs are not fully occupied by the enclosed mesh tiles, some detected tiles
might not contain any visible faces. Nonetheless, this approach significantly re-
duces the number of tiles that have to be processed by the ray casting procedure
for each image. We detect intersections of the camera pyramid and a MBB by
a three-stage check starting with the most likely and fastest test. At first, we
check for each tile if any corner point of the respective MBB is inside the cam-
era pyramid (point in polyhedron test). The second and third stage perform
edge face intersections, checking intersections of pyramid edges starting from
the projection center and any MBB face or intersections of MBB edges and any
pyramid face. Once a test succeeds, the respective enclosed tile is marked as
visible and the residual checks are omitted (check omission).

As a result of (I), we receive a list of visible tiles for each image, i.e. each
image \(img\) is linked to \(n_{tiles}^{img}\) tiles. Hence, there are \(n_{tiles}^{img}\) image-tile-pairs for
each \(img\). Vice versa, a list of visible images for each tile is stored. For each
image-tile-pair, visible faces are determined via ray casting. For this purpose,
for each pixel, a 3D projection ray is created and intersected with the mesh faces
of all linked tiles. The intersected faces are candidates for the final association
result. Concerning a single image-tile-pair, all candidate faces are truly visible.
However, an image probably covers multiple tiles, and consequently, some faces
might be occluded by faces of another tile (cf. Figure 5). For the final result,
we fuse the \(n_{tiles}^{img}\) ray casting results across the visible tiles into one final ray
casting result per image.

Implementationally, the fusion across the \(n_{tiles}^{img}\) image-tile-pairs is done im-
PLICITLY by depth updates. Initially, each pixel is associated with a near-infinite
depth value. The association information is updated whenever a candidate face
reduces the depth value. Hence, the final association information is steered by
faces of minimum depth that are truly visible (i.e. faces that mark the first
intersection along the respective ray). To speed up the implementation, steps
(II) and (III) are parallelized with respect to images. Each process handles
\(n_{tiles}^{img}\) image-tile-pairs per image. The implicit fusion reduces the memory foot-
print of the algorithm since only the final ray casting result has to be stored.

To minimize the memory footprint, we avoid storing the association information channel-wise due to the curse of dimensionality. Particularly for oblique images, only a small part of the image may be associated due to the limited reconstruction area of the mesh. Instead, we store the final association information as a sparse pixel cloud per image consisting only of pixels that have been linked with a face. The sparse pixel cloud contains tuples of associated pixel positions, the depth, the tile-dependent face index, and optionally, other attributes (e.g., labels) as transferred from the mesh.

The stored face indices trivialize the transfer of features and labels from the mesh to the images (Mesh $\mapsto$ Img in Table 2). We copy the desired quantities to the linked pixels of the associated images at a stroke due to the one-to-many-to-many relationship. Reversely, the many-to-many-to-one relationship calls for information aggregation (Img $\mapsto$ Mesh in Table 2). For each face, we derive robust median features as gathered from the associated pixels across the respective images. Features may embrace sensor-intrinsic multi-spectral information, handcrafted features, and features as derived by DL pipelines. Analogously, majority votes determine the per-face labels as transferred from the linked pixels.

4.3. Point Cloud Image Association (PCImgA)

The PCImgA aims for the linking of pixel locations and 3D points. Theoretically, the collinearity equations establish an explicit relationship between 3D points and pixels. Each point can be projected into the image space given the exterior and interior orientation of the respective image. However, the bare projection cannot check for visibility, and hence, links visible and non-visible points with imagery. Therefore, point visibility has to be checked prior to the linking of pixels and 3D points. To this end, we leverage the mesh representation by combining mechanisms PCMA and ImgMA implicitly and explicitly. The implicit linking is face-centered whereas the explicit linking is point-centered or pixel-centered (dependent on the transfer direction). Making a detour via the mesh largely solves the visibility problem for 3D points in image space.
The implicit linking couples the association mechanisms PCMA and ImgMA by simply executing them sequentially. Specifically, \( n_{px} \) pixels of \( n_{img} \) images and \( n_{pts} \) points are exclusively linked to the respective face. For the information transfer, the information from the starting representation is gathered per face and transduced to the target representation. Therefore, the joint face apparently establishes a linking of points and pixels. The per-face label and features as derived from the starting representation are determined via majority voting and median aggregation respectively (cf. Table 2). Subsequentially, the per-face aggregations are copied to the target modality (one-to-many relationship).

On the contrary, the explicit linking couples the association mechanisms PCMA and ImgMA by leveraging the stored association information and utilizing the collinearity equations. As a result of ImgMA, visible faces for each image are known. At the same time, mechanism PCMA delivers the associated points for each face. Consequently, associated points of visible faces are marked as visible. Therefore, the combination of PCMA and ImgMA results in a visible subset of the PC per image. The collinearity equations explicitly link the visible points and the pixel locations across all imagery. Therefore, explicit linking truly associates points and pixel locations.

For each point, there is an unambiguous pixel location in each image. However, the reverse situation is ambiguous. Depending on GSD and point density, each pixel of each image may enclose several visible points. Therefore, transferring information from the PC to imagery (PC \( \rightarrow \) Img in Table 2) demands a pixel-wise aggregation (many-to-one relationship per pixel and image). The features and labels are aggregated by median aggregation and majority voting respectively. To accelerate the process, we approximate the pixel-wise aggregation by transferring only information of the point of minimum depth. This approximation simplifies the association to a one-to-one relationship per pixel and image. Likewise, a one-to-one relationship holds if the GSD is smaller than the point distance. Since each point is covered by several images, the information transfer Img \( \rightarrow \) PC requires a point-wise aggregation across images.
4.4. Preconditions and Limitations

The proposed method connects 3D PCs, photographic imagery (following the central perspective), and textured meshes. By nature of the algorithm, it merely depends on the pure existence of those three modalities. Therefore, it is a generic approach that works with any photographic image and PC regardless of the acquisition platform (aerial, terrestrial, mobile), the image type (panchromatic, RGB, multi-spectral), and PC type (MVS cloud, LiDAR cloud, persistent scatterer cloud). However, we focus on the linking of aerial RGB imagery and ALS PCs along with the respective textured 3D mesh (cf. section 5). Furthermore, photogrammetric meshes, LiDAR meshes, or hybrid meshes can be processed. The linking is not constraint to a specific mesh generation algorithm or mesh geometry (2.5D or 3D). However, the entire association benefits from good geometric reconstruction. We are aware of the fact that proper meshing of our complex world is a hard task and still subject to research.

A proper reconstruction ensures an appropriate entity linking and information transfer. As a general rule, the better the mesh represents the true 3D structure of the real world, the better works the proposed association mechanism and the subsequent information transfer. Obviously, as a precondition to the association, the considered data representations have to cover the same area and have to be oriented in the same coordinate system.

Assuming a proper co-registration, entities across representations can only be associated when underlying real-world objects are captured or reconstructed in all representations. In other words, each face should at least enclose one point or one pixel respectively. Reversely, each point or pixel should be mapped onto a corresponding face. However, these relationships do not always exist due to inter-representation differences such as object penetration (cf. section 1, Laupheimer et al., 2020a). For instance, due to occlusion during data acquisition, a facade may not be captured in the ALS data, and hence the respective faces cannot link any points. Reversely, due to mesh simplification, a thin-structured object may not have been reconstructed entirely in the mesh, but the object is fully captured in the PC and imagery. Additionally, there might
be inter-representation discrepancies due to asynchronous data acquisition. Figure 6 shows both situations for the H3D data set.

Figure 6: Discrepancies and data gaps in 3D models of H3D. Top: Urban area. Overlay of textured mesh and ALS PC (height-coded). Facades are reconstructed entirely in the mesh but are captured only partially by the PC. Bottom: Ship lock area. The mesh (left) partially reconstructs the river but misses to fully reconstruct thin structures like light poles and power lines, which are captured entirely in the PC (right). The asynchronous data acquisition of images and ALS data cause inconsistencies between mesh and PC (e.g. for cars).

Misalignment among modalities is a decisive issue for multi-modality, wherefore proper co-registration is subject to current research. Ideally, imagery and PC data are co-registered simultaneously in a joint adjustment. Consequently, the derived mesh is aligned with both data sources and co-registration issues are obsolete. Nonetheless, reality shows that co-registration discrepancies are an important and real issue (cf. section 3). A good relative orientation of 3D data and imagery is beneficial to the linking of pixels with points and faces. We are aware of the fact that PCImgA and ImgMA depend on the quality of
the co-registration. However, the proper co-registration of both data sources is not the focus of this work. Figure 7 depicts the influence of the co-registration quality of imagery and 3D data.

Inherently, MVS meshes and MVS PCs are perfectly aligned with imagery. Hence, incorrect or missing associations between the representations are only due to the reconstruction quality and data gaps (as for V3D).

![Figure 7: Influence of the co-registration quality as visualized by the label transfer (left/right: good/bad relative orientation of 3D data and imagery). The yellow lines depict “epipolar lines” crossing roof corners in the RGB image (center). The orientation parameters as achieved by bundle adjustment have been slightly falsified artificially for the visualization on the right.](image)

Since the relationship of 3D space and image space is strictly defined by the collinearity equations, we highlight the discussion of PCMA. Figure 8 and Table 3 sketch discrepancies of PC and mesh despite representing the same real-world objects. These discrepancies are largely covered by the adaptive thresholding. Despite and due to this technique, not all points are associated with faces. There are three groups of unassociated points (cf. Table 3).

Visually, the association is done utilizing the per-face association prism which is limited by the range of manually set thresholds $\theta_i^+$ and $\theta_i^-$. The threshold-based approach filters all points whose orthogonal projections to the face plane exceed a specific value (cf. Table 3, A1). This prevents the linking of points and faces that most likely represent different surfaces. Furthermore, the adaptive thresholding breaks the association once a point is associated at any threshold level $l$ (”early stopping”). Hence, points that are closer to the
Figure 8: Discrepancies between PCs and the mesh (black line) caught by adaptive thresholding. *Left:* Black arrows indicate the discrepancy between the 2.5D mesh and the annotated 3D PC. *Center:* The noisy PC (blue) oscillates about the reconstructed mesh. *Right:* There might be misalignment between PCs (blue: MVS, orange: LiDAR) and the mesh as generated of a single source or complementary sources.

Table 3: The schematic drawings illustrate the three cases where faces (black lines, separated by black strokes) and points are not linked (side view with respect to the face). Non-associated points are marked in red, associated points in green. The association prisms are marked in blue. Adaptive thresholds are omitted except for A2. A2 depicts the increasing thresholds with increasing blueness. Diverging association prisms create dead zones like depicted in B (hatched in red). B1 mimics a perfect planar surface as dashed black line.
face than the maximum threshold may not be linked (cf. Table 3 A2). Levels \( l > 1 \) can be seen as a fall-back for scenarios where a proper association is not possible. The adaptive thresholding favors near-surface points by ensuring the association of points fulfilling the smallest threshold.

The adaptive thresholding facilitates a varying degree of freedom by the set thresholds and their inter-level spacing. Eventually, it balances the strictness of the point-face linking. Small-valued thresholds along with small-spaced levels enforce a tight coupling where only points close to the mesh surface are associated. Large values along with large level spacing loosen the coupling. Moreover, the asymmetric two-fold thresholding per level enables non-symmetric filtering improving flexibility and adaptiveness. Therefore, the presented association mechanism is agnostic to the geometric structure of mesh geometry: 2.5D or 3D meshes can be processed. The asymmetric adaptive thresholding allows associating faces with points where 2.5D and 3D geometry differ significantly while favoring the association of near-surface points (e.g. facades or tree stems, cf. Figure 8 on the left). Laupheimer et al. (2020a) discuss in detail the particular challenges for the PCMA using a 2.5D mesh.

The set of association prisms does not enclose all points of the PC. Particularly, points above and below the mesh surface may fall into dead zones not covered by the association prisms. The prisms of adjacent faces diverge when they form convex or concave surfaces, i.e. their normal vectors are not parallel. Non-perfect reconstructions of planar surfaces artificially introduce convex or concave structures (cf. Table 3 B1). Naturally, points above truly convex surfaces or below concave surfaces cannot be linked. Typically, points above the reconstructed roof ridge cannot be associated (cf. Table 3 B2). For this reason, co-registration discrepancies increase the number of non-associations.

The previously described missed associations are owed to the nature of the problem itself and the implementation aiming for a good trade-off of memory and speed (by adaptive thresholding). Besides, we declare points that are projected on the face edges or vertices to be out-of-face points (cf. Table 3 C). By these means, we avoid their linking by choice. Technically, these points belong
to two adjacent faces A and B. Hence, it is hard to decide whether to assign them to face A or its adjacent face B. Therefore, such points cannot be linked unambiguously and may cause ambiguity in the information transfer. Here, our reasoning is to link not all, but unambiguous points. As a side-effect, neglecting these points accelerates the association process. However, if desired, our implementation allows us to link these points, too [Laupheimer et al., 2020a].

To the best of our knowledge, meshing algorithms depend fully on geometry and do not incorporate semantics. Therefore, reconstructed faces do not necessarily represent semantic borders. For instance, consider the transition of a planar impervious surface to green space in the real world. The mesh representation may simplify this scenario to a single large face. On the contrary, the same scenario is captured properly in the PC and imagery. Consequently, too large triangles at class borders will associate points and pixels of different classes (cf. Figure 12). For this reason, semantically incorrect associations are unavoidable due to the meshing.

Naturally, the PCImgA inherits the limitations of its sub pipelines. Besides, there are specific issues regarding the PC visibility as derived via the mesh as a proxy. Faces are marked as visible once a pixel is associated with the face. Points, in turn, are marked as visible if they are associated with a visible face. However, a face marked as visible does not have to be entirely visible. Moreover, a face marked as visible does not have to be associated with points that are truly visible as well. The adaptive thresholding links points and faces along the normal directions whereas the line-of-sight is relevant for the point visibility. In other words, taken individually, the made associations by the sub pipelines are correct, but their composition does not guarantee a correct visibility check for each point-pixel relationship (using explicit PCImgA). On the contrary, the implicit PCImgA makes use of the correct point-face and face-pixel linking. Rephrased, the implicit linking overcomes georeference issues utilizing the adaptive thresholding and uses the correct visibility checks for faces. Figure 9 depicts point visibility and showcases apparently visible points. It is unlikely that all truly non-visible points that are marked as visible are occluded.
by truly visible points. For this reason, depth filtering helps only when truly non-visible and truly visible points are on the same projection ray. Otherwise, truly non-visible points are linked with pixels through collinearity equations causing incorrect information transfer (for the explicit linking of the PCImgA). Generally, smaller faces better represent the geometry and reduce the impact of truly non-visible points. However, large triangles are beneficial concerning processing time.

![Figure 9](image.png)

Figure 9: The sketches indicate truly (green) and apparently visible points (red) as detected by PCImgA utilizing the mesh (black wireframe) as a proxy. (a) showcases an apparently visible point due to being linked to a truly but non-fully visible face (blue). Fully visible faces are depicted in green. The dashed cubes on the right mimic the situation of the opaque cubes in “transparent” mode. (b) showcases apparently visible points due to the divergence of normal direction and line-of-sight. Points are associated along the normal direction of faces. Hence, visible faces might be linked to truly non-visible points.

5. Results and Analysis

We demonstrate the capability of our methodology, its flexibility, and adaptiveness to underlying data by deploying V3D and H3D (cf. section 3). To quantitatively analyze the linking methodology, GT data is necessary for each modality. However, to the best of our knowledge, there is no real-world data set that provides annotations for PC, mesh, and imagery at the same time (cf. subsection 2.3). For this reason, we qualitatively verify the effectiveness of the explicit entity linking visualizing the label transfer. We visualize the achieved
explicit entity linking by transferring labels from the manually annotated PCs to
the mesh, and therefrom, to image space. However, we want to emphasize that
the transferred information is not limited to labels. Features can be transferred
to other modalities, too. Figure 1 exemplarily shows the annotated modalities
for a dedicated tile from H3D as achieved by the proposed methodology. Figure
10 shows a selection of automatically annotated images of various GSDs
as derived via the implicit PCImgA. We opt for the implicit linking to create
densely labeled images since faces are projected to image space instead of sin-
gle points. Furthermore, for V3D, implicit PCImgA makes use of the perfect
co-registration of the MVS mesh and imagery. At the same time, the enclosed
PCMA is able to dampen the co-registration discrepancy in 3D space by lever-
aging the adaptive thresholding. Hence, the implicit linking avoids inconsistent
point-pixel-pairs. Both figures visually verify that the transfer operates reason-
able and smoothly on both data sets featuring different scales and resolutions.
Besides, Figure 10 reveals the dependence of synchronous acquisition and mesh
quality. The picture on the center-left shows different positions of a car due
to asynchronous capturing of nadir imagery and ALS data (cf. section 3). The
picture on the center-right depicts a ship (class vehicle) in the ship lock sur-
rrounded by water. Since the mesh does not properly reconstruct the ship, the
transferred labels do not cover the entire ship in image space.

Despite the absence of GT for all modalities, we try to quantify the entity
linking by a proxy analysis: forward and backward passing of labels. The rel-
ationship of image space and 3D space is well-known and strictly defined by
collinearity equations and hence, does not have to be validated. For this rea-
son, we highlight the label transfer from the PC to the mesh (forward pass)
and therefrom back to the PC (backward pass). During forward passing, we
aggregate labels on the face-level via majority vote. The backward pass is a
straightforward copy operation utilizing the stored association information. The
comparison of back-transferred and manual annotations allows us to validate the
effectiveness of PCMA (label consistency check).

For the used data sets, we found in an empirical process the association to
Figure 10: Automatically annotated images with labels as transferred from the PC to the mesh and, therefrom, to image space (implicit PCImgA) for H3D (top: oblique, center: nadir) and V3D (bottom). Non-associated pixels are depicted with RGB values. Original RGB images are shown on the left except for the nadir image of V3D which covers the entire labeled area.
perform best with thresholds $\theta_1 = \pm 30\,\text{cm}$, $\theta_2 = \pm 60\,\text{cm}$, $\theta_3 = \pm 120\,\text{cm}$ (V3D) and $\theta_1 = \pm 5\,\text{cm}$, $\theta_2 = \pm 10\,\text{cm}$, $\theta_3 = \pm 15\,\text{cm}$ (H3D). The chosen thresholds are guided by the shift between mesh and ALS data (cf. section 3). In particular, thresholds are fine-tuned to maximize the number of associations while keeping mismatches of back-transferred and manual labels at a minimum.

Figure 11 shows the fringe of the MVS mesh (bottom) and the respective ALS PC (top) for V3D. The height-coding indicates reconstruction errors in the leftmost part of the MVS mesh: the building and tree are not reconstructed. The adaptive thresholding guarantees a correct linking of faces and points where geometry is reconstructed correctly. Likewise, false reconstructed faces remain unlabeled and hence avoid label inconsistencies (after the backward pass).

For V3D, the adaptive thresholding associates 40.9% of faces covering 53.8% of the surface area with 75.6% of LiDAR points. The proxy analysis reveals that 98.9% of associated points show consistency in manual and back-transferred labels. 2.0% of associated faces are linked to points of different classes causing label inconsistencies for 1.1% of associated points. The achieved weighted average precision of the label consistency check is 98.9%.

For H3D, the adaptive thresholding associates 67.3% of faces covering 71.2% of
of the surface area with 55.9% of LiDAR points. 99.6% of points pass the label consistency check. Vice versa, 0.9% of associated faces are linked to points of different classes. The achieved weighted average precision is 99.9%.

Structural differences among 3D modalities prevent full association of points and faces for both data sets (cf. Figure 6 and facades in Figure 11). Particularly semi-transparent objects reduce the association rates on the point-level since sub-surface LiDAR points are not linked to the mesh surface. The majority of non-associated points belong to vegetational classes (V3D: 68%, H3D: 74%). In this regard, the high LiDAR density of H3D causes a low association rate on the point-level. In contrast, the comparatively high association rate on the face-level indicates proper co-registration and high-quality mesh. The majority of non-associated faces build the water surface where no LiDAR points are captured. For V3D, 15.5% of non-linked points belong to facade and roof pointing out the minor MVS mesh quality. To summarize, the association rates indicate the impact of the mesh quality and the point density.

The proxy analysis reveals that the majority of established point-face connections is correct for both data sets (98.9%/99.6%). Likewise, the forward-backward-pass shows that common meshing does not incorporate semantic borders. Hence, faces may be linked to points of different classes. In this case, the back-transferred majority vote causes inconsistencies. Figure 12 shows inconsistently labeled points for the entire H3D data set and as close-up. The overview at the top exhibits 0.4% of points failing the label consistency check. These points represent semantic borders and indicate an improper mesh reconstruction. We are aware of the fact that co-registration discrepancies in 3D space increase this effect. However, due to the high mesh quality and small co-registration discrepancy, H3D is less affected than V3D.

Furthermore, the proxy analysis helps to detect label noise. Figure 13 depicts a building from V3D as PC (left) and wireframe mesh (right). A few of the transferred labels to the mesh (lower right) seem not to match the given GT on the PC (upper left). For instance, some faces on the facade are marked as a roof. Consequently, the back-transferred labels to the PC (lower left) do not match
Figure 12: Overview of H3D points that show inconsistencies in manual annotations and back-transferred labels (top). The close-up at the bottom depicts inconsistently labeled points marked by the manually annotated GT on the textured mesh (left) and the back-transferred labels on the wireframe (right).

the initial annotation. Here, the appearance of false labels hints at label noise, since the inconsistencies cannot be explained by georeference issues. Figure 14 shows the GT along with its class-wise GT for classes facade and roof. The figure discloses that few points erroneously carry labels of both classes.

6. Conclusion and Future Work

To jointly leverage imagery and ALS data for semantic scene analysis, we propose a novel holistic methodology that explicitly integrates imagery and PC data via the mesh as the core representation. The multi-modal data fusion establishes explicit connections of points, faces, and pixels and enables the subsequent sharing of arbitrary information across modalities. The information transfer incorporates the established one-to-many relationships by aggregation. Hence, representation-specific features and (manual) annotations can be shared at a stroke across all modalities (cf. Table 2). Therefore, the proposed association mechanism can be seen as an integrator that functions as a labeling tool and a feature sharing tool. By these means, the novel method serves as a powerful
Figure 13: PC (left) and mesh representation (right) of a building in V3D. The top row shows the manually annotated GT and the textured mesh overlaid with its wireframe. The bottom row shows the automatically labeled mesh (PC \rightarrow Mesh) and the respectively labeled PC with back-transferred labels from the mesh (Mesh \rightarrow PC).

Figure 14: Label noise in form of duplicates in V3D for building of Figure 13. The right side shows GT separated by class roof (top) and facade (bottom) The label noise causes wrong label transfer and thus inconsistencies in the forward-backward-pass.
integrative backbone boosting multi-modal learning. In particular, the method underlines its utility for pixel-wise GT generation. Any labeled 3D data (PC or mesh) can be projected into image space to annotate multiple images at once while performing the visibility check. Hence, it minimizes the manual labeling effort. Consequently, the versatile applicability of the information transfer fosters modality-specific and multi-modal semantic segmentation.

The linking mechanism is designed to surpass imperfections of real-world data by adaptive thresholding. The tile-wise parallel processing aims for a trade-off of memory and speed. We qualitatively and quantitatively demonstrate its effectiveness and adaptiveness to underlying data by deploying two airborne data sets of different resolutions and scales. V3D is typical for large-scale country-wide mapping with moderate GSD of some centimeters and a considerable time shift between ALS and image data collection. H3D provides extremely high-resolution data with mainly synchronous data capture from a hybrid sensor system and is representative of data collection at small-scale complex built-up areas. Due to the absence of multi-modal GT, a strict quantitative analysis of the proposed method is difficult. As an alternative, we analyze the label consistency on the PC by forward-backward-passes of labels across entities of different modalities. The quantitative analysis shows that nearly 100% of the established connections are consistent. However, points of different classes that are linked to a common face might be useful for a subsequent semantically driven remeshing. Due to structural discrepancies, full association across entities is not possible. We discuss preconditions and limitations in detail highlighting the benefits of high-quality co-registration and high-quality reconstruction. The strength of our method is its simplicity and flexibility that immediately profits from advances in data acquisition, co-registration, and meshing. In the future, we aim for pixel-accurate co-registration of ALS data and imagery leveraging the hybrid strip adjustment (Glira et al., 2019). We claim that improved co-registration improves both geometric reconstruction and semantic analysis. To prove our assumption, we plan an ablation study for multi-modal features on different representations by analyzing the performance of a ML classifier.
7. Acknowledgements

V3D is provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010). H3D data originates from a research project in collaboration with the German Federal Institute of Hydrology (BfG) in Koblenz. We thank all our colleagues for insightful discussions. In particular, we thank our students Fangwen Shu, Mohamad Hakam Shams Ed-din, and Vishal Pani, who assisted the implementation. Furthermore, we thank the whole nFrames team for their support regarding the mesh generation. Special thanks are directed to Carmen Kaspar for proofreading.

References

Ahmed, E., Saint, A., Shabayek, A., Cherenkova, K., Das, R., Gusev, G., Aouada, D., Ottersten, B.E., 2018. Deep learning advances on different 3d data representations: A survey. CoRR abs/1808.01462. arXiv:1808.01462.

Ali Khan, S., Shi, Y., Shahzad, M., Xiang Zhu, X., 2020. FGCN: Deep Feature-based Graph Convolutional Network for Semantic Segmentation of Urban 3D Point Clouds, in: CVPRW, pp. 778–787.

Armeni, I., Sax, S., Zamir, A.R., Savarese, S., 2017. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. CoRR abs/1702.01105. arXiv:1702.01105.

Boulch, A., 2019. Generalizing Discrete Convolutions for Unstructured Point Clouds, in: Eurographics Workshop on 3D Object Retrieval, The Eurographics Association. doi:10.2312/3dor.20191064.

Boulch, A., Le Saux, B., Audebert, N., 2017. Unstructured Point Cloud Semantic Labeling Using Deep Segmentation Networks, in: Eurographics Workshop on 3D Object Retrieval, The Eurographics Association. doi:10.2312/3dor.20171047.

Boussaha, M., Vallet, B., Rives, P., 2018. Large Scale Textured Mesh Reconstruction From Mobile Mapping Images and LiDAR Scans. ISPRS Annals
Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P., 2016. Geometric deep learning: Going beyond euclidean data. CoRR abs/1611.08097.

Chang, J., Gu, J., Wang, L., Meng, G., Xiang, S., Pan, C., 2018. Structure-Aware Convolutional Neural Networks, in: Advances in Neural Information Processing Systems 31. Curran Associates, Inc., pp. 11–20.

Cramer, M., 2010. The dgpf-test on digital airborne camera evaluation overview and test design. PFG Photogrammetrie, Fernerkundung, Geoinformation 2010, 73–82. doi:10.1127/1432-8364/2010/0041

Cramer, M., Haala, N., Laupheimer, D., Mandlburger, G., Havel, P., 2018. Ultra-high precision uav-based LiDAR and dense image matching. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-1, 115–120. doi:10.5194/isprs-archives-XLII-1-115-2018

Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M., 2017. ScanNet: Richly-annotated 3d reconstructions of indoor scenes, in: CVPR.

Eitel, J., Höfle, B., Vierling, L., Abellán, A., Asner, G., Deems, J., Glennie, C., Joerg, P., LeWinter, A., Magney, T., Mandlburger, G., Morton, D., Müller, J., Vierling, K., 2016. Beyond 3-d: The new spectrum of lidar applications for earth and ecological sciences. Remote Sensing of Environment 186, 372–392. doi:10.1016/j.rse.2016.08.018

García-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martínez, V., García Rodríguez, J., 2017. A review on deep learning techniques applied to semantic segmentation. CoRR abs/1704.06857.

Gerke, M., Rottensteiner, F., Wegner, J., Sohn, G., 2014. ISPRS semantic labeling contest. doi:10.13140/2.1.3570.9445

39
Glira, P., Pfeifer, N., Mandlburger, G., 2019. Hybrid orientation of airborne lidar point clouds and aerial images. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W5, 567–574.

Graham, B., Engelcke, M., van der Maaten, L., 2018. 3d semantic segmentation with submanifold sparse convolutional networks, pp. 9224–9232. doi:10.1109/CVPR.2018.00961.

Griffiths, D., Boehm, J., 2019a. A Review on Deep Learning Techniques for 3D Sensed Data Classification. Remote Sensing 11. doi:10.3390/rs11121499.

Griffiths, D., Boehm, J., 2019b. Synthcity: A large scale synthetic point cloud. CoRR abs/1907.04758. arXiv:1907.04758.

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. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences V-2-2020, 727–734. doi:10.5194/isprs-annals-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, in: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 91–98.

Hackel, T., Wegner, J.D., Schindler, K., 2016. Fast Semantic Segmentation of 3D Point Clouds With Strongly Varying Density. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences III-3, 177 – 184. doi:10.3929/ethz-b-000126659.

Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D., 2019. MeshCNN: A Network with an Edge. ACM Transactions on Graphics (TOG) 38, 90.
He, H., Upcroft, B., 2013. Nonparametric semantic segmentation for 3d street scenes, in: Amato, N. (Ed.), IEEE/RSJ International Conference on Intelligent Robots and Systems: New Horizon, Tokyo, Japan.

Hua, B.S., Pham, Q.H., Nguyen, D., Tran, M.K., Yu, L.F., Yeung, S.K., 2016. SceneNN: A Scene Meshes Dataset with aNNotations, in: Fourth International Conference on 3D Vision, pp. 92–101. doi:10.1109/3DV.2016.18

Huang, J., You, S., 2016. Point Cloud Labeling Using 3D Convolutional Neural Network, in: 23rd International Conference on Pattern Recognition (ICPR), pp. 2670–2675. doi:10.1109/ICPR.2016.7900038

Kalogerakis, E., Hertzmann, A., Singh, K., 2010. Learning 3D Mesh Segmentation and Labeling. ACM Transactions on Graphics 29.

Kölle, M., Laupheimer, D., Haala, N., 2019. Klassifikation hochaufgelöster LiDAR- und MVS-Punktwolken zu Monitoringzwecken, in: 39. Wissenschaftlich-Technische Jahrestagung der OVG, DGPF und SGPF in Wien, pp. 692–701.

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. ISPRS Annals V-2-2020, 501–508. doi:10.5194/isprs-annals-V-2-2020-501-2020

Lambert, J., Liu, Z., Sener, O., Hays, J., Koltun, V., 2020. MSeg: A composite dataset for multi-domain semantic segmentation, in: CVPR.

Landrieu, L., Raguët, H.R., Vallet, B., Mallet, C., Weinmann, M., 2017. A structured regularization framework for spatially smoothing semantic labelings of 3D point clouds. ISPRS Journal of Photogrammetry and Remote Sensing 132, 102–118. doi:10.1016/j.isprsjprs.2017.08.010

Landrieu, L., Simonovsky, M., 2017. Large-scale point cloud semantic segmentation with superpoint graphs. CoRR abs/1711.09869. arXiv:1711.09869
Laupheimer, D., Shams Eddin, M.H., Haala, N., 2020a. On the association of lidar point clouds and textured meshes for multi-modal semantic segmentation. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences V-2-2020, 509–516. doi:10.5194/isprs-annals-V-2-2020-509-2020.

Laupheimer, D., Shams Eddin, M.H., Haala, N., 2020b. The Importance of Radiometric Feature Quality for Semantic Mesh Segmentation, in: 40. Wissenschaftlich-Technische Jahrestagung der DGPF in Stuttgart, pp. 205–218.

Lawin, F.J., Danelljan, M., Tosteberg, P., Bhat, G., Khan, F.S., Felsberg, M., 2017. Deep projective 3d semantic segmentation. CoRR abs/1705.03428.

Mandlburger, G., Wenzel, K., Spitzer, A., Haala, N., Glira, P., Pfeifer, N., 2017. Improved topographic models via concurrent airborne lidar and dense image matching. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W4, 259–266.

Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., Terzopoulos, D., 2020. Image segmentation using deep learning: A survey. arXiv:2001.05566.

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

Peters, T., Brenner, C., 2019. Automatic generation of large point cloud training datasets using label transfer. Tagungsband der 39. Wissenschaftlich-Technischen Jahrestagung der DGPF.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation, in: CVPR, pp. 77–85.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. Pointnet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, in: Advances in Neural Information Processing Systems, pp. 5105–5114.
Qiao, Y.L., Gao, L., Yang, J., Rosin, P.L., Lai, Y.K., Chen, X., 2019. LaplacianNet: Learning on 3D Meshes with Laplacian Encoding and Pooling. abs/1910.14063. arXiv:1910.14063.

Ramirez, P.Z., Paternesi, C., De Gregorio, D., Di Stefano, L., 2019. Shooting Labels: 3D Semantic Labeling by Virtual Reality. arXiv preprint abs/1910.05021. arXiv:1910.05021.

Rothermel, M., Wenzel, K., Fritsch, D., Haala, N., 2012. SURE: Photogrammetric Surface Reconstruction From Imagery, in: Proceedings LC3D Workshop, Berlin.

Rouhani, M., Lafarge, F., Alliez, P., 2017. Semantic Segmentation of 3D Textured Meshes for Urban Scene Analysis. ISPRS Journal of Photogrammetry and Remote Sensing 123, 124–139. doi:10.1016/j.isprsjprs.2016.12.001.

Schmohl, S., Soergel, U., 2019. Submanifold sparse convolutional networks for semantic segmentation of large-scale ALS point clouds, in: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 77–84. doi:10.5194/isprs-annals-IV-2-W5-77-2019.

Schult, J., Engelmann, F., Kontogianni, T., Leibe, B., 2020. DualConvMeshNet: Joint Geodesic and Euclidean Convolutions on 3D Meshes, in: CVPR.

Shilane, P., Min, P., Kazhdan, M., Funkhouser, T., 2004. The Princeton Shape Benchmark, in: Shape modeling applications, 2004. Proceedings, IEEE. pp. 167–178.

Su, H., Maji, S., Kalogerakis, E., Learned-Miller, E., 2015. Multi-View Convolutional Neural Networks for 3D Shape Recognition, in: ICCV, pp. 945–953. doi:10.1109/ICCV.2015.114.

Tutzauer, P., Laupheimer, D., Haala, N., 2019. Semantic urban mesh enhancement utilizing a hybrid model. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W7, 175–182.
Vosselman, G., Coenen, M., Rottensteiner, F., 2017. Contextual segment-based classification of airborne laser scanner data. ISPRS journal of photogrammetry and remote sensing 128, 354–371. doi:10.1016/j.isprsjprs.2017.03.010

Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS Journal of Photogrammetry and Remote Sensing 105, 286–304. doi:https://doi.org/10.1016/j.isprsjprs.2015.01.016

Wichmann, A., Agoub, A., Kada, M., 2018. Roofn3d: Deep learning training data for 3d building reconstruction. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-2, 1191–1198. doi:10.5194/isprs-archives-XLII-2-1191-2018

Winiwarter, L., Mandlburger, G., Schmohl, S., Pfeifer, N., 2019. Classification of ALS point clouds using end-to-end deep learning. PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science 87, 75–90. doi:10.1007/s41064-019-00073-0

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J., 2015. 3D ShapeNets: A Deep Representation for Volumetric Shapes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1912–1920.

Xie, Y., Tian, J., Zhu, X.X., 2020. Linking points with labels in 3d: A review of point cloud semantic segmentation. IEEE Geoscience and Remote Sensing Magazine 8, 38–59. doi:10.1109/MGRS.2019.2937630

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, BMVC. 30th British Machine Vision Conference.

Copyright: © 2021. This manuscript version is made available under the CC-BY-NC-ND 4.0 license http://creativecommons.org/licenses/by-nc-nd/4.0/