ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-2, 2018
ISPRS TC II Mid-term Symposium "Towards Photogrammetry 2020", 4–7 June 2018, Riva del Garda, Italy

SUPERVISED OUTLIER DETECTION IN LARGE-SCALE MVS POINT CLOUDS FOR 3D CITY MODELING APPLICATIONS

Corinne Stucker, Audrey Richard, Jan D. Wegner, Konrad Schindler

Photogrammetry and Remote Sensing, ETH Zürich, Switzerland
{firstname.lastname}@geod.baug.ethz.ch

Commission II, WG II/3

KEY WORDS: Supervised Filtering, Outlier Detection, Semantics, Scene Understanding, Point Clouds, City Modeling

ABSTRACT:

We propose to use a discriminative classifier for outlier detection in large-scale point clouds of cities generated via multi-view stereo (MVS) from densely acquired images. What makes outlier removal hard are varying distributions of inliers and outliers across a scene. Heuristic outlier removal using a specific feature that encodes point distribution often delivers unsatisfying results. Although most outliers can be identified correctly (high recall), many inliers are erroneously removed (low precision), too. This aggravates object 3D reconstruction due to missing data. We thus propose to discriminatively learn class-specific distributions directly from the data to achieve high precision. We apply a standard Random Forest classifier that infers a binary label (inlier or outlier) for each 3D point in the raw, unfiltered point cloud and test two approaches for training. In the first, non-semantic approach, features are extracted without considering the semantic interpretation of the 3D points. The trained model approximates the average distribution of inliers and outliers across all semantic classes. Second, semantic interpretation is incorporated into the learning process, *i.e.* we train separate inlier-outlier classifiers per semantic class (building facades, roof, ground, vegetation, fields, and water). Performance of learned filtering is evaluated on several large SfM point clouds of cities. We find that results confirm our underlying assumption that discriminatively learning inlier-outlier distributions does improve precision over global heuristics by up to ≈ 12 percent points. Moreover, semantically informed filtering that models class-specific distributions further improves precision by up to ≈ 10 percent points, being able to remove very isolated building, roof, and water points while preserving inliers on building facades and vegetation.

1. INTRODUCTION

Outlier detection refers to the process of identifying patterns in data that do not comply with the general or expected behavior of the data. Outliers can be very different in nature, and the exact definition depends on the target application and the underlying assumptions regarding the data structure and the data generating process. Since outlier definition depends on both, given data and task, we propose to learn discriminative classifiers for outlier removal in point clouds and, further, to model class-specific distributions. Our goal is to remove most outliers while retaining the large majority of inliers (high precision). For 3D object reconstruction, missing data (incorrectly removed inliers) is usually more harmful than some few remaining outliers close to the true surface. Parts without sufficient data cannot be reconstructed at all, whereas outliers close to the true object surface are handled with smoothing priors that are built into the 3D reconstruction approach (Häne et al., 2013, Bláha et al., 2016). In this paper, we thus aim for high precision and assume that few, remaining outliers are handled by regularizers of the 3D reconstruction pipeline.

The automatic detection and elimination of noise and outliers in point cloud data sets is a long-standing, active field of research (Cheng and Lau, 2017). Most of these point cloud filtering techniques are dedicated to applications in industrial metrology and hence, are tailored to point clouds with a relatively small proportion of outliers and homogeneous point densities. In contrast, point clouds generated by image-based, multi-view stereo (MVS) 3D reconstruction techniques feature large portions of outliers and very inhomogeneous point densities across a scene. A common strategy is trying to avoid outliers already at the depth map estimation stage through enforcing consistency across views (Goesele et al., 2007, Furukawa and Ponce, 2010, Wolff et al., 2016). Still, gross outliers in MVS point clouds pose significant challenges to surface reconstruction algorithms. Conventional meshing techniques fail in their presence and require substantial manual post-processing. Volumetric 3D reconstruction approaches risk losing many details if regularizers or visibility constraints are enforced strongly.

Here, we propose to view MVS point cloud filtering as a preprocessing step that is applied after depth map fusion and before 3D reconstruction. The main idea is to filter outliers in large MVS point clouds by learning class-specific inlier-outlier distributions with supervised classifiers. Our target application are semantically annotated 3D city models generated by MVS using aerial cameras. We build on recent works (Häne et al., 2013, Bláha et al., 2016, Bláha et al., 2017) that exploit the multi-view imaging setup to simultaneously reconstruct and segment 3D models into semantically meaningful 3D entities such as building facades, roofs, streets, and vegetation, where 3D shape and semantic class-labels are mutually supportive.

Although supervised approaches have proven to be effective for many classification tasks, supervised outlier detection approaches are difficult to realize in practice. First, it is demanding and often prohibitively expensive to obtain an accurate and representative training data set which comprises both normal and outlier data instances. Second, the outlier distribution of the data is sometimes unknown in advance and, third, it can be dynamic in nature. For MVS point clouds derived from aerial images, the inlier-outlier distributions are assumed to be static. Labeling inliers and outliers for training can be done efficiently by overlaying the raw point cloud with an already existing semantic 3D city model and imposing a fixed threshold on the point-to-mesh distances.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-2-263-2018 | © Authors 2018. CC BY 4.0 License.

We test two approaches: (i) inlier-outlier distributions are modelled globally regardless of semantic classes and (ii) class-specific distributions are learned. Both approaches are validated on large aerial MVS point clouds and compared to a conventional, unsupervised, heuristic baseline. We find that supervised machine learning achieves much higher precision than a heuristic baseline method. Moreover, class-specific filtering further improves results by retaining more inliers in low-density areas like vertical building facades, which will allow more accurate 3D reconstruction. Additionally, we check the generalization capability of our learned models. We show that once sufficiently trained on a larger scene, models can be applied to unseen aerial MVS scenarios and still achieve reasonably good results.

2. RELATED WORK

A large variety of point cloud filtering approaches exists. Point cloud denoising approaches are typically used in the context of 3D surface reconstruction and do not detect outliers directly. Instead, these methods aim at reducing the noise inherent in point clouds by adapting the position of raw points. In contrast, unsupervised and supervised point cloud filtering approaches are dedicated to detecting and removing outliers among the data without changing the position of raw points. In the following, we try to roughly classify related point cloud filtering approaches into these three categories.

Point cloud denoising has been approached in various ways. The seminal moving least squares (MLS) method of (Levin, 2004) reduces noise in point clouds implicitly by projecting the points onto a locally fitted low-degree bivariate polynomial. Several variants of the traditional MLS approach have been developed, mainly to reduce the filtering effect near sharp features and to handle sparse sampling and outliers. The modifications are based on an iterative refitting scheme to model locally piecewise smooth surfaces (Fleishman et al., 2005), adjust the polynomial fitting procedure (Guennebaud and Gross, 2007), introduce a parameterization-free projection operator (Lipman et al., 2007) or express the MLS procedure as a kernel regression process including robust statistics (Öztireli et al., 2009, Öztireli, 2015). Further point cloud denoising approaches are inspired by filtering techniques used in image processing (Deschaud and Goulette, 2010, Digne, 2012) or follow concepts developed in the field of differential geometry (Ma and Cripps, 2011) and spectral analysis (Öztireli et al., 2010).

Unsupervised outlier detection constitutes the majority of approaches and can be further subdivided into (i) statistical-based, (2) clustering-based, and (3) distance-based approaches. Statistical-based outlier detection assumes that the data is generated by a stochastic process. Any data instance that is unlikely to be generated from the estimated stochastic process according to some test statistic is then reported as an outlier (Barnett and Lewis, 1974, Eskin, 2000). The statistical outlier removal tool implemented in the Point Cloud Library (PCL)[1] assumes that the average distance of a point to its nearest neighbors follows a Gaussian distribution and performs statistical hypothesis testing to identify and discard points whose average distance to their neighbors is outside a certain confidence interval (Rusu and Cousins, 2011). Nonparametric outlier detection methods infer the underlying probability distribution of inliers and outliers directly from the data using clustering (He et al., 2003, Yu et al., 2002, Schall et al., 2005, Latecki et al., 2007). Further, early works often applied distance-based methods (Knorr and Ng, 1998) to find global outliers using the k-nearest neighborhood of a data instance to compute its outlier score. Typically, the outlier score of a data instance is constituted by the distance to its k-nearest neighbor (Ramaswamy et al., 2000) or by the average distance to all other data instances within the k-nearest neighborhood (Angiulli and Pizzuti, 2002). A strategy to identify local outliers is based on the assumption that local outliers are located in areas of relatively low density compared to their k-nearest neighbors. In (Breunig et al., 2000), the outlier score of a data instance is computed as the ratio of the average local density of the k-nearest neighbors to the local density of the data instance itself. Several extensions of this idea have been proposed, mainly to improve the density estimation procedure for linearly distributed data sets (Jin et al., 2006) and to better handle regions of different densities that are not clearly separated (Tang et al., 2002).

Supervised outlier detection refers to approaches where distributions are learned with labeled ground truth. One strategy is to only learn the inlier distribution of points and to view any data instance that deviates significantly from the trained model as an outlier. For example, one-class support vector machines (Rätsch et al., 2002, Amer et al., 2013) or one-class kernel Fisher discriminant analysis (Roth, 2004) have been applied with this strategy. In our paper, we propose to learn both inlier and outlier distributions with labeled ground truth from the data. Moreover, we learn class-specific models to better cope with the varying inlier-outlier distributions as a function of the object class.

3. OUTLIER DETECTION

Reasons for outliers are manifold. Typical sources are human or instrumental errors, and natural variations or unexpected changes in the behavior of a system. In practice, data sets are usually impacted by multiple types of outliers, and it is subject to the application whether a particular type of outlier is of interest or not. For 3D city modeling from aerial images, outlier removal is an essential pre-processing step to generate a cleaner data set for 3D reconstruction. The nature of outliers is one of the key aspects that needs to be considered when designing an outlier detection algorithm. According to (Chandola et al., 2009), outliers can be classified into the following three main categories:

- point outlier: a single data instance that deviates significantly from the remaining data set
- collective outlier: a group or sequence of data instances that deviates significantly from the remaining data set, even though the individual data instances may not be anomalous
- contextual outliers: a single data instance that is only anomalous in a specific context (*e.g.*, spatial or temporal context)

The primary type of outliers in point cloud data sets derived from image-based 3D reconstruction techniques are point outliers. They are induced by image imperfections (*e.g.*, lens distortion or sensor noise), matching ambiguities, errors in the camera calibration as well as in the camera pose and depth map estimation procedure. In contrast, collective and contextual outliers are not present in point cloud data sets.

Point outliers can be further subdivided into global and local outliers. A global outlier is a single data instance that deviates significantly from the entire data set. A single data instance is considered as a local outlier if it differs substantially from other data instances within its vicinity. This notion of global and local outliers is shown in Figure 1. P_1 and P_2 can be easily detected as global outliers, as these data instances exhibit a considerable

[1] `http://www.pointclouds.org`

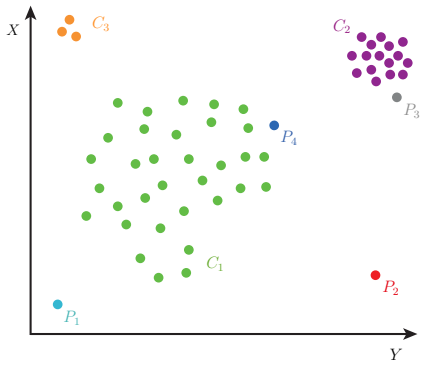


Figure 1: Illustration of the different types of point outliers by means of a two-dimensional synthetic data set. The data set encompasses three clusters C_1, C_2, and C_3 of normal data instances, two global point outliers P_1 and P_2, and one local point outlier P_3. Unlike P_3, the data instance P_4 is normal and belongs to cluster C_1.

distance to the remaining points. From a global perspective, P_3 would be classified as a normal data instance due to its proximity to cluster C_2. However, when examined locally, P_3 appears to be a local outlier because its distance to cluster C_2 is relatively large compared to the spacing between the data instances of cluster C_2. In comparison, data instance P_4 should be considered as normal, although its distance to the nearest cluster C_1 is roughly the same as the distance between P_3 and C_2. Lastly, the points forming cluster C_3 can be classified as either global outliers or as a small regular cluster. It depends on the application whether such micro clusters need to be detected as anomalous or not.

Most point outlier detection methods fail to capture both global and local outliers. Methods tailored to detect local outliers may be able to identify global outliers as well, provided that global outliers are sparsely distributed and do not form a micro cluster. However, methods tailored to detect global outliers can hardly be applied to detect local outliers (as illustrated in Fig. 1). In general, it is more challenging to detect local outliers than global outliers. First, the definition of locality is a non-trivial task and is often ill-defined, especially if the data exhibits clusters of varying densities. Second, statistical properties of a data instance are strongly affected if its spatial support includes nearby outliers or normal data instances of different distributions.

4. METHOD

Aerial MVS point clouds generated from nadir and oblique aerial images inevitably comprise a considerable amount of outliers. The purpose of point cloud filtering is to reduce outliers while preserving inliers. We develop a supervised binary classification scheme to assign each 3D point of a raw, unfiltered point cloud to one of the following two categories:

- 3D points assigned to the *inlier* point category are assumed to be located close to the underlying surface of the captured scene.
- 3D points assigned to the *outlier* point category are considered as either global or local outliers. Global outliers are caused by systematic deviations or gross errors in the point cloud generation process (*e.g.*, matching errors or inadequate camera calibration), whereas local outliers are induced by random deviations and uncertainties in the camera pose and depth map estimation procedure (*e.g.*, depth quantization).

Ultimately, the filtered point cloud is derived by discarding all 3D points that are predicted as outliers.

The decision whether a 3D point is deemed as an inlier or an outlier is primarily dependent on the local point distribution given by the 3D points within its vicinity. The neighborhood of inliers can be characterized by well-defined point distributions, even though the sampling density may vary locally due to the texture of the scene and the spatial configuration of the recorded images. In the context of urban scenes, these local point distributions display mainly planar (*e.g.*, ground, building facades, and roofs) or spherical (*e.g.*, vegetation) patterns. In contrast to these characteristic structures, the point neighborhood of global outliers is typically sparse and does not exhibit a distinct geometric layout.

The characteristic point distribution of inliers and outliers is not only an intrinsic property of urban point clouds in general but rather varies across different semantic classes of urban scenes. In particular, point cloud regions representing building roofs or ground commonly exhibit a low level of noise, as these scene structures are well captured by nadir and oblique aerial images. However, these point cloud regions may be incomplete and show a varying point density due to the low or missing texture of the underlying scene. Point cloud regions representing vegetated areas are usually densely sampled but are impaired by a considerable level of noise due to the repetitive texture of the underlying scene. Vertical scene structures like building facades exhibit more outliers and have often fewer inliers because they typically show repetitive textures and surface areas (*e.g.*, windows) corrupted by specular reflections – two properties which lead to mismatches during image matching within the structure-from-motion pipeline. Further, the orientation of building facades with respect to the viewing direction of the (nadir) camera poses additional challenges to the image matching and depth map estimation procedure (*e.g.*, invalid assumption of fronto-parallel surfaces).

We follow two approaches for supervised outlier detection in urban point clouds. In the first approach, a discriminative model is trained to distinguish between the local point distribution of inliers and outliers without considering the semantic interpretation of the 3D points. Thus, the trained model approximates the average behavior of inliers and outliers across different semantic classes. The second, class-specific approach postulates that the local point distribution of inliers and outliers is specific to each of the semantic classes (building facades, roof, ground, vegetation, fields, and water) for the reasons described previously. A discriminative model is trained for each of the semantic classes to better adapt to the individual inlier and outlier distributions per class.

4.1 Feature Extraction

We compute 24 standard features from literature per 3D point $\boldsymbol{P_i}$ that are either adapted from unsupervised outlier detection methods or deduced from LiDAR point cloud labeling methods. The reader is referred to the original works for an in-depth coverage of the applied features that can be grouped into the following five categories:

- density-based features (Ramaswamy et al., 2000, Breunig et al., 2000, Angiulli and Pizzuti, 2002, Kriegel et al., 2008, Zhang et al., 2009)
- 3D eigenvalue-based features (Weinmann et al., 2013)
- local plane-based features (Chehata et al., 2009)
- height-based features (Weinmann et al., 2015b)
- 2D features (Weinmann et al., 2013)

The local neighborhood $\mathcal{N}_i$ of a 3D point $\boldsymbol{P_i}$ is defined as the smallest sphere centered at $\boldsymbol{P_i}$ that encompasses the $k \in \mathbb{N}$ closest 3D points to $\boldsymbol{P_i}$ with respect to the Euclidean distance in 3D space. 3D points that are located at the same distance to $\boldsymbol{P_i}$ as its k-nearest neighbor are included in $\mathcal{N}_i$ as well. Consequently, the number of neighbors included in a local point neighborhood may vary among the 3D points but has a lower limit of at least k neighbors. Note that the 3D point $\boldsymbol{P_i}$ is excluded from its local point neighborhood $\mathcal{N}_i$.

Following recent trends in 3D scene understanding and classification (Brodu and Lague, 2012), the features are extracted at multiple scales by varying the size k of the local point neighborhood. The rationale behind this approach is threefold: First, it avoids using heuristic or empiric knowledge on the scene to select the scale parameter k. Second, the optimal scale parameter k depends heavily on the local point density and the local 3D structure of the scene and may thus not be identical for each local 3D point neighborhood. In particular, it is presumed that the optimal neighborhood size of both inliers and local outliers is smaller than of global outliers. Last, the feature extraction at multiple scales presents additional information of how the local 3D structure behaves across scales, which in turn may support the discrimination between inliers and outliers. Specifically, it is assumed that the local 3D structure of inliers and possibly of local outliers is stable over a range of scales, whereas the local 3D structure of global outliers alters with varying scale.

4.2 Supervised Filtering

We train a Random Forest classifier (Breiman, 2001) using the features listed in Section 4.1 to learn the average behavior of inliers and outliers across all semantic classes. Random Forests have been shown to yield good results for many point cloud classification tasks (Chehata et al., 2009, Weinmann et al., 2015a), run efficiently on large data sets and can cope with redundant features. The optimal hyperparameters are determined via grid search and cross-validation. The classifier outputs a binary label per point indicating whether the respective 3D point is predicted as an inlier or an outlier. Eventually, the filtered point cloud is derived by assembling all 3D points that are predicted as inliers.

4.3 Semantically Informed Filtering

In order to allow for different inlier-outlier distributions per object category, we make the supervised classification approach presented in Section 4.2 class-specific. We assume that each 3D point already comes with a class likelihood, which originates from previous image labeling and projection to 3D as described in (Bláha et al., 2016). This additional semantic information per point is used to train multiple classifiers, where each classifier learns the inlier-outlier distribution of a specific semantic class.

4.4 Implementation Details

Our point cloud filtering method is implemented in MATLAB. Initial tests showed that the Random Forest classifier provided in the MATLAB toolbox is incapable of processing large data sets. Furthermore, the hyperparameters of the Random Forest classifier cannot be accessed or modified easily. Because of these limitations, the *ETH Random Forest Template Library*[2] is incorporated into the implemented point cloud filtering routine. It is written in C++ and hence, is suited to process large data sets. Beyond a considerable decrease in computation time, it further enables to manually set the hyperparameters of the classifier.

[2] `http://www.prs.igp.ethz.ch/research/Source_code_and_datasets.html`

5. EXPERIMENTS

We evaluate our approach on three large-scale aerial MVS point clouds with different structure and semantic classes. Aerial image sets are Enschede (Netherlands)[3], Dortmund (Zeche Zollern, Germany), and Zurich (Switzerland)[4]. The three aerial image sets are acquired in the *Maltese cross* configuration (*i.e.* one nadir image and four oblique views to the north, south, east, and west per camera position) to mitigate visibility problems such as foreshortening or occlusion. We use the standard VisualSFM pipeline of (Wu, 2011) to orient the image blocks and the public implementation of plane-sweep stereo (Häne et al., 2014) with semi-global matching as smoothness prior (Hirschmüller, 2008) to estimate per-view depth maps. Further, we apply a multi-class boosting classifier (Benbouzid et al., 2012, Bláha et al., 2016) to predict pixelwise class-conditional likelihoods of the six semantic object classes building (facades), roofs, ground (impervious surfaces), vegetation (trees), fields, and water. Given the depth information and the class likelihoods at each pixel, we generate the input point clouds to our algorithm by back-projecting the pixels into 3D space and assigning a semantic label to each 3D point given by the maximal class likelihood of the corresponding image pixel.

5.1 Data Pre-Processing

Data sets Dortmund, Enschede, and Zurich differ in the number and resolution of the images. Consequently, generated MVS point clouds have different point densities. In order to ensure fair comparisons, point clouds need to have roughly the same average point density. We thus balance densities among data sets by adapting the percentage of back-projected image pixels (per view and site) such that the resulting point clouds exhibit a median distance of about 0.45m between the points. After this preprocessing, we have 5.2 million (Dortmund), 14.8 million (Enschede), and 5.8 million (Zurich) of points, respectively.

5.2 Ground Truth Labeling and Evaluation Strategy

A major shortcoming of the available data sets is their lack of ground truth, *i.e.* the actual segmentation of the point clouds into inliers and outliers is unknown. To generate ground truth labels, we take the semantic 3D models created by the approach of (Bláha et al., 2016) as a reference, despite them not reflecting reality perfectly well. For each 3D point of a raw point cloud, we compare its distance to the corresponding semantic mesh of the 3D model against a manually chosen threshold. If the point-to-mesh distance is below the threshold, we declare the point an inlier. A 3D point whose point-to-mesh distance exceeds the threshold is declared an outlier. We use a two-sided threshold of 0.6m across all three data sets, which is experimentally determined through visual inspection and corresponds to three times the resolution of the semantic 3D models.

We compute confusion matrices and derive the standard measures accuracy, precision, recall, and F_1-score for quantitative evaluation. Since we strive for outlier detection, an outlier correctly detected as such is defined a *true positive (TP)*, whereas a correctly detected inlier is a *true negative (TN)*. Accordingly, an outlier incorrectly classified as inlier is a *false negative (FN)*, while an inlier wrongly detected as outlier is a *false positive (FP)*. We consider our MVS point cloud filtering method as a pre-processing step to generate a cleaner data set as input to a 3D reconstruction

[3] data provided by Slagboom en Peeters Aerial Survey: `https://www.slagboomenpeeters.com`

[4] Dortmund and Zurich are from the ISPRS/EuroSDR Benchmark for Multi-Platform Photogrammetry: `https://www2.isprs.org/commissions/comm1/icwg15b/benchmark_main.html/`

Site	Unsupervised filtering			Supervised filtering			Semantically informed filtering		
	Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]
Dortmund	71.82	**62.30**	66.72	79.17	56.52	65.84	**82.54**	61.64	**70.34**
Enschede	69.32	69.18	69.25	81.45	70.44	75.53	**84.11**	**71.40**	**77.22**
Zurich	77.47	**62.96**	69.46	83.66	54.07	65.67	**88.02**	62.50	**72.97**

Table 1: Quantitative results (average numbers after 6-fold cross-validation) of the supervised, non-semantic filtering approach and the semantically informed filtering approach in comparison with the heuristic, unsupervised filtering approach of (Sotoodeh, 2006).

Dortmund	Supervised filtering			Semantically informed filtering		
	Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]
Building	80.39	53.17	63.82	**91.09**	**60.93**	**72.58**
Fields	80.50	57.11	66.70	**80.53**	**63.61**	**71.00**
Ground	79.70	57.20	66.57	**81.85**	**62.73**	**70.80**
Roof	83.17	55.40	66.38	**87.20**	**66.62**	**75.13**
Vegetation	77.14	56.25	64.88	**82.16**	**59.62**	**68.59**

Enschede	Supervised filtering			Semantically informed filtering		
	Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]
Building	79.81	**66.35**	72.42	**86.11**	65.38	**74.21**
Ground	82.52	70.49	76.02	**83.71**	**74.35**	**78.75**
Roof	**83.23**	71.53	76.92	80.84	**76.72**	**78.69**
Vegetation	79.53	**73.81**	76.55	**88.38**	68.65	**77.10**

Zurich	Supervised filtering			Semantically informed filtering		
	Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]
Building	79.02	57.42	66.47	**84.98**	**61.17**	**70.96**
Ground	80.00	49.26	60.95	**81.81**	**59.37**	**68.64**
Roof	81.78	48.74	61.07	**86.66**	**56.09**	**67.82**
Vegetation	85.63	58.54	69.50	**91.23**	**64.69**	**75.38**
Water	95.45	62.94	75.83	**99.41**	**79.41**	**88.24**

Table 2: Quantitative results per semantic class (average numbers after 6-fold cross-validation) of the supervised, non-semantic filtering approach and the semantically informed filtering approach.

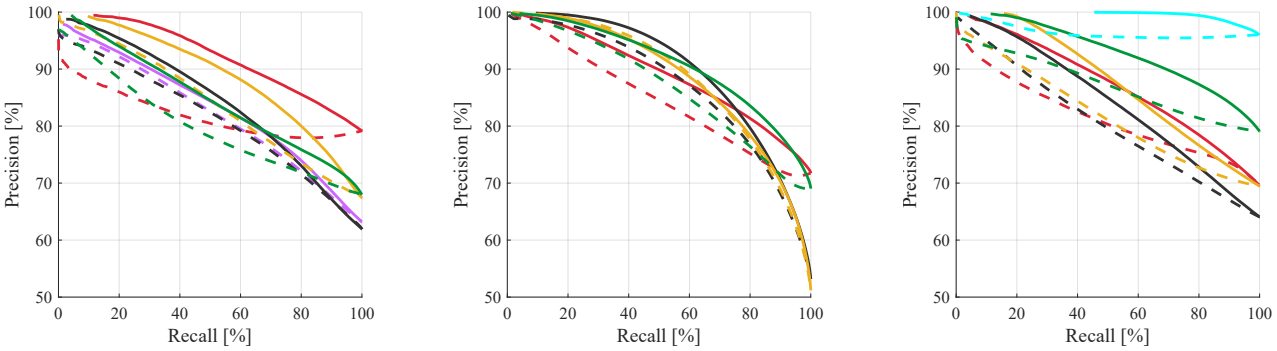
algorithm. Hence, we particularly aim at attaining high precision values as we do not want to erroneously remove inliers and lose valuable information for subsequent processing steps.

5.3 Results

The optimal hyperparameters of the Random Forest classifiers and optimal neighborhood sizes for feature extraction are determined through maximization of the F_1-score via grid search and cross-validation. We find 20 decision trees with maximum tree depth of 15 as optimal parameters for our application, where the Gini index is used as splitting criterion. For feature extraction, we test various single scales k and multiple scale combinations. We consider a single neighborhood size of $k = 100$ as a good compromise between computation time and classification accuracy.

Supervised filtering vs. unsupervised filtering Table 1 compares quantitative results of the supervised, non-semantic filtering approach to the unsupervised baseline (Sotoodeh, 2006). Note that (Sotoodeh, 2006) uses the local outlier factor proposed by (Breunig et al., 2000) as feature to encode the relative local density of a point. Segmentation into inliers and outliers is done by simple thresholding of the feature values. Compared to this baseline, the supervised approach improves precision between 7 and 11 percent points, albeit at the cost of lower recall. A visual comparison of the results in Figure 3 reveals that unsupervised filtering (second column from left) removes way too many inliers. Especially facades with low density are almost entirely removed, which would make 3D reconstruction impossible. Although this can also happen in particular cases with the supervised approach (see third column from left, third row), facade points are usually better retained. Furthermore, geometric structures of roofs and vegetation are better preserved with the supervised filtering approach. These results indicate that trading recall for precision is indeed necessary to ensure reconstructability while rejecting most outliers.

Supervised filtering vs. semantically informed filtering Table 1 and Table 2 compare quantitative results of the semantically informed filtering approach versus supervised filtering without explicit semantic knowledge (see Fig. 3 for a visual comparison). Semantic filtering performs best regarding precision, recall, and F_1-score in almost all cases. Precision for class building, which mainly comprises vertical facades, is consistently better for all cases if filtered with the semantic approach. A visual comparison in Figure 3 shows that semantically informed filtering (right column) keeps a much larger proportion of the inliers on facades compared to both baselines. These results indicate that an average classifier trained over all semantic classes is not able to



(a) Dortmund (b) Enschede (c) Zurich

Figure 2: Cross-validated precision-recall curves of the supervised, non-semantic filtering approach (dashed lines) and the semantically informed filtering approach (solid lines). The colors indicate the semantic classes building, roof, ground, vegetation, fields, and water.

separate inliers and outliers equally well as a class-specific one. While this effect is smaller for classes that show similar point distributions like vegetation and fields of Dortmund, it becomes apparent for classes with very different inlier-outlier distributions like building. Roofs, ground, fields, vegetation, and water bodies are mainly horizontally oriented. In contrast, building facades are mostly vertical and dense matching is hampered by textureless regions, repetitive textures, and surface parts corrupted by specular reflections. As a result, building points exhibit a fundamentally different inlier-outlier distribution, which is only inaccurately captured by a classification model averaged over all semantic classes.

We provide precision-recall curves in Figure 2. Semantically informed filtering consistently outperforms supervised, non-semantic filtering in terms of both recall and precision. The most striking improvement is observed for the building class (red lines).

Qualitative comparison Figure 3 shows detailed views of the raw, unfiltered point clouds and their filtered versions. Points located in free space are removed correctly by all filtering approaches. However, the quality of the filtering is improved in three different ways if inlier-outlier distributions are learned in a class-specific way. Firstly, isolated building, roof, and water points are correctly removed. Secondly, clustered outliers located between building fronts and erroneous building points in the vicinity of dense roof areas are successfully discarded, too. Thirdly and most importantly, considerably more inliers are retained in low-density areas like vertical building facades.

Generalization capability A general drawback of supervised methods compared to unsupervised ones is that a new model usually has to be trained from scratch per scene. To verify the extent to which our learned models generalize across different scenes, we train two data sets and test on the third (results are shown in Tab. 3).

Class-specific models consistently outperform average ones across scenes regarding precision and F_1-score. An interesting finding is that precision (and F_1-scores) of all classes are high compared to the unsupervised, heuristic baseline (*c.f.* Tab. 1). This indicates that a supervised outlier filter trained on a different scene might still work better than an unsupervised heuristic. However, this has to be taken with a grain of salt due to the limited number of data sets, similar acquisition properties, and scene content. As soon as point cloud distributions vary strongly across scenes, this might no longer hold. However, re-training a pre-trained classifier on a very small portion of the new scene might solve the problem. We leave this for future work.

6. CONCLUSION

In this paper we propose to formulate outlier filtering in MVS point clouds as a supervised classification problem. Further, given point-wise class likelihoods, we show that incorporating class-specific knowledge for outlier detection significantly improves precision while keeping inliers in low-density areas like building facades. Main insights of this work are that (i) inlier-outlier distributions in aerial MVS point clouds are class-specific, (ii) training supervised classifiers per class improves over learning average distributions across all classes, (iii) once classifiers have been trained on a sufficiently large amount of training data, models generalize relatively well to new scenes under the assumption that these have been acquired and pre-processed similarly.

Despite the generic nature of the developed point cloud filtering algorithm, a bottleneck is transferability of a trained model to a new scene and modality. As with any supervised classifier, learned inlier-outlier distributions are directly related to the acquisition technique (*e.g.*, active or passive measurement method, sensor type, aerial or terrestrial data acquisition, flight plan, *etc.*) as well as the scene content. Application to entirely new scenes that contain a very different set of classes or that have been acquired with a different sensor type need labeled reference data.

In future work we will investigate this transfer learning problem in more detail. We will experiment with point clouds of different modalities (*e.g.*, LiDAR) and replace the traditional classification pipeline with a 3D deep learning approach.

REFERENCES

Amer, M., Goldstein, M. and Abdennadher, S., 2013. Enhancing one-class support vector machines for unsupervised anomaly detection. In: ACM SIGKDD Workshop on Outlier Detection and Description.

Angiulli, F. and Pizzuti, C., 2002. Fast outlier detection in high dimensional spaces. In: European Conference on Principles of Data Mining and Knowledge Discovery.

Barnett, V. and Lewis, T., 1974. Outliers in statistical data. Wiley.

Benbouzid, D., Busa-Fekete, R., Casagrande, N., Collin, F.-D. and Kégl, B., 2012. Multiboost: A multi-purpose boosting package. The Journal of Machine Learning Research (JMLR) pp. 549–553.

Bláha, M., Rothermel, M., Oswald, M., Sattler, T., Richard, A., Wegner, J. D., Pollefeys, M. and Schindler, K., 2017. Semantically informed multiview surface refinement. In: ICCV.

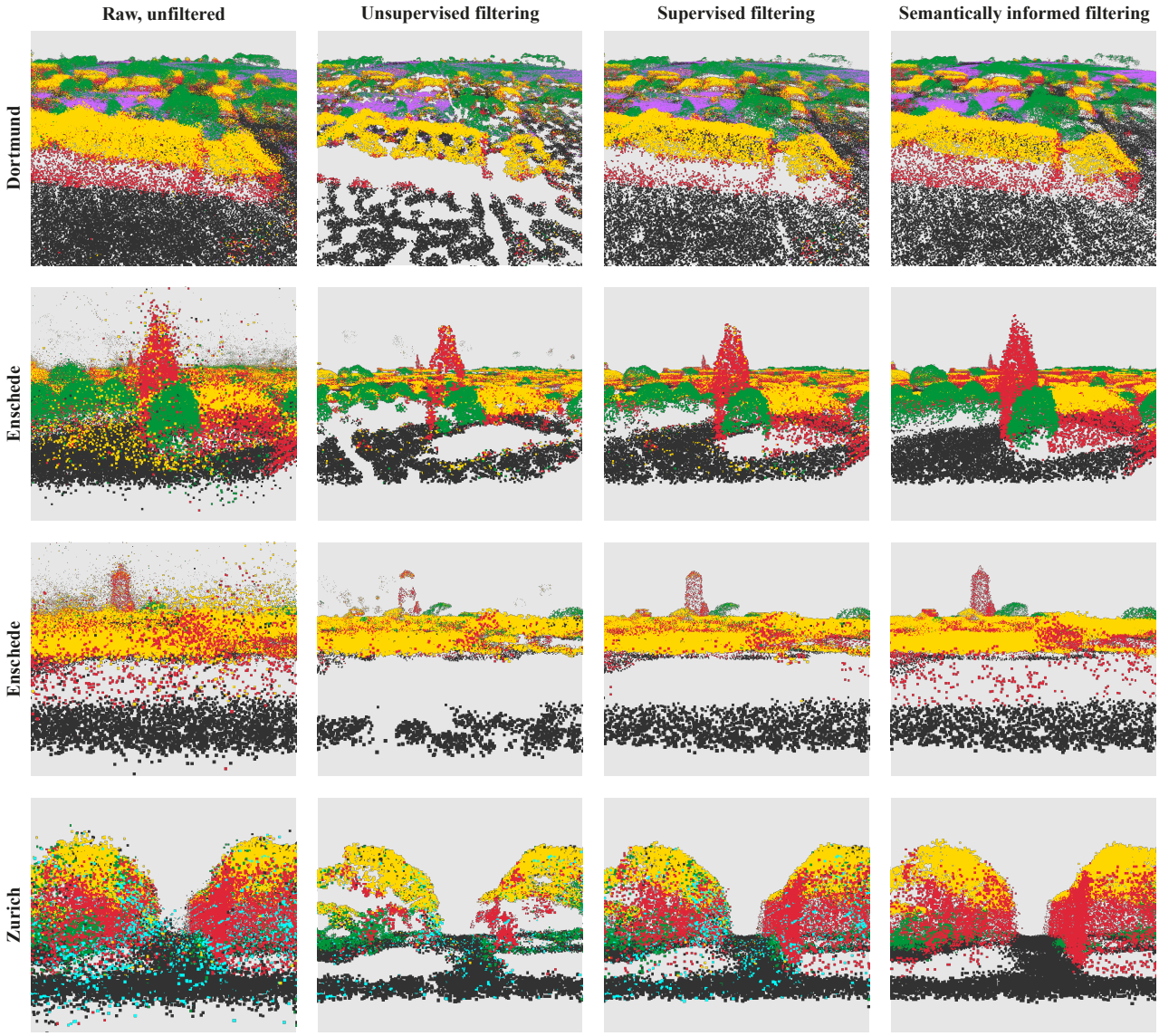


Figure 3: Qualitative filtering results. *Left:* Raw, unfiltered point cloud. *Middle left:* Filtered point cloud using unsupervised learning (Sotoodeh, 2006). *Middle right:* Filtered point cloud using supervised learning. *Right:* Filtered point cloud using semantically informed learning. The colors indicate the semantic classes building, roof, ground, vegetation, fields, and water.

Training sites	Testing site	Classes	Supervised filtering			Semantically informed filtering		
			Precision [%]	Recall [%]	F_1 [%]	Precision [%]	Recall [%]	F_1 [%]
Enschede	Dortmund	Building	81.42	54.15	65.04	86.91	77.09	81.71
Zurich		Ground	79.75	59.35	68.05	80.72	66.12	72.70
		Roof	82.79	58.23	68.37	83.62	71.31	76.98
		Vegetation	76.55	57.08	65.40	85.93	48.57	62.06
Dortmund	Enschede	Building	79.54	60.62	68.80	87.74	51.87	65.20
Zurich		Ground	82.03	66.25	73.30	84.78	66.59	74.59
		Roof	83.81	66.96	74.44	79.42	66.57	72.43
		Vegetation	79.91	71.29	75.35	85.20	69.53	76.57
Dortmund	Zurich	Building	73.78	67.93	70.74	84.85	59.31	69.82
Enschede		Ground	79.16	53.26	63.68	80.74	61.68	69.94
		Roof	80.74	51.95	63.22	83.03	62.04	71.02
		Vegetation	85.73	60.94	71.24	91.35	60.91	73.09

Table 3: Results for filtering approaches trained on two data sets and applied to the third. Note that classes that appear in only one of the data sets (fields in Dortmund, water in Zurich) are not shown.

Bláha, M., Vogel, C., Richard, A., Wegner, J. D., Pock, T. and Schindler, K., 2016. Large-scale semantic 3d reconstruction: an adaptive multi-resolution model for multi-class volumetric labeling. In: CVPR.

Breiman, L., 2001. Random forests. Machine Learning 45(1), pp. 5–32.

Breunig, M. M., Kriegel, H.-P., Ng, R. T. and Sander, J., 2000. LOF: identifying density-based local outliers. In: ACM Sigmod Record, pp. 93–104.

Brodu, N. and Lague, D., 2012. 3d terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. ISPRS Journal of Photogrammetry and Remote Sensing 68, pp. 121–134.

Chandola, V., Banerjee, A. and Kumar, V., 2009. Anomaly detection: A survey. ACM Computing Surveys.

Chehata, N., Guo, L. and Mallet, C., 2009. Airborne lidar feature selection for urban classification using random forests. ISPRS Archives.

Cheng, S.-W. and Lau, M.-K., 2017. Denoising a point cloud for surface reconstruction. arXiv preprint arXiv:1704.04038.

Deschaud, J.-E. and Goulette, F., 2010. Point cloud non local denoising using local surface descriptor similarity. ISPRS Archives 38(3A), pp. 109–114.

Digne, J., 2012. Similarity based filtering of point clouds. In: CVPR Workshops.

Eskin, E., 2000. Anomaly detection over noisy data using learned probability distributions. In: ICML.

Fleishman, S., Cohen-Or, D. and Silva, C. T., 2005. Robust moving least-squares fitting with sharp features. ACM Transactions on Graphics 24(3), pp. 544–552.

Furukawa, Y. and Ponce, J., 2010. Accurate, dense, and robust multiview stereopsis. IEEE TPAMI 32(8), pp. 1362–1376.

Goesele, M., Snavely, N., Curless, B., Hoppe, H. and Seitz, S. M., 2007. Multi-view stereo for community photo collections. In: ICCV.

Guennebaud, G. and Gross, M., 2007. Algebraic point set surfaces.

Häne, C., Heng, L., Lee, G. H., Sizov, A. and Pollefeys, M., 2014. Real-time direct dense matching on fisheye images using plane-sweeping stereo. International Conference on 3D Vision, pp. 57–64.

Häne, C., Zach, C., Cohen, A., Angst, R. and Pollefeys, M., 2013. Joint 3d scene reconstruction and class segmentation. In: CVPR.

He, Z., Xu, X. and Deng, S., 2003. Discovering cluster-based local outliers. Pattern Recognition Letters 24(9), pp. 1641–1650.

Hirschmüller, H., 2008. Stereo processing by semiglobal matching and mutual information. IEEE TPAMI 30, pp. 328–341.

Jin, W., Tung, A. K., Han, J. and Wang, W., 2006. Ranking outliers using symmetric neighborhood relationship. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining.

Knorr, E. M. and Ng, R. T., 1998. Algorithms for mining distance-based outliers in large datasets. In: International Conference on Very Large Data Bases.

Kriegel, H.-P., Zimek, A. et al., 2008. Angle-based outlier detection in high-dimensional data. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Latecki, L. J., Lazarevic, A. and Pokrajac, D., 2007. Outlier detection with kernel density functions. In: International Workshop on Machine Learning and Data Mining in Pattern Recognition.

Levin, D., 2004. Mesh-independent surface interpolation. In: Geometric Modeling for Scientific Visualization, Springer, pp. 37–49.

Lipman, Y., Cohen-Or, D., Levin, D. and Tal-Ezer, H., 2007. Parameterization-free projection for geometry reconstruction. ACM Transactions on Graphics.

Ma, X. and Cripps, R. J., 2011. Shape preserving data reduction for 3d surface points. Computer-Aided Design 43(8), pp. 902–909.

Öztireli, A. C., Alexa, M. and Gross, M., 2010. Spectral sampling of manifolds. ACM Transactions on Graphics.

Öztireli, A. C., Guennebaud, G. and Gross, M., 2009. Feature preserving point set surfaces based on non-linear kernel regression. In: Computer Graphics Forum, pp. 493–501.

Öztireli, C., 2015. Making sense of geometric data. IEEE Computer Graphics and Applications 35(4), pp. 100–106.

Ramaswamy, S., Rastogi, R. and Shim, K., 2000. Efficient algorithms for mining outliers from large data sets. In: ACM SIGMOD International Conference on Management of Data, New York, NY, USA.

Rätsch, G., Mika, S., Schölkopf, B. and Müller, K.-R., 2002. Constructing boosting algorithms from SVMs: an application to one-class classification. IEEE TPAMI 24(9), pp. 1184–1199.

Roth, V., 2004. Outlier detection with one-class kernel fisher discriminants. In: NIPS.

Rusu, R. B. and Cousins, S., 2011. 3d is here: Point cloud library (PCL). In: EEE International Conference on Robotics and Automation (ICRA).

Schall, O., Belyaev, A. and Seidel, H.-P., 2005. Robust filtering of noisy scattered point data. In: Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium.

Sotoodeh, S., 2006. Outlier detection in laser scanner point clouds. ISPRS Archives.

Tang, J., Chen, Z., Fu, A. W.-C. and Cheung, D. W., 2002. Enhancing effectiveness of outlier detections for low density patterns. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining.

Weinmann, M., Jutzi, B. and Mallet, C., 2013. Feature relevance assessment for the semantic interpretation of 3d point cloud data. ISPRS Annals 5, pp. W2.

Weinmann, M., Jutzi, B., Hinz, S. and Mallet, C., 2015a. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS Journal of Photogrammetry and Remote Sensing 105, pp. 286–304.

Weinmann, M., Urban, S., Hinz, S., Jutzi, B. and Mallet, C., 2015b. Distinctive 2d and 3d features for automated large-scale scene analysis in urban areas. Computers & Graphics 49, pp. 47–57.

Wolff, K., Kim, C., Zimmer, H., Schroers, C., Botsch, M., Sorkine-Hornung, O. and Sorkine-Hornung, A., 2016. Point cloud noise and outlier removal for image-based 3d reconstruction. In: IEEE International Conference on 3D Vision.

Wu, C., 2011. VisualSFM: A visual structure from motion system.

Yu, D., Sheikholeslami, G. and Zhang, A., 2002. Findout: Finding outliers in very large datasets. Knowledge and Information Systems 4(4), pp. 387–412.

Zhang, K., Hutter, M. and Jin, H., 2009. A new local distance-based outlier detection approach for scattered real-world data. Advances in Knowledge Discovery and Data Mining pp. 813–822.