

MIN-CUT BASED SEMANTIC BUILDING LABELING FOR AIRBORNE LIDAR DATA

Serkan Ural^{a,b,*} and Jie Shan^a

^aLyles School of Civil Engineering, Purdue University, West Lafayette, IN 47907, USA - jshan@purdue.edu

^bInstitute for Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland - sural@phys.ethz.ch

Commission II, WG II/3

KEY WORDS: Building extraction, lidar, graph-cuts, semantic segmentation, local feature histogram

ABSTRACT:

Classification and segmentation of buildings from airborne lidar point clouds commonly involve point features calculated within a local neighborhood. The relative change of the features in the immediate surrounding of each point as well as the spatial relationships between neighboring points also need to be examined to account for spatial coherence. In this study we formulate the point labeling problem under a global graph-cut optimization solution. We construct the energy function through a graph representing a Markov Random Field (MRF). The solution to the labeling problem is obtained by finding the minimum-cut on this graph. We have employed this framework for three different labeling tasks on airborne lidar point clouds. Ground filtering, building classification, and roof-plane segmentation. As a follow-up study on our previous ground filtering work, this paper examines our building extraction approach on two airborne lidar datasets with different point densities containing approximately 930K points in one dataset and 750K points in the other. Test results for building vs. non-building point labeling show a 97.9% overall accuracy with a kappa value of 0.91 for the dataset with 1.18 pts/m² average point density and a 96.8% accuracy with a kappa value of 0.90 for the dataset with 8.83 pts/m² average point density. We can achieve 91.2% overall average accuracy in roof plane segmentation with respect to the reference segmentation of 20 building roofs involving 74 individual roof planes. In summary, the presented framework can successfully label points in airborne lidar point clouds with different characteristics for all three labeling problems we have introduced. It is robust to noise in the calculated features due to the use of global optimization. Furthermore, the framework achieves these results with a small training sample size.

1. INTRODUCTION

Simpler representations derived from airborne lidar point clouds usually provide more practical and manageable input for efficient analysis in many applications. Attaining such useful simpler representations as final products usually demands extensive efforts for processing very large unstructured point cloud data (Yang et al., 2016; Guan et al., 2013). Labeling of point clouds either for classification or segmentation provides a level of organization which helps achieve such representations.

Earlier research focuses more on handling lidar data as 2.5D by resampling the point clouds into raster grids. Such approaches may be convenient for employing a wide range of well-established image processing algorithms, but loss of information is inevitable due to resampling. More recently, this limitation encouraged more research that exploited lidar data in 3D. Promising results have been reported in labeling 3D point clouds within the last decade. Among others, approaches employing features calculated for each point by considering nearby points within a spatial neighborhood are especially notable. Features suitable for discriminating various properties of points have been utilized for 3D point labeling using a broad range of methods (Guo et al., 2016). In this study, we present an approach for the labeling of lidar point clouds either for classification or segmentation with their 3D coordinates as the only input.

While classification and segmentation are usually considered separately in numerous algorithms developed over the years for either task, some methods (Martinez et al., 2016; Carlberg et al., 2009) also perform semantic classification of the segments after segmentation. Vilariño et al. (2017) employ a two-phase region growing algorithm for the segmentation of point clouds and then classify the segments by a rule-based classification method. In contrast to classification after segmentation, classification of individual points without segmentation (Weinmann et al., 2015a,

Blomley and Weinmann, 2017, Bauchet and Lafarge, 2019) and segmentation as a refinement step after point-wise classification (Weinmann et al., 2017) have also been explored. Polewski et al. (2014) similarly employ point classification as the initial step of segmentation for the detection of fallen trees. Considerably more recently, a shift towards achieving optimal labeling is observed in published research.

While some approaches disregard any contextual information, significant amount of research also focus on considering local dependencies for the segmentation or classification of lidar point clouds while simultaneously relying on global conformance. Xu and Stilla (2019), as a recent example, use global graph-based clustering for an unsupervised automatic segmentation of point clouds to extract contours of planar elements of building facades. Markov random fields (MRF) and conditional random fields (CRF) provide a convenient structure for the formulation of point labeling problems by taking contextual information into consideration. An early example of research on labeling 3D point clouds with an MRF based approach by Anguelov et al. (2005) uses associative Markov networks (AMNs) coupled with linear programming inference for segmentation of objects and object classes from data acquired by a mobile laser scanning system. Research employing MRFs for point classification often employ point features derived from the eigenvalues of the covariance matrix of points within a neighborhood. Ural and Shan (2012) label points as surface or scatter via min-cut optimization using point features derived from the eigenvalues of the structure tensor. Similarly, Sun and Salvaggio (2013), classify trees by graph-cut optimization for which the data costs are calculated by thresholding the smallest eigenvalue. Du et al. (2017) employ grid based features along with the flatness feature calculated with the eigenvalues to perform a graph-cut based point labeling for building extraction. Bae and Mekurjev (2017) employ their proposed convex relaxation and optimization framework on

* Corresponding author

graphs for labeling point clouds as ground, man-made structures, and vegetation using an energy function combining region and edge-based features. MRF formulation can also be applied to groups of points in an object-based manner rather than individual points. Zhu et al. (2017) first generate supervoxels by clustering points using homogeneity constraints and classify the supervoxels via graph-cut based energy minimization. Lafarge and Mallet (2012) employ an MRF based label propagation procedure for planimetric arrangement of geometric shapes and urban components with input from unsupervised classification of 3D points using a five-parameter energy function and a Potts model for pairwise interactions.

Along with MRF based methods, CRF formulation of point cloud labeling has also advanced within the last decade. As one of the earlier examples, Lim and Suter (2007) propose a method employing conditional random fields for the classification of 3D point clouds that are adaptively reduced by omitting geometrically similar features. Niemeyer et al. (2012) also utilize CRF formulation for the supervised classification of lidar point clouds using loopy belief propagation (LBP) for inference. In their later research, Niemeyer et al. (2014) employ random forest (RF) classification to calculate the pairwise potentials of CRF model for labeling airborne point clouds. Similarly, Weinmann et al. (2015b) label mobile laser scanning data in a CRF framework using LBP for inference with optimal neighborhood estimation. They use RFs to calculate association potentials in the CRF model but prefer a simpler Potts model for calculating interaction potentials. Lang et al. (2016) apply an adaptive graph down-sampling to reduce the computational burden of large datasets in their CRF model and LBP inference-based framework for point labeling with histogram of oriented residuals (HOR) as point features. Smooth classification of 3D points clouds can also be achieved through regularization based on structured optimization as proposed by Landrieu et al. (2017).

In this study, we propose a framework which takes advantage of the contextual formulation capabilities of MRFs coupled with powerful graph-cut optimization for semantically labeling point clouds with focus on building extraction. We introduce a new approach for utilizing the point features for the calculation of data and smoothness costs of the energy function.

2. GRAPH-CUT APPROACH FOR POINT LABELING

We propose a graph-cut point labeling framework for various point labeling tasks. In the case of building extraction from airborne point clouds, different stages of the process can be formulated within this framework, separately or in combination: 1) Ground filtering, in which the points are labeled as ground or off-ground. 2) Labeling surface vs non-surface points for extracting the points that fall on planar or curved surfaces like building roofs. 3) Segmentation of roof planes by labeling the points on the same roof facet. The generic workflow for semantic point labeling with this graph-cut based approach is given in Figure 1.

Point labeling is achieved in each stage by formulating the individual problem on a graph with data cost and smoothness cost designed specific to its nature. In Ural and Shan (2016), we presented the details of our min-cut based filter for handling the ground filtering task as one of the main steps of this framework. This paper provides the details for the remaining two tasks, namely, building extraction, and roof segmentation through the proposed framework. They differ from the unsupervised ground filtering approach in Ural and Shan (2016) regarding the local point features and energy functions employed. We describe the implementation details of each stage including graph

construction, energy functions, optimization methods, and local point features used for calculating data and smoothness terms of the energy function. We also introduce a novel feature, Multi-level Local Feature Histograms (MLFH), used as a descriptor in the point labeling process for building extraction.

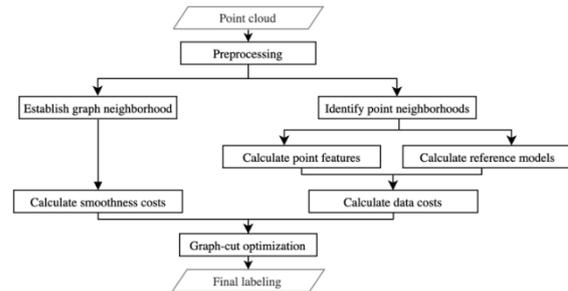


Figure 1. Workflow for min-cut based semantic point labeling from airborne lidar data.

2.1. Graph-based Formulation of the Problem

We construct a graph model for labeling the points in the lidar point cloud by considering each lidar point as a node of a graph. The edges between the nodes are defined such that each node is connected to its Voronoi neighbors with an edge. 2D neighborhood is used in case of ground filtering and roof segmentation, and 3D neighborhood for surface classification. The special nodes representing the labels are also connected to each node with an edge. Labels differ depending on the required outcome. They are determined as ground and off-ground in the case of ground filtering. Labels for differentiating the points sampling a surface vs. others are identified to be surface and non-surface. Sequential integers are used as labels for each segment during the segmentation of the roof facets. Edges connecting each point to the label nodes are given the weights calculated as the costs for assigning the corresponding label to each point. These constitute the data cost term of the energy function. Smoothness costs between pairs of nodes connected with an edge on the graph are calculated and assigned as weights to the corresponding edges. The minimum-cut severs the edges on the graph corresponding to the minimum energy. Thus, points that remain connected to either the source (S) or the sink (T) node are labeled accordingly.

2.2. Graph-cut Solution

In this research, we employ the graph-cut interpretation of energy minimization that is commonly used for image pixel labeling as extended to the geometry of unstructured 3D point clouds. We adapt the fast approximate energy minimization algorithm via graph-cuts introduced by Boykov et al. (2001) to formalize the labeling problems of unstructured 3D point clouds. We employ the α -expansion move they proposed on graphs in our framework when more than two labels are involved (Boykov et al., 2011).

2.3. Data Costs

Calculating the data cost requires a distance measure between the proposed histogram of the multi-level local features of a point and the histogram representing the class label. The details of the point features involved and how these histograms are calculated are explained in section 3.2. A suitable distance measure for evaluating their similarity is the Jeffries-Matusita (JM) distance which is a measure commonly used to calculate the dissimilarity between two probability distributions (Richards and Jia, 2006; Dabboor et al., 2014), described by Jeffreys (1946) as one of the two “invariants expressing the difference between two

distributions of chance". We calculate the data cost for surface vs. non-surface point classification as the JM distance between two histograms to measure their dissimilarity. The data cost is then

$$D(H_{ip}) = s \sqrt{\sum_b [\sqrt{H_i(b)} - \sqrt{H_p(b)}]^2} \quad (1)$$

where H_i is the feature histogram of the training points representing a class model, H_p is the histogram of the local point features for point p , and b refers to the histogram bins. Once calculated, we scale the data cost values between zero and 100 to be utilized in graph-cut optimization (GCO) for labeling the points. The higher the data cost, the more dissimilar the points are to the surface class.

Data cost function for the segmentation of the surface points considers the Euclidean distance $d_{f_{pi}}$ in the feature space between the point's feature vector f_p and the label means μ_{f_i} . The labels here are the initial labels acquired as a result of a watershed segmentation of the 3D surface normal feature space.

$$d_{f_{pi}} = \|\mu_{f_i} - f_p\| \quad (2)$$

2.4. Smoothness Costs

Smoothness cost functions are defined such that a smoothness penalty is introduced when two nodes of the graph are labeled different than each other. This allows the parameter σ to control the range within which the cost will decrease rapidly with respect to the difference of the feature values of the two nodes. It regulates that any configuration which assigns different labels to two connected nodes having similar properties are less likely to be preferred over labeling them the same. We use a function of three parameters as the smoothness cost for the classification of surface vs. non-surface points in the point cloud. These three parameters are the smoothness parameter σ , the feature distance measure $D(H_{pq})$ between the MLFHs of the two nodes connected with an edge on the graph calculated in the same way as in Equation 1, and the 3-D spatial Euclidean distance $d(p, q)$ between them. Smoothness energy for the classification of surface vs non-surface points is then calculated as

$$V_{p,q}(f_p, f_q) = \frac{1}{d(p,q)} \exp\left(-\frac{(D(H_{pq}))^2}{2\sigma^2}\right) \quad (3)$$

where $d(p, q)$ is the Euclidean distance between points p and q in the spatial domain.

Smoothness costs for roof plane segmentation are calculated as a function of the Euclidean distance $d(p, q)$ between the two points connected to each other with an edge and the angle between their surface normals which replaces the term $D(H_{pq})$ in Equation 3.

3. BUILDING CLASSIFICATION

3.1. Labeling Building Points

Identifying building roof points is an essential task for building extraction from airborne lidar point clouds. This task may be regarded as the identification of points that are on the same planar surface since most building roofs can be modeled as a combination of planar patches. This subsection describes the process of labeling surface and non-surface points using the proposed graph-cut labeling approach. Calculating data and smoothness costs for the framework as described in the previous section requires the calculation of point feature descriptors for each point. We calculate features for the local neighborhood of

each point and then label the points using these calculated features. We employ the features calculated for each point as described below and introduce multi-level local feature histograms (MLFH) as a new approach for utilizing these features as point descriptors for labeling surface vs. non-surface points.

We establish a feature vector for each point using all the points within their individual neighborhoods and then calculate the MLFH for each point from these feature vectors as described in the following section. Next, each point is labeled via graph-cut optimization using the costs calculated with the MLFH of the training samples collected for surface and non-surface classes.

3.2. Multi-level Local Feature Histograms

Eigenvalues of the covariance matrix of a point's neighborhood have been investigated in various studies to understand the geometric properties and structure of the data. West et al. (2004) use a set feature descriptions, including structure tensor omnivariance (S.T.O.), structure tensor anisotropy (S.T.A.), structure tensor planarity (S.T.P.), structure tensor eigenentropy (S.T.E.), structure tensor linearity (S.T.L.), structure tensor sphericity (S.T.S.), all functions of the eigenvalues, $(\lambda_1, \lambda_2, \lambda_3)$ of the covariance matrix of the point's local neighborhood where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$. Among these, we employ the S.T.P. and S.T.S. features calculated for the local neighborhood of each point for labeling surface and non-surface points. Every point with enough number of points within their predefined neighborhood then has a feature vector with two elements which represents the geometric properties of that point.

$$S.T.P. = (\lambda_2 - \lambda_3)/\lambda_1; \quad S.T.S. = 1 - (\lambda_1 - \lambda_3)/\lambda_1 \quad (4)$$

Using point feature vectors allows a classifier to be trained and then each point can be assigned a label through a classification algorithm. One shortcoming of using local point features in this manner directly for labeling airborne lidar points is that the feature vectors may vary due to noise in the dataset. As a result, individual points may be labeled in contradiction with the labels of their neighboring points even though they are of the same nature.

As a more robust approach, accumulating a histogram by counting various properties that are calculated based on the geometric relationships of points within a reference point's neighborhood has been widely used as a method of generating point descriptors. Such histograms may be generated both by binning the spatial domain or the feature domain (Guo et al., 2014). One popular method for binning the feature space is the Point Feature Histograms (PFH) by Rusu et al. (2008) which considers pairwise features among the points in the neighborhoods calculated with reference to a local reference frame. This allows the evaluation of pairwise geometric relationships of points through a histogram generated for all pair combinations in the neighborhood. Fast Point Feature Histograms (FPFH), a more computationally efficient version of this method was also introduced later (Rusu et al., 2009). This approach emerges as a solution to the weakness of single point features not being descriptive enough, and multi-valued point features not being robust enough in the existence of noise in the point clouds (Rusu et al., 2009). Many other histograms have been proposed in literature as local point descriptors. A detailed review can be found in (Guo et al., 2014).

We propose the use of histograms at multiple levels. As mentioned before, we employ a multi-valued feature vector as a descriptor for the geometry of points' local neighborhoods. First,

we calculate the feature vector for a point of interest, p_i , using all the points in its predefined spherical neighborhood with radius R . Next, we calculate the feature vectors for each of the points p_{ij} , that are within the neighborhood of point p_i , using the points p_{jk} , in their respective predefined individual neighborhoods. Then, we divide the 2D feature space into 10x10 bins and count the number of points which fall into each bin to generate a histogram of calculated feature vectors. Due to the involvement of point neighborhoods at multiple levels, we call these histograms multi-level local feature histograms (MLFH). For each point with enough neighbors in the dataset, we calculate the features in their respective neighborhoods. Points which don't have a sufficient number of neighbors are excluded from processing. The histogram is normalized with respect to the points falling into each bin over the total number of points.

We also generate MLFH to serve as models for the training samples we collect from the point clouds that are representative of the semantic classes that we'd like to extract. In our building extraction framework, the two classes are: i) the surface class, which includes the points on the surfaces of man-made structures like building roofs as well as natural surfaces like the ground, ii) non-surface class which includes the points that are not on a surface, such as the trees. Figure 2 below presents examples of mean MLFH generated for these two classes.

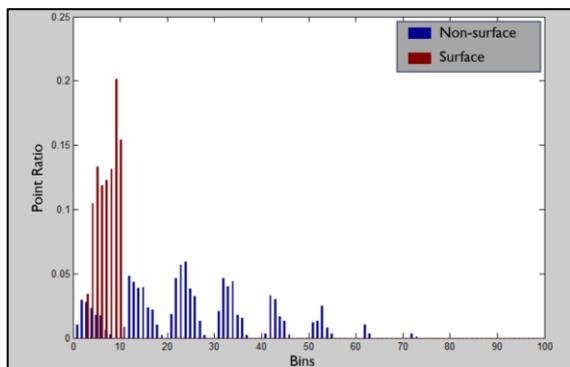


Figure 2. Examples of mean MLFH for training samples of the non-surface and surface classes. The horizontal axis is the bins in the feature space, while the vertical axis is the ratio of points in each bin to the number of points in the entire neighborhood.

4. BUILDING AND ROOF SEGMENTATION

4.1. Labeling Individual Buildings

Once off-ground surface points are acquired by GCO labeling, we identify individual buildings with the assumption that off-ground surface points mainly constitute buildings. We then remove the points that are aligned vertically (e.g. building walls) as well as the noise, points that are not considered as buildings, from building extraction results. At this stage, the focus is on the building points. The result includes noise-free building points. We calculate the surface normal for each points' spherical neighborhood with radius R to obtain the angular divergence of each point's surface normal from the horizontal vector at that point. Each point with a divergence angle smaller than a threshold of 10° is then removed from the dataset since they mainly constitute the building walls. Next, we employ DBSCAN (Density Based Spatial Clustering of Applications with Noise) clustering algorithm to label individual buildings.

4.2. Labeling Individual Roof Planes

Until this stage, our framework addressed two common point cloud labeling tasks; ground filtering and building extraction. As

a third common task, we employ our framework for the segmentation of roof planes by labeling the building points. Our approach for roof plane segmentation relies on the optimization of a noisy over-segmentation achieved by watershed segmentation of the normal feature space. After the labelling, each remaining segment corresponds to a roof facet. We first remove all points with less than two neighbors within a predefined spherical neighborhood. Then, we calculate the surface normal for each point and ensure the consistency of their orientations (up/down). After the reorientation of the normals, we divide the 3D surface normal feature space into bins and generate a 3D histogram of bin counts. Then we apply watershed segmentation (Meyer, 1994) on the inverse of this histogram and assign each point the labels that are the result of this segmentation. This labeling serves as the initialization which is later optimized via graph-cut optimization to acquire the final roof plane labels.

5. EXPERIMENTS AND RESULTS

5.1. Test Data

The first dataset we used in this research is the lidar point cloud of part of Bloomington, Indiana of USA obtained from the Indiana Spatial Data Portal. It was collected as part of Monroe County orthophotography and lidar data acquisition carried out by MJ Harden on April 11-12, 2010 with an Optech Gemini system. The size of the study area is approximately 4200 x 2080 sq ft (1280 x 634 sq m). It spans a typical U.S. Midwest suburban area with residential settlement including houses and apartment buildings of various types and sizes as well as vegetation with varying density. There is no abrupt elevation change in this noticeably flat area. Reported vertical RMSE is 0.347 ft/10.58 cm. The point coordinates are in NAD 1983 HARN horizontal datum and NAVD 88 vertical datum projected on the Indiana West State Plane Coordinate System. The point cloud includes approximately 930K points and is provided in LAS 1.2 format. The average point density is 1.13 pts/m² when calculated for all points in the study area, and 0.97 pts/m² for the last returns only. Data specifications close to these values are more common typically for lidar acquisitions with the purpose of generating base topographic products in the U.S. for counties or larger geographic areas. Figure 3a shows the study area which covers approximately 2.25 km².

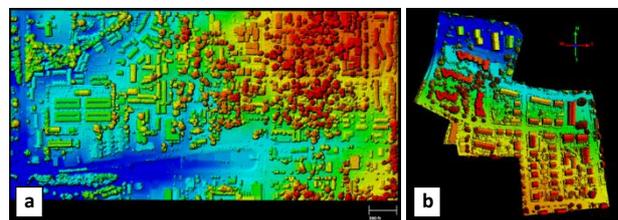


Figure 3. DSMs of (a) Bloomington with 5ft/1.5m and (b) Vaihingen with 0.3m GSD. Blue-to-red color ramp represents low-to-high elevation values relative for each dataset.

The second dataset is the airborne lidar point cloud over Vaihingen in Germany, part of the test dataset for ISPRS Test Project on Urban Classification and 3D Building Reconstruction (Niemeyer et al., 2014). It was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010). The dataset was acquired using a Leica ALS50 system at a mean flying height of 500 m above ground on August 21, 2008. The part of the dataset we used to test our method includes approximately 750K points in UTM Zone 32N horizontal coordinate system. Average point density we calculated for the study area is 8.83 pts/m² for all points and 8.61 pts/m² for last returns. The point cloud is provided in ASCII

format including labels for each point corresponding to one of eight classes including, powerline, low vegetation, impervious surfaces, car, fence/hedge, roof, facade, shrub, and tree. The terrain is considerably flat. The scene as presented in Figure 3b is a built-up urban area covering approximately 83,700 m². It includes buildings of various sizes and styles along with moderate number of trees and low vegetation.

5.2. Labeling of Building Points

We calculated the S.T.P. and S.T.S. local point features using a spherical neighborhood with radius R . It was determined as the approximate radius of a circle which would cover an $n \times n$ grid around each point. Given the point density d_p of a dataset, the grid size Δ_g is calculated as $\Delta_g = 1/\sqrt{d_p}$, so that there are approximately the same number of grid cells in an area as the number of points (Kim and Shan, 2011). We used $R = 3.6$ m (~12 ft) for Bloomington dataset which corresponds to a 5x5 neighborhood around each point and $R = 1.7$ m for the Vaihingen dataset which corresponds to a 7x7 neighborhood.

We then generated the MLFH for each point's neighborhood using the local point features. We also calculated the MLFH for the training samples we collected representing surface and non-surface features. Bloomington and Vaihingen training datasets include 22 and 5 samples of surface patches having 7658 and 1655 points along with 34 and 6 samples of non-surface point clusters of 12199 and 7780 points respectively. We excluded the training data from the dataset when evaluating the results. Using the MLFH of off-ground points and the training datasets for surface and non-surface classes, we calculated the data costs for each point. We also calculated the smoothness costs for point pairs connected with an edge on the graph generated using the Voronoi neighborhood of points with each other. Smoothness parameter $\sigma = 0.8$ was used for both datasets. We assigned the weight $\lambda = 1$ for the smoothness cost term of Bloomington dataset and $\lambda = 5$ for the Vaihingen dataset. Each point in either dataset was then assigned one of two labels: surface or non-surface as a result of GCO. Final labeling of surface and non-surface points in Bloomington and Vaihingen datasets as are presented in Figure 4.

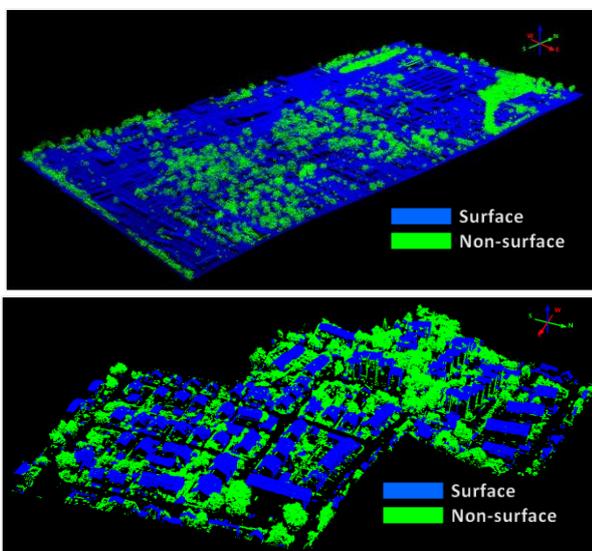


Figure 4. Bloomington lidar point cloud (top) and Vaihingen off-ground lidar points (bottom) labeled as surface (blue) and non-surface (green) points.

5.3. Labeling of Individual Buildings

The distance thresholds we used for clustering the building points with DBSCAN were 3 m and 1 m for Bloomington and Vaihingen datasets respectively. These values were determined approximately based on the $n \times n$ grid neighborhood around the points. Since along and cross-track point spacing may be different in airborne lidar data, neighborhood is identified to be large enough to provide the connection between the scan lines. Minimum number of points we required to exist in a cluster was four points for both datasets. The labels which consist of less than a minimum number of points that would not constitute a building roof were removed from the building points once all clustering was done and all tiles were combined in one dataset. We set the minimum number of building points to 20 for the Bloomington dataset and 30 for the Vaihingen dataset. Finally, any remaining point with less than the desired number of minimum neighbors within a distance R to the point were removed as the post process noise removal step. Figure 5 below shows the final labeling of individual buildings for Bloomington and Vaihingen datasets respectively. Surface points in the Bloomington dataset include the ground points since ground filtering is applied after this classification.

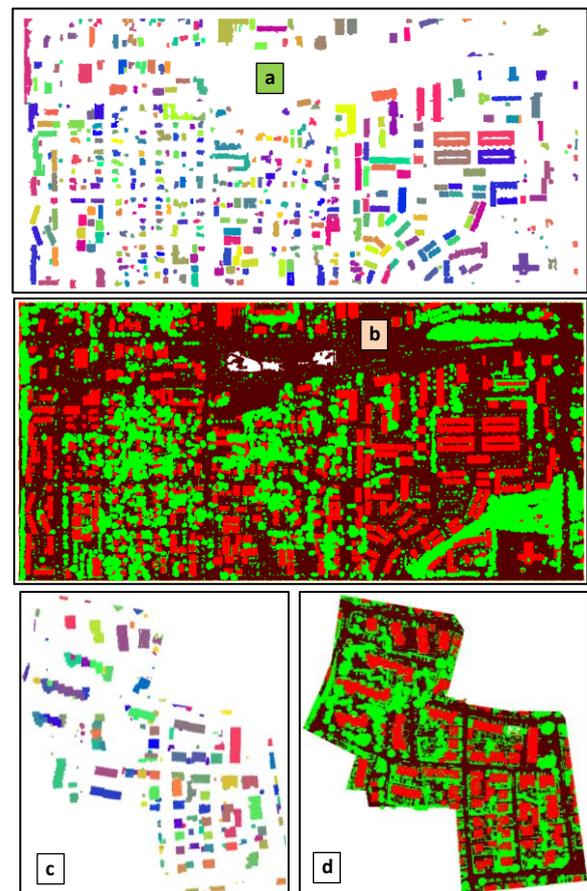


Figure 5. Building extraction result for Bloomington (a,b) and Vaihingen (c,d) datasets. (a,c): Points randomly colored for each individual building. (b,d): Points depicted as ground (brown), off-ground non-building (green), and building (red) classes.

5.4. Labeling of Individual Roof Planes

For the Bloomington and Vaihingen datasets, we have calculated the surface normal for each point within the spherical neighborhood with $R = 3.6$ m and $R = 1.2$ m respectively which correspond to a 5x5 neighborhood based on point density. Smoothness cost parameter was set to be $\sigma = 5$ for both datasets.

We applied the smoothness costs weights as twice and four times the data cost weights for Bloomington and Vaihingen datasets respectively. Several examples comparing the initial watershed segmentation and the GCO optimization results for Bloomington and Vaihingen datasets are presented in Figure 6.

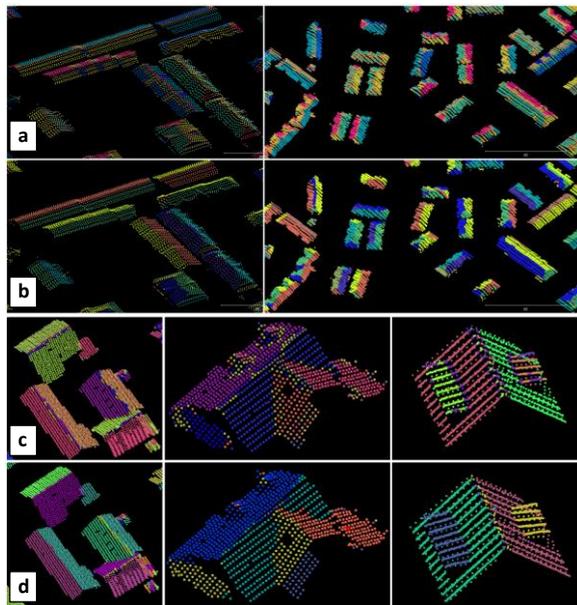


Figure 6. Examples of the initial watershed segmentation (a, c) and GCO (b, d) results for roof plane segmentation of the Bloomington (a,b) and Vaihingen (c,d) datasets.

6. EVALUATION AND DISCUSSION

6.1. Building Classification

Each point in the Vaihingen dataset provided by the ISPRS are labeled with one of nine different classes including a “Roof” class. We merged all the classes except the “Roof” class into one “Not Building” class and validated our results with these labels excluding the training samples. For the Bloomington dataset, we labeled all the roof points in the dataset manually for validation excluding the points used in training. We counted the true positive (TP), false positive (FP), true negative (TN), and false negative (FN) number of points and calculated the true positive rate (TPR) - sensitivity/recall-, true negative rate (TNR) – specificity-, false positive rate (FPR), false negative rate (FNR), overall accuracy, and Kappa values. Validation results are provided in Table 1. Figure 7 presents the building points that are missed during classification and points that are actually not building but misclassified as such for Vaihingen and Bloomington datasets.

		Reference				
<i>Bloomington</i>		Building	Not Building		Σ	
Classification	Building	115342	0.934	11556	0.014	126898
	Not Building	8139	0.066	794766	0.986	802905
	Σ	123481		806322		929803
	Accuracy:	0.979		Kappa:	0.909	
	<i>Vaihingen</i>					
Classification	Building	134257	0.931	12599	0.023	146856
	Not Building	9883	0.069	540543	0.977	550426
	Σ	144140		553142		697282
	Accuracy:	0.968		Kappa:	0.902	

Table 1. Confusion matrix presenting the TP, FP, TN, and FN number of points, TPR (sensitivity/recall), TNR (specificity), FPR, FNR, accuracy, and Kappa values for the Bloomington (top) and Vaihingen (bottom) building extraction results.

Typical mislabeling that we have observed were mainly due to i) trees that are too close to the rooftops, ii) dense treetops that resemble surfaces, iii) variable vertical sampling of the trees which results with insufficient points in the points’ neighborhoods to reflect the non-surface nature of the trees. The first two issues are closely related to the physical properties of the environment. For the third issue where the neighborhood is not large enough to define the geometric properties of the trees that are relatively sparsely sampled, one may choose to expand the neighborhood as long as this expansion doesn’t exceed the distances that affect the separation of the objects in the spatial domain. One may expand the neighborhood if the average point density is high enough. Then, the neighborhood will cover the 3D structure of relatively sparsely sampled trees when expanded. Also, it won’t be exceeding the distance that two objects may comfortably be separated from each other. One potential solution to the neighborhood size problem even for sparse sampling would be adjusting it to the local characteristics of the point cloud. This would in return add to the computational cost.

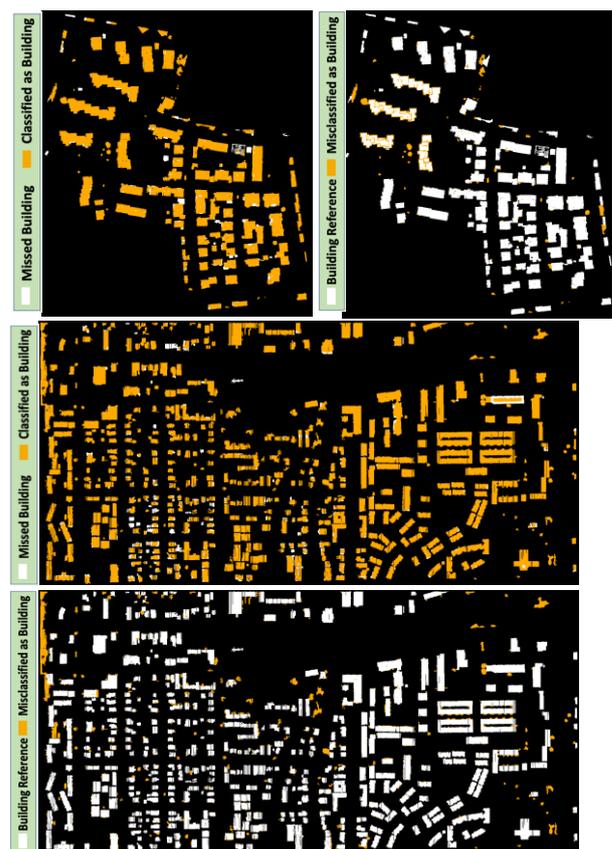


Figure 7. Building extraction results for Vaihingen (top) and Bloomington (middle & bottom) datasets with reference labeling.

6.2. Roof Plane Segmentation

We have selected 20 buildings from the reference dataset each with at least two roof planes and manually labeled each roof plane. Then we have calculated a confusion matrix for each building roof by comparing the hand labeled roof planes and the roof plane segmentation results. Table 2 below provides overall accuracy values calculated for each building and the average accuracy for 20 buildings. Accuracy value for each building is calculated as the ratio of the points identified on the correct reference plane to the total number of points. A visual inspection of the results shows that there is a trade-off between smoothness and the level of detail that needs to be preserved. This trade-off varies based on the point density. The two attachments as seen

in Figure 8 that are circled in red were not identified as different roof planes mainly due to their scale considering the point density of the dataset. Further investigation with more labeled reference data would be needed for a more robust evaluation that would provide better insight on the generalization potential of the approach. It should also be noted that the accuracy values in Table 2 is expected to be biased towards higher accuracies since larger planes with higher number of points are likely to be well detected.

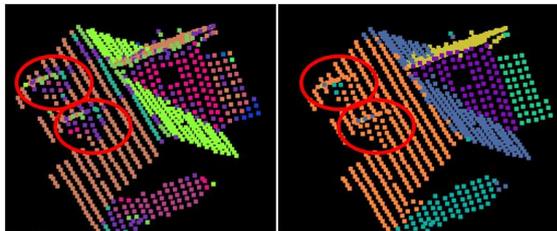


Figure 8. Example of a partially failed case of roof plane segmentation in the Vaihingen dataset due to the scale of the detail to be preserved. Left: Initial watershed segmentation result. Right: Final segmentation after GCO optimization.

Building ID	# Points	# Planes (Reference)	# Planes (Segmentation)	% Accuracy
1	1672	3	7	97.0
2	1280	4	5	83.8
3	805	2	3	97.5
4	1413	2	2	98.4
5	5854	2	4	94.6
6	2406	3	6	94.6
7	2198	3	7	87.5
8	2250	3	7	99.5
9	4200	4	8	67.4
10	1961	4	6	93.7
11	3728	2	6	97.9
12	2436	4	8	96.1
13	988	10	13	88.7
14	6648	3	7	94.9
15	1620	5	5	97.5
16	1390	3	5	94.8
17	687	6	7	66.8
18	3675	3	7	96.0
19	774	5	5	84.2
20	2613	3	8	93.3
Avg. Accuracy:				91.2

Table 2. Roof plane segmentation accuracies for 20 reference building roofs in the Vaihingen dataset.

7. CONCLUSION

This study has established a methodology for the point labeling problem based on the MRF formulation coupled with graph-cut optimization with the final objective of building extraction. We specified three different labeling tasks, namely, ground filtering, surface and non-surface classification, and roof plane segmentation, framed as an optimization problem on graphs and employed an efficient graph-cut algorithm to determine buildings using only 3D coordinates of airborne lidar points. At each labeling stage, we identified relevant point features and used these features to calculate the data costs and smoothness costs that are designed to calculate the global cost of labeling all points according to the nature of each labeling problem. Finally, we have labeled each point by minimizing the overall cost via graph-cut optimization on the graphs that we have formulated our labeling problems with.

Our framework could successfully label points in point clouds with different characteristics for all three labeling problems we have introduced. We have tested our approach for building extraction in two airborne lidar point cloud datasets with different point densities. Test results for building vs. non-

building point labeling show that we could achieve a 97.9% overall accuracy with a kappa value of 0.91 for the lower point density dataset (1.18 pts/m²) and a 96.8% accuracy with a kappa value of 0.90 for the higher point density dataset (8.83 pts/m²). Roof plane segmentation results provided 91.2% overall accuracy calculated with reference to the manually labeled roof planes of 20 buildings.

In surface and non-surface point classification, we have used feature vectors that represent the most fundamental geometric properties we were interested in. Even though training data were used to calculate data costs, distinctive class separation that we have observed in our tests suggests the possibility of using the same training data for similar datasets. Instead of directly using features extracted in each point's neighborhood for classification, we introduced Multi-level Local Feature Histograms (MLFH), as more robust descriptors in the point labeling process. MLFHs consider the distribution of the features calculated for each point in a point's neighborhood instead of relying only on one multi-valued feature vector for that point.

MRF formulation coupled with graph-cut optimization enabled the points to be labeled with the consideration of spatial coherence. Points which would otherwise be mislabeled were penalized to conform to their surrounding points' labeling within determined smoothness criteria.

Several issues arise in our framework regarding its labeling performance. In surface classification, misclassifications occurred for both surface and non-surface classes. Some of these misclassifications were due to insufficient sampling. Some others were due to a combination of point neighborhood radius selection and proximity of surface and non-surface features. Incorporating an adaptive neighborhood in our framework can possibly eliminate part of these misclassifications. Another adjustment to our method with potential improvement prospect is using additional point features which can help identify additional properties of the local neighborhood like edges and corners. In the next stage, we also plan to make a thorough quantitative evaluation in comparison with other approaches.

ACKNOWLEDGEMENTS

The Vaihingen dataset was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) [Cramer, 2010]: <http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html>. This research was mainly funded by Hacettepe University. Partial funding was provided by U.S. Army Research Office and Indiana Department of Transportation.

REFERENCES

- Anguelov, D., Taskarf, B., Chatalbashev, V., Koller, D., Gupta, D., Heitz, G., & Ng, A., 2005. Discriminative learning of Markov random fields for segmentation of 3D scan data. *CVPR 2005*, 2, 169–176.
- Bae, E., & Merkurjev, E., 2017. Convex Variational Methods on Graphs for Multiclass Segmentation of High-Dimensional Data and Point Clouds. *J. Math. Imaging Vision*, 58(3), 468–493.
- Bauchet, J. P., Lafarge, F., 2019. City reconstruction from airborne LiDAR: A computational geometry approach. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-4/W8, 19–26.
- Blomley, R., & Weinmann, M., 2017. Using multi-scale features for the 3D semantic labeling of airborne laser scanning data.

- ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., 4(2W4), 43–50.
- Boykov, Y., Veksler, O., & Zabih, R., 2001. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11), 1222–1239.
- Boykov, Y., Veksler, O., & Zabih, R., 2011. Optimizing Multilabel MRFs Using Move-Making Algorithms. In A. Blake, P. Kohli, & C. Rother (Eds.), *Markov Random Fields for Vision and Image Processing*, (51–64). The MIT Press, Cambridge.
- Carlberg, M., Gao, P., Chen, G., & Zakhor, A., 2009. Classifying urban landscape in aerial lidar using 3d shape analysis. *16th IEEE International Conference on Image Processing (ICIP), 2009*, 1701–1704.
- Cramer, M., 2010. The DGPF-test on digital airborne camera evaluation – overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation 2010*, 73–82.
- Dabboor, M., Howell, S., Shokr, M., & Yackel, J., 2014. The Jeffries-Matusita distance for the case of complex Wishart distribution as a separability criterion for fully polarimetric SAR data. *Int. J. Remote Sens.*, 35(19), 6859–6873.
- Du, S., Zhang, Y., Zou, Z., Xu, S., He, X., & Chen, S., 2017. Automatic building extraction from LiDAR data fusion of point and grid-based features. *ISPRS J. Photogramm. Remote Sens.*, 130, 294–307.
- Guan, H., Li, J., Zhong, L., Yongtao, Y., & Chapman, M., 2013. Process virtualization of large-scale lidar data in a cloud computing environment. *Comput. Geosci.*, 60, 109–116.
- Guo, Y., Bennamoun, M., Sohel, F., Lu, M., & Wan, J., 2014. 3D object recognition in cluttered scenes with local surface features: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(11), 2270–2287.
- Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., Kwok, N.M., 2016. A Comprehensive Performance Evaluation of 3D Local Feature Descriptors. *Int. J. Comput. Vis.*, 116, 66–89.
- Jeffreys, H., 1946. An Invariant Form for the Prior Probability in Estimation Problems. *Proc. of the Royal Society of London. Series A, Mathematical and Physical*, 186(1007), 453–461.
- Kim, K. & Shan, J., 2011. Building roof modeling from airborne laser scanning data based on level set approach. *ISPRS J. of Photogramm. Remote Sens.*, 66(4), 484–497.
- Lafarge, F., & Mallet, C., 2012. Creating large-scale city models from 3D-point clouds: A robust approach with hybrid representation. *Int. J. Comput. Vis.*, 99(1), 69–85.
- Landrieu, L., Raguét, H., Vallet, B., Mallet, C., & Weinmann, M., 2017. A structured regularization framework for spatially smoothing semantic labelings of 3D point clouds. *ISPRS J. of Photogramm. Remote Sens.*, 132, 102–118.
- Lang, D., Friedmann, S., & Paulus, D., 2016. Adaptivity of conditional random field based outdoor point cloud classification. *Pattern Recogn. and Image Anal.*, 26(2), 309–315.
- Lim, E.H. & Suter, D., 2007. Conditional Random Field for 3D Point Clouds with Adaptive Data Reduction. *2007 International Conference on Cyberworlds (CW'07)*, 404–408.
- Martínez, J., Rivera, F. F., Cabaleiro, J. C., Vilariño, D. L., Pena, T. F., & Miranda B., D., 2016. A rule-based classification from a region-growing segmentation of airborne lidar. In *Proc. of SPIE Image and Signal Processing for Remote Sensing XXII*, 10004, 100040F.
- Meyer, F., 1994. Topographic distance and watershed lines. *Signal Processing*, 38(1), 113–125.
- Niemeyer, J., Rottensteiner, F., & Soergel, U., 2012. Conditional random fields for LiDAR point cloud classification in complex urban areas. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, I-3, 263–268.
- Niemeyer, J., Rottensteiner, F., & Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS J. of Photogramm. Remote Sens.*, 87, 152–165.
- Polewski, P., Yao, W., Heurich, M., Krzystek, P., & Stilla, U., 2014. Detection of fallen trees in ALS point clouds by learning the Normalized Cut similarity function from simulated samples. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-3, 111–118.
- Richards, J. A., & Jia, X., 2006. *Remote Sensing Digital Image Analysis*. Springer-Verlag, Berlin Heidelberg.
- Rusu, R. B., Marton, Z. C., Blodow, N., & Beetz, M., 2008. Learning informative point classes for the acquisition of object model maps. *2008 10th International Conference on Control, Automation, Robotics and Vision*, 643–650.
- Rusu, R. B., Blodow, N., & Beetz, M., 2009. Fast Point Feature Histograms (FPFH) for 3D registration. *2009 IEEE International Conference on Robotics and Automation*, 3212–3217.
- Sun, S., & Salvaggio, C., 2013. Aerial 3D building detection and modeling from airborne LiDAR point clouds. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 6(3), 1440–1449.
- Ural, S., and Shan, J., 2012. Min-cut based segmentation of airborne lidar point clouds. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XXXIX-B3, 167–172.
- Ural, S., & Shan, J., 2016. A min-cut based filter for airborne Lidar data. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLI-B3, 395–401.
- Veksler, O., 1999. *Efficient graph-based energy minimization methods in computer vision*. Cornell University.
- Vilariño, D. L., Cabaleiro, J. C., Martínez, J., Rivera, F. F., & Pena, T. F., 2017. Graph-based approach for airborne light detection and ranging segmentation. *J. Appl. Remote Sens.*, 11(1), 15020–1–13.
- Weinmann, M., Urban, S., Hinz, S., Jutzi, B., & Mallet, C., 2015a. Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas. *Computers and Graphics (Pergamon)*, 49, 47–57.
- Weinmann, M., Schmidt, A., Mallet, C., Hinz, S., Rottensteiner, F., & Jutzi, B., 2015b. Contextual Classification of Point Cloud Data By Exploiting Individual 3D Neighbourhoods. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, II-3/W4, 271–278.
- Weinmann, M., Hinz, S., & Weinmann, M., 2017. A Hybrid Semantic Point Cloud Classification-Segmentation Framework Based on Geometric Features and Semantic Rules. *PFG – J. of Photogramm., Remote Sens. Geoinf. Sci.*
- West, K. F., Webb, B. N., Lersch, J. R., Pothier, S., Triscari, J. M., & Iverson, A. E., 2004. Context-driven automated target detection in 3D data. *Proc. SPIE Int. Soc. Opt. Eng.*, 5426, 133–143.
- Xu, Y., U. Stilla, , 2019. Contour extraction of planar elements of building facades from point clouds using global graph-based clustering. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-2/W7.
- Yang, J., Cao, Z., & Zhang, Q., 2016. A fast and robust local descriptor for 3D point cloud registration. *Information Sciences*, 346–347, 163–179.
- Zhu, Q., Li, Y., Hu, H., & Wu, B., 2017. Robust point cloud classification based on multi-level semantic relationships for urban scenes. *ISPRS J. of Photogramm. Remote Sens.*, 129, 86–102.