Interreg Alpine Space project - **NEWFOR**

Project number 2-3-2-FR

NEW technologies for a better mountain **FOR**est timber mobilization

Priority axis 2 - Accessibility and Connectivity

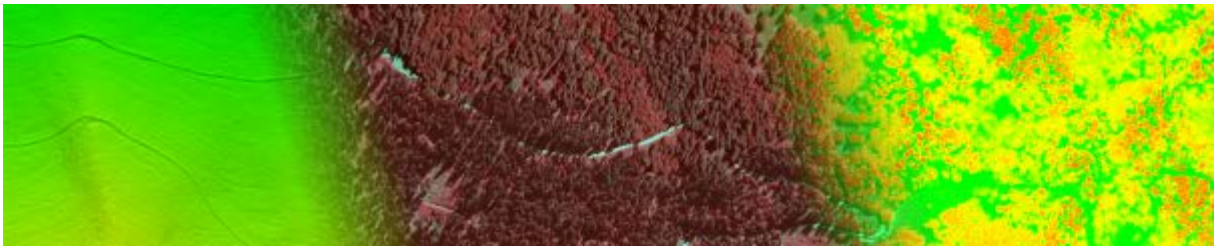
Workpackage 5: Forest accessibility

Airborne Laser Scanning for Forest Road Detection

State-of-the-Art

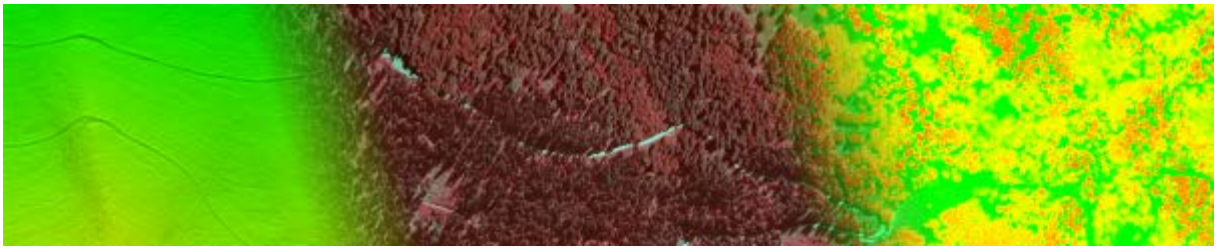
Coordinator: Markus Hollaus (Vienna University of Technology – AUSTRIA, Department of Geodesy and Geoinformation, Research Group for Photogrammetry (E120-7))
Contributors: Sajid Ghuffar, Markus Hollaus, Norbert Pfeifer (all Vienna University of Technology – AUSTRIA, Department of Geodesy and Geoinformation, Research Group for Photogrammetry (E120-7))

Final version 30/November/2014

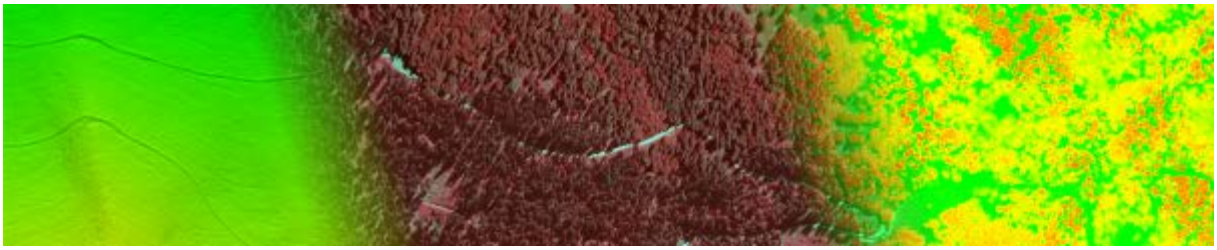


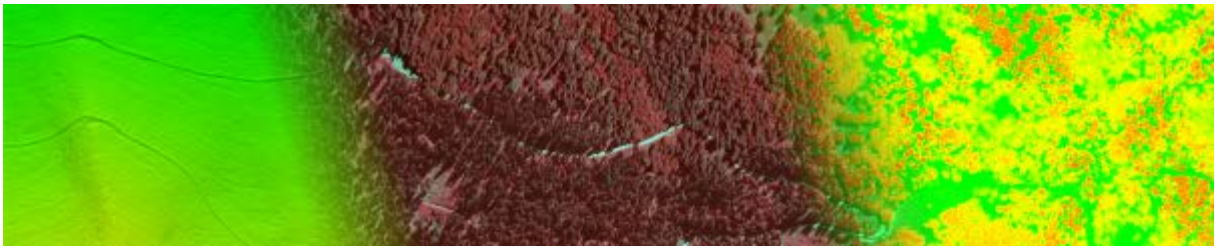
The consortium of the project Interreg Alpine Space NEWFOR

This project has been, co-funded by the European Regional Development Funds, and achieved under the third call of the European Territorial Cooperation Alpine Space Programme 2007-2013.



1 CONTENTS

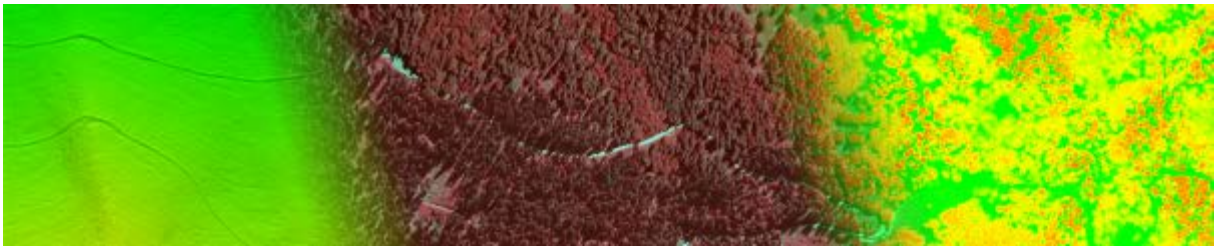




2 ABSTRACT

In this SoA report different techniques for road extraction from Airborne Laserscanning data (ALS) are summarized and discussed. Furthermore, several examples of road extraction from point clouds as well as images are presented.

The state of the art of these techniques shows that airborne laser scanning has a great potential in extracting forest roads, however there is a lack of robust and automatic methods in achieving this goal. The planarity of the road surface and the fact that roads are on ground level have been exploited by authors for forest roads detection. The fusion of intensity data along with plane fitting residuals have so far given the best results in forest road extraction. However intensity of ALS signal is strongly dependent on the properties of the surface and the flight mission. Generally it can be stated that there is a need for more robust measures to extract roads in varying terrain types.



3 INTRODUCTION

Forest roads are essential for forest management and monitoring and provides vital passage for forest products, infrastructure and fire protection. Therefore it is imperative to develop tools which help to automatically detect forest roads. Airborne laser scanning (ALS) has a great potential for forest road detection due to its ability to penetrate into dense vegetation. Unlike ALS, aerial and space borne imagery cannot see under dense canopy, as a result do not provide satisfactory results in such areas. On the other hand terrestrial laser scanning (TLS) and in situ measurement using tacheometers [1] can be expensive especially for large scale forest road detection.

The process of road extraction can be seen as a segmentation process which groups the data into different categories based on some decision rules. As mentioned previously ALS is the most suitable data source for detection of forest roads. Therefore in context of forest road extraction it is essential to study the segmentation methods for point cloud data which can then be utilized for forest road detection.

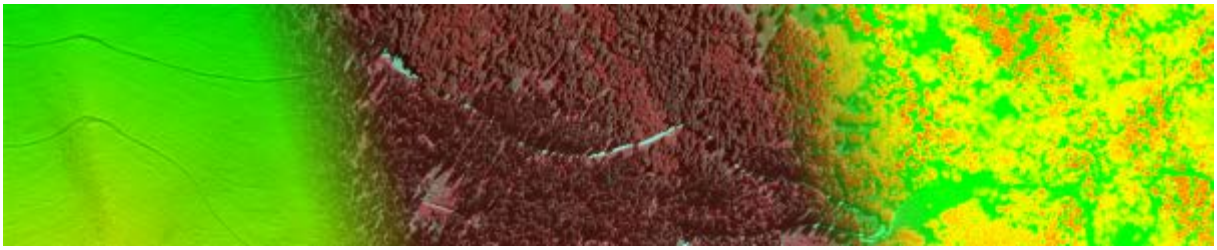
In section 2 we present a survey of segmentation methods which can potentially be utilized for extraction of forest roads from point cloud data. We do not restrict ourselves to methods and techniques which are presented in literature with reference and applications in forestry. This enables to cover a variety of segmentation techniques, most of which were originally not discussed in relation to forestry but in future might help in developing tools for forest monitoring. This may also help the reader in understanding the advantages of various segmentation techniques in a broader perspective and facilitate as a reference for further development of segmentation methods for forestry, especially the forest road extraction. In section 3, we present methods for road extraction using ALS data with focus on forest road extraction. Again the aim is to present a broader view on the point features used and the methods applied. Furthermore in section 4 we present a brief overview of image based road extraction methods to develop a correlation between the point cloud based and image based techniques for road extraction.

4 SEGMENTATION

Segmentation is the process of partitioning certain identities into segments, where elements of each segment share closely related characteristics. The formal definition of segmentation as given in Hoover et al. [2] is as follows:

Let P represent the entire point cloud. The segmentation is a process of partitioning P in to sub groups P_i such that

1. $\bigcup_i^n P_i = P$
2. $P_i \cap P_j = \emptyset$ (empty set) for all i and j, $i \neq j$
3. P_i is a connected in space



4. Predicate P_i = TRUE for i=1,2,..,n and
5. Predicate $P_i \cup P_j$ = FALSE for P_i neighboring P_j

In literature there exists a variety of point cloud segmentation algorithms with focus on different applications like segmentation of indoor [3] and industrial scenes [4-6], tree detection [7-9], shape extraction [10], segmentation of urban areas [11]. Most of these segmentation methods can be categorized into the following four types:

1. Clustering
2. Parametric Shapes
3. Region Growing
4. Graph Cut (Graph partitioning)

In following sections a brief description of these segmentation methods is presented along with some examples of application of these methods for point cloud segmentation.

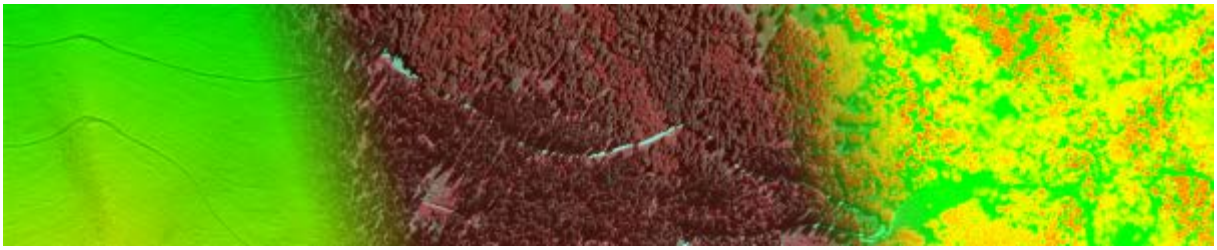
4.1 CLUSTERING

Clustering is the process of sub dividing data into similar clusters or groups where entities of each cluster shares similar properties. Clustering has been widely applied in data mining, machine learning and image analysis community. Conceptually clustering only involves grouping data into clusters based on some features. Therefore clustering may not satisfy spatial connectivity condition as mentioned above in the definition of segmentation. In context of point cloud segmentation, if position of the point is included in the feature space, clustering may result in segmentation. However if the position or the connectivity of points is not included in the clustering procedure then a point neighborhood condition is subsequently applied to obtain segmentation. A review of clustering algorithms can be found in [12].

The features commonly employed for point cloud clustering are based on the geometrical properties of points like position and normal vector and the radiometric properties like reflectance. A variety of clustering algorithms have been applied to point cloud data e.g. k-mean clustering [13], mean shift clustering [14], hierarchical clustering [15] and mode-seeking [16]. In the next section, some of these examples are discussed in detail in context of point cloud segmentation.

4.1.1 K-MEAN CLUSTERING

K-mean algorithm is a widely used data clustering algorithm. It divides data points in feature space into “K” clusters by minimizing the mean square distance between the data point and its center [13, 17, 18]. The initial cluster centers are often randomly selected and data points are assigned to clusters based on their distances to the cluster centers. The centroid of each cluster



is then recomputed using the points assigned to the cluster in the first step. The distance from each cluster center to data points is then recomputed and data points are assigned to a cluster whose center is the nearest. This process is applied iteratively until the cluster centers don't change anymore.

One disadvantage of k mean algorithm is that the number of clusters has to be specified in advance. Additionally the grouping into clusters depends on the initially selected cluster centers and the algorithm is sensitive to outliers.

4.1.2 "CLUSTERING IN AIRBORNE LASER SCANNING RAW DATA FOR SEGMENTATION OF SINGLE TREES"

Morsdorf et al. [9] perform k-mean clustering on raw point cloud data to obtain single trees. K mean clustering result is sensitive to the initial seed points which may result in unintended result. To obtain seed points, an nDSM based threshold value is applied to find seed points for individual trees. Additionally the pine trees crown size is quite smaller compared to the height distribution of points. Therefore z axis was scaled with an empirical derived factor. Individual trees segmented using region growing and k-mean clustering are shown in Fig. 1. It is mentioned that in locations where the trees are very close to each other the approach of local maxima in DSM underestimates the number of trees, as a result a single cluster was assigned to trees present very close to each other.

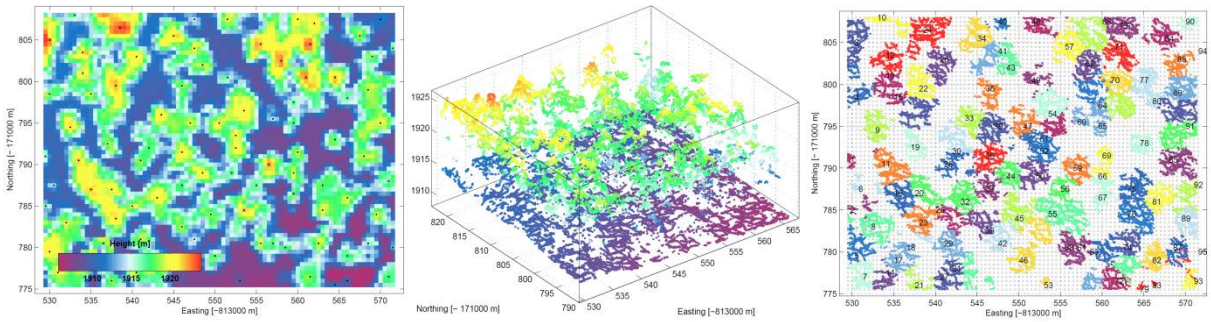
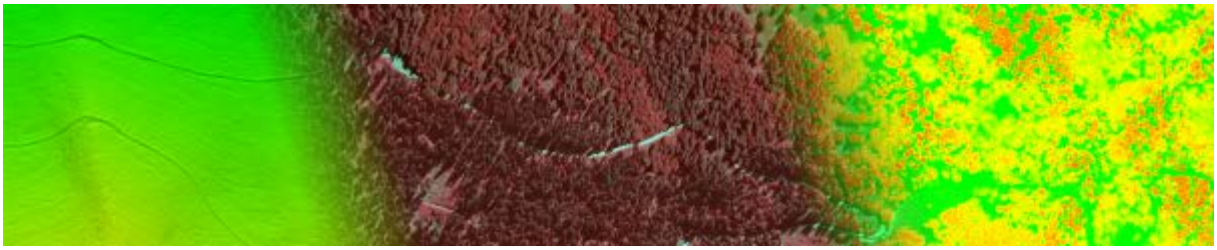


Fig. 1 Left: Seed points computed automatically from local maxima of nDSM, Middle: Side view of raw 3D data, Right: Segmented trees



4.1.3 MEAN SHIFT ALGORITHM

Mean shift algorithm, originally proposed by Fukunaga and Hostetler [14], is a non-parametric method, which can be used for clustering of data [14, 19, 20] . Unlike k-mean clustering mean shift algorithm doesn't require any prior knowledge about the number of clusters. In mean shift algorithm each point in feature space is considered as a sample of the underlying probability density function, which can be estimated using data points in a neighborhood. This probability density function describes the probability of distribution for each cluster in the feature space. Each data point in the feature space is then shifted via gradient ascent of the estimated probability density kernel. This way each point is iteratively shifted towards the maxima of the local probability density kernel. The only free parameter in the mean shift algorithm is the kernel width also called bandwidth of the kernel density function. The kernel width of the probability density kernel is important parameter for determining correct number of clusters. A high value of kernel width might result in merging of individual clusters (over-smoothing), while a small value might cause too many clusters. Melzer [11] write that there exists theoretical results to find optimal value of kernel width but only for univariate case.

4.1.4 "NON-PARAMETRIC SEGMENTATION OF ALS POINT CLOUDS USING MEAN SHIFT"

Melzer [11] use mean shift procedure for segmentation of city areas. Mean shift algorithm is applied on an ALS dataset over an area of Vienna, Austria. Results from using two different set of features points have been presented. In the first case only the 3D location of the points have been used for segmentation, while in the second case, amplitude and the pulse width of the last echo have been used in addition to the 3D location of the point. Results shows that using additional attributes of amplitude and echo width provide finer segmentation, particularly more individual buildings are characterized as a separate segment. The streets, vegetation and the buildings are segmented correctly. The only problematic zones are the courtyards where the vegetation and buildings are too close to each other. The results of the segmentation are shown in Fig. 2.

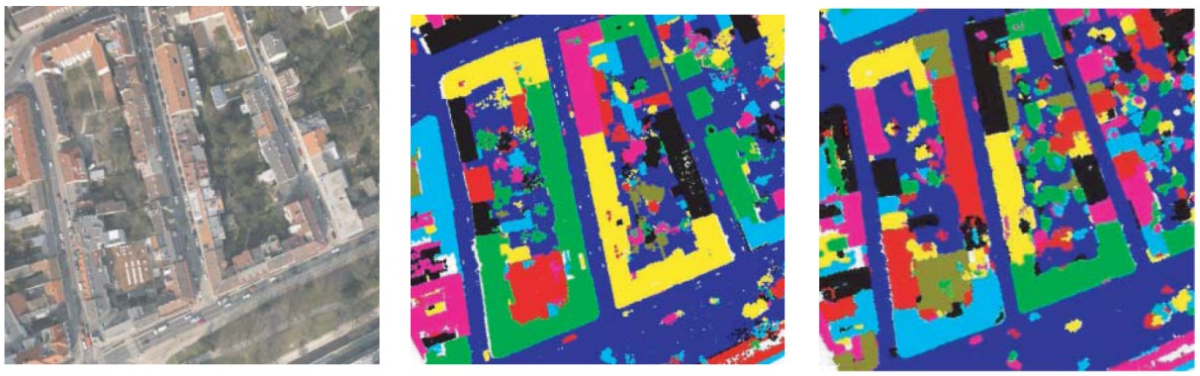
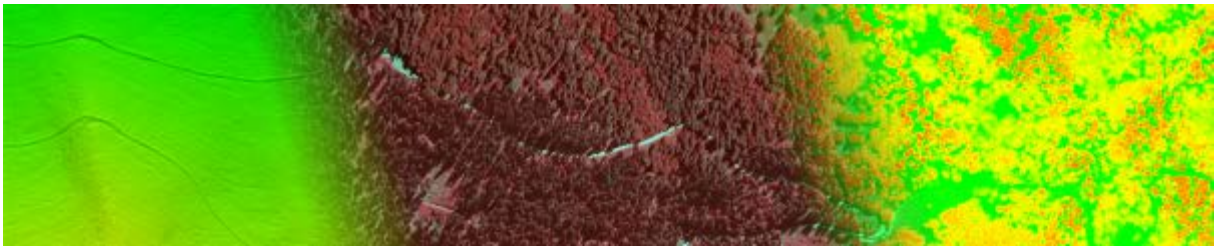


Fig. 2: Left: Aerial photograph of test area, Middle: Mean shift using only 3D point location, Right: Mean shift using additional attributes of amplitude and echo width



In addition to the segmentation of the urban areas, Melzer [11] present a method of extracting power lines: vegetation filtering using the mean shift procedure. If a DTM is available, terrain points can easily be removed using a height threshold. Then mean shift procedure is applied to the 3D point which produces separate clusters for vegetation and the points on the power lines. The elongation ratio the clusters, turns out to be good measure in differentiating between the vegetation clusters and power lines clusters. The results of the clustering are shown in the Fig. 3.

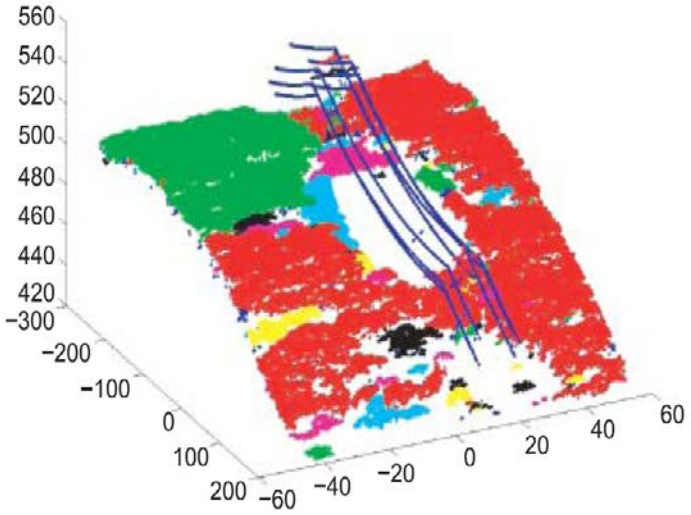


Fig. 3: Results of mean shift clustering for power lines extraction

4.1.5 "SEGMENTATION OF AIRBORNE LASER SCANNING DATA USING A SLOPE ADAPTIVE NEIGHBORHOOD"

Filin and Pfeifer [16] present a segmentation method which involves clustering using mode-seeking approach [21]. They define a seven dimensional feature vector for each point. These features include position, parameters of a plane fitted to the neighborhood of a point and the relative height difference between the point and its neighbors. Instead of creating 7 dimensional feature space the authors separate positional information to create 4 dimensional feature space. This feature space is clustered using a mode-seeking algorithm [21] to identify the surface classes. After extracting surface classes, the points are grouped in object space utilizing spatial proximity measure. The Fig. 5 shows distinct clusters for saddle-back roofs and dormer. The points belonging to these individual clusters are then segmented using spatial proximity measure.

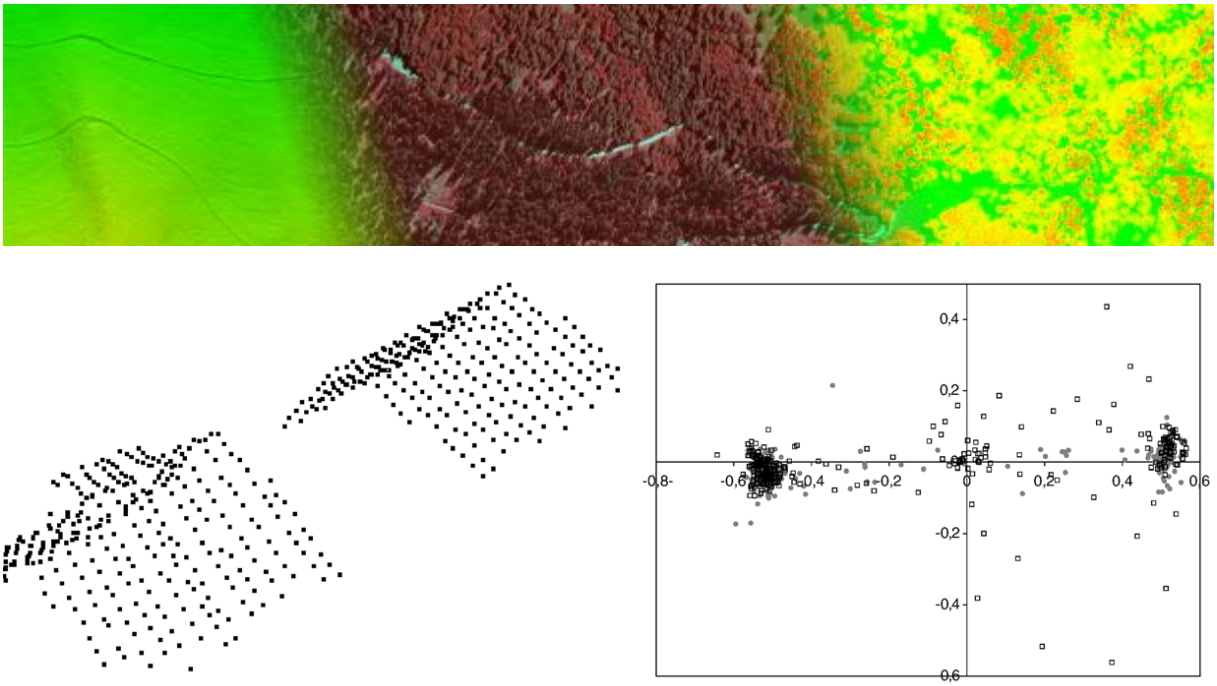
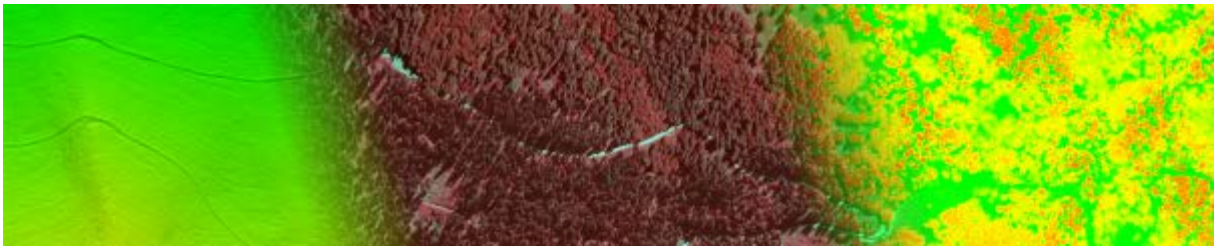


Fig. 4 Left: Point cloud of two saddle-back roofs and one dormer, Right: x and y component of normal vector for each point in feature space

4.2 PARAMETRIC SHAPE EXTRACTION

Extraction of parametric shapes like plane, cylinders and spheres from point cloud data has been a widely addressed topic. Generally a prior knowledge of expected shapes in the data is available. Random Consensus Sampling (RANSAC) [14] and Hough transform [22] are the most commonly used methods for parametric shape extraction. Both these methods are robust to noise and can be used for both 2D and 3D data [10].



4.2.1 HOUGH TRANSFORM

Hough transform is a method of finding shapes, lines and curves in a parameters space. It has been widely used in 2D images to find lines, circles and ellipses. 2D Hough transform has been extended to 3D for detecting cylinders [23], building modeling [24-26], roof detection [27], segmentation of industrial scenes [4].

A line in 2D can be represented by two parameters m (slope of the line) and c (the y intercept). Using this parameterization one can write the equation of line as:

$y = mx + c$

Therefore each line in xy plane can be represented as a point in (m,c) space (also known as the parameter space or Hough space). Therefore for each given point in object space we can assign m values between [0,π] and subsequently find the corresponding c values. These (m,c) pairs form a line in Hough space. By applying this procedure to data points we will have a set of lines in (m,c) space. These lines will intersect at a certain point in (m,c) space which represents the line present in the 2D object space.

Similar to the 2D case Hough transform can be extended to detect 3D objects in point clouds. A plane in 3D space is represented as

$$ax + by + cz + d = 0$$

And we apply a similar procedure as described before to compute the parameters a,b,c in the parameter space.

4.2.2 "RECOGNIZING STRUCTURE IN LASER SCANNER POINT CLOUDS"

Vosselman et al. [4] discuss segmentation of point cloud using smoothness, iterative extraction of planar surfaces and parameterized shapes using Hough transform. The 3D Hough transform using normal vectors where Cylinders and spheres are detected in point clouds. This technique is then applied to for automatic segmentation of industrial scene, city landscapes, digital elevation models and trees.

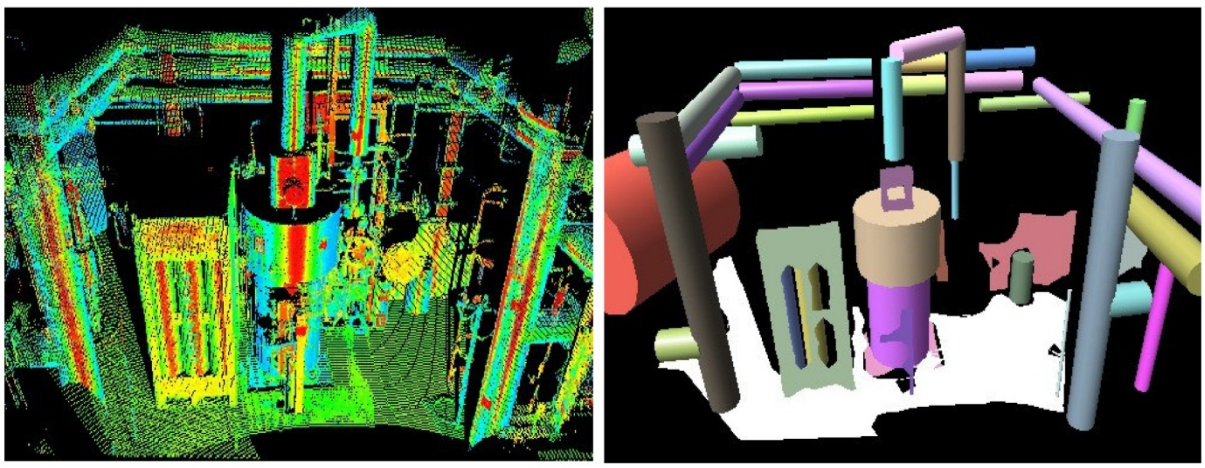
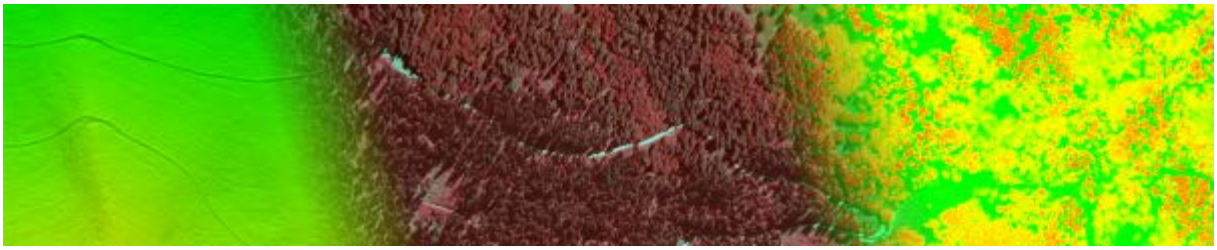


Fig. 5 Industrial scene, Left: Color coded points according to their surface normal direction, Right: Segmentation result

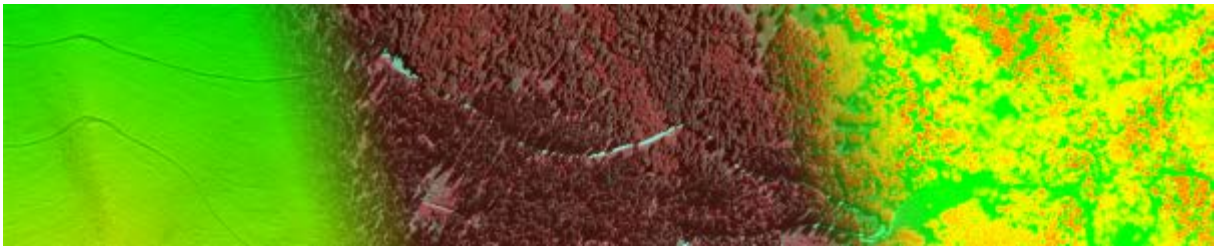


4.2.3 RANSAC

RANdom SAmple Consensus (RANSAC) paradigm is used for model fitting of data. RANSAC has been used for finding primitive shapes in point cloud data [3, 10, 28], road parameters extraction [29], point cloud registration [30], building façade extraction [31], roof plane detection [27]. RANSAC can handle large amount of outliers [32]. RANSAC takes minimum number of points possible to initially estimate the model parameters and then enlarges this set by including points which are consistent with the model parameters. For example in order to fit a plane to a collection of points, RANSAC will take 3 points and estimate the parameters of the plane and calculate the number of points that are compatible to the fitted plane. The formal definition of the RANSAC paradigm as stated in [32] is as follows

”Given a model that requires a minimum of n data points to instantiate its free parameters, and a set of data points P such that the number of points in P is greater than n [#(P)≥n], randomly select a subset SI of n data points from P and instantiate the model. Use the instantiated model M1 to determine the subset SI* of points in P that are within some error tolerance of Ml. The set SI* is called the consensus set of S1.

If #(SI*) is greater than some threshold t, which is a function of the estimate of the number of gross errors in P, use SI* to compute (possibly using least squares) a new model MI *.If #(SI*) is less than t, randomly select a new subset $2 and repeat the above process. If, after some predetermined number of trials, no consensus set with t or more members has been found, either solve the model with the largest consensus set found, or terminate in failure”.



4.2.4 "EFFICIENT RANSAC FOR POINT CLOUD SHAPE DETECTION"

When the number of primitive shapes classes is larger and the size of the point cloud is large as well, the computational cost of the shape extraction using RANSAC can become prohibitive. Schnabel et al. [10] propose a sampling strategy to increase the performance of the RANSAC. They extract planes, spheres, cylinders, cones and tori in the point cloud, which require between three to seven parameters. Schnabel et al. [10], use point normals along with the 3D point to reduce the minimal number of points required for estimating shape parameters. However they remark that using an additional point helps to eliminate shapes with low scores.

They estimate a plane using 3 points and then check if the difference between the normal of the plane and the normal vectors of the three points are within a threshold. A sphere is estimated using two points and their normal vectors. The centre of the sphere is the computed using the mid point of the shortest line segment between the two lines consisting of points and their normal vectors and the radius is the estimated as the mean of the distance between the points and the center. Similarly the parameters for the cylinders, cone and tori are estimated using the information from the 3D points and their normal vectors

The computational cost of the RANSAC mainly depends on two factors: the number of minimal sets drawn and the score evaluation for each shape. The sampling of the points is directly linked to the number of minimal sets which are drawn, therefore they introduce a sampling strategy based on octree to increase the performance of the algorithm. When choosing the candidate points the first sample is drawn without any restriction but the next sample points are chosen from a randomly drawn level of octree which contains the first points. This exploits the fact that the points close to each other have high probability that they belong to the same shape. The computational cost for the score evaluation is reduced by dividing the point cloud into disjoint random subsets and evaluating the score functions for the subsets of points.

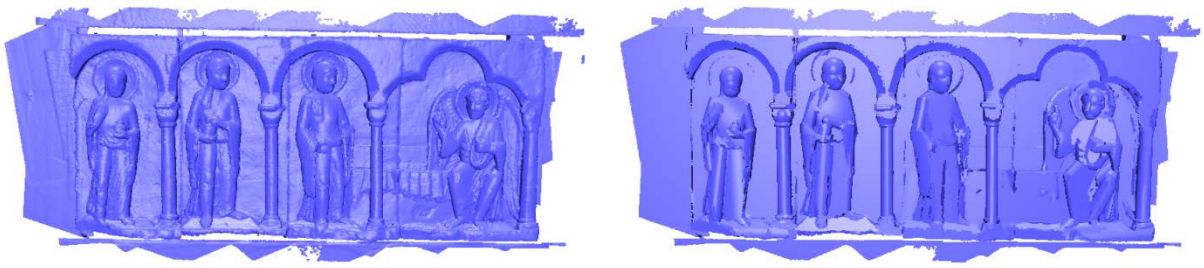
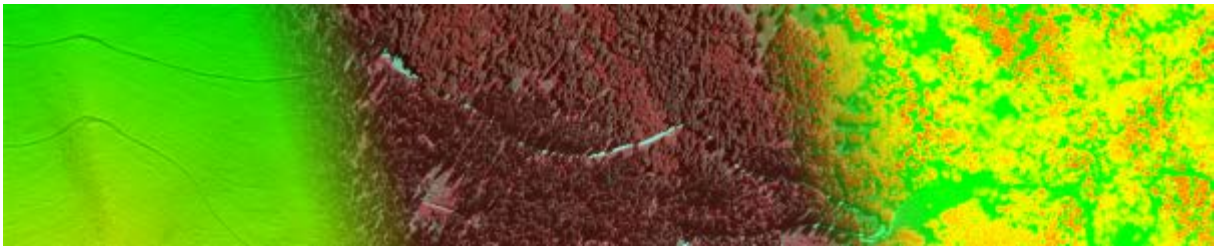


Fig. 6: Left: Original Right: Approximation of the surface from 372 detected shapes



4.3 REGION GROWING

Region growing is a bottom-up approach for segmentation, which starts by selecting a seed point and sequentially grows the segment following certain homogeneity criterion. If a neighboring pixel is consistent with the homogeneity criterion then it is added to the segment. When a segment stops growing a new seeds point is selected and then same region growing procedure is applied to each seed point. Region growing has been extensively used in segmentation of 2D images. A comparative study related to region growing based segmentation algorithms is presented in [33].

The two important factors in the region growing algorithms are the selection of the seeds and the definition of homogeneity criterion. In the context of the point cloud segmentation, the homogeneity criterion is commonly based on geometric properties like planarity, curvature and surface normals. Seed points are quite often selected randomly [34, 35], but it can also be chosen based on other criterion e.g. low residuals of plane fitting [5, 36], local maxima of DSM [9].

4.3.1 "SEGMENTATION BASED ROBUST INTERPOLATION- A NEW APPROACH TO LASER DATA FILTERING"

Tovari and Pfeifer [35] present a segmentation based interpolation method by classification of terrain and object points. In the first step a region growing based segmentation is performed and then robust filtering is applied to the segments. This method combines the strengths of point based and segment based filtering approaches. In the preprocessing step normal vectors are estimated for each point. Then the region growing algorithm randomly picks a seed point and checks the n nearest neighbors for the following criterion after estimating the plane for the current segment.

- Similarity of normal vectors
- Distance of candidate point to the adjusting plane
- Distance between current point and candidate point

Points are added to the segment if they are within some pre-defined threshold for these criterions. The growing of the segment continues until no more points fulfill the above mentioned criterion. The ground, roofs and walls are correctly segmented. Also objects e.g. chimneys, power lines, vegetation and vehicles result in different segments. Additionally the different parts of the terrain have been split into different segments based on the respective break lines as shown in Fig. 6.

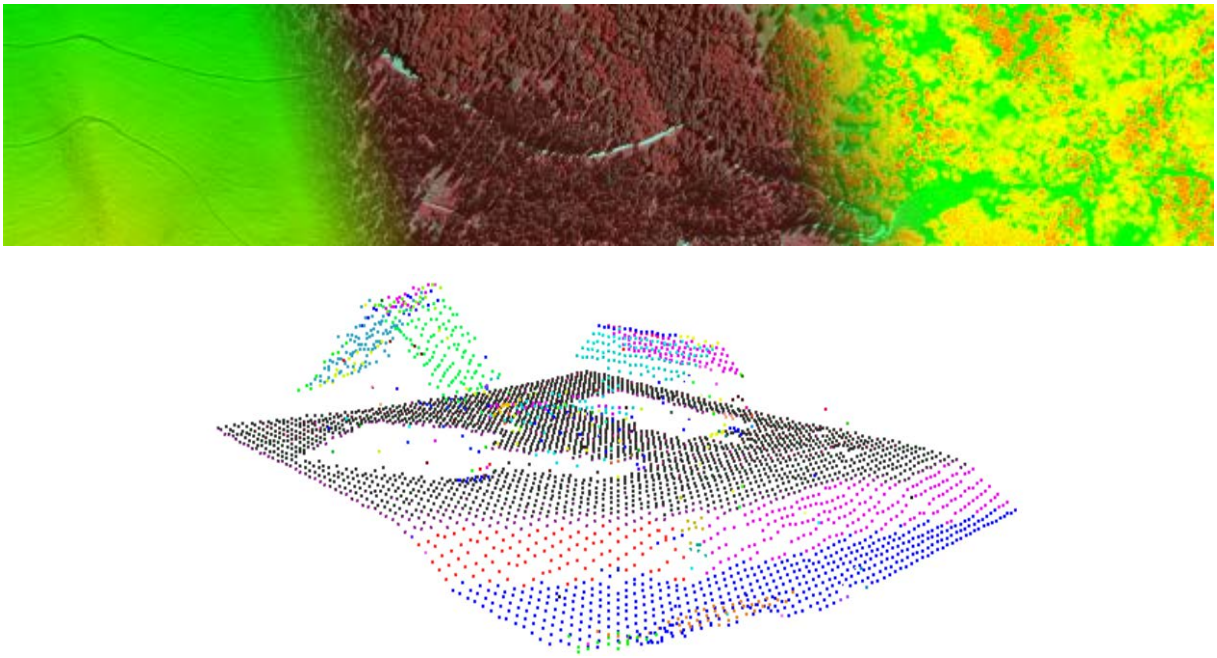


Fig. 7: Segmentation result of the region growing algorithm

The segments from the region growing method are then used for robust interpolation of the terrain. Initially all the segmentation are assigned equal weight. Then the ground surface is iteratively determined and the weights for each segment are recomputed. A moving least squares method using a first order polynomial is used for surface computation. Weights for the next iteration are computed based on the distance from the observed point to the interpolated surface, which is positive if the point is above the surface and negative when the point is below the surface. The average value of this distance for object segments would be high positive value. Thus iteratively segments are classified as objects or terrain and for the object segments the weight is set to zero for moving least squares surface interpolation.

4.3.2 "SEGMENTATION OF POINT CLOUDS USING SMOOTHNESS CONSTRAINT"

Rabbani et al. [5] present an automatic point cloud segmentation method where the local smoothness value is used for segmentation. The smoothness value is expressed as residual computed from the normal estimation for each point using either a fixed distance or k-nearest neighbors. It has been shown that the residuals of the plane fitting are indicator of curvature of cylinders (cylinders of different radii). A bottom up approach for segmentation using region growing is employed.

Two parameters a) residuals of normal estimation and the angle between the normal vectors are used to provide a tradeoff between under and over segmentation. The algorithm is tested on point clouds from industrial scenes which mainly contain planar and curved surfaces of different radii. The proposed algorithm performs well in segmenting objects of different curvature.

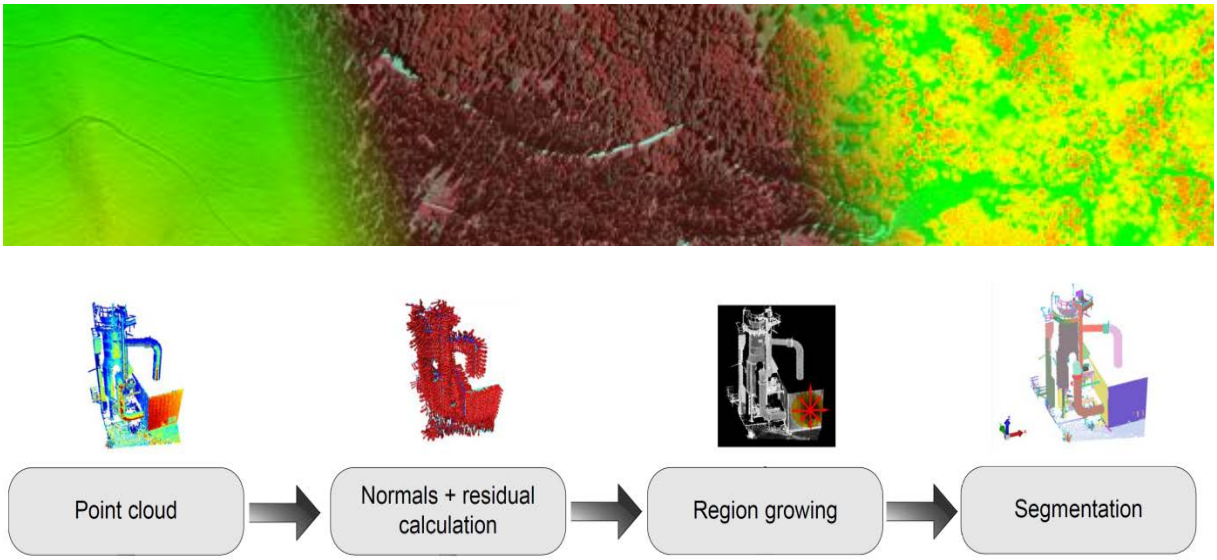


Fig. 8 Flow chart of the segmentation algorithm

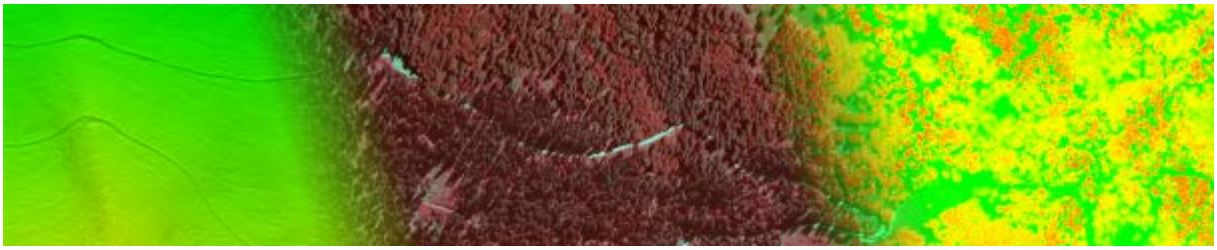
4.3.3 "OBJECT SEGMENTATION WITH REGION GROWING AND PRINCIPAL COMPONENT ANALYSIS"

Roggero [34] propose a segmentation technique which combines region growing with principal component analysis (PCA). The distribution of the points in the 3D space for an object is representative of its geometrical properties. PCA finds dominant patterns in the observations by representing the observations in terms of orthogonal vectors called principal components, where the first principal component represents the direction of the largest variance in data and the successive principal components represents descending variance along the orthogonal directions.

Segmentation is based on region growing procedure where a seed point is selected arbitrarily and new points are added to the segment if they fulfill certain criterion. These criterion involve geometric descriptors of the points which were computed with the help of PCA. The geometric descriptors mentioned are static moment, curvature and junction of surfaces. In the first step geometrical descriptors are computed for each point and then using the region growing algorithm points were aggregated into the segments based on the distance from certain descriptor.

4.3.4 "CLASSIFICATION AND SEGMENTATION OF TERRESTRIAL LASER SCANNER POINT CLOUDS USING LOCAL VARIANCE INFORMATION"

Belton and Lichti [37] present a segmentation method based on the covariance analysis of a local neighborhood of points similar to Roggero's [34] approach mentioned before. The eigenvectors of the covariance matrix correspond to the principal components of the point neighborhood and the eigenvalues correspond to the variance in the corresponding direction. The amount of variance in the normal direction is an indication if the points belong to a surface. However the difference in the density and the distribution of the points from laser scanning makes it difficult to find appropriate threshold for variance. The ratio of the variance in the normal direction to the sum of the total variance in the neighborhood is often used to solve this problem. This quantity represents the curvature in the neighborhood, which can be used to determine if the point belongs to a planar surface. However this method is sensitive to highly curved surfaces



which may lead to misclassification. Therefore Belton and Lichti classify points based on the variance of the curvature in a neighborhood. This will classify objects with similar curvature into one class. The boundary points are detected by using the distance between the centroid of the neighborhood and the location of the subject point. This classification of the surface and boundary points is then transformed into a segmentation using a region growing algorithm. Additionally the intersection points between two surfaces are detected by computing the intersection points of two fitted planes on the surface edges.

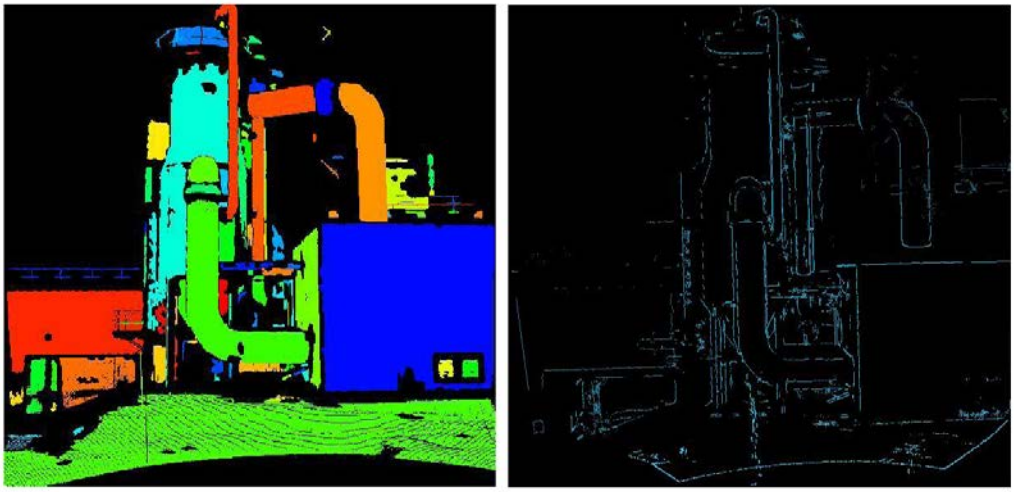
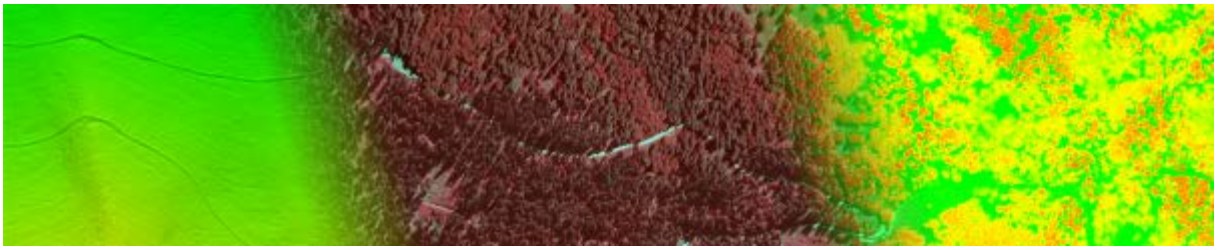


Fig. 9 Left: Segmentation result on a scene from industrial plant, Right: Detected intersection points

4.4 GRAPH PARTITIONING

The concept of Graph cuts for image segmentation was introduced in [38], which has been extended to Normalized Graph Cuts in [39]. The idea of graph based segmentation has also been applied to point cloud segmentation in [40-42] . The pixels in the image or the points in a point cloud can be represented as nodes of a graph and the edges of the graph can represent a similarity measure between the nodes of the graph based on geometric or radiometric primitives. A large similarity between the nodes means large weights. A graph (V,E), where V denotes the nodes of the graph and E denotes the edges (e.g. based on 4-neighbourhood in images), can be partitioned into disjoint A,B by removing the edges joining the two parts. A graph cut is the total weight of the edges of the graph that have been removed for partitioning [39].

$$cut(A,B) = \sum_{u \in A, v \in B} w(u,v)$$



The optimal partitioning of the graph minimizes this cut value. Graph cuts is a top down approach which divides the input graph into two segments. For further partitioning, graph cuts can then be applied iteratively to further subdivide each of these segments. The graph cut partitioning encourages separation of small set of nodes [39]. A normalized graph cuts algorithm obviates this behavior by also taking in consideration the total number of the edge connection in the graph [39].

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$

where $assoc(A,V) = \sum_{u \in A, t \in V} w(u,t)$ is the weight of connections from A to all other nodes in the graph and $assoc(B,V) = \sum_{u \in A, t \in V} w(u,t)$ is the weight of connections from B to all other nodes in the graph. The denominators value discourages the partitioning with only few nodes [43].

4.4.1 "MIN-CUT BASED SEGMENTATION OF POINT CLOUDS"

Golovinskiy and Funkhouser [42] present a graph min cut based method to segment a point cloud into a foreground and background object. Given an object location, this method finds points which belong to this object and the rest of the points are treated as background. First a ground plane is estimated by iterative plane fitting and the points close to the ground are removed. Then k-nearest neighbor graph is constructed where the edges of the graph has weights which decrease with distance. The algorithm requires two inputs: a 2D location and a background radius (a horizontal distance where it is assumed that background begins). Furthermore two methods of computation are presented. In the first method the background radius is selected automatically within a range and in the second method a user defines hard foreground and background constraints which are taken into account by min-cut algorithm.
The results of min-cut based segmentation are compared with ground truth segmentation, which is created using an interactive segmentation tool. The quantitative analysis of the results are given using the ratio of correct predicted foreground points to the total number of predicted foreground points and the ratio of foreground points which are correctly predicted to the number of foreground points in the ground truth.

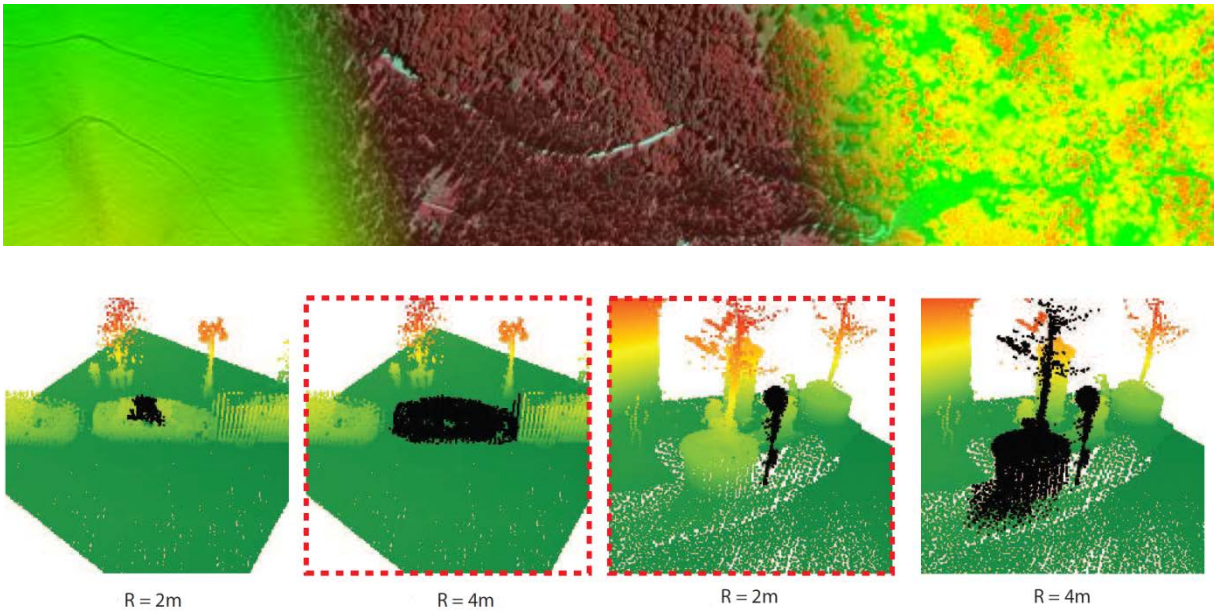


Fig. 10 Effect of different background radius: Depending on the object the background radius needs to be adopted

4.4.2 "SEGMENTATION OF FULL WAVEFORM LIDAR DATA FOR SINGLE TREE DETECTION USING NORMALIZED CUT"

Reitberger et al. [7] use normalized graph cut based segmentation for detection of single trees. The region of interest is sub-divided into voxels and these voxels are represented as a graph. The similarity between the voxels is represented by the weights between the nodes of the graph. These weights are computed based on the Euclidean distance between the voxels and the full waveform feature similarity. The experimental and reference dataset is from Bavarian Forest National Park. The full waveform ALS data was acquired in 2006 with a point density of 25 points/m^2.

The normalized graph cut is applied hierarchically over the voxel space represented as a graph. Each graph *G* is divided into two new graphs *G1* and *G2* using normalized cuts, if the number of voxels in each graph is greater then a predefined number. Then *G1* and *G2* are further subdivided into two graphs using normalized cuts. This procedure is stopped if the value of normalized cuts exceeds certain threshold.

This method of single trees detection using normalized cuts is compared with the conventional method of detecting single trees using canopy height model (CHM) [44]. The method of normalized cut is able to detect small trees under the crown of a tall tree. Additionally normalized cuts distinguished individual trees in close neighborhood better than the method based on CHM.

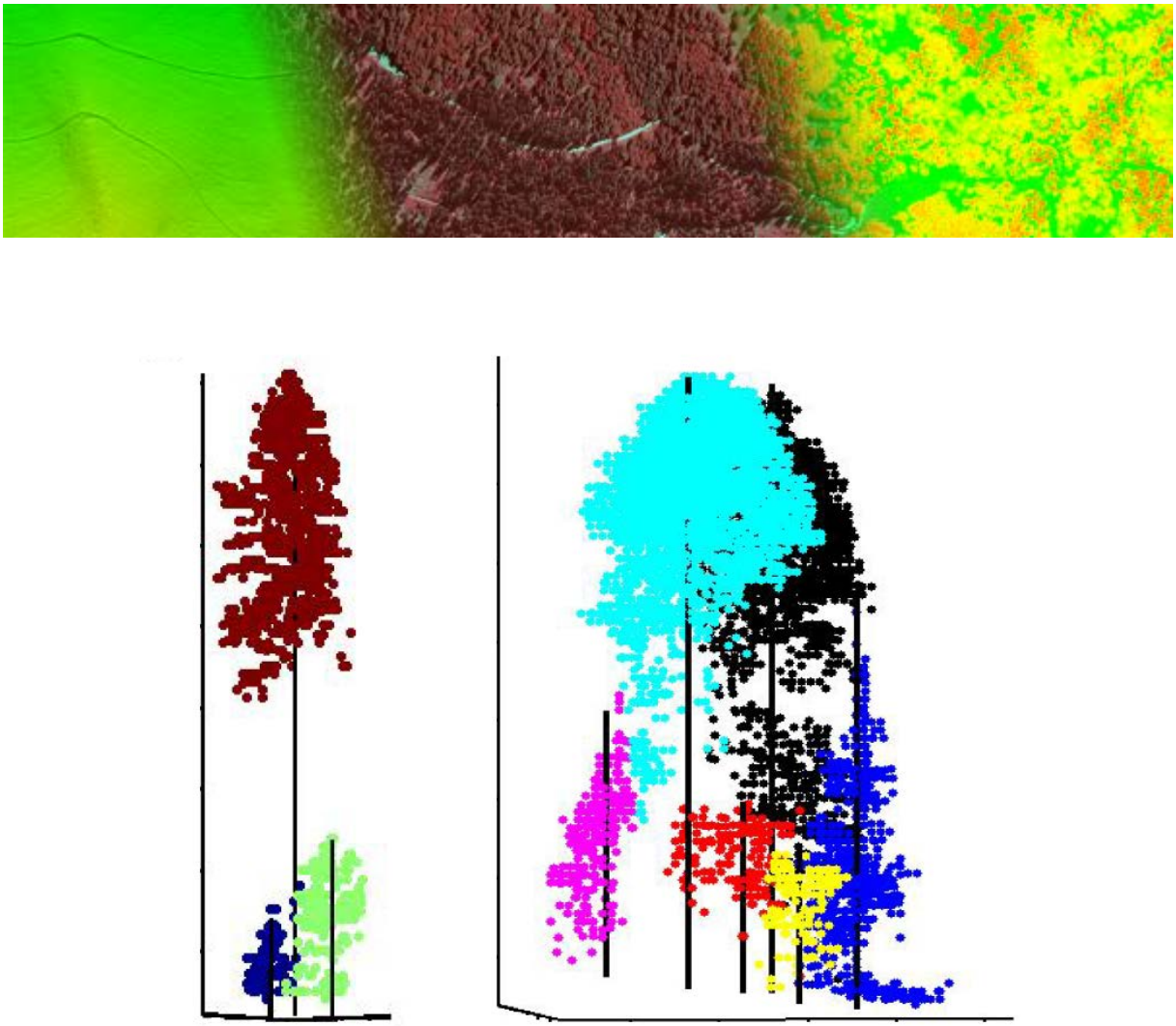


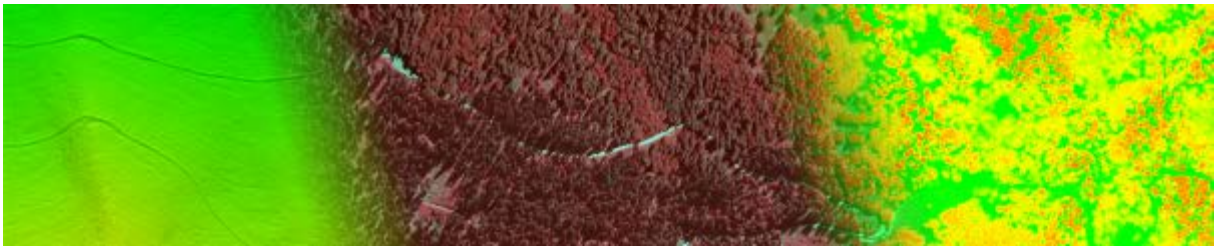
Fig. 11 Segmented individual trees

4.5 MISCELLANEOUS

4.5.1 TENSOR VOTING

The concept of Tensor voting for extracting geometric features was introduced by Guy and Medioni [45]. Tensor voting is a model free non iterative algorithm, which can be used to categorize the point-ness, curve-ness and surface-ness of each data point [46]. Each input site is represented as a second order symmetric tensor and the communication between the sites takes place through the process of voting. Tensor votes from other sites are added at each site location. The resultant tensor then encodes the geometrical information about that site. The free parameter in the tensor voting is the scale of the field. Tensor voting has been applied for point cloud segmentation [47, 48], motion segmentation [49], registration of 3D points [50] and epipolar geometry [49, 51].

The second order tensor can be visualized as a 3D ellipsoid. This can be written in the following form [46, 48]:



$$T = [e_1\ e_2\ e_3]\begin{bmatrix}\lambda_1 & 0 & 0\\ 0 & \lambda_2 & 0\\ 0 & 0 & \lambda_3\end{bmatrix}\begin{bmatrix}e_1^T\\ e_2^T\\ e_3^T\end{bmatrix}$$

$$T = \lambda_1 e_1 e_1^T + \lambda_2 e_2 e_2^T + \lambda_3 e_3 e_3^T$$

Where λ_1, λ_2 and λ_3 are the eigenvalues $(\lambda_1 \geq \lambda_2 \geq \lambda_3)$ and vectors e_1, e_2 and e_3 are the corresponding eigenvectors, representing the directions of the principal axes of the ellipsoid. The above form of the tensor can also be written as:

$$T = (\lambda_1 - \lambda_2)e_1 e_1^T + (\lambda_2 - \lambda_3)(e_1 e_1^T + e_2 e_2^T) + \lambda_3(e_1 e_1^T + e_2 e_2^T + e_3 e_3^T)$$

In this form $e_1 e_1^T$ represents a stick, $(e_1 e_1^T + e_2 e_2^T)$ represents a plate and $(e_1 e_1^T + e_2 e_2^T + e_3 e_3^T)$ represents a ball. While λ_3 represents point-ness, $(\lambda_2 - \lambda_3)$ represent curve-ness and $(\lambda_1 - \lambda_2)$ represents surface-ness [45, 51]. This is graphically shown in the Fig. 11. This tensor information at each data point is exchanged with the neighbors using ball, plate and stick fields whose magnitude depends on the orientation and decreases with distance from the source. For each point in space these tensor fields of individual sources are added up which gives the resultant tensor values at each point. These dense vector fields can then be used to generate features like junction, curves and surfaces.

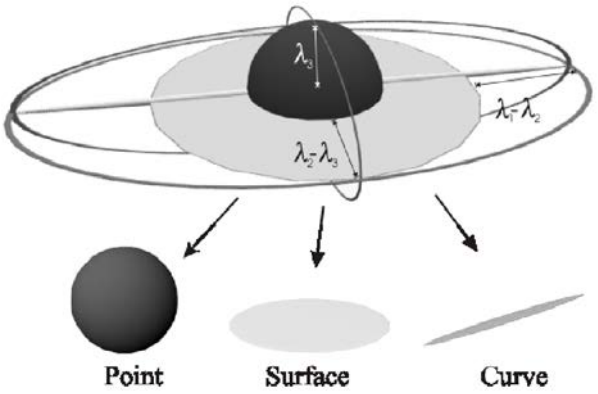
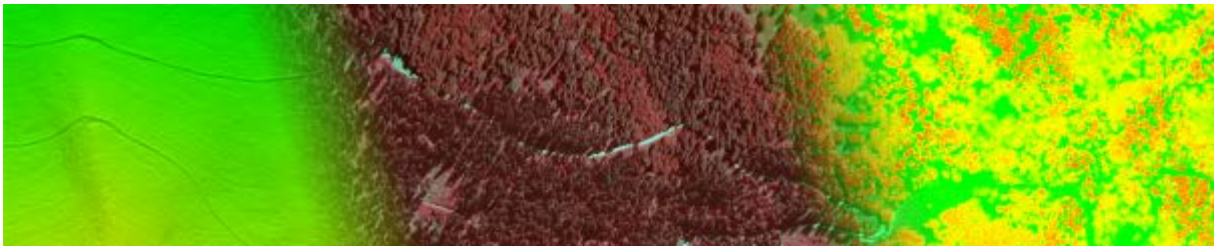


Fig. 12: Decomposition of tensor into point, surface and curve elements

4.5.2 "SEGMENTATION OF LIDAR DATA USING THE TENSOR VOTING FRAMEWORK"

Schuster [48] presents a Tensor voting based method for segmentation of Lidar into line elements, surface patches and volumetric elements. The data points are encoded as initial ellipsoid based on the confidence ellipse of the measured data. In order to obtain a dense tensor field, the space of the point cloud is sampled in a grid and the tensor values are interpolated at



each grid point using the tensor voting from original sparse data. After obtaining the dense tensor field the tensor values at each grid point are decomposed into point, surface and curve field part, which are then investigated separately to compute e.g. line and surface features. The maxima of the point field can be computed by simply finding the maxima of the smallest eigenvalue of the tensor. The most probable surface and curve points are computed using the gradient of the tensor field strength projected on the normal or the tangent vector.

In order to segment planes, a region growing algorithm is used which merges neighbors of the points based on homogeneity criterion (minimum description length criterion [52] on the orientation vector of the surface part of the tensor. Fig. 14 shows the segmented facade using tensor voting and region growing applied on TLS data. Similarly curves are segmented using the difference of orientation vectors of the curve part of the tensor as a criterion for region growing. This method is similar to the methods of Roggero [34] and Belton and Lichti [37], where region growing is used for segmentation by using homogeneity criterion based on local point distribution.

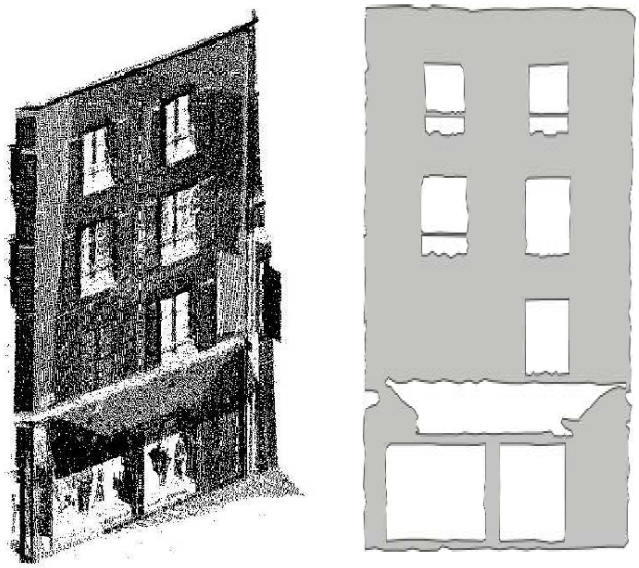
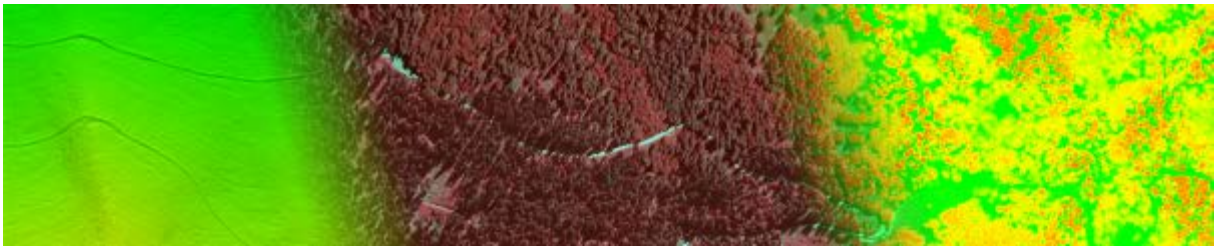


Fig. 13 Left: Terrestrial laser scan of facade Right: Segmented facade

4.5.3 "STRUCTURING LASER SCANNED TREES USING 3D MATHEMATICAL MORPHOLOGY"

Gorte and Pfeifer [53] present a 3D mathematical morphology based method for segmentation of tree stem and branches from laser scanner data. The analysis of the tree structure is performed in 3D raster domain comprised of voxels. For the experiments, resolution between 2 cm to 5 cm was used. The value assigned to each voxel is the number of original laser scanning points which fall in the voxel space. After rasterization, 3D mathematical morphological operation of closing is applied to fill in gaps due to occlusions. Similarly morphological closing is applied to fill in



hollow trees because the points on the tree appear on the surface of the wood. Afterwards a skeletonization procedure is applied to convert thickness of tree trunks and branches into single voxel. Then the skeleton of each tree is converted into a graph where each node is a voxel and the weight of the edge between the nodes depends on if the voxels share a surface, an edge or a point.

Dijkstra's algorithm [54] is applied to the individual graph to compute the shortest route from each node of the graph to the root node. The shortest paths from each node to the root collectively form a spanning tree of the graph. The spanning tree removes any loops which may have formed during the previous steps. After this step each voxel is assigned a unique branch identification number. At this stage skeleton of each tree has been labeled into unique branches. Now these labels are extended to the remaining voxels from the original voxel space using nearest neighbor search. In the last step each of the voxel is then transferred back to laser points. Fig. 15 shows final result of the segmentation in to individual branches.

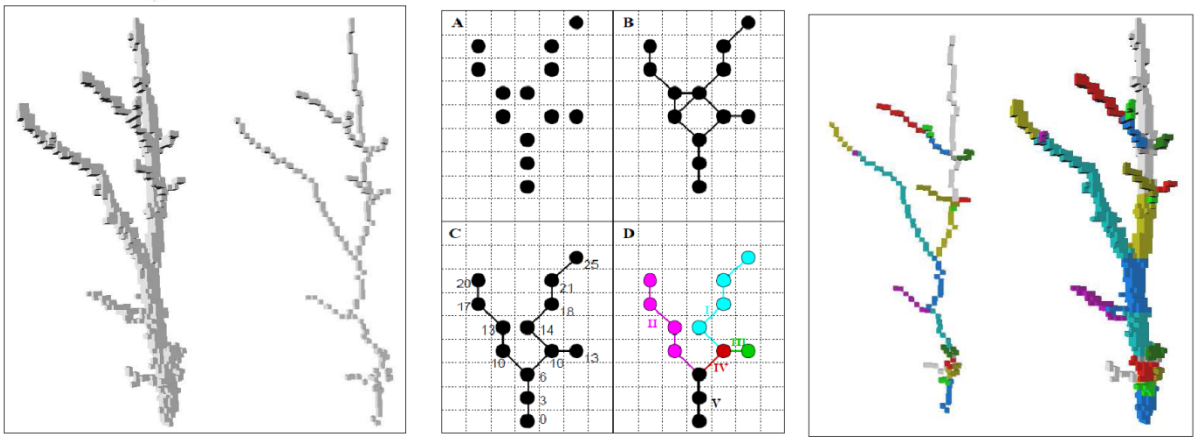
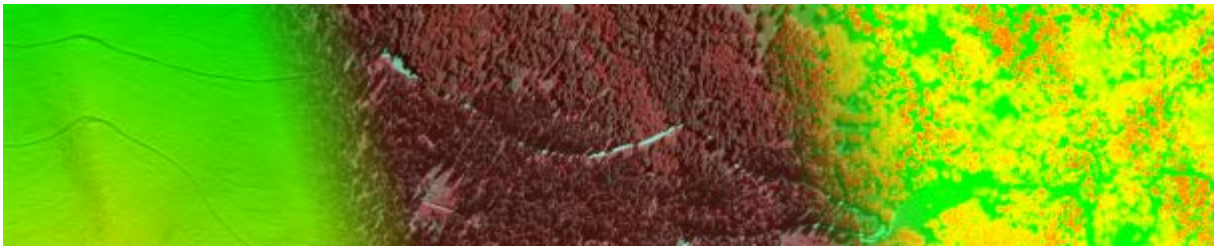


Fig. 14 Left: Rasterized points (Voxels) and skeleton Middle: Skeleton segmentation algorithm Right: Segmented skeleton and segmented voxel space

5 SURVEY OF CURRENT METHODS FOR FOREST ROADS DETECTION

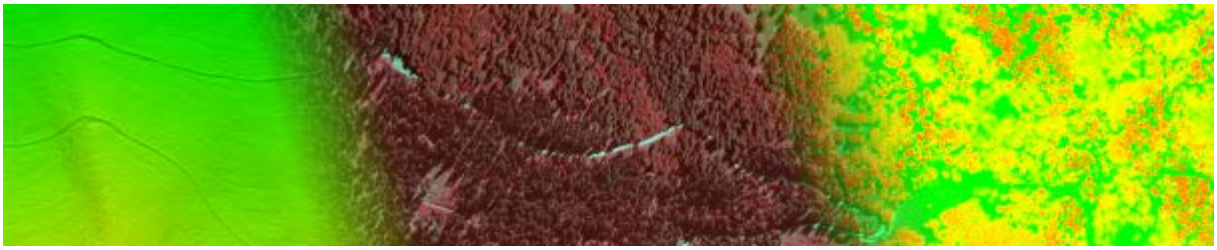
In this section we present some methods for road detection with focus on forest road extraction using lidar, which have been published in the literature. The planarity of the road surface and the fact that roads are on ground level have been exploited by authors for forest roads detection.

5.1 "THE AUTOMATIC EXTRACTION OF ROADS FROM LIDAR DATA "



The first step in the hierarchical classification of Clode [55] is to build a DSM using the last echo from the LiDAR. In order to filter out the non-terrain points a height threshold is applied to the DTM which removes points on trees and houses. An intensity threshold in then applied to differentiate grid points that lie on the roads from points that on surfaces like grass. If two or more road surface types are present in the subject area than separate threshold are applied to obtain two subsets which represents the points/pixels on the road. Morphological operations are then applied to fill in small holes. In order to remove car parks a threshold is applied on the width of the road, which successfully removes some of the car parks. A connected component analysis is performed to label the roads and all those patches whose area is less than pre-defined threshold are removed.

Fig. 15 Left: Point on DTM, Middle: Points after intensity filtering, Right: Resulting road network



5.2 "ROADS AND BUILDINGS FROM LASER SCANNER DATA WITHIN A FOREST ENTERPRISE"

Rieger et al. [56] propose a semi-automatic method to detect roads in dense forest based on digital terrain models. ALS data over a hilly region south of Vienna is used as the test data. In the first step a DTM is created from the laser pulses, which is then used to create a digital slope model which represents the elevation angle of surface normal at each grid point. This slope model is then converted into a digital image where the grey levels represent the slope of the terrain. The break lines in the DTM represent abrupt changes in the slope, therefore they can be used to detect roads in the DTM. These break lines can be detected in the slope image using an edge detection filter. An edge preserving smoothing filter is used to pre smooth the edge in order to achieve better results from edge detection. After applying the edge detection filter, lines are detected using a line extracting algorithm. Depending on the topography of the area these lines would appear broken. Therefore a semi-automatic process exploiting "snakes" [57] is used to bridge the gaps and provide longer line segments. Furthermore as the roads would be represented by two parallel line contour's an extension to the original snake algorithm known as "twin snakes" [58] which gives line segments representing the sides of the roads.

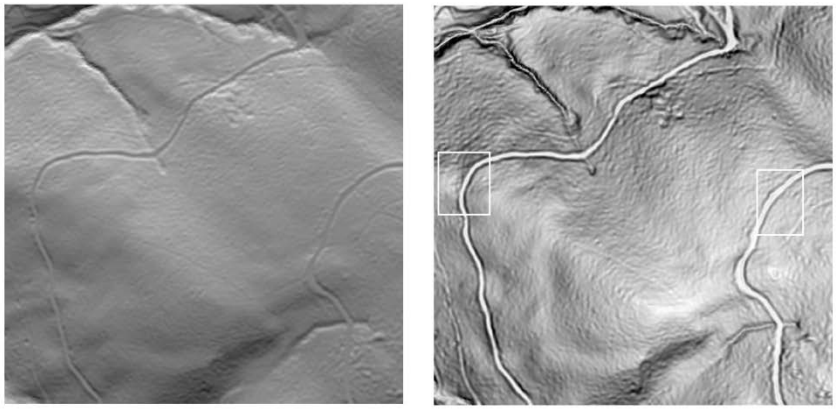


Fig. 16 Hill shaded view of the terrain model (Left), Slope model (Right)

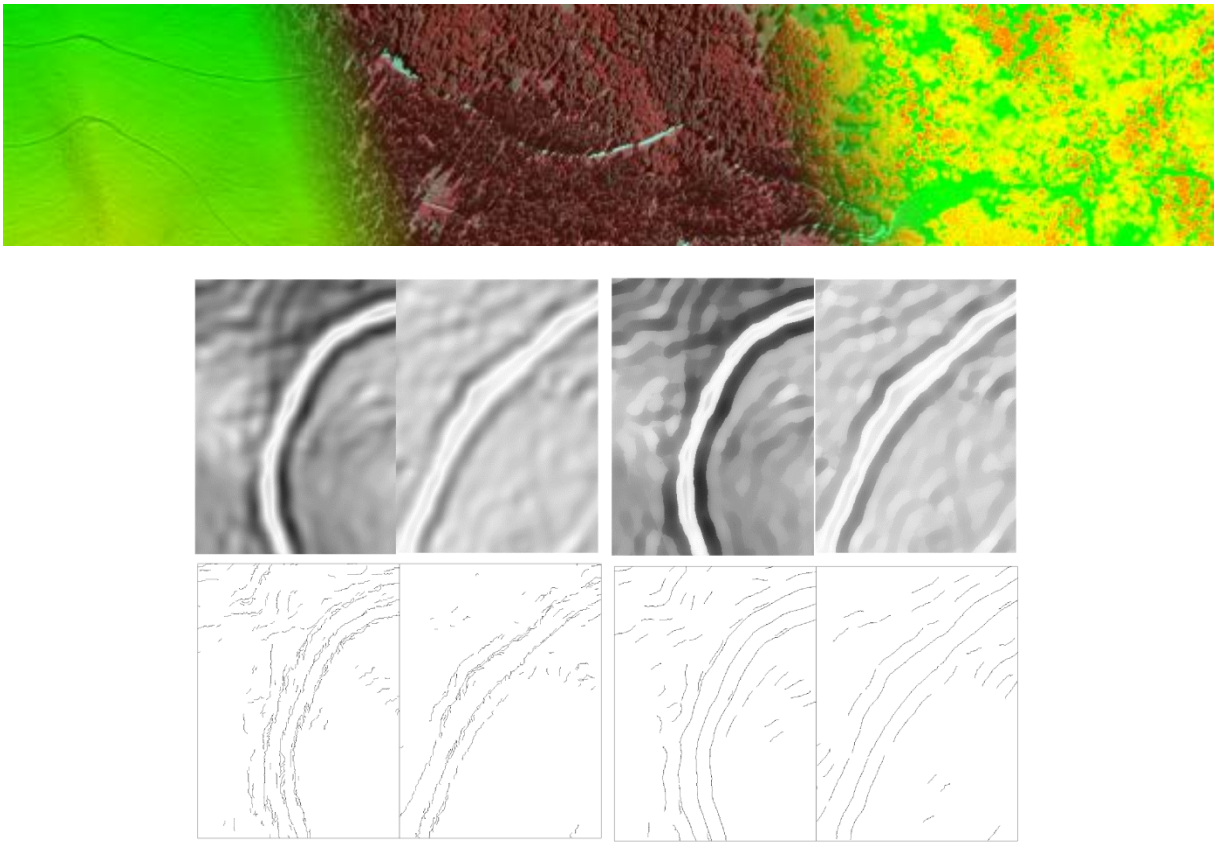


Fig. 17 Detailed Slope model (Top Left), Slope model after preprocessing (Top Right), Edges extracted from slope model (Bottom Left), Edges extracted after preprocessed slope model (Bottom Right)

5.3 "FOREST ROADS MAPPED USING LIDAR IN STEEP FORESTED TERRAIN"

Russel et al. [59] present a manual method of extracting roads in forests using DEM and compare the results to field surveyed control points. The test site is an area, 19 km north of the City of Santa Cruz in the Little Creek watershed. A road centerline survey was done in the test area to determine the location and elevation of the target road and this data is used to analyze the accuracy of the road detection. In order to compute a DEM off-terrain points (e.g. vegetation) were removed to obtain ground points. The filtered off-terrain points comprised of approx. 94% of the total points. A 1m DEM was computed from the remaining points. The topographic features like roads, stream banks, are easily visible in the rasterized grids of slope and shaded relief. Over 30km of road and trail features were manually digitized using ArcMap through visual interpretation of these grids. A comparison with the surveyed centerline showed total road length of the road centerline was measured to within 0.2% and road segment slopes measured with a mean absolute difference of 0.53%.

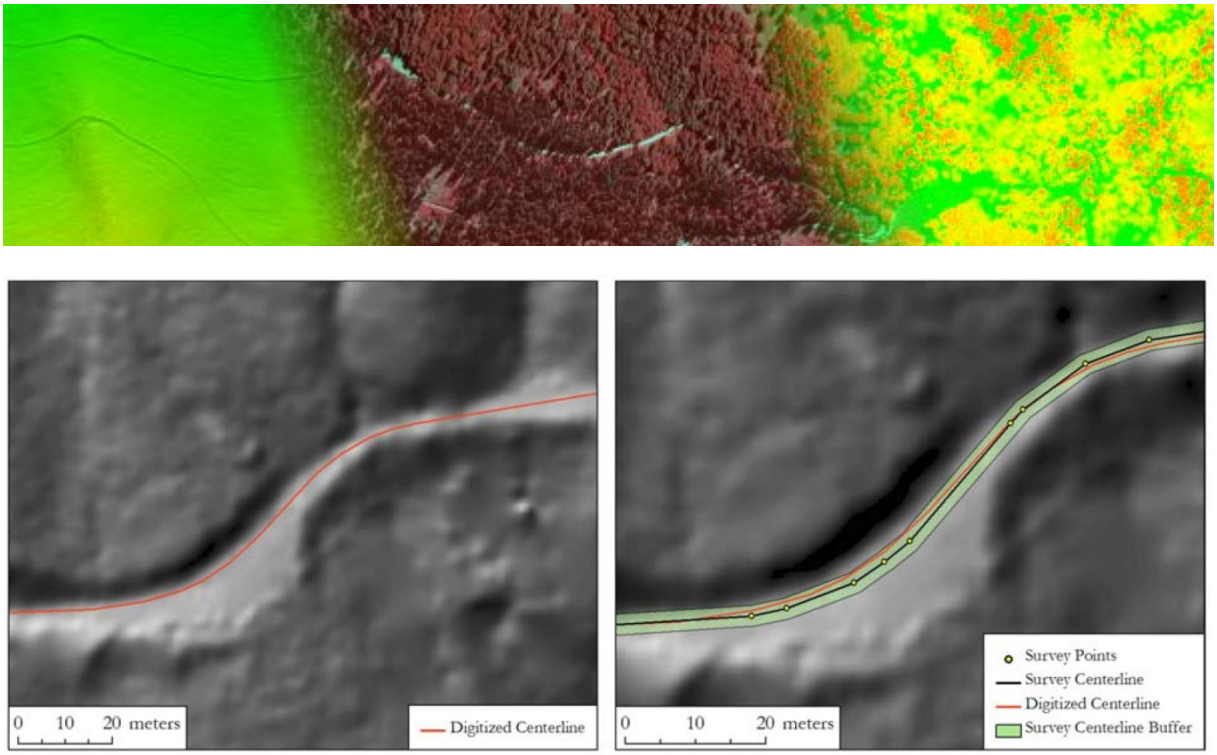


Fig. 18 Left: Road centerline digitized from LiDAR shaded relief grid, Right: Field Surveyed centerline (black) and digitized centerline (red)

5.4 "PATHWAY DETECTION AND GEOMETRICAL DESCRIPTION FROM ALS DATA IN FORESTED MOUNTAINOUS AREA"

David et al. [1] deals with detection of roads and pathways using features like intensity, height variance in rasterized images of the subject area. The study area is 108ha state forest located in southern French Alps and the flight campaign was done in April 2007. The roads and pathways are ground based objects which are planar and show radiometric contrast to the vegetation. Therefore for detection of roads and pathways, rasterized images (1m grid size) based on following three features are computed

1) Height - Pathways are ground objects
2) Height variance – Roads and pathways are planar
3) Intensity- Roads and pathways show different radio radiometric properties from vegetation

Afterwards seeds points are manually or automatically chosen and the candidate pixels in each image are selected using a region growing algorithm based on the mean and variance of the region. The selected pixels in each of the three images are then merged together to produce a single image representing the pathways. It has been stated that the intersection of the three images produce the best results. Afterwards these pathways are vectorized by first applying a median filter on the resulting image and then applying morphological operations to obtain smooth pathway borders. Small pathways which are less than 5m are filtered out. Finally to estimate the road width and centers, 1D sections within the lidar point cloud are successively generated for each point of the pathways.

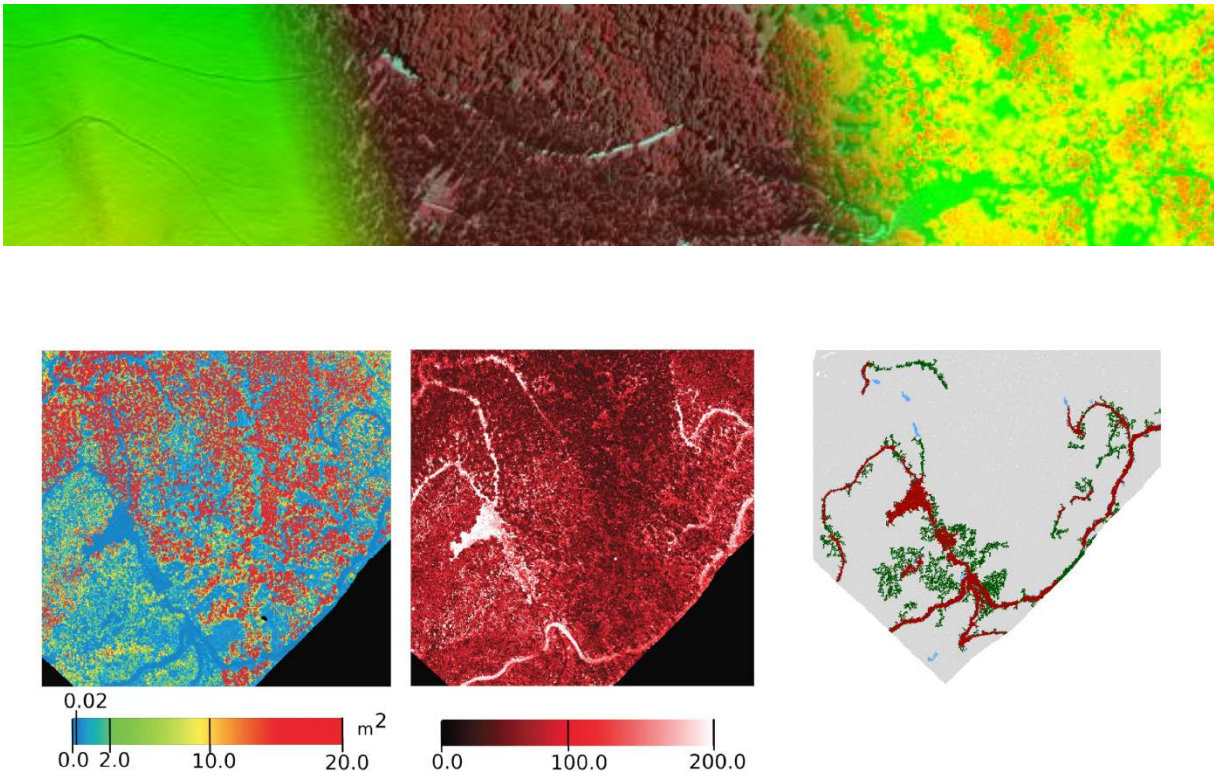


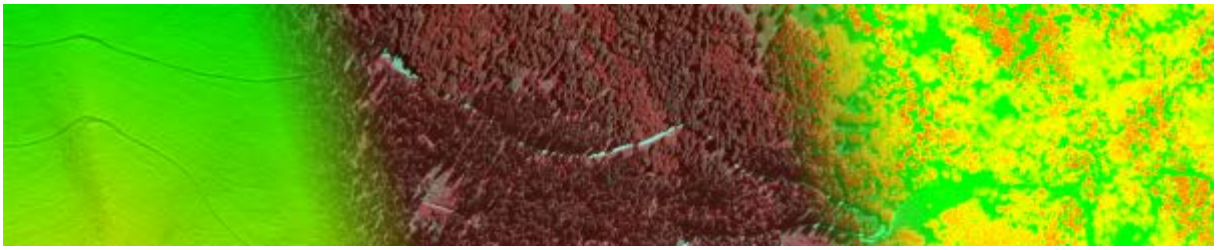
Fig. 19 Left: Elevation Image, Middle: Intensity Image , Right: Image showing masks (nDSM, variance σz and Intensity I), Red= [σz ∩ I ∩ nDSM], Green=[σz ∩ nDSM], Blue=[I ∩ nDSM]

6 IMAGE BASED TECHNIQUES

In this section a brief summary of the road extraction from images using segmentation is presented. A survey of image based road extraction methods can be found in [60] and [61]. The main characteristics used for detecting roads in point cloud are smoothness and height from ground. In images roads are mainly detected using characteristics like constant width, low curvature, connectivity and consistent reflection properties due to similar materials used in manufacturing [61]. Fortier et al. [61] categorize these characteristics into Spectral properties: surface reflectance, Geometric properties: width and curvature and topological properties: connectivity and links.

Zhang et al. [62] classify roads based on the gray value in the intensity images. Afterwards mathematical morphological operations are performed to remove noise and to obtain continuous roads surfaces. Laptev et al. [63] propose a road detection from images using scale space and snakes [57]. The centerlines of the roads are extracted as lines in coarser scale, which initializes the snakes at fine scales. Further constraints of low curvature, consistent width and connectivity are checked to obtain roads in the image.

Heipke et al. [64] present a hierarchical approach to road extraction where roads are extracted in low and high resolution separately and the results are combined to improve the quality of road detection. In low resolution roads are detected using a gray level threshold and connectivity. In high resolution roads are detected using edge detection and exploiting the fact that edges on sides of the roads should remain parallel and have constant distance. Similar strategy has been used by Baumgartner et al. [65, 66] to detect roads from aerial imagery. Lines are detected in low resolution while parallel edges are detected in high resolution. The results from multiple resolutions are combined to extract roads.



Agouris et al. [67] present a spatio-spectral clustering based method for detecting roads in multispectral images. In the first step clustering is applied to group pixels into clusters based on spectral properties. Afterwards morphological opening and component linking is applied to points belonging to the spectral class corresponding to roads. Haala and Vosselman [68] detect roads using classification using thresholding and region growing on hue of the observed image.

7 SUMMARY

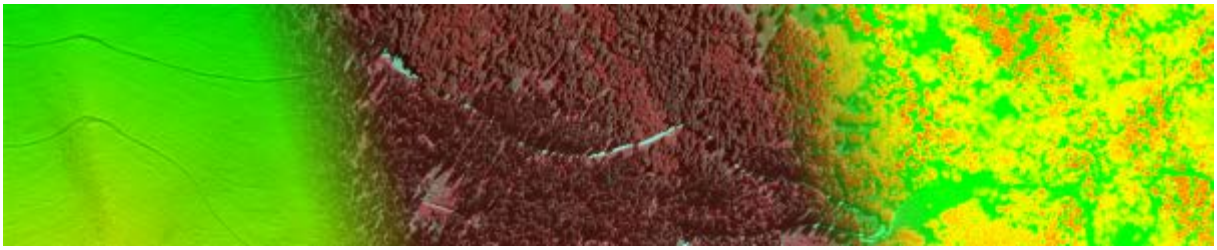
In this report several point cloud segmentation techniques were discussed and the some examples of road extraction from point cloud were presented. In the end a brief overview of road extraction from images was presented.

Most of the segmentation methods do not strictly fulfill the definition put forward at start of section 2. Often only segments (object) of interest are built, whereas the remaining point cloud forms one , non-connected rejection segment (class). As neighborhood systems either k-nearest neighbor, fixed distance neighbor, pixels or voxels are used. Features are often only spatial distance, distance and normal vectors, additionally local point distribution ("roughness"), radiometric and full waveform features are used. Comparison of segmentation results to ground truth is hardly performed. One reason may be that segmentation is part of a workflow for object identification and reconstruction which allows evaluation of higher level results.

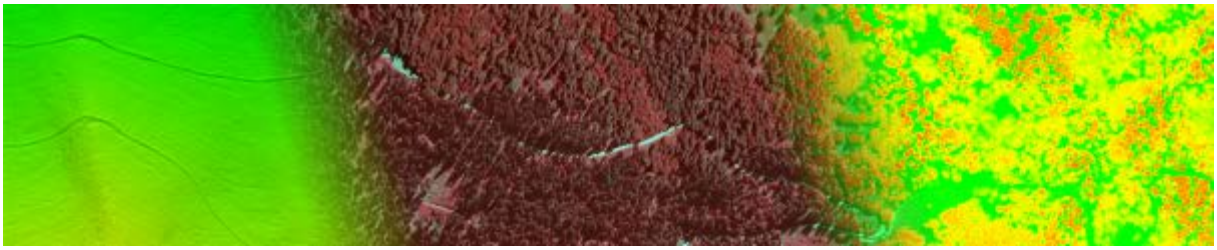
Airborne laser scanning has a great potential in extracting forest roads, however there is a lack of robust and automatic methods in achieving this goal. . The planarity of the road surface and the fact that roads are on ground level have been exploited by authors for forest roads detection. The fusion of intensity data along with plane fitting residuals have so far given the best results in forest road extraction. However intensity of lidar signal is strongly dependent on the properties of the surface and the flight mission. Generally we need more robust measures to extract roads in varying terrain types. Using the calibrated surface reflectance [69] should be investigated as a potential remedy.

Table 1: Neighborhood, distance and features used in given methods

			Features		
Methods	**Neighborhood**	**Distance**	**Position**	**Intensity**	**Normal**
Morsdorf et al. [9]	Points				
Melzer [11]	Points	Yes	Yes	Yes	Yes
Filin and Pfeifer [16]	points				

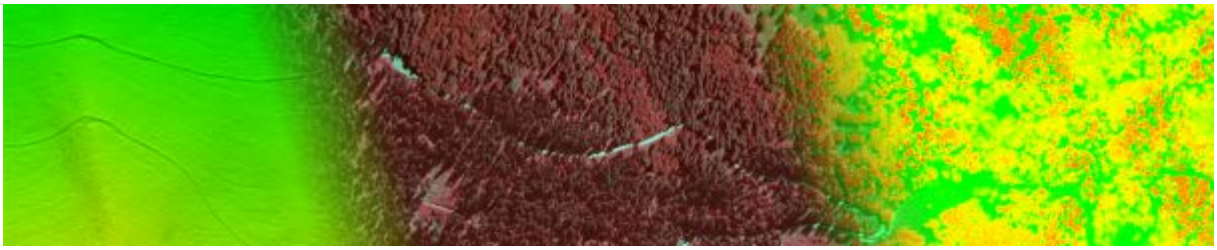


Vosselman et al.[70]	Points				
Schnabel et al.[10]	points				
Toavri and Pfeifer [35]	Points KNN	Yes			
Rabbani [6]	Points KNN	Yes			Yes
Roggero [34]	Points				
Belton and Lichti [37]	Points				
Golovinskiy and Funkhouser [42]	Points				
Reitberger et al. [7]	Points				
Schuster [48]	Points				
Gorte and Pfeifer [53]	Voxel 26 neighbors	Yes			
Clode [55]	Pixels 8 neigh			Yes	

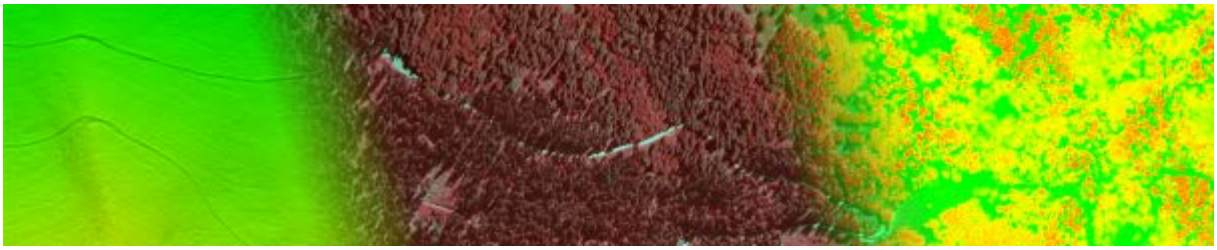


8 REFERENCES

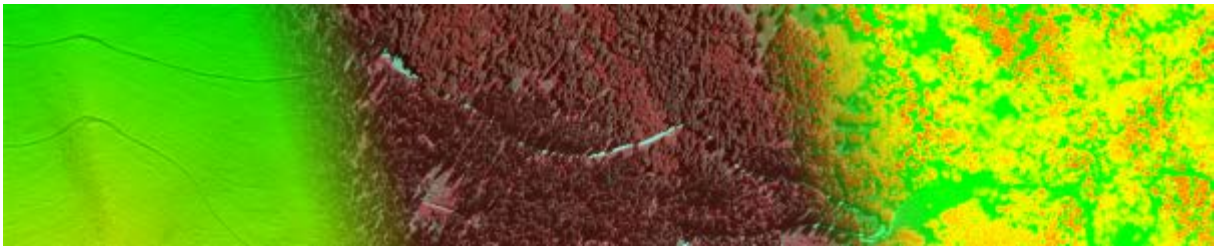
1. David, N., et al., *Pathway detection and geometrical description from ALS data in forested mountaneous area.* Proceedings of Laser Scanning 2009, 2009. **38**: p. 242-247.
2. Hoover, A., G. Jean-Baptiste, and X. Jiang, *An experimental comparison of range image segmentation algorithms.* Pattern Analysis, 1996. **18**: p. 673-689.
3. Reisner-Kollmann, I., *Segmenting multiple range images with primitive shapes.* Systems, Signals and, 2012.
4. Vosselman, G., et al., *Recognising structure in laser scanner point clouds.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2004. **46**: p. 33-38.
5. Rabbani, T., F. van Den Heuvel, and G. Vosselmann, *Segmentation of point clouds using smoothness constraint.* IEVM06, 2006. **1**: p. 1-6.
6. Rabbani, T., F. van Den Heuvel, and G. Vosselmann, *Segmentation of point clouds using smoothness constraint.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2006. **36**(5): p. 248-253.
7. Reitberger, J. *3D segmentation of full waveform Lidar data for single tree detection using normalized cut*. in *ISPRS Congress Beijing 2008*. 2008.
8. Tittmann, P., et al., *Tree Detection and Delineation from LiDAR point clouds using RANSAC.* Scanning, 2011: p. 1-23.
9. Morsdorf, F., et al., *Clustering in airborne laser scanning raw data for segmentation of single trees.* International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2003. **34**(part 3): p. W13-W13.
10. Schnabel, R., R. Wahl, and R. Klein, *Efficient RANSAC for Point-Cloud Shape Detection.* Computer Graphics Forum, 2007. **26**(2): p. 214-226.
11. Melzer, T., *Non-parametric segmentation of ALS point clouds using mean shift.* Journal of Applied Geodesy, 2007. **1**: p. 159-170.
12. Jain, A.K., M.N. Murty, and P.J. Flynn, *Data clustering: A review.* ACM Computing Surveys, 1999. **31**(3): p. 264-323.
13. Duda, R.O., P.E. Hart, and D.G. Stork, *Pattern classification*. 2nd ed2001, New York: Wiley. xx, 654 p.
14. Fukunaga, K. and L. Hostetler, *The estimation of the gradient of a density function, with applications in pattern recognition.* IEEE Transactions on Information Theory, 1975. **21**(1): p. 32-40.
15. Sotoodeh, S., *Hierarchical clustered outlier detection in laser scanner point clouds.* Laser07, 2007. **36**: p. 383.
16. Filin, S. and N. Pfeifer, *Segmentation of airborne laser scanning data using a slope adaptive neighborhood.* ISPRS Journal of Photogrammetry and Remote Sensing, 2006. **60**: p. 71-80.
17. Kanungo, T., et al., *An efficient k-means clustering algorithm: Analysis and implementation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002. **24**(7): p. 881-892.
18. MacQueen, J., *Some methods for classification and analysis of multivariate observations.* Proceedings of the fifth Berkeley symposium, 1967. **233**: p. 281-297.



19. Cheng, Y. and K.S. Fu, *Conceptual clustering in knowledge organization.* IEEE transactions on pattern analysis and machine intelligence, 1985. **7**: p. 592-8.
20. Cheng, Y., *Mean shift, mode seeking, and clustering.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995. **17**(8): p. 790-799.
21. Haralick, R.M. and L.G. Shapiro, *Computer and robot vision*1992, Reading, Mass.: Addison-Wesley Pub. Co. v. <1-2 >.
22. Hough, P.C., *Method and means for recognizing complex patterns*, 1962, US Patent.
23. Rabbani, T., *Efficient hough transform for automatic detection of cylinders in point clouds.* ISPRS WG III/3, III/4, 2005.
24. Oda, K., et al. *Automatic building extraction and 3-D city modeling from Lidar data based on hough transformation*. in *Commission III, PS WG III/3*. 2004.
25. Vosselman, G. and S. Dijkman, *3D building model reconstruction from point clouds and ground plans.* International Archives of Photogrammetry and Remote Sensing, 2001. **34**(3/W4): p. 22-24.
26. Overby, J., L. Bodum, and E. Kjems, *Automatic 3D building reconstruction from airborne laser scanning and cadastral data using Hough transform.* ISPRS Congress, July, 2004.
27. Tarsha-Kurdi, F., T. Landes, and P. Grussenmeyer. *Hough-transform and extended RANSAC algorithms for automatic detection of 3D building roof planes from LiDAR data*. in *Proceedings of International Society of Photogrammetry Remote Sensing*. 2007.
28. Bolles, R. and M. Fischler. *A RANSAC-based approach to model fitting and its application to finding cylinders in range data*. in *Proceedings of the 7th International Joint Conference on Artificial Intelligence*. 1981.
29. Hatger, C. and C. Brenner, *Extraction of road geometry parameters from laser scanning and existing databases.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2003. **34**(3/W13): p. 225-230.
30. Yang, S.-w., C.-c. Wang, and C.-h. Chang. *RANSAC matching: Simultaneous registration and segmentation*. in *2010 IEEE International Conference on Robotics and Automation*. 2010. IEEE.
31. Boulaassal, H., T. Landes, and P. Grussenmeyer. *Automatic segmentation of building facades using terrestrial laser data*. in *SilviLaser*. 2007.
32. Fischler, M.A. and R.C. Bolles, *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.* Communications of the ACM, 1981. **24**(6): p. 381-395.
33. Fan, J., et al., *Seeded region growing: an extensive and comparative study.* Pattern Recognition Letters, 2005. **26**: p. 1139-1156.
34. Roggero, M., *Object segmentation with region growing and principal component analysis.* International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, 2002. **34**(Part 3A): p. 289–294.
35. Tóvári, D. and N. Pfeifer. *Segmentation based robust interpolation-a new approach to laser data filtering*. in *ISPRS WG III/3, III/4, V/3 Workshop "Laser scanning 2005*. 2005.
36. Rottensteiner, F. and C. Briese, *A new method for building extraction in urban areas from high-resolution LIDAR data.* International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, 2002. **34**(3/A): p. 295-301.
37. Belton, D. *Classification and segmentation of terrestrial laserscanner point clouds using local variance information*. in *IAPRS, XXXVI*. 2006. Dresden, Germany.



38. Wu, Z. and R. Leahy, *An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1993. **15**(11): p. 1101-1113.
39. Shi, J. and J. Malik, *Normalized cuts and image segmentation.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000. **22**: p. 888-905.
40. Yao, W. and S. Hinz. *Object extraction based on 3d-segmentation of LiDAR data by combining mean shift with normalized cuts: two examples from urban areas*. in *Urban Remote Sensing Event, 2009*. 2009.
41. Sedlacek, D. and J. Zara, *Graph cut based point-cloud segmentation for polygonal reconstruction.* Advances in Visual Computing, 2009: p. 218-227.
42. Golovinskiy, A. and T. Funkhouser. *Min-cut based segmentation of point clouds*. in *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), 2009* 2009. IEEE.
43. Wang, L. and H. Chu. *Graph theoretic segmentation of airborne lidar data*. in *Proceedings of SPIE*. 2008. Spie.
44. Reitberger, J., P. Krzystek, and U. Stilla, *Combined tree segmentation and stem detection using full waveform lidar data.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2007. **36**: p. 332-337.
45. Guy, G., *Inference of surfaces, 3D curves, and junctions from sparse, noisy, 3D data.* Pattern Analysis and Machine Intelligence, 1997. **19**: p. 1265-1277.
46. Medioni, G. and C. Tang. *Tensor voting: Theory and applications*. in *Proceedings of RFIA, Paris, France*. 2000.
47. Kim, E. and G. Medioni, *Urban scene understanding from aerial and ground LIDAR data.* Machine Vision and Applications, 2010. **22**: p. 691-703.
48. Schuster, H. *Segmentation of lidar data using the tensor voting framework*. in *XXth ISPRS Congress*. 2004.
49. Tong, W.-S., C.-K. Tang, and G. Medioni, *Simultaneous two-view epipolar geometry estimation and motion segmentation by 4D tensor voting.* IEEE transactions on pattern analysis and machine intelligence, 2004. **26**: p. 1167-84.
50. Reyes, L., G. Medioni, and E. Bayro, *Registration of 3D Points Using Geometric Algebra and Tensor Voting.* International Journal of Computer Vision, 2007. **75**: p. 351-369.
51. Tang, C. and G. Medioni, *N-dimensional tensor voting and application to epipolar geometry estimation.* Pattern Analysis and Machine, 2001. **23**: p. 829-844.
52. Rissanen, I., *Minimum Description Length Principle*, in *Encyclopedia of Statistical Sciences*1987. p. 523–527.
53. Gorte, B. and N. Pfeifer, *Structuring laser-scanned trees using 3D mathematical morphology.* International Archives of Photogrammetry and Remote Sensing, 2004. **35**(B5): p. 929-933.
54. Dijkstra, E., *A note on two problems in connexion with graphs.* Numerische mathematik, 1959. **1**: p. 269-271.
55. Clode, S., P. Kootsookos, and F. Rottensteiner, *The automatic extraction of roads from LIDAR data.* The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science, 2004. **35**: p. 231-237.
56. Rieger, W., et al., *Roads and buildings from laser scanner data within a forest enterprise.* International Archives of Photogrammetry and Remote Sensing, 1999. **32**(Part 3-W14): p. 185 –191.
57. Kass, M., A. Witkin, and D. Terzopoulos, *Snakes: Active contour models.* International journal of computer vision, 1988. **1**(4): p. 321-331.



58. Kerschner, M., *Homologous twin snakes integrated in a bundle block adjustment.* International Archives of Photogrammetry and Remote Sensing, 1998. **32**: p. 244-249.
59. White, R.a., et al., *Forest Roads Mapped Using LiDAR in Steep Forested Terrain.* Remote Sensing, 2010. **2**(4): p. 1120-1141.
60. Mena, J.B., *State of the art on automatic road extraction for GIS update: a novel classification.* Pattern Recognition Letters, 2003. **24**: p. 3037-3058.
61. Fortier, A., et al., *Survey of work on road extraction in aerial and satellite images*, 1999, Center for Topographic Information Geomatics, Ontario, Canada. Technical Report.
62. Zhang, C., et al., *Road network detection by mathematical morphology.* 1999: p. 185-200.
63. Laptev, I., et al., *Automatic extraction of roads from aerial images based on scale space and snakes.* Machine Vision and Applications, 2000. **12**(1): p. 23-31.
64. Heipke, C., C. Steger, and R. Multhammer. *A hierarchical approach to automatic road extraction fron aerial imagery*. in *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II, Proc. SPIE,*. 1995.
65. Baumgartner, A., et al., *Update of roads in gis from aerial imagery: Verification and multi-resolution extraction.* International Archives of Photogrammetry and Remote Sensing, 1996. **31**: p. 53-58.
66. Baumgartner, A., C. Steger, and H. Mayer, *Multi-resolution, semantic objects, and context for road extraction*, in *Semantic Modeling for the Acquisition of Topographic Information from Images and Maps*, W. Förstner, Plümer, L. , Editor 1997, Birkhäuser-Verlag. p. 150-156.
67. Agouris, P., P. Doucette, and A. Stefanidis. *Spatiospectral cluster analysis of elongated regions in aerial imagery* in *Proceedings of International Conference on Image Processing*. 2001. Thessaloniki.
68. Haala, N. and G. Vosselman, *Recognition of road and river patterns by relational matching.* International Archives of Photogrammetry and Remote Sensing, 1992. **29**: p. 969-975.
69. Briese, C., et al. *Calibration of full-waveform airborne laser scanning data for object classification*. in *Laser Radar Technology and Applications XIII, SPIE Proceedings*. 2008. Orlando.
70. Vosselman, G., et al., *Recognising structure in laser scanner point clouds.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2004. **46**(8): p. 33-38.