ISPRS Journal of Photogrammetry and Remote Sensing 128 (2017) 354–371

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Contextual segment-based classification of airborne laser scanner data

George Vosselman [a,*], Maximilian Coenen [b], Franz Rottensteiner [b]

[a] Faculty of Geo-Information Science and Earth Observation (ITC), Department of Earth Observation Science, University of Twente, Enschede, The Netherlands
[b] Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Nienburger Straße 1, 30167 Hannover, Germany

ARTICLE INFO

Article history:
Received 14 December 2016
Received in revised form 17 March 2017
Accepted 21 March 2017
Available online 26 April 2017

Keywords:
Point cloud
Segmentation
Classification
CRF

ABSTRACT

Classification of point clouds is needed as a first step in the extraction of various types of geo-information from point clouds. We present a new approach to contextual classification of segmented airborne laser scanning data. Potential advantages of segment-based classification are easily offset by segmentation errors. We combine different point cloud segmentation methods to minimise both under- and over-segmentation. We propose a contextual segment-based classification using a Conditional Random Field. Segment adjacencies are represented by edges in the graphical model and characterised by a range of features of points along the segment borders. A mix of small and large segments allows the interaction between nearby and distant points. Results of the segment-based classification are compared to results of a point-based CRF classification. Whereas only a small advantage of the segment-based classification is observed for the ISPRS Vaihingen dataset with 4–7 points/m^2, the percentage of correctly classified points in a 30 points/m^2 dataset of Rotterdam amounts to 91.0% for the segment-based classification vs. 82.8% for the point-based classification.
© 2017 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

1. Introduction

Point clouds have become a standard type of data in production processes of digital terrain models, 3D city and landscape models, and land use maps. Whether produced by airborne laser scanning (ALS) or by dense matching of aerial photographs, classification of point clouds is required as the first step to extract the geo-information to be produced. While DTM production only requires a classification into ground and non-ground points, other processes typically require discrimination between multiple classes. Developed methods for point cloud classification can be categorised as point-based or segment-based, e.g. (Xu et al., 2012). Point-based classification makes use of point features, like pulse reflectance strength (lidar) or colour (image), as well as features characterising the point distribution in the local neighbourhood of the points, like surface smoothness or local normal vector direction. These features are calculated for every point and class labels are assigned point by point. In contrast, segment-based approaches first divide a point cloud into segments and assign class labels to segments such that all points within a segment obtain the same class label.

Potentially, segment-based classification has several advantages over point-based classification. First, segment feature values can be calculated by averaging over feature values of the points in a segment and may therefore better represent class characteristics. Secondly, segments contain more information than a point with its local neighbourhood. Therefore, a segment-based classification can use more features, like segment size, shape, percentage of last pulse echoes, or (variations in) point density. Such additional features may improve the class separability. Thirdly, segment-based classifications may be more effective in utilising context information. In point-based classification using probabilistic relaxation (Smeeckaert et al., 2013) or Conditional Random Fields (CRF; Niemeyer et al., 2014) the main use of context information is to ensure a local consistency of the point labels, e.g. based on models favouring a smooth labelling. In a segment-based classification, context is already intrinsically considered by the use of segments, which are expected to consist of points having similar properties and/or belonging to the same object. As segments can be very large, interactions between points that may be far apart from each other (*long range interactions*) are intrinsically modelled, which is even extended by an explicit model of context between segments belonging to different classes. Finally, as the number of segments is much smaller than the number of points, the classification can be much faster. This will, however, be partially offset by the time needed to segment the point cloud.

* Corresponding author.
E-mail addresses: george.vosselman@utwente.nl (G. Vosselman), coenen@ipi.uni-hannover.de (M. Coenen), rottensteiner@ipi.uni-hannover.de (F. Rottensteiner).

http://dx.doi.org/10.1016/j.isprsjprs.2017.03.010
0924-2716/© 2017 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

The full potential of the above advantages can only be realised with a good segmentation. Under- and over-segmentation errors will negatively affect the classification accuracy. In particular, under-segmentation will inevitably lead to classification errors as a segment-based classification will assign the same class label to all points of a segment. To ensure under-segmentation is minimal, a strong over-segmentation is often preferred (Lim and Suter, 2009), similar to the use of super-pixels in image classification. However, features, like segment size or point density variation, become less useful in case of over-segmentation and in general all advantages diminish.

In this paper our goal is to improve point cloud classification by segmenting point clouds into as large as possible segments with rich descriptors and by making optimal use of these segments in a contextual classification. The scientific contributions are as follows:

- We propose the integration of multiple segmentation approaches for segment- and context-based classification. Combining different segmentation approaches is advantageous for a segmentation of point clouds containing objects of different point distributions. We expect this to provide a good starting point for contextual segment-based classification.
- For the contextual segment-based classification of point clouds based on a CRF, we propose the use of a local definition of the neighbourhood relations as the basis of the graph underlying the CRF, which is more flexible with respect to handling segmentations delivering segments of different shapes and spatial extents than a definition based on the arrangement of representative points of the segments as, for instance, in (Shapovalov et al., 2010).
- In this context, we also propose a new set of interaction features based on a local analysis of the point cloud in the vicinity of the segment boundaries, which will be used for a context model based on a generic classifier.
- We argue that very large segments corresponding to the ground and to standing water bodies can be identified reliably by simple heuristics. Nevertheless, we propose to consider these segments in the CRF-based classification as *fixed nodes* that will not change their class labels, but will contribute valuable context information.

This paper is structured as follows. We start with a discussion of related work in Section 2, focussing on methods for the segmentation and contextual classification of ALS point clouds. In Section 3, the new segmentation algorithm is described, and the methodology for CRF-based classification will be explained in Section 4. Section 5 presents a thorough evaluation of the approach using two data sets, comparing the results of the new method with a point-based classification and showing the advantages of considering context in the classification procedure. Finally, conclusions and an outlook on future work are given in Section 6.

2. Related work

This review on the related work focusses on the two main aspects of our new approach. We start with a discussion of related work in the field of ALS segmentation. After that, work on the classification of point clouds will be presented with a focus on contextual methods and on methods designed for ALS data.

2.1. Segmentation of ALS point clouds

Many algorithms have been developed for the extraction of surfaces from point clouds. Efficient RANSAC (Schnabel et al., 2007) and 3D Hough transform combined with surface growing (Vosselman, 2012) are often used in work to extract roof faces and other surfaces from airborne laser scanning data. While these methods often serve their purpose, they are not well suitable for segmenting the parts of a point cloud that cannot be described by surfaces. This review focusses on such non-parametric methods for point cloud segmentation. A more extensive survey of point cloud segmentation methods is provided by Nguyen and Le (2013).

Melzer (2007) presented a first study to apply mean shift (Comaniciu and Meer, 2002) to the segmentation of urban point clouds. Points on buildings, vegetation and terrain were already grouped by using mode seeking with only the X-. Y-and Z-coordinates. Finer segmentations were obtained when also making use of amplitude and pulse width of the echoes. Ferraz et al. (2010) used mean shift to separate surface vegetation, understory and overstory in forested areas. Yao et al. (2009) combined mean shift with normalised cuts to extract vehicles and flyovers. Rutzinger et al. (2008) used segment growing to cluster and classify vegetation points in an urban environment. Only the homogeneity in echo widths was used as a criterion for clustering neighbouring points. This feature typically distinguishes vegetation from smooth surfaces. Some over-segmentation in vegetation was observed because of variation in the echo widths within the vegetation.

More work on segmenting point clouds into non-planar segments has been performed with mobile laser scanning data. A typical workflow is to determine the points on the ground surface, remove those points from the dataset and then determine the connected components in the remaining point set (Douillard et al., 2010). Pu et al. (2011) and Velizhev et al. (2012) in addition incorporated scene knowledge to select components for further classification. Pu et al. (2011) eliminated large vertical components (walls) when extracting street furniture whereas Velizhev et al. (2012) selected on component size and distance to the ground when selecting cars and street lights. Golovinskiy and Funkhouser (2009) made initial estimates of background points (street level) and foreground points (street furniture, cars) and then used a min-cut based segmentation to improve the initial estimates.

Aijazi et al. (2013) segmented a point cloud generated by mobile laser scanning in two steps. After removing points on the ground the remaining connected components are segmented based on colour and reflectance strength. A two-step approach applied to airborne laser scanning point clouds has been presented by Xu et al. (2012). After an initial segmentation and classification of planar point sets, connected components of points with a doubtful classification were re-segmented using mean shift to generate new segments for a further classification. Vilariño et al. (2016) discuss a graph-based approach to segment point clouds in which each point instantiates a segment and segments are merged based on an analysis of within-segment and between-segment distances, similar to the graph-based image segmentation by Felzenszwalb and Huttenlocher (2004). Sima and Nüchter (2013) also include the differences in pulse reflection strengths in the calculation of distances between points when segmenting point clouds of indoor environments. As plane orientation is not included in the distance calculation, intersecting roof planes, but also adjacent roof and wall planes and adjacent wall and terrain surfaces may be merged to single segments.

2.2. Classification of ALS data

Techniques for the classification of ALS data can be characterised according to whether the classification is applied to the ALS points or to segments of ALS points, e.g. (Xu et al., 2012). Point-based classification has the advantage that a decision is taken for each point, so that the result will not be affected negatively by segmentation errors. On the other hand, as an individual

3D point does not carry much information, the definition of discriminative features typically involves an analysis of the point distribution in a local neighbourhood of each point, and the feature vectors thus determined may become unstable at the boundaries of objects, where the neighbourhood contains points from different object classes. For a thorough discussion of features for point-based classification, refer to (Weinmann et al. 2015a). The authors also discuss the selection of an appropriate size of the local neighbourhood for feature computation and come up with two methods for determining an individual optimal neighbourhood size for each individual point. Lin et al. (2014) try to overcome the problems related to neighbourhood selection by defining features based on a weighted co-variance matrix considering the local characteristics of the point cloud. Brodu and Lague (2012) argue that a set of feature values computed with multiple neighbourhood sizes better characterize a local point distribution. Nevertheless, the classification results can still be improved by tapping *context* as an additional information source.

One way to consider context in object detection is to apply heuristics such as those used in (Horvat et al., 2016) for the classification of vegetation in ALS data. However, a principled statistical approach leads to the application of probabilistic graphical models such as Conditional Random Fields (CRF; Kumar and Hebert, 2006). A review of methods for the classification of point clouds based on CRF can be found in (Niemeyer et al., 2014). The authors carry out a point-based classification of ALS data using a CRF. Thus, the nodes of the underlying graph correspond to the ALS points, whereas the edges (responsible for the context model) link each points to its nearest neighbours in 2D. The context model, represented by the interaction potentials of the CRF, is based on a generic classifier simultaneously predicting the pair of class labels of neighbouring points. Point-based classification based on CRF in general leads to a smoother result compared to an individual classification of points, the most obvious effect being a considerable improvement of the class-specific quality indices for classes having a small number of instances, e.g. (Niemeyer et al., 2014) and (Weinmann et al., 2015b).

A major problem of such approaches is the fact that in pairwise CRF, interactions only occur at a very local level, so that the result will still contain isolated clusters of points assigned to the wrong class. This is known as the problem of missing long-range interactions, and several strategies have been proposed for its solution. Luo and Sohn (2014) combine two CRF models, one considering a very local neighbourhood and the other considering long-range interactions, to classify terrestrial laser point clouds. The method is tailored to the specific structure of terrestrial point clouds, only considering long-range interactions in vertical directions. Other techniques try to incorporate segmentation results into a point-based classification. Xiong et al. (2011), rather than relying on a graphical model, apply a series of stacked classification processes, propagating point-based classification to segments and using the segment-based results to consider context in a final point-based classification. Kohli et al. (2009) propose the robust P^N Potts model for pixel-based classification using a CRF. Segmentation results are considered by higher-order-potentials encouraging all pixels inside a segment to take the same class label. This model is used for the contextual classification of ALS point clouds in (Niemeyer et al., 2016). The underlying segments are determined by a supervoxel segmentation (Papon et al., 2013). Niemeyer et al. (2016) use the output of the point-based classification for a second CRF-based classification in which the nodes of the graph correspond to segments that are generated by a procedure similar to region growing, but only combining points associated to the same class in the point-based results. In this way, interactions at object level should be considered. It has to be noted that the gain from such a procedure comes at the cost of a very complex inference procedure. Furthermore, the application of the P^N Potts model places restrictions the models for the local spatial interactions. Guo et al. (2015) carry out a point-based classification. After that, they determine clusters of points assigned to a specific object class and refine the classification of such objects using a foreground-background segmentation similar to (Golovinskiy and Funkhouser 2009), based on a graphical model. Such a procedure would be difficult to apply in a scenario as in ours where the classes may correspond to object parts such as walls, roofs and roof superstructures.

An alternative to point-based classification is the classification of segments. Segments may be seen as a natural neighbourhood for the definition of features, resulting in more stable features. In addition, as the number of segments is considerably smaller than the number of points, the classification becomes much faster, though this has to be traded against the time required for segmentation. Xu et al. (2012) combine the classification of segments obtained by two different algorithms with a point-based classification for isolated points. The classification itself is based on a set of heuristic rules. Segment-based classification can also be interpreted as a way of considering context, and by linking all points inside a segment the range of interactions can become relatively large. However, it may nevertheless be advantageous to incorporate models of context beyond the segment boundaries, which can, again, be done using pairwise CRF. This has been done by Shapovalov et al. (2010), who used a segmentation technique based on a tree decomposition of space for defining the segments. This leads to a strong over-segmentation in which the segments are of a similar size. In the resultant graph, each segment is represented by one single point (“medoid”), whereas the edges are based on the analysis of the k nearest neighbours (knn) among the medoids in 2D and in 3D. The authors use a naïve Bayes classifier for defining the interaction potentials, taking into account features describing the deviations of the surface normal of the segments and the geometrical arrangement of the medoids. It would seem that a more local definition of both, the neighbourhoods and the feature vectors related to the interaction potential could be able to capture the local arrangement of classes in a better way, in particular if the segment size can vary. Najafi et al. (2014) first carry out a point-based classification using a simple CRF-based technique and then use k-means clustering for separately segmenting the subsets of the point cloud assigned to a specific label. The corresponding segmentation results are merged and form the basis for a second, segment-based CRF which does not only consider a pairwise smoothing term, but also a higher-order potential linking segments in different vertical layers with the goal of a better representation of the 3D structure of the scene in areas with, e.g., overhanging trees. Again, it would seem that the combination of a point-based CRF with the segment-based one leads to a considerable computational burden. Pham et al. (2015) also propose a CRF based on a segmentation and involving higher-order cliques. Their method is designed for indoor scenes captured by RGB-D sensors, so that it is not applicable in our context.

Some techniques for segment-based point cloud classification do not apply a generic segmentation technique, but they rely on an initial foreground-background segmentation and then disregard the points assigned to the background, just assigning class labels to segments that correspond to local clusters of foreground points. For instance, Golovinskiy et al. (2009), interested in the detection of objects such as cars or street furniture in a combination of terrestrial and airborne laser scanner data, eliminate ground and building façade points based on simple heuristics. After detecting object locations in the remaining point cloud, a graph-based segmentation is applied for a fine discrimination of foreground and background points in the vicinity of objects. Clusters of foreground points are considered to be objects, and they are classified in a procedure consisting of two stages, where the results of the first stage

are used to define context features for the second one. It would seem to be difficult to apply such a strategy for a classification where the classes may correspond to object parts.

In this paper, we build on the observations made by Xu et al. (2012) that different segmentation techniques may be favourable for different object classes, combining two methods in a hierarchical procedure to obtain a generic segmentation technique for ALS data that is suitable for a large variety of urban object classes. This delivers the basis of a segment-based CRF. Due to the properties of the segmentation algorithm, we will get segments of very different extent, so that we suggest defining the edges of the CRF on the basis of local neighbourhoods of points at segment boundaries rather than on the configuration of medoids (Shapovalov et al., 2010), and we also propose a set of interaction features derived from a local analysis of the point clouds near the segment boundaries. We define simple heuristics for classifying very large segments corresponding to the ground or to standing water bodies, but unlike Golovinskiy et al. (2009) we include them in our CRF as fixed nodes because they carry valuable context information.

3. Segmentation of point clouds

The goal of the segmentation is to split the point cloud into as large as possible segments of points belonging to the same class. In this way we expect to obtain the maximum benefits of the segment-based classification, as discussed in the introduction. In the next two sections we first discuss results of segmentation using parametric and non-parametric approaches. We then develop the strategy to segment large point clouds by combining different approaches, following Vosselman (2013).

3.1. Segmentation into planar surfaces

Segmentation into planar surfaces often starts by using RANSAC or a 3D Hough transform for detecting planar patches in a local neighbourhood. The detected patches are then used as seeds for a surface growing procedure to determine the maximum extent of the planar surfaces (Vosselman, 2012). Fig. 1 shows the result of this segmentation approach to a point cloud of an urban scene. Both roof surfaces and walls are well captured in segments despite the large difference in point density. To obtain this result the local neighbourhood used for seed detection as well as surface growing was solely defined by the k (20) nearest neighbours without a constraint on the proximity of the neighbours. The road surface breaks up into nearly co-planar larger segments as the terrain is not exactly planar. The 2D shape of those segments can be rather arbitrary. A small change in a segmentation parameter could lead to a very different segmentation of the road surface. This could potentially also have an effect on the robustness of the features of the segment and those of the segment relationships that are used for the classification.

Obviously, vegetation cannot be properly captured by planar segments. Vegetation is mostly split into many small sets of points that by co-incidence satisfy the planarity constraint of the segmentation method. For a segment-based classification this is undesirable as the segment features will become very unreliable. For instance, a feature like the percentage of last echoes which is high on roofs and terrain, but relatively low in vegetation is less informative when segments in vegetation are very small.

3.2. Segmentation based on point features

An alternative to segmentations into surfaces would be to group points based on locally computed point features. Object classes like roof, wall, terrain, and water can be characterised by rather smooth

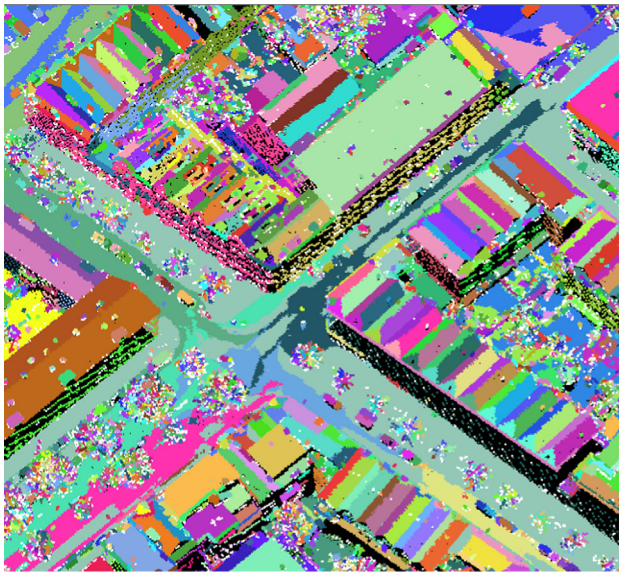


Fig. 1. Segmentation of a point cloud of an urban scene into planar segments using Hough transform combined with surface growing. Points shown in white could not be grouped to a segment of a minimum size (10 points) and did not obtain a segment label.

fields of locally computed normal vectors. At the transition of one class to another, there is usually a clear change in the normal vector direction or a large difference in height. By grouping nearby points with similar normal vector directions, points of the mentioned object classes are typically well segmented. Some over-segmentation occurs on roofs containing roof faces with different orientations, but under-segmentation is rare. Vegetation, however, is characterised by rather arbitrary directions of locally computed normal vectors. This is shown in Fig. 2(a) and (b), showing a hand-labelled point set with a wall, terrain, and trees (a) and the corresponding normal vector directions plotted on a Gaussian sphere (b). A clear difference between vegetation and the other classes is the planarity of local neighbourhoods. Combining normal vector directions with planarity yields a feature space in which points of different adjacent objects can usually be separated. This is visualized in Fig. 2(c) in which the normal vector directions have been multiplied with the local point cloud planarity. Vegetation points are now also well clustered.

These features were used to segment point clouds with two approaches: mean shift (Comaniciu and Meer, 2002) and segment growing (Rutzinger et al., 2008). Comaniciu and Meer (2002) suggest a joint representation for the spatial and (pixel) attribute domain. The balance between similarity in coordinates and similarity in attributes can be tuned by specifying a bandwidth for each domain. For the segmentation of the point cloud the mean shift algorithm did not lead to satisfactory results (Fig. 2(d)). Emphasizing the spatial domain leads to a strong over-segmentation, whereas emphasizing similarity in normal vector directions and planarity results in segments with patches of points that are spatially disconnected, though nearby in feature space. The latter effect then makes the computation of segment features as required for the classification less reliable.

Results with a segment growing algorithm are shown in Fig. 2 (e). As a segment will only extend to immediate neighbouring points with similar attributes (scaled normal vector), the connectivity of the segment points is inherent to the procedure. Some points in vegetation are shown in white. Those points were not included in surrounding segments because of a locally higher planarity value, which may arise from points on a few coplanar

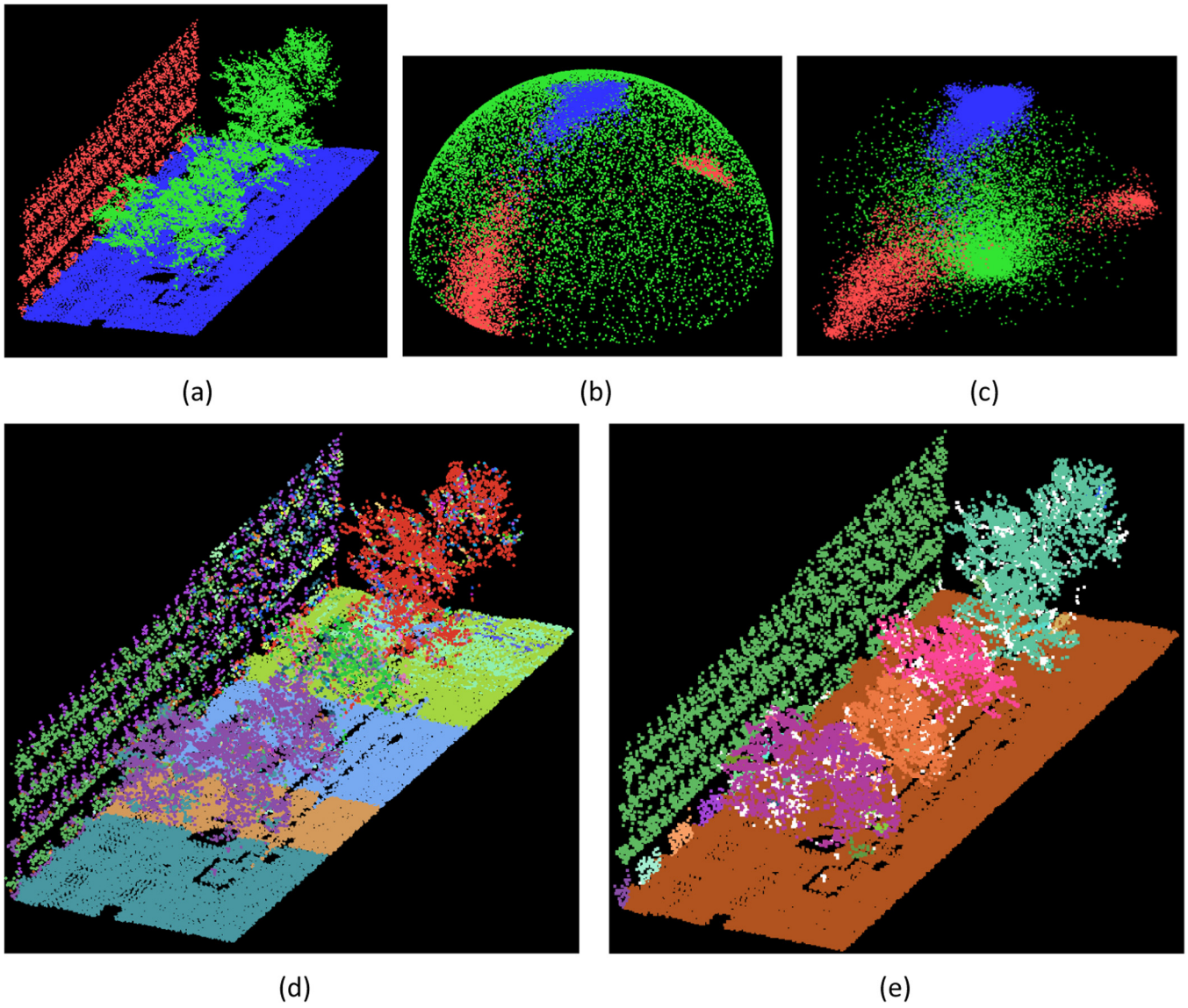


Fig. 2. Segmentations based on point features. (a) Hand-labelled scene with *terrain* (blue), *wall* (red) and *trees* (green). (b) Normal vectors plotted on a Gaussian sphere. (c) Normal vectors scaled by the local point cloud planarity. (d) Mean-shift segmentation based on normal vector directions and planarity. (e) Segment growing based on normal vector directions and planarity. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

branches. Both under- and over-segmentation were observed near the intersections of surfaces, like walls and terrain patches, where local normal vector directions are computed in neighbourhoods with points from two surfaces. This may lead to additional segments with points from two different classes.

3.3. Combining segmentation methods

Reviewing the above discussed methods both the extraction of planar surfaces and the segmentation based on point features clearly show advantages and disadvantages. The following procedure was designed to make use of the advantages and repair the disadvantages of both methods as far as possible.

1. Initially the point cloud is segmented into planar patches using a Hough transform for local seed detection followed by surface growing. Planar object pieces, like roofs and walls, are well extracted in this step.
2. As the first step segments vegetation into many small pieces (Fig. 1), only the larger planar segments are kept and segment labels are removed from segments below a certain size. All points without a segment label are re-segmented with the segment growing method based on the similarity in the feature space of normal vector directions and planarity (Fig. 2(c)). Problems with normal vectors at the intersection of otherwise well-defined planar patches are thereby circumvented, as these patches are usually larger than the segment size threshold. Most points on vegetation or on smaller objects extruding from a surface, like chimneys or cars, are grouped to single segments.
3. Slightly non-planar surfaces, like the ground surface in Fig. 1, break up into multiple larger planar segments. This is the reason why we merge adjacent segments if certain constraints are fulfilled: Two adjacent segments A and B are merged if the normal vector directions of the two planes are nearly parallel and, in a local neighbourhood along their common border, points in A fit well to the plane of segment B and vice versa.

4. In the last step points without a segment label are given the most frequent label of the points in their neighbourhood. Points remain unlabelled if there are no other points around them. This majority voting for unlabelled points effectively groups points in vegetation with dissimilar point feature values (white points in Fig. 2(e)).

Application of the above procedure onto the dataset used in Fig. 1 results in the segmentation of Fig. 3. The segmentation shows only little under- or over-segmentation. Some nearby trees or cars are grouped in a single segment. This could potentially impact the classification accuracy if shape descriptors are used as segment features.

To take full advantage of the context provided by the segments and their neighbouring segments it is required to make the segments as large as possible. Memory limits, however, restrict the size of point sets that can be segmented in one process. We therefore divide large datasets into tiles that can be processed in core memory. After the tile-wise segmentation all segments along either side of the tile borders are considered for merging using the same criteria as in step 3 described above. In this step it is determined which planar segments that were created in step 1 should be merged. In a similar procedure segments created in step 2 and situated along the tile boundary are compared based on similarity in point feature values. After identifying all segments that should be merged across tile boundaries, all segments are re-labelled. An example for nine tiles is shown in Fig. 4.

4. Context-based classification of point cloud segments

The goal of classification is to assign a unique label corresponding to an object class out of a set of predefined classes to each 3D point. In segment-based approaches, the entities to be classified are the segments, and the class label of a segment is transferred to all points belonging to the segment; points not belonging to any segment are marked as *not classified*. We apply a Conditional Random Field (CRF) for segment-based classification. We start the description of our classification methodology with a short review of the CRF framework in Section 4.1. In the subsequent sections, we will describe how this framework is applied in the context of the task at hand.

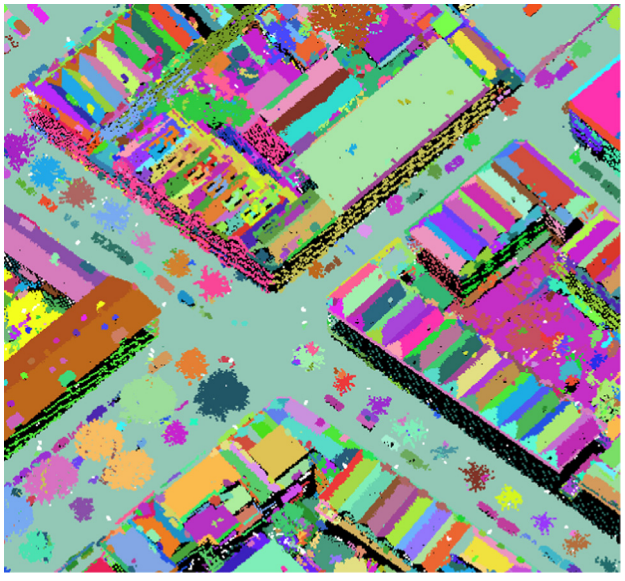


Fig. 3. Segmentation of a point cloud of an urban scene with the combination of surface growing and feature based segment growing, segment merging and majority voting.

4.1. Conditional Random Fields (CRF)

Conditional Random Fields (CRF) are undirected graphical models that provide a probabilistic framework for context-based classification. A CRF is defined on an underlying graph $G = (\boldsymbol{n}, \boldsymbol{e})$ consisting of a set of nodes $\boldsymbol{n}$ and a set of edges $\boldsymbol{e}$. Each node $i \in \boldsymbol{n}$ corresponds to a primitive to be classified, in our case to a point cloud segment. In the classification process, a class label C_i is to be determined for every node i. The class labels for all nodes are collected in a vector $\mathbf{C} = (C_1, C_2, \ldots, C_N)$, where N is the number of nodes in $\boldsymbol{n}$. Each edge $e_{ij} \in \boldsymbol{e}$ links a node $i \in \boldsymbol{n}$ with its neighbour $j \in \boldsymbol{n}$. Nodes linked by an edge are supposed to be statistically dependent, so that the edges represent the contextual relations in the classification process. An application of the CRF framework is free to define which nodes are to be linked by an edge. The goal of classification is to find the most probable configuration $\boldsymbol{C}$ of class labels given the observed point cloud, the latter represented by a data vector $\boldsymbol{x}$. That is, we have to determine the vector $\boldsymbol{C}$ that maximises the posterior probability $P(\boldsymbol{C}|\boldsymbol{x})$. In a CRF, this posterior is modelled according to Eq. (1) (Kumar and Hebert, 2006):

$$P(\boldsymbol{C}|\boldsymbol{x}) = \frac{1}{Z}\left(\prod_{i\in\boldsymbol{n}}\varphi(\boldsymbol{x}, C_i)\cdot\prod_{e_{ij}\in\boldsymbol{e}}\psi(\boldsymbol{x}, C_i, C_j)\right). \quad (1)$$

In Eq. (1), Z is a normalisation constant referred to as the partition function that does not depend on the class labels. The terms $\varphi(\boldsymbol{x}, C_i)$ are called the *association potentials*. They link the class label C_i of each node i to the observed data and can be based on any discriminative classifier with a probabilistic output. The terms $\psi(\boldsymbol{x}, C_i, C_j)$ are called the *interaction potentials*. They are responsible for the context model; again, arbitrary discriminative classifiers with a probabilistic output can be applied to model these potentials (Kumar and Hebert, 2006). CRF just provides a general statistical framework which can be adapted to specific classification tasks by defining specific graph structures, models for the potentials, and features. For our classification problem, these specific definitions are described in the subsequent sections.

4.2. Definition of the graph

As pointed out previously, we carry out a segment-based classification, and consequently, the graph nodes of our proposed method correspond to the 3D point cloud segments extracted in the way described in Section 3. The resulting class labels are transferred to the 3D points associated to the segments after classification. Unlike segmentation methods based on mean-shift (Melzer, 2007) or tree decomposition of space (Shapovalov et al., 2010), the method used here does not necessarily result in segments of a limited extent. In particular, in addition to many segments related to roof planes or small objects, it usually will also deliver a few very large segments that correspond to the ground and, where applicable, to large water bodies. It is very simple to identify these segments by their size as indicated by the number of points in a segment. These segments are assigned to a class before the CRF-based classification on the basis of simple heuristics related to size, average height, and average point spacing. If a segment contains more than a pre-defined number θ_{pg} of points and if its average height above ground is smaller than a threshold θ_{hg}, it is supposed to be a *ground* segment. If large *water* bodies are expected in the scene, a piece of information that has to be provided by the user, segments containing more than a pre-defined number θ_{pw} of points and having an average point spacing lower than a user-defined threshold θ_{sw} are classified as *water*. Average

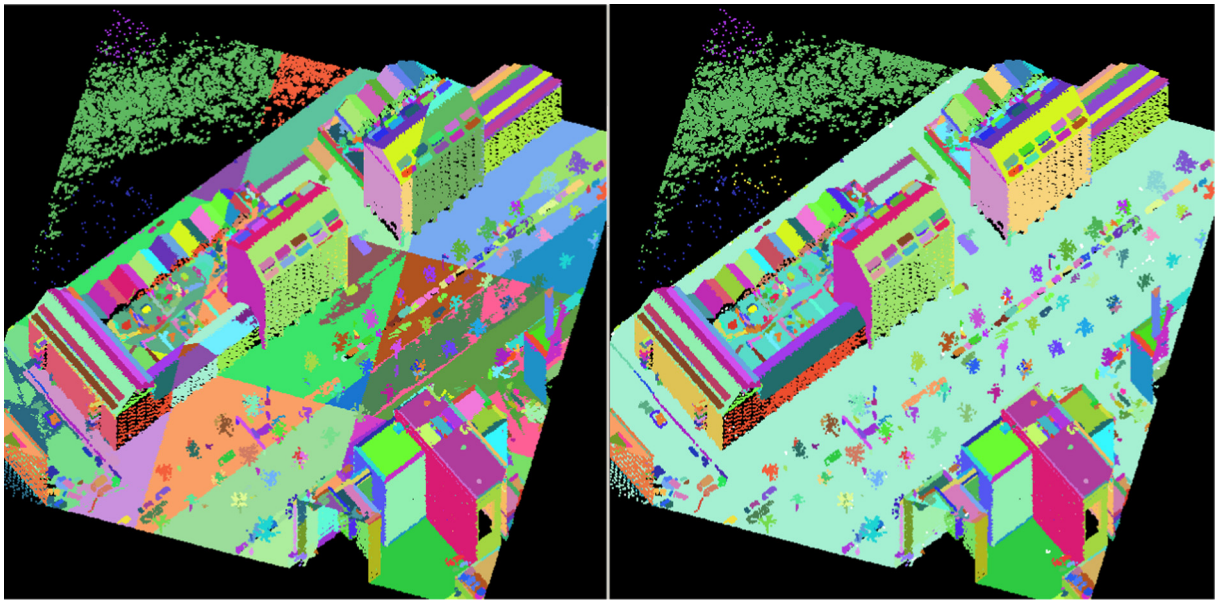


Fig. 4. (a) Tile-wise segmentation of nine tiles (subset of a dataset with 960 tiles). (b) Segmentation result after merging segments across tile boundaries.

point-spacing is used here because large water bodies usually contain patches with relatively low point density, so that this feature is better-suited for a separation than height. Note that this does not mean that all *ground* or *water* segments are pre-classified at this stage; there may remain small segments, e.g. corresponding to roads near the boundaries of an urban scene that are cut off the main road network or small standing water bodies. Whereas typically only a few segments (1–10, depending on factors such as scene size or the presence of large water bodies) will be pre-classified in this way, they may nevertheless contain a considerable portion of the entire point cloud. As these segments carry a lot of context information, they are included in the CRF, but as *fixed nodes*, i.e., their class labels are fixed as determined using the heuristics just described. Thus, unlike Golovinskiy et al. (2009), who eliminate *ground* points before classification, we still exploit this information in the classification process.

The definition of the edges of the CRF is based on a definition of neighbouring segments. Two segments i and j are considered neighbours if there is a point $\boldsymbol{p}_l \in j$ which is contained in the local point neighbourhood of a point $\boldsymbol{p}_m \in i$ or vice versa. The point neighbourhoods are defined as consisting of the 20 nearest neighbours in 2D. A definition of these point neighbourhoods in 2D is preferred to ensure that roof segments and segments of the surrounding terrain will be considered neighbouring. For all neighbouring segments (nodes) an edge e_{ij} connecting the two nodes will be inserted into the graph. If one of the two neighbouring nodes is a fixed node, the edge will be directed, leading from the fixed node to the unknown one and, thus, being responsible to transfer context information from the fixed (*ground* or *water*) node to its neighbour. Otherwise, the edge will be undirected. Of course, no edge is required between two fixed nodes in case they happen to be neighbours. The resulting graph structure is shown schematically in Fig. 5.

This definition of the graph structure has several advantages over a method based on a definition of neighbourhood that only considers one representative point per segment such as the medoid (Shapovalov et al., 2010). Firstly, defining segments to be neighbours by the distance between medoids only works well if all segments have a similar size. In case of a mix of large and small segments the medoids of two large adjacent segments will be too far apart for the segments to be considered neighbours. At the same time, two small segments separated by several other small segments in between may be considered to be neighbours. Furthermore, the large segments corresponding to the fixed nodes may have thousands of neighbouring segments; if the neighbourhood definition were based on one representative point, probably only a few segments in the scene centre would be connected to such a fixed node. Secondly, we consider the inclusion of these large segments as fixed nodes to be very useful. Their class labels can be determined with a very high certainty from the heuristics described above. If they were included as unknown nodes, each neighbour in the graph would have a small impact on the class label of such a node, and as a consequence, the computation of the overall impact could lead to numerical problems (cf. Section 4.5). On the other hand, the information that a segment is a neighbour of a large ground or water segment in combination with a classification of the related features may help to improve the classification of these segments. Finally, in the process of finding out which segments should be connected by an edge, for each edge, the points at the boundary of the two nodes involved are identified, which can be used for the definition of meaningful edge features for the determination of the interaction potentials (cf. Section 4.6.2). Note that for an efficient local neighbourhood analysis as required for our edge definition, a spatial index over the point cloud, e.g. a kd-tree, is required.

4.3. Association potential

The association potential $\varphi(\boldsymbol{x}, C_i)$ is related to the probabilistic output of a discriminative classifier. As in Kumar and Hebert (2006), the dependency of the potential for node i from all data $\boldsymbol{x}$ is considered by defining a site-specific feature vector $\boldsymbol{f}_i(\boldsymbol{x})$, whose components are (in principle) arbitrary functions of a subset of $\boldsymbol{x}$, for each node i (cf. Section 4.6.1 for a detailed definition of these node feature vectors). The association potentials are related to the posterior probability of a discriminative classifier according to $\varphi(\boldsymbol{x}, C_i) = P(C_i|\boldsymbol{f}_i(\boldsymbol{x}))$. In this work, *Random Forests* (RF) (Breiman, 2001) are used for this purpose as they are considered to be one of the best classifiers while being computationally efficient and directly applicable for multiclass problems, e.g. (Tokarczyk et al., 2012). A RF is a bootstrap ensemble classifier consisting of a number T of randomized decision trees. Each tree is trained indepen-

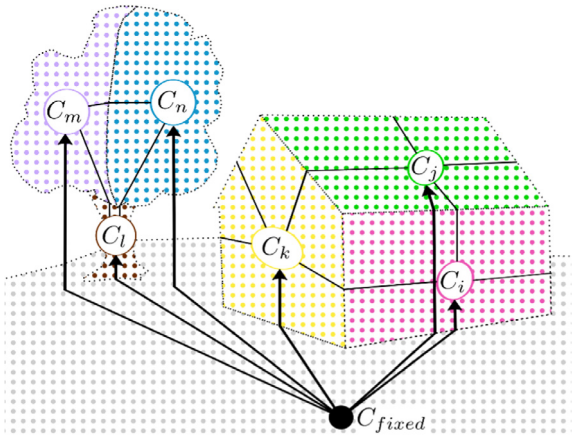


Fig. 5. Graph structure of the CRF (schematic). The nodes correspond to the point cloud segments, which are also indicated by the colours of the laser scanner points. The segment on the ground corresponds to a fixed node. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

dently on a randomly selected subset of the training data (bootstrap dataset). The most important parameters are the maximum depth d_{RF} of a tree in the forest and the minimum number of training samples $n_{s,min}$ required for a node to be split in the training process. In the classification process, the features of an unknown sample are presented to all trees in the RF, and each tree casts a vote for the class it considers to be the most likely one. Normally, the RF assigns the feature vector to the class receiving the largest number of votes. We use the number of votes N_l cast for a specific class C^l as the basis for defining the posterior probability:

$$P(C_i = C^l|\boldsymbol{f}_i(\boldsymbol{x})) = \frac{N_l}{T}. \quad (2)$$

The definition of Eq. (2) is used for all nodes (i.e., point cloud segments) whose class labels are to be determined by the CRF. In order to be able to apply standard inference algorithms for defining the optimal configuration of classes in the CRF (cf. Section 4.5), we also define an association potential for the *fixed nodes* according to $\varphi(\boldsymbol{x}, C_i = C^l) = \delta(C^l = C^{fixed})$, where $C^{fixed} \in \{ground, water\}$ is the class label the fixed node has been assigned to and $\delta(\cdot)$ is the Kronecker delta function delivering 1 if the argument is true and 0 otherwise. This definition is based on the assumption that the pre-defined class labels of these nodes are considered to be absolutely certain.

4.4. Interaction potential

In many applications of CRF, the interaction potentials are designed to achieve a data-dependent smoothing of the classification results, which is, for instance, achieved by the contrast-sensitive Potts model (Boykov and Jolly, 2001). Whereas such a definition also makes sense for the classification of point clouds if the nodes of the CRF correspond to individual points (Weinmann et al., 2015b), it is not the optimal choice in the case considered here, where the nodes are segments which do already correspond to objects or to larger object parts, so that a change of class labels between neighbouring segments is very likely. Fortunately, the CRF framework (Kumar and Hebert, 2006) allows for a very generic formulation of the interaction potentials on the basis of the output of another discriminative classifier, this time a classifier that predicts the local configuration of class labels (C_i, C_j) at two neighbouring nodes i and j given the data:

$$\psi(\boldsymbol{x}, C_i, C_j) = [P(C_i, C_j|\boldsymbol{g}_{ij}(\boldsymbol{x}))]^w. \quad (3)$$

In Eq. (3), w is a parameter modulating the relative impact of the interaction potentials on the classification output. The dependency of the potential from the data is considered by defining edge-specific feature vectors $\boldsymbol{g}_{ij}(\boldsymbol{x})$ (cf. Section 4.6.2) as the basis of classification. Such a model was used in (Shapovalov et al., 2010), using a naïve Bayes classifier and three features for that purpose. Niemeyer et al. (2014) applied such a model in the context of point-based classification, comparing two different models and concluding that a model based on RF had a slightly better performance than a linear model. Consequently, we also apply a RF for computing the posterior $P(C_i, C_j|\boldsymbol{g}_{ij}(\boldsymbol{x}))$ in this paper. Such a model allows learning that certain class relations are more likely than others based on the edge feature vectors. Note that for a classification problem discerning N_C classes, the classifier has to be able to differentiate any of the $N_C{}^2$ possible configurations of labels at neighbouring nodes.

4.5. Training and classification

In the training process, the parameters of the potentials according to Eqs. (2) and (3) have to be determined from annotated training data. In our case, this involves the supervised training of two RF classifiers, one for each type of potential. In this work, these classifiers are trained independently from each other. The weight parameter w could be determined by a procedure such as cross validation, e.g., (Shotton et al., 2009), but here it is set to a value that was determined empirically. Fully labelled point clouds are required for training. In order to train the RF classifier for the association potentials, the point cloud is segmented using the algorithm described in Section 3. After computing the node features (cf. Section 4.6.1), the class label of each training segment is determined by a majority vote of the reference class labels of the points assigned to that segment. In order to avoid unbalanced training data sets, the same number n_s of training samples is drawn randomly for each class to be discerned. For the association potentials, a training sample corresponds to a node feature vector $\boldsymbol{f}_i(\boldsymbol{x})$ (cf. Section 4.6.1) and the according class label. For the interaction potential, we analyse the segmented point clouds for finding pairs of neighbouring segments according to the definition given in Section 4.2; each pair of neighbouring segments i and j can be used to define a feature vector $\boldsymbol{g}_{ij}(\boldsymbol{x})$ and a vector $\boldsymbol{g}_{ji}(\boldsymbol{x})$ (cf. Section 4.6.2) and results in two training samples for the RF classifier, namely (C_i, C_j, $\boldsymbol{g}_{ij}(\boldsymbol{x})$) and ($C_j$, C_i, $\boldsymbol{g}_{ji}(\boldsymbol{x})$).

Given the model for the posterior $P(\boldsymbol{C}|\boldsymbol{x})$ according to Eq. (1) and the parameters of the potentials according to Eqs. (2) and (3) as determined in the training phase, it is the goal of optimisation to determine the label configuration $\boldsymbol{C}_{opt}$ for which $P(\boldsymbol{C}|\boldsymbol{x})$ becomes a maximum. For multi-class problems, exact optimisation is computationally intractable, so that approximate algorithms have to be applied. We apply the max-sum variant of loopy belief propagation (LBP) (Frey and MacKay, 1998; Szeliski et al., 2008), an iterative message passing algorithm that can be applied to CRF with arbitrary formulations of the interaction potentials. This is important for our application, because for the best alternative, algorithms based on graph-cuts, the interaction potentials have to fulfil certain constraints (Szeliski et al., 2008) that cannot be guaranteed by the model according to Eq. (3).

4.6. Features

4.6.1. Node features

The node feature vectors $\boldsymbol{f}_i(\boldsymbol{x})$ are determined on the basis of all the points $\boldsymbol{p}_{ik}$ belonging to segment i. Several features are based on the covariance matrices $\mathbf{M}_i$ of all points inside the segment i or of the points $\boldsymbol{p}_{ik}$ in a local neighbourhood of a point $\boldsymbol{p}_i$:

$$\mathbf{M}_i = \sum_k (\boldsymbol{p}_{ik} - \boldsymbol{p}_C) \cdot (\boldsymbol{p}_{ik} - \boldsymbol{p}_C)^{\mathrm{T}}, \quad (4)$$

where $\boldsymbol{p}_C$ is the centre of gravity of the points under consideration. Note that the eigenvector corresponding to the smallest eigenvalue of $\mathbf{M}$ is the normal vector of a plane fitted to the points $\boldsymbol{p}_n$ used to compute $\mathbf{M}$.

In this work we use 19 features for each segment i, which constitute the node feature vector $\boldsymbol{f}_i(\boldsymbol{x})$:

(1) Segment size s_i: The number of points in the segment i
(2) Average point spacing dp_i: A (2D) Delaunay triangulation is computed of all points in a segment, and dp_i is defined as the average length of a TIN edge between two points contained in segment i. Edges connected to the segment's convex hull (that may bridge concavities) are not used in the computation of the average point spacing. When a segment is covering multiple tiles, TINs are calculated for the segment parts in every tile separately and the average is computed over all interior TIN edges in all tiles. This feature is not only useful to separate ground from water surfaces, but also helps to distinguish between roof surfaces and larger sets of coplanar points in vegetation (Xu et al., 2012). In the latter case the point spacing within the segment is much larger, because many surrounding points are not located in the plane of the segment.
(3) Percentage np_i of points in segment i that are situated in the vicinity of neighbouring segments according to the neighbourhood criterion described in Section 4.2.
(4) Features derived from the height above ground: Average, minimum and maximum heights above ground $h_{avg,i}$, $h_{min,i}$, $h_{max,i}$ of the points in segment i. LAStools[1] was used to classify points into ground and non-ground. The surface fitted to the ground points is used to determine the height above the ground for every point.
(5) Features derived from the covariance matrix $\mathbf{M}_i$ of all points belonging to segment i:
  (a) Original eigenvalues $\lambda_{0,i}$, $\lambda_{1,i}$, $\lambda_{2,i}$ of $\mathbf{M}_i$
  (b) Scaled eigenvalues $\lambda_{0s,i}$, $\lambda_{1s,i}$, $\lambda_{2s,i}$, derived by dividing the original eigenvalues $\lambda_{0,i}$, $\lambda_{1,i}$, $\lambda_{2,i}$ by their sum. The scaled eigenvalues give the relative size of the eigenvalues and are expected to characterize the geometrical arrangement of the points inside a segment (linear/planar/volumetric). The original eigenvalues contain this information as well, but they are also affected by the geometrical extents of a segment. As there is no linear relationship between the vector of all eigenvalues and the vector of all scaled eigenvalues, the scaled values might add useful information to the classification process.
  (c) Inclination angle α_i of the adjusting plane passing through all points of segment i.
(6) Features derived from local covariance matrices $\mathbf{M}_{ik}$. To derive these features, such a matrix is computed for each point $\boldsymbol{p}_{ik}$ of the segment i, taking into account a local neighbourhood.
  (a) Average scaled eigenvalues $\lambda_{0ls,i}$, $\lambda_{1ls,i}$, $\lambda_{2ls,i}$ obtained by dividing the eigenvalues of the local covariance matrices $\mathbf{M}_{ik}$ by their sum.
  (b) Average local flatness f_i and linearity l_i (Gross and Thoennessen, 2006), computed on the basis of the eigenvalues of $\mathbf{M}_{ik}$. In literature, flatness, or planarity, is often defined as $(\lambda_2 - \lambda_3)/\lambda_1$. This, however, leads to low flatness values in case of points on elongated walls or fences for which $\lambda_1 \gg \lambda_2$. Normalisation by λ_2 is therefore preferred.
  (c) Variance of local slope angles $v_{\alpha i}$: Each local covariance matrix is used to derive a slope value; $v_{\alpha i}$ is the variance of these slope values.

All the features are scaled linearly into the interval between 0 and 1, truncating the feature values by the β and $1-\beta$ percentiles of the feature distribution to mitigate the impact of outliers (we used $\beta = 0.5\%$ in our experiments).

4.6.2. Edge features

The edge feature vectors $\boldsymbol{g}_{ij}(\boldsymbol{x})$ used in this work have three different components. First, we follow Niemeyer et al. (2014), who, in the context of point-based classification, use the concatenation of the node feature vectors $\boldsymbol{f}_i(\boldsymbol{x})$ and $\boldsymbol{f}_j(\boldsymbol{x})$ of the nodes i and j connected by the edge e_{ij} in the graphical model. In addition, we use 12 handcrafted features derived from the points at the boundary between the segments i and j. These features are collected in a vector $\boldsymbol{\mu}_{ij}(\boldsymbol{x})$, so that the overall edge feature vector becomes $\boldsymbol{g}_{ij}(\boldsymbol{x}) = [\boldsymbol{f}_i^T(\boldsymbol{x}) \boldsymbol{f}_j^T(\boldsymbol{x}) \boldsymbol{\mu}_{ij}{}^T(\boldsymbol{x})]^T$. The twelve edge features collected in $\boldsymbol{\mu}_{ij}(\boldsymbol{x})$ are as follows:

(1) The number n_{ij} of points along the common border of the two neighbouring segments according to the neighbourhood definition in Section 4.2.
(2) The average height difference Δh_{ij} along the common segment border. It is defined as the average of the height differences between all pairs of points $\boldsymbol{p}_m \in i$ and $\boldsymbol{p}_l \in j$ that are within each other's local point neighbourhood.
(3) Average angle α_{ij} between local normal vectors. For computing α_{ij}, angles between the normal vectors of points $\boldsymbol{p}_m \in i$ and $\boldsymbol{p}_l \in j$ are computed when these points are within each other's local point neighbourhood.
(4) Features derived from the covariance matrix $\mathbf{M}_{ij}$ of all points along the segment border:
  (a) Original eigenvalues $\lambda_{0,ij}$, $\lambda_{1,ij}$, $\lambda_{2,ij}$ of $\mathbf{M}_{ij}$
  (b) Scaled eigenvalues $\lambda_{0s,ij}$, $\lambda_{1s,ij}$, $\lambda_{2s,ij}$, derived by dividing the original eigenvalues $\lambda_{0,i}$, $\lambda_{1,i}$, $\lambda_{2,i}$ by their sum.
(5) Features derived from local covariance matrices $\mathbf{M}_{ijk}$ of all points along the segment border
  (a) Average local eigenvalues $\lambda_{0l,ij}$, $\lambda_{1l,ij}$, $\lambda_{2l,ij}$ of the local covariance matrices $\mathbf{M}_{ijk}$
  (b) Scaled local eigenvalues $\lambda_{0ls,ij}$, $\lambda_{1ls,ij}$, $\lambda_{2ls,ij}$, derived by dividing $\lambda_{0l,ij}$, $\lambda_{1l,ij}$, $\lambda_{2l,ij}$ by their sum.

The elements of $\boldsymbol{\mu}_{ij}(\boldsymbol{x})$ are scaled linearly into the interval between 0 and 1 similarly to the node features. Note that according to this definition, $\boldsymbol{\mu}_{ij}(\boldsymbol{x})$ and $\boldsymbol{\mu}_{ji}(\boldsymbol{x})$ only differ by the signs of Δh_{ij} and Δh_{ji}. The feature vector $\boldsymbol{g}_{ji}(\boldsymbol{x})$ used for training becomes $\boldsymbol{g}_{ji}(\boldsymbol{x}) = [\boldsymbol{f}_j^T(\boldsymbol{x}) \boldsymbol{f}_i^T(\boldsymbol{x}) \boldsymbol{\mu}_{ji}^T(\boldsymbol{x})]^T$.

This definition of the edge feature vectors combines features related to the properties of the nodes connected by an edge as well as to the properties of the points at the common boundary of these nodes.

5. Experiments

5.1. Test data and test setup

5.1.1. Test data

For the evaluation of our approach we conduct experiments using two data sets. The first one is the LiDAR data set of Vaihingen, Germany, which is also part of the ISPRS 3D labelling benchmark.[2] The ALS data were acquired in August 2008 by a Leica ALS50 system with 45° field of view and a mean flying height above ground of

[1] https://rapidlasso.com/lastools/ (accessed 13/12/2016).

[2] http://www2.isprs.org/commissions/comm3/wg4/3d-semantic-labelling.html (accessed 13/12/2016).

Table 1
Class distribution (number of points per class) in the reference of the Vaihingen test data sets.

	Area 1	[%]	Area 2	[%]	Area 3	[%]	All areas	[%]
ground	94,494	(51.7)	139,573	(51.9)	191,966	(58.4)	426,033	(54.6)
building	59,999	(32.8)	52,742	(19.6)	87,118	(26.5)	199,859	(25.6)
trees	24,480	(13.4)	71,319	(26.5)	41,550	(12.6)	137,349	(17.6)
other	3944	(2.2)	5424	(2.0)	8270	(2.5)	17,638	(2.3)
Sum	182,917		269,058		328,904		780,879	

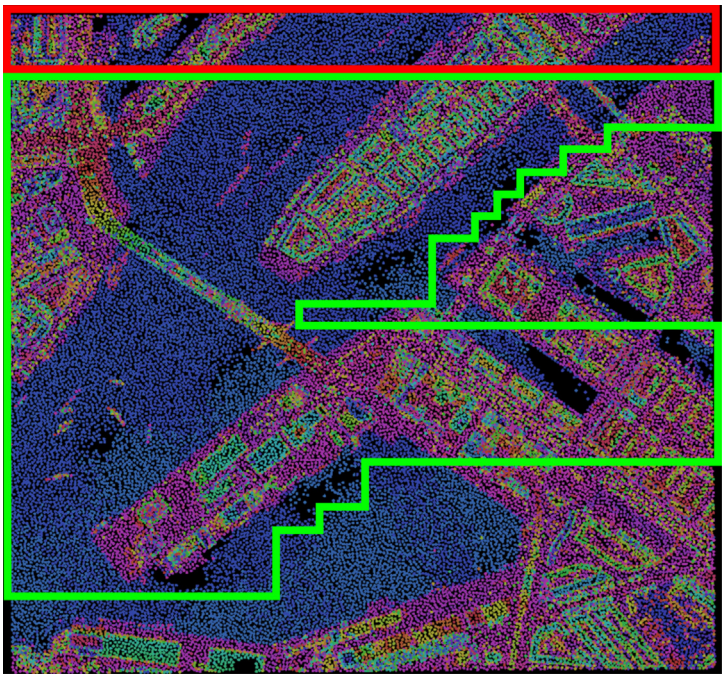


Fig. 6. Top view of the Rotterdam data set. Area with available reference labels is framed. (Red box: training area, green polygon: test area). The colours are related to height. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

500 m (Haala et al., 2010). The point density is between 4 and 7 points/m^2. For the classification task, five areas, labelled manually by Niemeyer et al. (2014), are available: two training sites and three test sites with different scenes. Test area 1 is situated in an urban area of dense development consisting of historic buildings with complex shapes, roads and trees. Area 2 is characterised by high-rising residential buildings that are surrounded by trees, and area 3 is a purely residential neighbourhood with small detached houses and many surrounding trees. As we use the same set of features and because the second data set does not contain LiDAR intensity data, we restrict ourselves to the use of the point co-ordinates only. Consequently, it is not feasible to discern all the classes differentiated in Niemeyer et al. (2014); we restrict ourselves to four classes, namely *ground*, *building*, *tree* and *other*. The distribution of class labels in the reference of the Vaihingen test data is shown in Table 1. The two training areas (not shown in Table 1) consist of altogether 428,182 labelled points.

The second data set is a LiDAR point cloud which covers an area of approximately 2.2 × 2.2 km^2 of the city centre of Rotterdam, The Netherlands. The data were acquired with the FLI-MAP laser scanner in April 2010. The survey was designed to capture at least 30 points/m^2. Here, seven classes, namely *ground*, *water*, *tree*, *building roof*, *roof structure*, *building facade* and *other*, are distinguished. Fig. 6 shows a top view on the Rotterdam data set. The red and green frames indicate areas for which manually assigned reference labels are available. An area of 210 × 2200 m^2 in the northern part of the point cloud (red box in Fig. 6) was used for training (14.6% of the labelled data), whereas the remaining points (those inside the green polygon in Fig. 6) are used for evaluation. The point distribution of the available reference data of the Rotterdam test set is shown in Table 2. Note that many laser pulses were absorbed by water surfaces. This explains the low percentage of points in the class water compared to the relative large area covered by water.

5.1.2. Parameter settings

Several parameters need to be set for the segmentation procedure discussed in Section 3.3. For the surface growing in the first step seeds were considered detected if at least 10 out of the 20 points in a local 3D neighbourhood were within 15 cm of a plane extracted with the 3D Hough transform. The seed segments were extended to other points located within 2 m distance and within 15 cm to the fitted plane. In the second step segment labels were removed from small segments and the resulting unsegmented points were re-segmented based on the normal vector directions scaled with the local planarity. Points within a distance of 0.4 in this attribute space were considered similar. In the experiments of the Vaihingen data three different settings were used to analyse (1) the benefits of the combined use of multiple segmentation methods and (2) the sensitivity of the results to the maximum size of the planar segments to be re-segmented. In the first variant,

Table 2
Class distribution of the reference (points per class) of the Rotterdam training and test data.

Class	*ground*	*water*	*roof*	*tree*	*other*	*roof structure*	*façade*	Sum
Points (training)	3,086,036	735,921	919,396	490,514	533,449	65,132	268,968	6,099,416
[%]	50.6	12.1	15.1	8.0	8.8	1.1	4.4	100.0
Points (testing)	18,210,744	3,795,446	6,996,222	1,190,184	2,114,423	1,043,501	2,269,551	35,620,071
[%]	51.1	10.7	19.6	3.3	5.9	2.9	6.4	100.0

referred to as variant *all_planar*, we segmented the point cloud into planar segments only. Here we only used the first step of the method described in Section 3.3. For the other two variants, re-segmentation of small planar segments was carried out, using different values for the maximum size of planar segments to be re-segmented, namely 40 in variant *planar_40* and 100 in variant *planar_100*. For the segmentation of the Rotterdam data we chose the parameter setting *planar_40*. Nearly parallel segments were merged if their normal vectors were within 5 deg and at least 5 points of one segment (with points of the other segment among their 20 nearest neighbours) were within a perpendicular distance of 0.15 m to the plane of the other segment. In the final majority voting step points that were still not assigned to a segment were only considered if there were segmented points within a 1 m radius.

The two independent RF classifiers used to model the unary and the pairwise potentials of the CRF, respectively, were trained separately. The parameters for the training of the RF were found empirically and are summarized in Table 3. The value for the number of training samples to be used per class n_s was set depending on the amount of available training samples for the nodes and edges, respectively. We selected the value of the number of samples of the most common class (the most common class combination in case of the binary potentials) in the training data. The resulting values for the number of training samples are shown in Table 4. The parameter w of the interaction potential was set to 0.7, a value which was found empirically.

The thresholds for segments to be considered as fixed *ground* or *water* segments, respectively, are defined as follows: For the Vaihingen data we choose the size threshold $\theta_{pg} = 25{,}000$ and the height threshold $\theta_{hg} = 0.1$ m; no *water* segments are considered in this case. In the case of the Rotterdam data, we use $\theta_{pg} = 335{,}000$ points and $\theta_{hg} = 0.5$ m for the definition of *ground* segments; the thresholds for defining fixed *water* segments are $\theta_{pw} = 100{,}000$ points and $\theta_{sw} = 0.3$ m.

Table 5 shows the statistics of the segmentation results for the three variants applied to the Vaihingen data and for the Rotterdam data as well as the statistics for the heuristic classification of the large *ground* and *water* segments. The table also presents the number of points that are not assigned to any segment and, thus, will not be assigned to any class in the segment-based classification procedure. In most cases, this number is in the order of 1% of the data or even less, but in case of variant *all_planar* in Vaihingen, it is as high as 6.8%, indicating that a purely planar segmentation may be too restrictive for capturing urban scenes completely. The row labelled *PB vs. SB* shows the correspondence of the point-based reference to the majority labels of the segments transferred to the 3D points in [%]. These values can be interpreted to represent the upper limit for the point-based overall accuracy that can be achieved by a segment-based classification. That is, even if the segment-based classification worked perfectly (Overall Accuracy 100%), a small percentage of the points would be classified incorrectly due to small segmentation failures (on top of the points not classified at all due to being isolated and not assigned to a segment). Additionally, those values deliver an indication for how good the segmentation preserves the object class boundaries. A comparison of the values achieved for the setting *planar_40* in Vaihingen and Rotterdam clearly shows that in Rotterdam a smaller number of points is affected by segmentation failures (2.1% vs. 4.6% in Vaihingen), This is probably due to the higher point density in the Rotterdam data, which allows for a better segmentation of smaller *roof*, *wall*, and *ground* patches. There are only a few fixed *ground* and (in case of Rotterdam) *water* segments, which, nevertheless, contain about 36% and 49% of the points for Vaihingen and Rotterdam, respectively. Note that the class assignment of all fixed segments is correct, though the segments may contain a few points belonging to other classes at their boundaries.

5.1.3. Test setup

Using the data sets described in Section 5.1.1, we carry out a set of experiments to evaluate our method. In the classification, our method assigns an object class to each segment, and the class label of each segment is transferred to the related 3D points. Thus, for the evaluation we compare the results to the reference for the seg-

Table 3
Parameters chosen for the RF classifiers for both potentials and both test data sets.

	Vaihingen		Rotterdam	
	Unary Potential	Binary Potential	Unary Potential	Binary potential
T	150	150	180	100
d_{RF}	15	20	15	20
$n_{s,min}$	5	5	5	10

Table 4
Number of samples n_s chosen for the RF classifiers for both potentials in our tests. In Vaihingen, the number of samples depended on the parameter settings for segmentation (see main text for the meaning of the three variants).

	Vaihingen						Rotterdam	
	Unary potential			Binary potential			Unary potential	Binary potential
	All_planar	Planar_40	Planar_100	All_planar	Planar_40	Planar_100	Planar_40	Planar_40
n_s	1900	600	500	19,000	2800	1650	3100	15,000

Table 5
Statistics about the numbers of segments and fixed points. *PB vs. SB*: Overall accuracy of the segment-based reference compared to the point-based reference.

Test data set	Vaihingen			Rotterdam
Segmentation	*all_planar*	*planar_40*	*planar_100*	*planar_40*
Number of segments	8340	3142	1785	41,951
Number of isolated points	53,330	5543	14,802	62,244
PB vs. SB [%]	98.0	95.4	92.9	97.9
Number of fixed segments (*ground*)	0	4	4	2
Number of fixed segments (*water*)	0	0	0	6
Number of fixed points (*ground*)	0	283,223	282,138	16,959,198
Number of fixed points (*water*)	0	0	0	3,475,395

Table 6
Comparison of the overall accuracy (SB) achieved by our classification with and without considering the interaction potentials. The numbers in parentheses are the improvements in OA compared to classification without context.

Test data set	Vaihingen			Rotterdam
Segmentation	*all_planar*	*planar_40*	*planar_100*	
No context	72.9%	65.3%	67.2%	59.8%
With context	82.1% (+9.2%)	73.9% (+8.6%)	76.0% (+8.8%)	68.8% (+9.0%)

ments (whose reference labels are determined by a majority vote of the points contained in it) and to the reference of the individual points. We refer to these evaluation methods as segment-based (SB) vs. point-based (PB) evaluation. In both cases, we also include the *ground* and *water* segments that were considered to be fixed. All of these segments were classified correctly by the heuristics described above, but as the number of such segments is small, their impact on the SB evaluation is small. However, they contain a considerable part of the points, so that the impact of these segments on the PB evaluation is relatively large. The comparison of the classification results to the reference delivers a confusion matrix, from which we determine the overall accuracy (*OA*) as well as *completeness*, *correctness* and *quality* per class (Heipke et al., 1997). The segmentation process will result in a certain number of points that cannot be assigned to any segment and which, consequently, will not be classified at all. In order to obtain a fair comparison of different methods and/or different variants of our own method, we also report the number of such points for each experiment.

As mentioned above, edges between two segments are defined if points of one segment were among the 20 nearest points of the other segment. We also considered to define any two segments as neighbouring based on an analysis of segment labels in fixed size neighbourhoods of 1 and 2 m radius. This increased the number of edges by 10–20%, but did not have a significant effect on the classification accuracy. Results are therefore only reported for the first mentioned definition of neighbouring segments.

Using the above evaluation protocol, we apply our methodology both with and without considering the interaction potentials in order to assess the impact of the context model; the variant not considering context is easily derived by setting the parameter w of the interaction potential to zero. For the Vaihingen data set, we also compare the results achieved for different settings of the segmentation parameters. Finally, we compare our method to a point-based classification technique. For that purpose, we adapt the first stage of the two-stage classification approach of Niemeyer et al. (2015), restricting the point-based feature vectors to features that can be obtained from the 3D information only to allow for a fair comparison to our technique.

5.2. Results and discussion

The classification results for the three test areas of the Vaihingen data set are combined to one single value for the OA. Table 6 shows the SB classification results of Vaihingen for the three applied variants of segmentation as well as the results obtained for the Rotterdam data. Table 7 contains the PB evaluation results. Rather than OA, we report the number of all points (including the isolated ones) assigned to the correct class. This implies that the percentages of points assigned to the correct class, points assigned to a wrong class, and isolated points have to sum to 100%.

5.2.1. Vaihingen

Fig. 7 shows the three variants of the segmentation for a part of Area 3 of the Vaihingen dataset. Vegetation is clearly split into many small segments in variant *all_planar*. The differences between variants *planar_40* and *planar_100* can be seen in vegeta-

Table 7
Number of points assigned to the correct class, to a wrong class or to no class at all. The numbers in parentheses are the improvements compared to classification without context. PB: Point-based classification.

Test area	Values in [%]	Isolated points	Without context		With context	
			Correct class	Wrong class	Correct class	Wrong class
Vaihingen	*all_planar*	6.8	83.1	10.1	86.3 (+3.2)	6.9
	planar_40	0.7	84.1	15.2	88.3 (+4.2)	11.0
	planar_100	1.9	84.2	13.9	86.8 (+2.6)	11.3
	PB	–	–	–	86.8	13.2
Rotterdam	Our method	0.2	90.3	9.5	91.0 (+0.7)	8.8
	PB	–	–	–	82.8	17.2

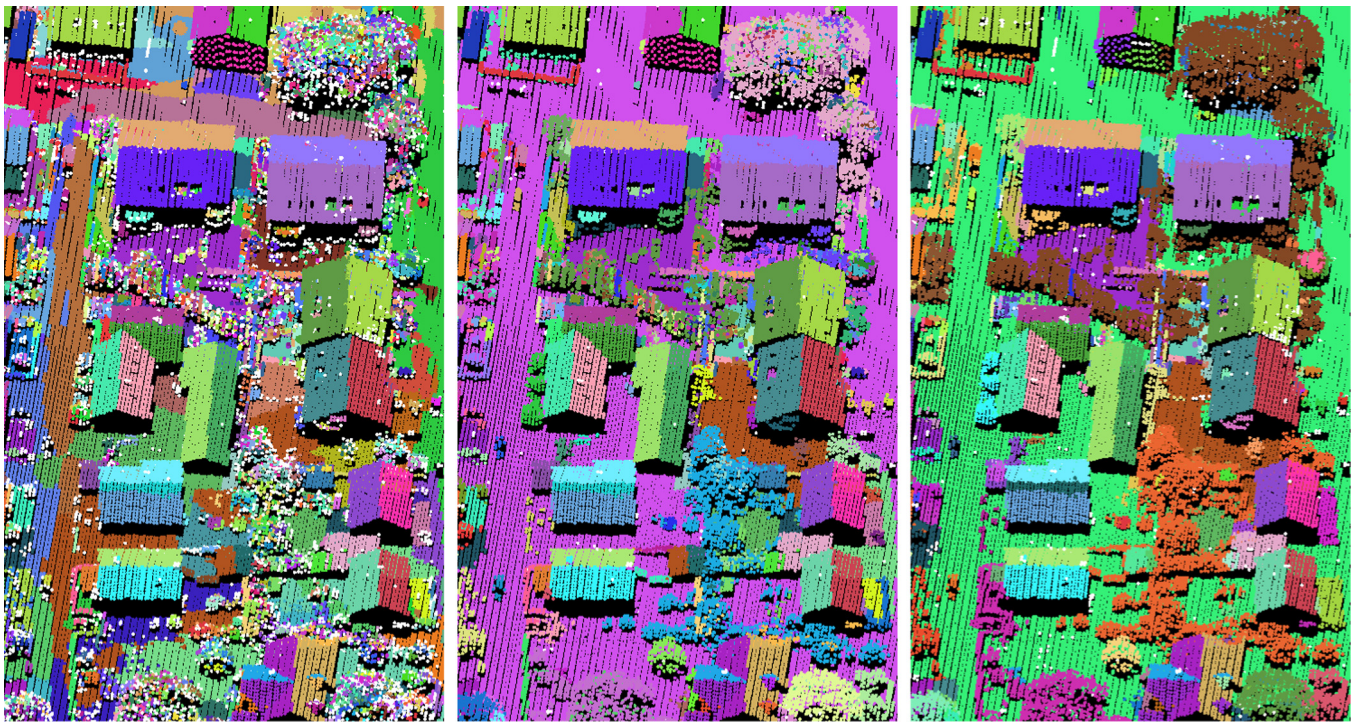


Fig. 7. Segmentations of a part of Area 3 of the Vaihingen dataset in the variants (a) *all_planar*, (b) *planar_40*, and (c) *planar_100*.

tion where more planar segments are found in the *planar_40* variant, and in planar structures with in between 40 and 100 points that are better segmented in the *planar_40* variant. Examples of the latter differences are visible in the wall of the gable roof building at the top and in the fences in the lower left corner.

Table 5 shows that the three variants lead to distinctly different numbers of segments, which consequently also differ in their size. The segmentation variant leading to most segments is the *all_planar* segmentation with 8340 segments, which is about 2.7 and 4.7 times more than resulting from the *planar_40* and the *planar_100* segmentations, respectively. Note that in the case of the *all_planar* segmentation, no *fixed nodes* can be defined, because this setting results in very small segments. Variant *all_planar* delivers the best overall accuracy in terms of the SB evaluation (82.1%), compared to 73.9% and 76.0% for the other variants. In all cases, the classification considering context information leads to distinct improvements in OA up to 9.2%. However, the comparison of the results on the basis of the point-based reference (Table 7) puts the advantage of variant *all_planar* into perspective: here, the three variants lead to very similar percentages of correctly classified points, with a slight advantage of 2% for variant *planar_40* (88.3%). This variant also leads to the most complete segmentation, with only 0.7% of unsegmented points. The segmentation with the highest number of isolated points is *all_planar* (6.8%). Variant *planar_40* is also the one achieving the largest improvement in terms of the number of correctly classified points due to context (4.2%). This improvement is only about half of the improvement of OA in the SB evaluation, which indicates that the consideration of context in the classification mainly improves the classification accuracy of small segments which contain a relatively small number of points.

Table 7 also shows the results of the point-based classification. It becomes obvious that its results are comparable to those

Table 8
Comparison of the computation time for contextual segment-based and point-based classification of the Vaihingen data set on a standard desktop computer.

	Our approach	Point-based classification (Niemeyer et al., 2015)
Segmentation	6:30 min	–
Feature calculation	7:30 min	2:30 min
Training	1:20 min	2:00 min
Classification	0:40 min	13:40 min
Sum	16:00 min	18:10 min

Table 9
Confusion matrices (SB) of the classification of the segments from variant *planar_40* [%].

[%]	Without context						With context					
	R/C	*grd*	*bld*	*tree*	*other*	Comp.	R/C	*grd*	*bld*	*tree*	*other*	Comp.
27.7	*grd*	18.7	4.2	3.2	1.6	67.4	*grd*	19.9	4.5	3.0	0.4	71.5
34.0	*bld*	1.5	29.1	3.0	0.4	85.7	*bld*	1.4	29.9	2.4	0.2	88.3
30.7	*tree*	3.4	10.8	16.0	0.5	52.1	*tree*	3.3	4.1	23.2	0.1	75.5
7.6	*other*	3.9	1.7	0.5	1.5	19.4	*other*	4.3	2.0	0.5	0.8	11.0
	Corr.	68.3	63.4	70.4	37.1		Corr.	68.9	73.8	79.8	56.5	
	Qual.	51.3	57.4	42.8	14.6		Qual.	54.1	67.2	63.4	10.1	
	Overall accuracy					65.3	Overall accuracy					73.9

achieved for the *all_planar* and the *planar_100* segmentation variants, but 1.5% inferior to our best segment-based classification results (*planar_40*). Table 8 shows the approximate computation times for the individual steps of segment-based and point-based classification. The table indicates that, apart from delivering slightly better results, the segment-based classification is also somewhat faster than the point-based approach.

Tables 9 and 10 show the SB and PB confusion matrices, respectively, resulting from the classification of the segments of variant *planar_40*. The leftmost columns show the percentage of SB (Table 9) and PB (Table 10) reference data per class. Table 9 shows that the classification without context shows the highest confusions between *tree* segments and *building* segments, followed by misclassifications of *ground* segments as *building* and *tree* segments. The class *tree* constitutes 30.7% of all segments. In the classification without context, nearly one third of those segments, namely 10.8%, are erroneously classified as *building*. The consideration of context distinctly reduces these misclassifications to 4.1% and improves the amount of correctly classified *tree* segments by 7.2%. This accounts for the major part of the improvement in OA of 8.6%. Concerning the classification of the class *ground*, the incorporation of context in the classification only leads to slight improvements. The worst results, comparing the values for the quality, are achieved for the class *other*. Here, the contextual classification even decreases the quality while the values for the remaining classes increase compared to the classification without context. Similar observations can be made regarding the PB evaluation in Table 11, with the difference that the values for the mentioned confusion are less distinct than in the SB evaluation. Again, the reason for that are the variations in segment sizes. Obviously, most segment-based misclassifications affect small segments, so that the impact on the point-based results is smaller in comparison. The largest amount of segment-based misclassifications occurs between the class *tree* and *building*. Class *tree* constitutes 30.7% of all segments, but only 17.6% of all points, so that this confusion has a smaller influence on the PB evaluation. This, along with the influence of the large amount of points assigned to fixed nodes, is also the main reason why larger values for the OA are achieved in point-based evaluation.

Fig. 8 shows a section of the point cloud of Area 3 of the Vaihingen data coloured by class labels resulting from the point-based classification (b) as well as from our segment-based classification with (d) and without (c) applying the interaction potential. The result obtained by the point-based classification exhibit a more heterogeneous distribution of the class labels: some scattered points on building roofs are misclassified as *tree* points and a relatively large amount of points which are *ground* or *tree* in the reference are classified as *other*. Compared to the point-based classification, the segment-based classification delivers a smoother label distribution. Here, the observations from the confusion matrices which were discussed above are confirmed: in the centre of the figure, a larger *ground* area is falsely classified as *building* in the classification without context (c) but classified correctly by applying the interaction potential (d). Further, several *tree* points in the upper left area of Fig. 8(c) are misclassified as *building* by the classification without interaction potential. Here the consideration of context also improves the result. As seen in the confusion matrices, applying the interaction potential decreases the results of the class *other* compared to the classification ignoring contextual information. This is also observable in the example of Fig. 8(d), where none of the points belonging to the class *other* in the reference are classified correctly by the contextual classification, in contrast to the results of the classification without context (Fig. 8(c)).

5.2.2. Rotterdam

By applying our classification method on the Rotterdam data set we achieve segment-based OAs of 68.8% and 59.8% with and without using the interaction potential for the classification, respectively (see Table 6). Compared to the OA that we achieve for the Vaihingen data, these results are distinctly inferior. This can probably be attributed to the distinction of seven rather than only four classes in case of the Rotterdam data set. Concerning the point-based OAs (Table 7), the results of the Rotterdam data set are better compared to the results for Vaihingen. On the one hand, this may be due to a better agreement of the PB and SB references (cf. Table 5), on the other hand the percentage of points belonging to fixed *ground* and *water* segments is considerably larger in Rotterdam than in Vaihingen.

The point-based improvement of the classification by additionally incorporating contextual information is relatively low at 0.7%. However, the improvements in the segment-based evaluation are similar to those of the Vaihingen data set (9.0%). Again, this is an indicator that the interaction model mainly improves the classification accuracy for smaller segments. To evaluate this assumption, we performed a segment-based evaluation of the classification results independently for segments of different size. Fig. 9 shows the segment-based classification results for eight classes of segments sizes; Fig. 10 shows the improvement due to the interaction potential for these categories of segments. It becomes obvious that the classification accuracy of the segments increases with the segment size. More than 90% of all segments have a segment size smaller than one thousand points. The classification result without using contextual information is 60% or lower for those segments and lies between 60% and 70% when taking into account contextual information. On the other hand, all segments consisting of more than 100,000 points are classified correctly no matter whether the interaction potential is used or not. This analysis shows that our conjecture that modelling interactions mainly supports the classification of small segments is correct.

Table 7 shows that the segmentation of the Rotterdam data only leads to 0.2% of unsegmented points. Here, the point-based classification of Niemeyer et al. (2015) achieves an OA of 82.8%, which is

Table 10
Confusion matrices (PB) of the classification of the segments from variant *planar_40* [%].

[%]	Without context						With context					
	R/C	*grd*	*bld*	*tree*	*other*	Comp.	R/C	*grd*	*bld*	*tree*	*other*	Comp.
54.8	*grd*	47.1	4.2	3.0	0.5	86.0	*grd*	49.2	2.5	3.0	0.1	89.8
25.4	*bld*	0.5	23.9	0.9	0.1	94.0	*bld*	0.5	24.1	0.8	0.0	94.6
17.6	*tree*	1.1	3.1	13.3	0.1	75.7	*tree*	0.9	1.2	15.5	0.0	88.0
2.2	*other*	0.9	0.5	0.4	0.4	16.1	*other*	1.0	0.6	0.4	0.2	7.0
	Corr.	94.8	75.6	75.5	36.0		Corr.	95.3	84.5	78.8	57.8	
	Qual.	82.2	72.1	60.8	12.5		Qual.	86.0	80.6	71.1	6.7	
	Overall accuracy					84.7	Overall accuracy					88.9

Table 11
Confusion matrix of the point-based classification of the Rotterdam data set. Values in [%]. Rows correspond to the classes in the reference (R), columns correspond to the classes in the classification result (C).

R/C	grd	wa	roof	tree	other	roofStr	facade	Comp.
grd	46.6	0.1	2.2	0.0	1.5	0.2	0.0	91.9
wa	1.3	7.4	0.0	0.0	1.9	0.0	0.0	69.9
roof	1.5	0.0	16.5	0.1	0.5	1.2	0.2	82.5
tree	0.0	0.0	0.1	2.4	0.7	0.1	0.1	71.5
other	0.4	0.1	0.4	0.3	3.9	0.3	0.7	65.9
roofStr	0.0	0.0	1.3	0.1	0.1	1.3	0.1	44.3
facade	0.1	0.0	0.4	0.3	0.3	0.7	4.7	72.0
Corr.	93.4	96.8	79.2	76.8	43.5	34.5	81.0	
Qual.	86.4	68.3	67.8	58.8	35.5	24.1	61.6	
Overall accuracy								82.8

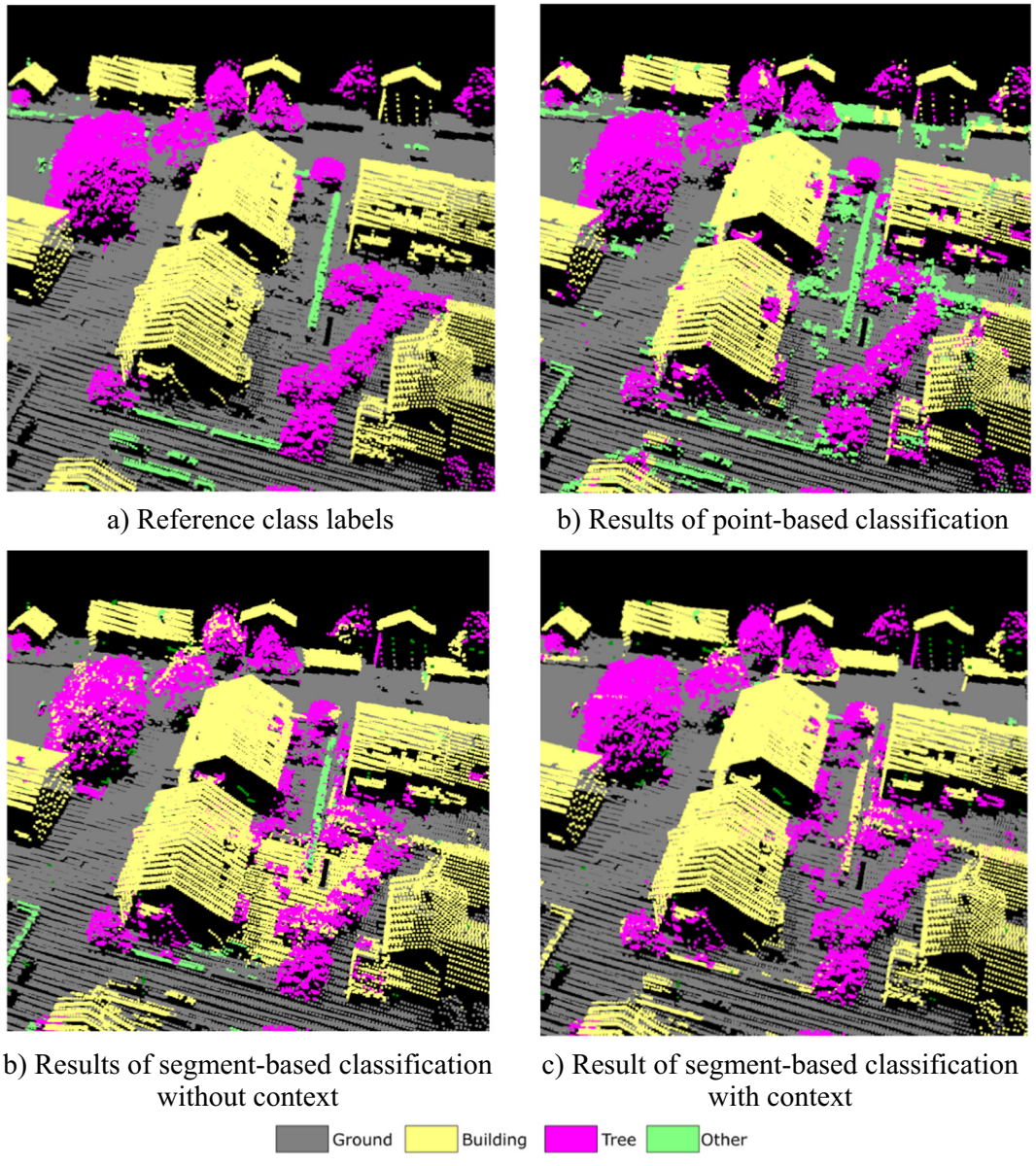


Fig. 8. Comparison of results obtained by classification with and without context (example from Area 3).

lower than the OA achieved by our approach by a margin of 8.2%. To achieve a more detailed comparison between the point-based and our segment-based classification approach, Tables 11 and 12 show the respective confusion matrices. Comparing the main diagonal of the confusion matrices, the classes which are classified distinctly better in our segment-based classification are *ground* and

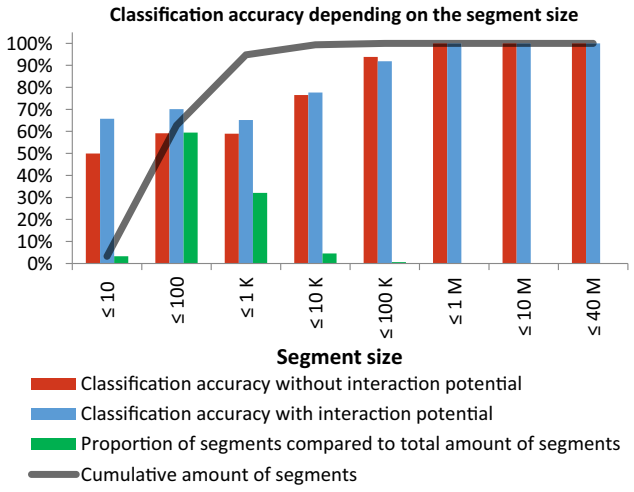


Fig. 9. Segment-based classification accuracy as a function of segment size.

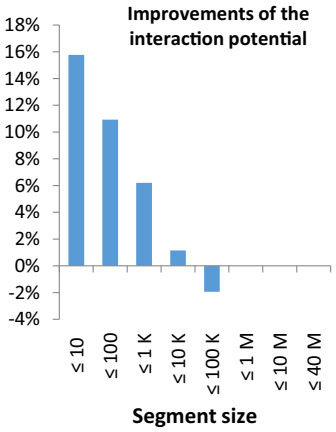


Fig. 10. Improvement of the classification accuracy due to the interaction potential as a function of segment size.

water. In the point-based classification the class *ground* has the largest confusion with the classes *roof* and *other*, while points of the class water are mainly misclassified as *other* and *ground*. In our segment-based classification these confusions are nearly elimi-

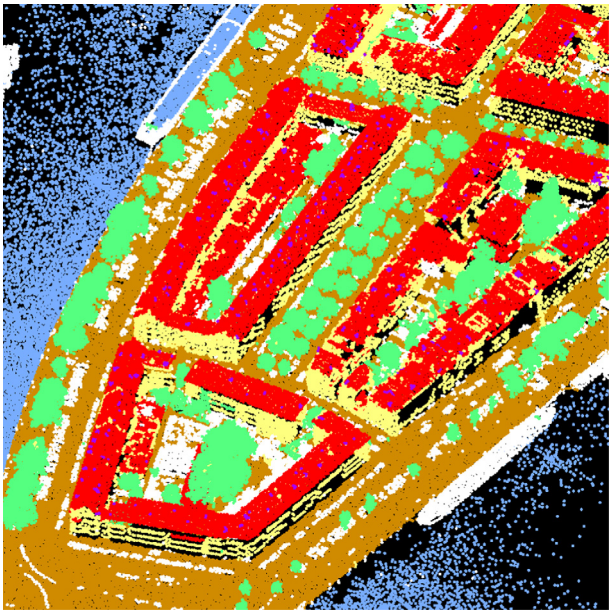
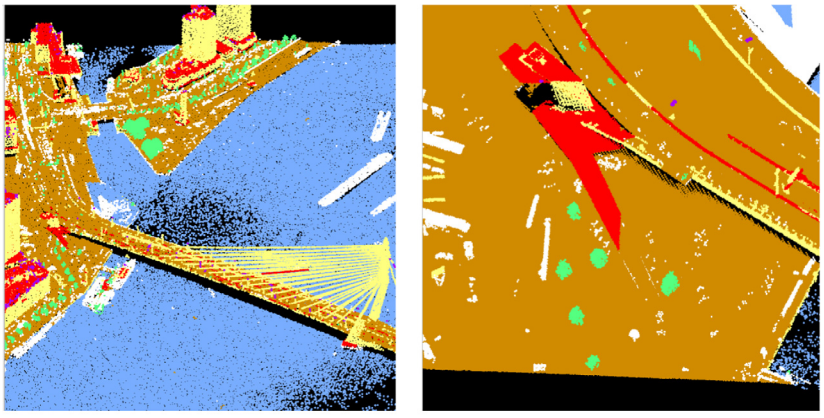


Fig. 11. Some subsets of the classification results for Rotterdam. Colours: red: *building roof*; purple: *roof structure*; blue: *water*; yellow: *building façade*; green: *tree*; ochre: *ground*; white: *other*. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

nated completely. Keeping in mind that the large majority of the *ground* and *water* points belong to the large segments that are classified based on simple heuristics, this is an indication that the segmentation preserves the boundaries of such objects rather well and that the classification works well for the remaining smaller *ground* and *water* segments.

Fig. 11 shows a subset of the classification results of Rotterdam. Two cases in complex surroundings are shown in Fig. 12. Fig. 12(a) shows the case of a bridge. The segmentation procedure merges the road surface on the bridge with the surrounding large *ground* segment so that there is no problem in the classification of that part of the bridge, even though the bridge surface is 10 m above the ground surface near the shoreline. However, the wires holding the bridge construction are classified as *wall* rather than *other*, which is probably due to the fact that they are the only structures of that type in the data set, so that they are not represented in the training data. Fig. 12(b) shows steep stairs connecting the ground and the bridge level at the western end of the same bridge. Here



(a) (b)

Fig. 12. Two problematic cases w ith complex structures in the Rotterdam data set. The colour code is identical to the one in Fig. 11. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 12
Confusion matrix (PB) of our segment-based classification of the Rotterdam data set [%]. Rows correspond to the classes in the reference (R), columns correspond to the classes in the classification result (C). The values only refer to the points assigned to a segment, thus the value for OA slightly differs from the one in Table 7.

R/C	grd	wa	roof	tree	other	roofStr	facade	Comp.
grd	50.3	0.0	0.2	0.0	0.4	–	0.2	98.3
wa	0.0	10.6	0.0	0.0	0.1	0.0	0.0	99.1
roof	0.5	0.0	17.9	0.0	0.7	0.1	0.4	90.9
tree	0.0	–	0.0	2.5	0.7	0.0	0.1	75.4
other	0.4	0.1	0.3	0.1	3.9	0.0	1.0	65.7
roofStr	0.0	–	1.2	0.0	0.2	0.6	0.9	21.4
facade	0.1	0.0	0.4	0.0	0.3	0.2	5.4	85.0
Corr.	97.9	98.7	89.1	93.1	62.3	65.0	68.1	
Qual.	96.3	97.9	81.8	71.4	47.0	19.2	60.8	
Overall accuracy								91.2

the merging of segments across tiles failed. The part attached to the ground surface is classified as *ground*. The higher parts are smooth surfaces above the ground level and therefore classified as *building roof*.

6. Conclusions

We have presented a method for the segment-based contextual classification of ALS point clouds. For that purpose, a hybrid segmentation technique to obtain planar segments as well as segments of arbitrary shape is designed, which delivers segments of quite different size. A CRF is applied to these segments, defining the underlying graph on the basis of a local 2D neighbourhood of points near the segment boundaries. We defined a set of segment-based features and, additionally, interaction features which are based on the local distribution of points in the vicinity of the segment boundaries. Using a simple set of heuristics, few very large *ground* and/or *water* segments could be identified. These segments were nevertheless considered in the classification as fixed nodes in order to propagate information to their neighbours for context-based classification.

Our method considers context at different levels. Firstly, as pointed out in Section 2.2, context is considered implicitly by carrying out a segment-based classification: points having similar properties are grouped to meaningful entities, and all points inside a segment are assigned same class labels. In view of this argumentation, the range of interactions between points will become rather large for large segments (though remaining rather limited for small ones). Secondly, we consider context by modelling the interactions between neighbouring segments in the CRF. While limited to first-order neighbours of each segment, it still expands the range of context beyond sets of points. Thirdly, the integration of the fixed nodes in the CRF can be seen as a way of considering context across the entire scene, though again being limited to interactions of these nodes with direct neighbours. The latter two levels both rely on the same interaction model based on a classifier that is adapted to the distribution of the features by training. However, our method does not consider global types of context that could, for instance, consist of models for the alignment of buildings or trees in rows that are parallel to roads.

Our evaluation has shown that the hybrid segmentation technique is quite effective, though it requires some parameter tuning for obtaining optimal results. The consideration of context had a large impact on the correct class assignment to segments, which could be increased by about 9% compared to the independent classification of the segments. The increase in classification accuracy was not as pronounced as that when the results were evaluated at the level of points, but an improvement in the order of 1–4% could still be achieved. This difference was shown to be the result of the fact that the improvement is largely due to a better classification of small segments. Compared to a point-based contextual classification, our segment-based technique could be shown to result in a slight improvement in the number of correctly classified points for the Vaihingen data set and a larger one (8%) for the Rotterdam data set. This improvement could be achieved despite of a small percentage of points that could not be classified because it was not assigned to any segment. In addition, the segment-based technique was shown to be somewhat faster than the point-based approach.

Future work could concentrate on an expansion of the proposed method to incorporate additional features delivered from ALS sensors, in particular intensity, echo counts, or full waveform features. This would require work on the consideration of intensity in the hybrid segmentation framework presented here, but also work on additional interaction features based on the local properties of points at the segment boundaries. Furthermore, the performance of our method when discerning different class structures, or using data of different point density and distribution needs to be investigated. This includes work on the generalisation of our method for point clouds that are derived from terrestrial sensors. The integration of context at an even larger scale, for instance in a hierarchical approach integrating segmentation results at different levels of detail, could be another direction of future research.

Acknowledgements

The Vaihingen data set 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.

The Rotterdam data set was provided by the Municipality of Rotterdam.

References

Aijazi, A.K., Checchin, P., Trassoudaine, L., 2013. Segmentation based classification of 3D urban point clouds: a super-voxel based approach with evaluation. Rem. Sens. 5, 1624–1650.

Boykov, Y.Y., Jolly, M.-P., 2001. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), vol. 1, pp. 105–112.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Brodu, N., Lague, D., 2012. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: applications in geomorphology. ISPRS J. Photogr. Rem. Sens. 68 (2012), 121–134.

Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 24 (5), 603–619.

Cramer, M., 2010. The DGPF-test on digital airborne camera evaluation – overview and test design. Photogr. Fernerkundung Geoinform. 2 (2010), 73–82.

Douillard, B., Underwood, J., Vlaskine, V., Quadros, A., Singh, S., 2010. A pipeline for the segmentation and classification of 3D point clouds. In: Proceedings of the International Symposium on Experimental Robotics (ISER 2010).

Felzenszwalb, P., Huttenlocher, D., 2004. Efficient graph–based image segmentation. Int. J. Comput. Vision 59 (2), 167–181.
Ferraz, A., Bretar, F., Jacquemoud, S., Goncalves, G., Pereira, L., 2010. 3D segmentation of forest structure using a mean-shift based algorithm. In: Proceedings IEEE 17th International Conference on Image Processing, 26–29 September, Hong Kong, pp. 1413–1416.
Frey, B., MacKay, D., 1998. A revolution: belief propagation in graphs with cycles. Adv. Neural. Inf. Process. Syst. 10, 479–485.
Golovinskiy, A., Funkhouser, T., 2009. Min-Cut based segmentation of point clouds. In: Proceedings of the IEEE Workshop on Search in 3D and Video (S3DV), ICCV Workshops, pp. 39–46.
Golovinskiy, A., Kim, V.G., Funkhouser, T., 2009. Shape-based recognition of 3D point clouds in urban environments. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2154–2161.
Gross, H., Thoennessen, U., 2006. Extraction of lines from laser point clouds. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVI-3A, pp. 87–91.
Guo, B., Huang, X., Zhang, F., Sohn, G., 2015. Classification of airborne laser scanning data using JointBoost. ISPRS J. Photogr. Rem. Sens. 100 (2015), 71–83.
Haala, N., Hastedt, H., Wolf, K., Ressl, C., Baltrusch, S., 2010. Digital photogrammetric camera evaluation – generation of digital elevation models. Photogr. Fernerkundung Geoinform. 2 (2010), 99–115.
C., Heipke, H., Mayer, C., Wiedemann, O., Jamet, 1997. Evaluation of automatic road extraction. In: International Archives of the Photogrammetry, Remote Sensing XXXII/3–4W2, pp. 151–160.
Horvat, D., Žalik, B., Mongus, D., 2016. Context-dependent detection of non-linearly distributed points for vegetation classification in airborne LiDAR. ISPRS J. Photogr. Rem. Sens. 116 (2016), 1–14.
Kohli, P., Ladický, L.L., Torr, P.H.S., 2009. Robust higher order potentials for enforcing label consistency. Int. J. Comput. Vision 82 (3), 302–324.
Kumar, S., Hebert, M., 2006. Discriminative random fields. Int. J. Comput. Vision 68 (2), 179–201.
Lim, E., Suter, D., 2009. 3D terrestrial lidar classifications with super-voxels and multi-scale conditional random fields. Comput. Aided Des. 41 (10), 701–710.
Lin, C.-H., Chen, J.-Y., Su, P.-L., Chen, C.-H., 2014. Eigen-feature analysis of weighted covariance matrices for LiDAR point cloud classification. ISPRS J. Photogr. Rem. Sens. 94 (1014), 70–79.
Melzer, T., 2007. Non-parametric segmentation of ALS point clouds using mean shift. J. Appl. Geod. 1 (2007), 159–170.
Luo, C., Sohn, G., 2014. Scene-layout compatible Conditional Random Field for classifying terrestrial laser point clouds. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-3, pp. 79–86.
Najafi, M., Namin, S.T., Salzmann, M., Petersson, L., 2014. Non-associative higher-order markov networks for point cloud classification. In: Proceedings of the European Conference on Computer Vision. Springer, pp. 500–515.
Nguyen, A., Le, B., 2013. 3D point cloud segmentation: a survey. In: Proceedings 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM), 12–15 November, pp. 225–230.
Niemeyer, J., Rottensteiner, F., Sörgel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. ISPRS J. Photogr. Rem. Sens. 87 (2014), 152–165.
Niemeyer, J., Rottensteiner, F., Sörgel, U., Heipke, C., 2015. Contextual classification of point clouds using a two-stage CRF. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XL-3/W2, pp. 141–148.
Niemeyer, J., Rottensteiner, F., Sörgel, U., Heipke, C., 2016. Hierarchical higher order CRF for the classification of airborne LiDAR point clouds in urban areas. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLI-B3, pp. 655–662.
Papon, J., Abramov, A., Schoeler, M., Worgotter, F., 2013. Voxel cloud connectivity segmentation – supervoxels for point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2027–2034.
Pham, T.T., Reid, I., Latif, Y., Gould, S., 2015. Hierarchical higher-order regression forest fields: an application to 3D indoor scene labelling. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2246–2254.
Pu, S., Rutzinger, M., Vosselman, G., Oude Elberink, S.J., 2011. Recognizing basic structures from mobile laser scanning data for road inventory studies. ISPRS J. Photogr. Rem. Sens. 66 (6 Suppl.), S28–S39.
Rutzinger, M., Höfle, B., Hollaus, M., Pfeifer, N., 2008. Object-based point cloud analysis of full-waveform airborne laser scanning data for urban vegetation classification. Sensors 2008 (8), 4505–4528.
Schnabel, R., Wahl, R., Klein, R., 2007. Efficient RANSAC for point-cloud shape detection. Comput. Graph. Forum 26 (2), 214–226.
Shapovalov, R., Velizhev, A., Barinova, O., 2010. Non-associative Markov networks for 3D point cloud classification. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXVIII-3A, pp. 103–108.
Shotton, J., Winn, J., Rother, C., Criminisi, A., 2009. Textonboost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context. Int. J. Comput. Vision 81 (1), 2–23.
Sima, M.-C., Nüchter, A., 2013. An extension of the Felzenszwalb-Huttenlocher segmentation to 3D point clouds. In: Proceedings Fifth International Conference on Machine Vision (ICMV 2012): Computer Vision, Image Analysis and Processing, Wuhan, 20 October 2012. SPIE Proceedings 8783 (13 March, 2013).
Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C., 2008. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. IEEE Trans. Pattern Anal. Mach. Intell. 30 (6), 1068–1080.
Smeeckaert, J., Mallet, C., David, N., Chehata, N., Ferraz, A., 2013. Large-scale water classification of coastal areas using airborne topographic lidar data. In: Proceedings of the 33rd IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2013), pp. 61–64.
Tokarczyk, P., Montoya, J., Schindler, K., 2012. An evaluation of feature learning methods for high resolution image classification. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences I-3, pp. 389–394.
Velizhev, A., Shapovalov, R., Schindler, K., 2012. Implicit shape models for object detection in 3D point clouds. In: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Science, Melbourne, Australia, vol. I-3, pp. 179–184.
Vilariño, D.L., Martínez, J., Rivera, F.F., Cabaleiro, Pena, T.F., 2016. Graph-based segmentation of airborne LiDAR point clouds. In: Proceedings Image and Signal Processing for Remote Sensing XXII. SPIE Proceedings, vol. 10004, 8 p.
Vosselman, G., 2012. Automated planimetric quality control in high accuracy airborne laser scanning surveys. ISPRS J. Photogr. Rem. Sens. 74, 90–100.
Vosselman, G., 2013. Point cloud segmentation for urban scene classification. In: ISPRS Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 40, part 7/W2, Antalya, 11–13 November, pp. 257–262.
Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015a. Contextual classification of point cloud data by exploiting individual 3D neighbourhoods. ISPRS J. Photogr. Rem. Sens. 105 (2015), 286–304.
Weinmann, M., Schmidt, A., Mallet, C., Hinz, S., Rottensteiner, F., Jutzi, B., 2015b. Contextual classification of point cloud data by exploiting individual 3D neighbourhoods. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-3/W4, pp. 271–278.
Xiong, X., Munoz, D., Bagnell, J.A., Hebert, M., 2011. 3-D scene analysis via sequenced predictions over points and regions. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2609–2616.
Xu, S., Oude Elberink, S., Vosselman, G., 2012. Entities and features for classification of airborne laser scanning data in urban area. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences I-4, pp. 257–262.
Yao, W., Hinz, S., Stilla, W., 2009. Object extraction based on 3D-segmentation of LiDAR data by combining mean shift with normalized cuts: two examples from urban areas. In: Proceedings of 2009 Joint Urban Remote Sensing Event (URBAN2009 – URS2009), Shanghai, China.