



## Special Section on Processing Large Geospatial Data

## Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas

M. Weinmann<sup>a,\*</sup>, S. Urban<sup>a</sup>, S. Hinz<sup>a</sup>, B. Jutzi<sup>a</sup>, C. Mallet<sup>b</sup><sup>a</sup> Institute of Photogrammetry and Remote Sensing, Karlsruhe Institute of Technology (KIT), Englerstr. 7, 76131 Karlsruhe, Germany<sup>b</sup> Université Paris-Est, IGN, SRIG, MATIS, 73 Avenue de Paris, 94160 Saint-Mandé, France

## ARTICLE INFO

## Article history:

Received 1 September 2014

Received in revised form

23 December 2014

Accepted 29 January 2015

Available online 7 February 2015

## Keywords:

3D scene analysis

Point cloud

Feature

Classification

Large-scale

Urban

## ABSTRACT

We propose a new methodology for large-scale urban 3D scene analysis in terms of automatically assigning 3D points the respective semantic labels. The methodology focuses on simplicity and reproducibility of the involved components as well as performance in terms of accuracy and computational efficiency. Exploiting a variety of low-level 2D and 3D geometric features, we further improve their distinctiveness by involving individual neighborhoods of optimal size. Due to the use of individual neighborhoods, the methodology is not tailored to a specific dataset, but in principle designed to process point clouds with a few millions of 3D points. Consequently, an extension has to be introduced for analyzing huge 3D point clouds with possibly billions of points for a whole city. For this purpose, we propose an extension which is based on an appropriate partitioning of the scene and thus allows a successive processing in a reasonable time without affecting the quality of the classification results. We demonstrate the performance of our methodology on two labeled benchmark datasets with respect to robustness, efficiency, and scalability.

© 2015 Elsevier Ltd. All rights reserved.

## 1. Introduction

The automated analysis of 3D point clouds has become a topic of great importance in photogrammetry, remote sensing, computer vision and robotics. One avenue of research directly addresses the analysis of urban environments, where recent investigations focus on 3D reconstruction [1–3], consolidation of imperfect scan data [4,5], object detection [6–9], extraction of roads and curbstones or road markings [10–12], urban accessibility analysis [13], recognition of power-line objects [14], extraction of building structures [15], vegetation mapping [16], large-scale city modeling [17], semantic perception for ground robotics [18] and semantization of complex 3D scenes [19]. A common task for many of these different applications consists of point cloud classification [20,21], where each 3D point is assigned a specific (semantic) class label.

Addressing the task of urban point cloud classification – where the spatial 3D data may be collected via dense matching as well as airborne, terrestrial and/or mobile laser scanning – we face a variety of challenges arising from the complexity of respective 3D scenes caused by an irregular sampling and very different types of objects. Since the results of urban 3D scene analysis may vary from

one dataset to another, publicly available standard datasets are desirable in order to compare the performance of different methodologies. Consequently, there has been a steadily increasing availability of 3D point cloud datasets in recent years [22]. However, urban point clouds with respective point-wise manual annotations in terms of semantic class labels are still rarely available, although this represents a prerequisite for supervised point cloud classification. One of the most widely used datasets is the Oakland 3D Point Cloud Dataset [23] which, however, only contains approximately 1.6 million labeled 3D points. Hence, this dataset is not tailored to designing large-scale processing pipelines.

Due to the recent technological advancements, it is meanwhile possible to collect geospatial data in a fast and efficient way via terrestrial and mobile laser scanning. In order to foster research in advanced 3D point cloud processing, two labeled point cloud datasets representing densely sampled urban environments with a significantly higher number of 3D points have been presented recently [24,25]. These can be considered as a first step towards large geospatial datasets in terms of city-scale or even larger. The availability of such datasets is important for comparing large-scale processing workflows which is the core issue of a recent benchmark [25] and addressed in this paper.

In our work, we consider each individual 3D point and its local 3D neighborhood for extracting respective geometric features. By exploiting a fully generic approach for optimizing the neighborhood

\* Corresponding author. Tel.: +49 721 608 47302.

E-mail address: [martin.weinmann@kit.edu](mailto:martin.weinmann@kit.edu) (M. Weinmann).

size, our approach is generally applicable and not tailored to a specific dataset. Furthermore, our approach represents a basic requirement for either smooth labeling techniques or methods involving contextual information, since both of them are based on the results of individual point classification. In summary, our contributions extend [26] and consist of

- a new methodology for large-scale urban 3D point cloud classification,
- an in-depth analysis of a powerful strategy for recovering individual 3D neighborhoods of optimal size,
- efficient feature extraction and classification, and
- an extension towards data-intensive processing.

In the following, we first reflect related work in Section 2 and provide an up-to-date view of approaches for processing 3D point cloud data in order to efficiently obtain significant information contained in the data. Subsequently, in Section 3, we present a methodology which is closely linked to recent investigations on 3D scene analysis involving optimal neighborhoods and different classifiers [27,28]. Based on these investigations with a very detailed evaluation, we can directly select the most appropriate solution with respect to urban 3D scene analysis. The considered criteria address feasibility in terms of simplicity and reproducibility of the involved components as well as performance in terms of accuracy and computational effort. We further introduce an increase in efficiency resulting from efficient neighborhood recovery. In order to extend the applicability of the selected methodology towards huge datasets, in Section 4, an extension towards large-scale urban point cloud classification is presented which does not affect the quality of the results, but allows the successive processing of huge point clouds in a reasonable time. Afterwards, the datasets involved in our experiments and the experimental results are presented in Sections 5 and 6. The derived results are subsequently discussed in Section 7. Finally, in Section 8, concluding remarks are provided and suggestions for future work are outlined.

## 2. Related work

Although modern devices nowadays allow the acquisition of additional information such as echo-based features or full-waveform features [29,30] which may alleviate 3D scene analysis, we focus on the use of geometric features as most of the available point cloud datasets only contain spatial 3D information. Other features may however easily be appended to the feature vectors defined in the scope of our work. In the following, we first present fundamental concepts for defining appropriate features for 3D scene analysis. Subsequently, we discuss approaches for (i) optimizing the derived feature vectors by involving feature relevance or (ii) describing the local 3D structure either at a single, but optimized neighborhood or at multiple neighborhoods of different size. Finally, we briefly reflect related work on classification methods for 3D scene analysis.

### 2.1. Feature design

A crucial issue for 3D scene analysis consists of designing appropriate features. For this purpose, numerous histogram-based methods have been proposed which accumulate information about the spatial 3D geometry into a histogram according to a specific quantized domain [31]. A still popular approach has been presented with Spin Images [32], where 2D histograms are derived by spinning a 2D plane patch around the surface normal and counting the number of points falling into each bin of the 2D

patch. Powerful alternatives have recently been presented with Point Feature Histograms (PFHs) [33] and their modification denoted as Fast Point Feature Histograms (FPPHs) [34]. Considering a point  $\mathbf{X}$  and all points within its local neighborhood, these approaches first assign the respective surface normal to  $\mathbf{X}$  and then sample geometric relations between the nearest neighbors in terms of angular variations and point distances into histograms. A different strategy has been presented with shape distributions [35] which are based on the idea of randomly sampling simple geometric measures such as distances or angles in order to obtain a descriptor characterizing the neighborhood around a point  $\mathbf{X}$  [36]. Furthermore, a combination of histograms with signatures has been presented with the Signature of Histograms of Orientations (SHOT) descriptor [31] in order to achieve a better balance between descriptiveness and robustness. The performance of different histogram descriptors has been compared in [37]. However, for all these approaches, single entries of the derived feature vectors are hardly interpretable.

Alternatively, the local 3D structure can be described by deriving the 3D covariance matrix from the spatial coordinates of a point  $\mathbf{X}$  and its neighbors. Based on the respective eigenvalues, a direct scene analysis may be conducted [38], or a set of features may be defined [39] which encapsulate geometric information about the local 3D structure. In particular the latter approach is nowadays commonly applied in lidar data processing, and the respective features or feature subsets are typically complemented with other geometric features [40,23,41,27,42,43]. A specific advantage consists of the fact that the respective entries in the feature vector are interpretable as they address local 3D shape primitives. In contrast to 3D covariance matrices encoding the relationships among points within a local neighborhood, covariance matrices of higher dimension have been used to combine multiple features such as angular measures and point distances to a compact representation [44]. Further information such as radiometric information may also be taken into account in this representation.

### 2.2. Feature relevance assessment

For compensating a lack of knowledge, often as many features as possible are extracted and involved in the classification process, although some of these features may be more and others less suitable. Consequently, investigations focusing on feature selection have also been introduced for 3D point cloud processing in order to improve the classification accuracy while simultaneously reducing both computational effort and memory consumption. Respective approaches allow a ranking of single features according to their relevance and the selection of a subset of the best-ranked features. Whereas the ranking can be obtained by involving a classifier [29,30,45], classifier-independent approaches are in the focus of filter-based feature selection which offers both simplicity and efficiency. Herein, univariate filter-based feature selection methods rely on a score function which simply evaluates feature-class relations based on the training data. A general relevance metric composed of several score functions addressing different intrinsic properties of the given training data has recently been proposed [27]. In contrast, multivariate filter-based feature selection methods rely on both feature-class and feature-feature relations in order to discriminate between relevant, irrelevant and redundant features [28].

### 2.3. Single-scale vs. multi-scale representation

The heuristic determination of local 3D neighborhoods is conducted either with respect to the absolute size [38,46] or with respect to the scale parameter [27]. However, in order to avoid heuristically determining a suitable neighborhood size – which

may even be specific for each dataset and which may not be identical across all 3D points in consideration – there have been few attempts to automatically derive individual neighborhoods of optimal size. Respective approaches rely on the local surface variation [47,48], a combination involving curvature, point density and noise of normal vector estimation [49,40], dimensionality features [41] or the measure of eigenentropy [28]. The need for involving such techniques becomes for instance apparent when analyzing the behavior of features derived from the 3D structure tensor and shape distribution features across different scales [36] or when observing a significant improvement in comparison to neighborhoods of fixed scale [28].

Instead of focusing on the concept of optimal neighborhoods, a consideration of features at multiple scales may be applied. For instance, it has been proposed to calculate features at different scales and involve a training procedure in order to define which combination of scales allows the best separation of different classes [50]. Further approaches even extract features based on different entities such as points and regions [51,52].

#### 2.4. Individual vs. contextual classification

When selecting an appropriate classifier, we may follow the strategy of individual point classification by exploiting respective feature vectors, where Support Vector Machines [53], Random Forests [29], AdaBoost [54] or classical Maximum Likelihood (ML) classifiers exploiting Gaussian Mixture Models (GMMs) [40] represent the most commonly applied approaches for point cloud classification. Alternatively, contextual learning approaches may be applied which address the idea that semantic labels of nearby 3D points tend to be correlated and hence involve relationships among 3D points within a local neighborhood in addition to the respective feature vectors. Respective approaches applied for point cloud classification are represented by Associative and non-Associative Markov Networks [23,55,56], Conditional Random Fields [21], multi-stage inference procedures focusing on point cloud statistics and relational information over different scales [51], and spatial inference machines modeling mid- and long-range dependencies inherent in the data [57].

When applying contextual learning approaches, it has to be taken into account that the relationships are inferred from the training data. Hereby, the local neighborhood is typically different from the neighborhood used for feature extraction. Furthermore, since the training data is limited, exact inference is computationally intractable and therefore either approximate inference techniques or smoothing techniques are commonly applied. The selection of an approximate inference technique remains challenging as there is no indication towards an optimal inference strategy, and such techniques quickly reach their limitations if the considered neighborhood becomes too large. In contrast, smoothing techniques enforce the desirable smooth labeling of nearby 3D points and may thus provide a significant improvement with respect to classification accuracy [58]. Such smoothing techniques, however, exploit either

the estimated probability of a 3D point belonging to each of the defined classes or the direct assignment of the respective label, and thus the results of an individual point classification.

### 3. Methodology

For 3D scene analysis in terms of uniquely assigning each 3D point a semantic label, we propose a fully automatic and generic methodology which consists of three successive steps (Fig. 1). In the first step, each 3D point is characterized with an individual local 3D neighborhood of optimal size (Section 3.1). This allows an extraction of highly distinctive features which is pursued in the second step, where various geometric 3D and 2D features are taken into consideration (Section 3.2). Finally, in the third step, the distinctive features and a given set of training examples are provided to a supervised classification scheme (Section 3.3). The main focus of our investigations is put on feature design in terms of deriving distinctive geometric features from individual neighborhoods of optimal size.

#### 3.1. Neighborhood selection

Considering a point  $\mathbf{X}$  in a point cloud  $\mathcal{P}$ , the respective neighborhood selection generally involves (i) a suitable neighborhood definition, (ii) an efficient recovery of the local neighborhood and (iii) an optimal parameterization of the neighborhood in terms of neighborhood size. These aspects are addressed in the following subsections.

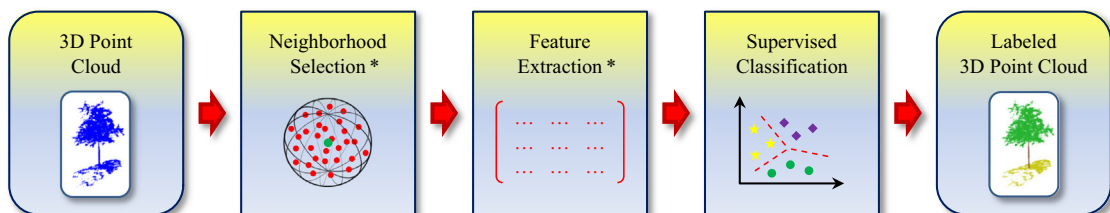
##### 3.1.1. Neighborhood definitions

In general, very different approaches may be applied in order to define the local 3D neighborhood  $\mathcal{N}$  of a given 3D point  $\mathbf{X} \in \mathbb{R}^3$ . For instance, a spherical neighborhood definition may be applied, where the local neighborhood is formed by all 3D points in a sphere with a fixed radius [59]. An alternative consists of applying a cylindrical neighborhood definition, where the local neighborhood is formed by all those 3D points whose 2D projections onto the ground plane are within a circle with a fixed radius [60]. A further neighborhood definition involves a fixed number of  $k$  closest 3D points for a given query point [61], which results in spherical neighborhoods of variable absolute size. Note that all these neighborhood definitions rely on the specification of one free scale parameter.

Since we want to account for more flexibility in case of varying point density and thereby avoid including a-priori knowledge on the scene, we employ a neighborhood definition based on the  $k$  closest neighbors of a given 3D point. Consequently, the nearest neighbors have to be recovered and an appropriate  $k$  has to be selected.

##### 3.1.2. Exact vs. approximate nearest neighbors

As a consequence of the selected neighborhood definition, a computationally quite expensive part consists of the calculation of



**Fig. 1.** The proposed methodology: after calculating an individual local 3D neighborhood of optimal size for each 3D point, highly distinctive 3D and 2D features are extracted and provided to a supervised classification scheme in order to obtain a semantically labeled 3D point cloud (\*The respective implementation (Matlab, C++ and binaries) is released with this paper and available at <http://www.ipf.kit.edu/code.php>).

the nearest neighbors for each 3D point. This nearest neighbor search can formally be described as follows: given a point set  $\mathcal{P} = \{\mathbf{X}_1, \dots, \mathbf{X}_N\}$  in a 3-dimensional Euclidean vector space, those points  $\mathbf{X} \in \mathcal{P}$  that are nearest to a given query point  $\mathbf{X}_Q$  should be recovered efficiently. The commonly used approach for nearest neighbor search is based on a Kd-tree [62] which represents a compact, hierarchical data structure for point sets sampled from a  $K$ -dimensional manifold. A point in a Kd-tree with  $N$  points can thus be localized with an average complexity of  $O(\log N)$  and, in the worst case, with a complexity of  $O(N)$  [63]. In order to further increase efficiency, an approximate nearest neighbor search has been proposed [64] which can be much faster than the exact nearest neighbor search with only little loss in accuracy since non-optimal neighbors may be returned. For details on these approaches, we refer to an extensive survey on data structures [65]. Addressing the criteria of query time and accuracy, a powerful approach with public availability and fully automatic parameter selection has been presented in the Fast Library for Approximate Nearest Neighbors (FLANN) [66] which is based on either searching hierarchical  $K$ -means trees with a priority search order or using multiple randomized Kd-trees. Hence, we apply the FLANN for nearest neighborhood search.

### 3.1.3. Optimal parameterization

Besides an efficient nearest neighbor search, it is desirable to automatically find the optimal parameterization for the selected neighborhood definition which is based on the  $k$  closest neighbors in our case. When addressing this issue, we may also take into account that the optimal choice of the parameter  $k$  (which is also commonly referred to as scale) may vary within a dataset since  $k$  certainly depends on the respective 3D structures and thus also on the respective class label.

Seminal work addressing the selection of an optimal value for the scale parameter  $k$  and thus the selection of the optimal neighborhood size is based on fundamental geometric properties of the point cloud data. For instance, the local surface variation (i.e. the change of curvature) may be exploited since a critical neighborhood size is indicated by a significant change of curvature when successively increasing the neighborhood by adding the next closest 3D point [47,48]. Furthermore, an iterative scheme involving curvature, point density and noise of normal vector estimation has been proposed [49,40]. Whereas these approaches are tailored for smoothly varying surfaces, they may face severe challenges when considering surfaces acquired with lidar systems or dense matching. Consequently, we focus on the use of multiple low-level geometric features which adequately capture the variability of natural environments.

Our approach is inspired by dimensionality based scale selection [41], where optimal neighborhood size selection is based on the idea that the optimal neighborhood size favors one dimensionality the most. More specifically, for describing the local 3D structure around a 3D point  $\mathbf{X} = \mathbf{X}_0$ , the dimensionality features of linearity  $L_\lambda$ , planarity  $P_\lambda$  and scattering  $S_\lambda$  are derived from the set of neighboring 3D points  $\mathbf{X}_i$  with  $i = 1, \dots, k$  by considering the respective 3D covariance matrix

$$\mathbf{C} = \frac{1}{k+1} \sum_{i=0}^k (\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T \quad (1)$$

with

$$\bar{\mathbf{X}} = \frac{1}{k+1} \sum_{i=0}^k \mathbf{X}_i \quad (2)$$

which is also known as the 3D structure tensor. This 3D structure tensor  $\mathbf{C}$  represents a symmetric positive-definite matrix. Consequently, its eigenvalues exist, are non-negative and correspond to

an orthogonal system of eigenvectors. For the sake of generality, we assume that there might not necessarily be a preferred variation with respect to the eigenvectors. This results in the general case of a structure tensor with rank 3 as well as eigenvalues  $\lambda_1, \lambda_2$  and  $\lambda_3$  with  $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$  and  $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ . Based on these eigenvalues, the dimensionality features are defined as

$$L_\lambda = \frac{\lambda_1 - \lambda_2}{\lambda_1} \quad P_\lambda = \frac{\lambda_2 - \lambda_3}{\lambda_1} \quad S_\lambda = \frac{\lambda_3}{\lambda_1} \quad (3)$$

and, as these features sum up to 1, they may be considered as the probabilities of a 3D point to be labeled as 1D, 2D or 3D structure [41]. Favoring one dimensionality the most thus corresponds to minimizing a measure given by the Shannon entropy [67] as

$$E_D = -L_\lambda \ln(L_\lambda) - P_\lambda \ln(P_\lambda) - S_\lambda \ln(S_\lambda) \quad (4)$$

across different scales  $k$ , and the optimal neighborhood size corresponds to the respective  $k$  with the minimal Shannon entropy. Instead of directly varying the scale parameter  $k$  as later tested in [28], however, the respective radius has been taken into account in [41]. Sampling the interval  $[r_1, r_2]$  between specified radii  $r_1$  and  $r_2$  into 16 scales, where the radii are not linearly increased since the radius of interest is usually closer to  $r_1$ , the optimal neighborhood size corresponds to the radius yielding the minimal Shannon entropy. However, the two radii  $r_1$  and  $r_2$  depend on various characteristics of the given data and are therefore specific for each dataset [41].

In order to avoid strong assumptions on the presence of specific geometric structures in the scene and to get rid of heuristic parameter selection, a more general solution for optimal neighborhood size selection has been proposed very recently [28] which has proven to outperform dimensionality based scale selection. Instead of exploiting the three dimensionality features, this approach directly exploits the eigenvalues of the 3D structure tensor which correspond to the principal components and thus span a 3D covariance ellipsoid. Normalizing the three eigenvalues  $\lambda_1, \lambda_2$  and  $\lambda_3$  by their sum  $\Sigma_\lambda$  yields normalized eigenvalues  $e_1, e_2$  and  $e_3$  summing up to 1. Thus, in analogy to the dimensionality based scale selection, the measure of eigenentropy  $E_\lambda$  given by the Shannon entropy according to

$$E_\lambda = -e_1 \ln(e_1) - e_2 \ln(e_2) - e_3 \ln(e_3) \quad (5)$$

is calculated and minimized across different scales  $k$  which, in turn, relates to minimizing the disorder of points within a 3D covariance ellipsoid. The optimal neighborhood size finally corresponds to the respective  $k$  with the minimal eigenentropy. For our experiments, we consider relevant statistics to start with  $k_1 = 10$  – which is in accordance to [41] – and successively increase the scale parameter  $k$  with a step size of  $\Delta k = 1$  up to an upper bound of  $k_2 = 100$  as proposed in [28], which already represents a relatively high number. Note that  $k_2$  can be arbitrary, and hence we will later focus on this issue in the experimental results.

The resulting optimal neighborhood size thus depends on contextual information preserved in the spatial arrangement of neighboring 3D points and may even be different for each individual 3D point. Even though optimal neighborhood size selection causes a higher computational effort with respect to both processing time and memory consumption, it should be taken into consideration since the classification accuracy is significantly improved according to recent investigations [28].

## 3.2. Feature extraction

Since many of the publicly available 3D point cloud datasets only contain information about the spatial 3D geometry in terms of XYZ coordinates, we focus on the use of geometric features. Such



geometric features typically rely on a local 3D neighborhood which has already been derived in the previous step. Involving the optimal neighborhood size for a respective 3D point, we may assume that highly distinctive geometric features can be derived from the set of 3D points within the neighborhood. Due to the high point density of recently published lidar point cloud datasets, the selected scale  $k$  tends to correspond to a relatively small absolute size, and the respective local 3D structure can therefore only be described with low-level geometric features. Such features show a specific behavior for planar patches, ridges, edges and vertices. By involving a 2D projection, we may also account for structures with larger extent, e.g. in the vertical direction.

In order to define adequate low-level geometric features, we follow the strategy involving a variety of geometric 3D and 2D features [27], but we additionally increase their distinctiveness by taking into account the optimal neighborhood size for each individual 3D point [28]. In total, a set of 21 distinctive low-level geometric features is thus calculated for each 3D point. For the sake of clarity, we briefly describe the involved 3D and 2D features in the following subsections. The respective code (Matlab, C++ and binaries) is released with this paper.

### 3.2.1. 3D features

Obviously, a variety of 3D features can directly be derived by describing basic geometric properties of the considered 3D neighborhood such as the absolute height  $Z$  of the considered 3D point  $\mathbf{X}$ , the radius  $r_{k-NN}$  of the sphere encompassing the local 3D neighborhood, the local point density  $D$  defined as number of points per unit volume and the verticality  $V$  which is based on the vertical component of the local normal vector. Hereby, the local normal vector is related to the eigenvector corresponding to the smallest eigenvalue of the respective 3D structure tensor. Furthermore, all 3D points within the neighborhood may be considered in order to calculate the maximum height difference  $\Delta Z_{k-NN}$  and the standard deviation  $\sigma_{Z,k-NN}$  of height values.

Additionally, we take into account that the 3D structure tensor encodes the general distribution of 3D points within the local neighborhood. Consequently, the normalized eigenvalues  $e_1$ ,  $e_2$  and  $e_3$  of the 3D structure tensor may also be exploited to define local 3D shape features. Besides the aforementioned dimensionality features of linearity  $L_\lambda$ , planarity  $P_\lambda$  and scattering  $S_\lambda$ , further features are represented by omnivariance  $O_\lambda$ , anisotropy  $A_\lambda$ , eigenentropy  $E_\lambda$ , the sum  $\Sigma_\lambda$  of eigenvalues and the local surface variation  $C_\lambda$ .

### 3.2.2. 2D features

Taking into account that urban areas are typically characterized by an aggregation of man-made structures, specific geometric relations in terms of symmetry and orthogonality are likely to occur. In particular, we may face a large number of structures which are oriented perpendicular to a horizontally oriented plane, e.g. building façades, traffic signs or curbstone edges. Consequently, a projection of the 3D point cloud onto a horizontally oriented plane might reveal additional information and possibly also clear evidence about the presence of specific structures in the observed scene. Respective features which are based on this 2D projection can easily be defined as 2D properties of the neighborhood such as the radius  $r_{k-NN,2D}$  and the local point density  $D_{2D}$ .

Furthermore, the coordinates resulting from the 2D projection of all 3D points within the neighborhood may be exploited to derive the 2D covariance matrix also known as the 2D structure tensor. In analogy to the 3D case, the respective two eigenvalues may be used to define characteristic features such as the sum  $\Sigma_{\lambda,2D}$  of the eigenvalues and their ratio  $R_{\lambda,2D}$ .

Interestingly, the aforementioned 2D features are based on spherical neighborhoods of relatively small absolute metric size, whereas man-made structures tend to provide a similar behavior across different height levels, i.e. across several meters of height. Consequently, a discretization of the 2D projection in terms of a 2D accumulation map [68] with discrete, quadratic bins (here with a side length of 0.25 m) may provide further interesting properties about local 3D structures which are not yet covered by the already defined features. Respective features based on the accumulation map have been proposed with the number  $M$  of points falling in the respective bin as well as the maximum height difference  $\Delta Z$  and standard deviation  $\sigma_Z$  of height values within the respective bin. Particularly, the feature  $M$  provides clear evidence on the existence of building façades and, if the point density is sufficiently high, also on the existence of curbstone edges. The detection of such man-made structures, in turn, is important for urban accessibility analysis which is one of the main intentions of current research.

## 3.3. Supervised classification

For classification, we apply a standard supervised classification scheme involving a set of training examples which, in turn, encapsulate feature vectors as well as the respective class labels. An adequate choice among a variety of classification strategies and respective approaches, however, should directly address applicability, reproducibility, and scalability in order to facilitate 3D scene analysis in large-scale urban environments. In the following, we first motivate our choice for the classifier and subsequently address the issue of how to obtain suitable training data.

### 3.3.1. Classifier selection

Since we focus on the applicability and reproducibility of all involved components, the involved classification strategy should be easy-to-use without crucial parameter selection, and respective implementations should be available in different software packages. For this reason, we focus on individual point classification based on a set of derived features. With the intention of processing huge point clouds, it is mandatory to apply a powerful but still computationally efficient classifier. In particular for scalability towards huge datasets, combining a set of weak classifiers such as decision trees via bootstrap aggregating which is commonly referred to as bagging [69] has proven to be successful. Using bootstrapped replica of the training data, i.e. subsets of the complete training data which are randomly drawn with replacement [70], diversity is obtained by training a weak learner of the same type for each subset of the training data.

The most popular example for bagging is represented by Random Forests [71] which represent a modern discriminative method and provide efficiency in case of a large amount of input data. Efficiency in this context covers simplicity and a high degree of parallelization which results in a fast classification scheme. Additionally, robustness to outliers, noise and missing data is provided. More specifically, a Random Forest represents an ensemble of randomly trained decision trees and thus aggregates hypotheses derived via decision trees which, in turn, are trained over different distributions of the training data. Thereby, each decision tree is constructed based on a top-down strategy successively selecting the variable which best splits the respective training data. Consequently, a split function and a stopping criterion have to be defined. Since, in the training phase, individual decision trees are assumed to be trained on randomly selected subsets of the given training data, the trees may be expected to be randomly different from one another which results in a decorrelation between individual tree predictions and thus improved

generalization and robustness [72]. Once a Random Forest is trained, the classification phase consists of letting each decision tree vote for a single class and assigning the respective label according to the majority vote of all decision trees. The parameters to be specified are the number of involved decision trees and the tree depth. In the experiments, we use a Random Forest with 100 trees and a tree depth of  $\lfloor \sqrt{d} \rfloor$ , where  $d$  equals the number of extracted features, i.e.  $d=21$ .

### 3.3.2. Training data

Given a set of training examples, we have to take into account that an unbalanced distribution of training examples across all classes may have a detrimental effect on the training process [72]. Consequently, we involve a class re-balancing in terms of randomly selecting the same number of training examples for each class. Alternatively, the known prior class distribution of the training set could be used for weighting the contribution of each class.

## 4. Extension towards large-scale urban point cloud classification

Recent investigations clearly show that the use of individual neighborhoods with optimal size has a significant, beneficial impact on the classification results [28]. However, the additional calculations cause a drastic increase in computational effort. Consequently, the described methodology is suited to process point clouds containing only up to a few millions of 3D points. When considering huge point clouds at city scale with possibly billions of points – which is the aim of recent effort in order to obtain an adequate 3D model of a whole city like Paris – an extension of the presented methodology has to be introduced.

The extension described in this paper does not affect the quality of 3D scene analysis, but only the scalability of the methodology in order to process larger datasets. Specifically, it focuses on successively processing a huge point cloud by applying a 2D sliding window function which is shifted within a horizontally oriented plane in discrete steps and involves a small padding region in order to avoid discontinuities at its borders. The size of the window should be chosen in a way that, for each step, a still reasonable number of 3D points is in consideration since the window size is practically limited depending on the available memory size. The partial results are subsequently merged together to obtain the results for the full scene. Accordingly, the approach represents a partitioning of the scene into subparts which, in turn, are extended by small padding regions at the borders and can be processed in parallel. For the sake of simplicity, we focus on two specific scenarios and select the one which is suited best with respect to the given data:

- For the more general scenario, we propose the use of a tiling approach. Taking a defined area within a horizontally oriented plane, e.g. a small quadratic area of  $10\text{ m} \times 10\text{ m}$ , allows a successive processing of data and thus also the analysis of huge point clouds at city-scale, where directly applying the methodology is intractable due to the computational burden with respect to computational effort and memory consumption.
- Without loss of generality, we may also take into account that the recently published benchmark datasets describe straight street sections with a length of approximately 160–200 m [24,25]. Consequently, the tiling approach can be substituted by a slicing approach, where the slices have a specified width, e.g. a width of 10 m, along the street direction and infinite extent along the two perpendicular directions.

Note that for both partitioning schemes, those 3D points within the small padding around the considered tile or slice are also used if they are within the neighborhood of other 3D points within the considered part of the scene in order to avoid artifacts at boundaries between tiles or slices. Due to the high point density of recent point cloud datasets, a padding with a width of 0.50 m is considered to be sufficient. As alternative to scene partitioning, streaming methods could be applied [73].

Consequently, we may repeat the proposed workflow consisting of (i) optimal neighborhood selection, (ii) feature extraction and (iii) classification independently for each tile or for each slice. Thus, large-scale urban 3D scene analysis is composed of successive steps which exploit the same methodology and could be parallelized if respective hardware is available.

## 5. Datasets

Since we focus on the issue of urban 3D point cloud classification and want to facilitate an objective comparison to other methodologies, we consider two publicly available and labeled 3D point cloud datasets representing densely sampled urban environments. Both datasets have been acquired in the city of Paris, France, via mobile laser scanning (MLS) systems:

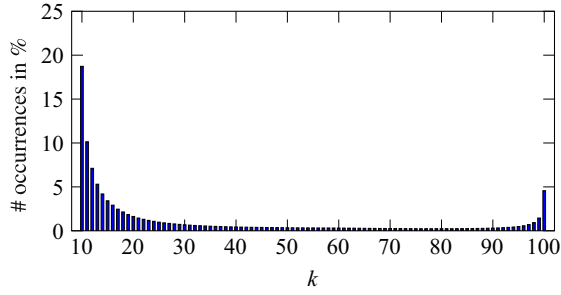
*Paris-rue-Madame database*<sup>1</sup> [24]: This 3D point cloud dataset has been acquired with the mobile laser scanning system L3D2 [74] equipped with a Velodyne HDL32. It contains 20 million points corresponding to a digitized street section with a length of approximately 160 m. A respective annotation has been conducted in a manually assisted way and includes both point-wise labels (26 different classes) and segmented objects (642 objects in total). This annotation relies on an initial segmentation based on elevation images [9] which is followed by a manual refinement [24].

*Paris-rue-Cassette database* [25]: This point cloud dataset has been acquired with the mobile laser scanning system called Stereopolis II [75] in January 2013. This system captures the local 3D geometry of the scene with two plane sweep lidars (Riegl LMS-Q120i) placed on each side of the vehicle in order to mainly observe the building façades with a centimeter accuracy and a 3D lidar (Velodyne HDL-64E) to observe the bottom part in between. The dataset contains 12 million points corresponding to a digitized street section with a length of approximately 200 m, and a manually assisted annotation is available which includes both point-wise labels and segmented objects. The respective annotation is based on recovering a regular 2D topology for the point cloud stream during data acquisition and an offline human interaction via a graph editing tool based on standard 2D image segmentation techniques [76]. A further extension in the form of 10 different zones with a total number of about 100 million points has been released in the scope of a recent contest [25].

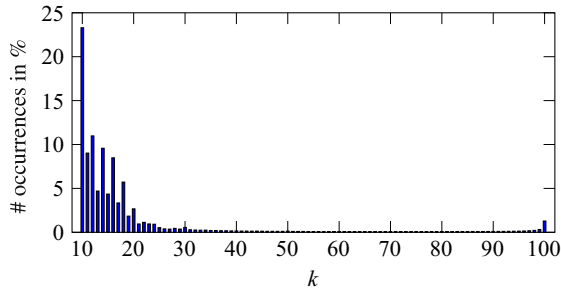
## 6. Experimental results

In the following, we first provide an in-depth analysis addressing the results of optimal neighborhood size selection for both involved datasets. Afterwards, we focus on the respective results for individual point classification and, finally, we demonstrate the performance of the new approach with respect to computational effort.

<sup>1</sup> *Paris-rue-Madame database: MINES ParisTech 3D mobile laser scanner dataset from Madame street in Paris.* ©2014 MINES ParisTech. MINES ParisTech created this special set of 3D MLS data for the purpose of detection-segmentation-classification research activities, but does not endorse the way they are used in this project or the conclusions put forward. The database is publicly available at <http://cmm.ensmp.fr/~serna/rueMadameDataset.html> (last access: 30 August 2014).



**Fig. 2.** Distribution of the assigned optimal neighborhood size  $k$  for all 3D points in the Paris-rue-Madame database.



**Fig. 3.** Distribution of the assigned optimal neighborhood size  $k$  for all 3D points in the Paris-rue-Cassette database.

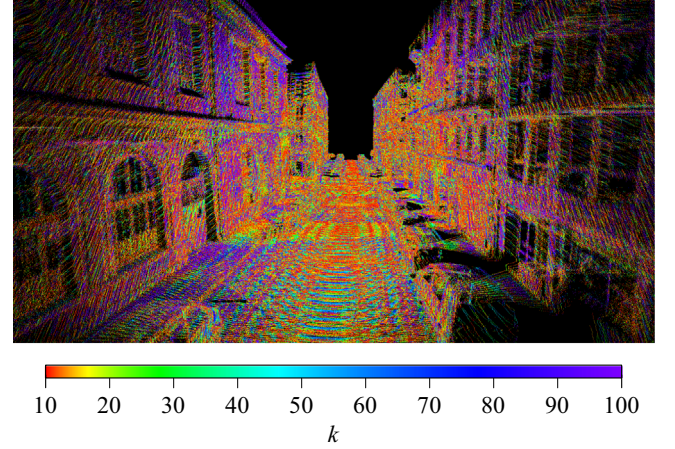
### 6.1. Optimal neighborhood size selection

In order to obtain more insights into the process of optimal neighborhood size selection, we first focus on a detailed analysis of the respective results. Note that each 3D point is assigned an individual value for  $k$  and that all integer values in  $[k_1, k_2]$  with  $k_1 = 10$  and  $k_2 = 100$  are taken into consideration. Since the upper boundary  $k_2$  has been selected for reasons of computational costs and may principally be set to an arbitrary value, we first consider the distribution of the parameter  $k$  across all 3D points. For this purpose, the respective distributions of the assigned optimal neighborhood size across all 3D points obtained for the Paris-rue-Madame database with 20 million 3D points and the Paris-rue-Cassette database with 12 million 3D points are visualized as histograms in Figs. 2 and 3. These figures clearly reveal a trend towards small values of  $k$ . Since the last bin in the histograms ( $k_2 = 100$ ) is likely to also represent those 3D points which might have a higher value than  $k_2 = 100$ , a small increase can be observed. However, the percentage of 3D points which are assigned an optimal neighborhood with less than  $k_2$  neighbors is 95.44% and 98.72% for the Paris-rue-Madame database and the Paris-rue-Cassette database, respectively. This shows that our selection of  $k_2$  is appropriate.

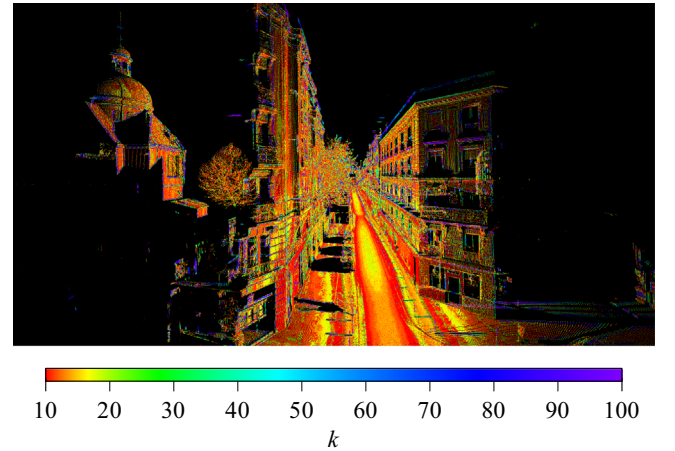
A qualitative visualization of the distributions of the assigned optimal neighborhood size across all 3D points of both involved datasets is depicted in Figs. 4 and 5. For the Paris-rue-Cassette database, a significantly smoother behavior can be observed.

### 6.2. Individual point classification

In order to assign a semantic label to each individual 3D point, we exploit only those 3D points with labels corresponding to the most dominant semantic classes *façade*, *ground*, *cars*, *motorcycles*/*2 wheelers*, *traffic signs/road inventory*, *pedestrians* and *vegetation* (Table 1), which represent a fraction of 99.81% of the Paris-rue-Madame database and 99.56% of the Paris-rue-Cassette database. All 3D points belonging to the other classes are removed since the respective number of samples per class is not considered to be



**Fig. 4.** Qualitative distribution of the assigned optimal neighborhood size  $k$  for all 3D points in the Paris-rue-Madame database.



**Fig. 5.** Qualitative distribution of the assigned optimal neighborhood size  $k$  for all 3D points in the Paris-rue-Cassette database.

representative. For training, we randomly select a small, balanced training set  $\mathcal{X}$  with 1,000 training examples per class, and the remaining data is used as test set  $\mathcal{Y}$ .

In order to allow a comparison to recent work, we design the evaluation scheme in analogy to [28]. Thus, our evaluation is based on (i) overall accuracy which indicates the performance of the classifier on the test set, (ii) recall which represents a measure of completeness or quantity, (iii) precision which represents a measure of correctness or quality, (iv)  $F_1$ -score which combines recall and precision with equal weights, (v) mean class recall which represents an averaged measure of completeness/quantity across all classes and (vi) a visual inspection of the derived results.

Evaluating our approach on both datasets, the overall accuracy is 88.82% for the Paris-rue-Madame database and 89.60% for the Paris-rue-Cassette database. The resulting recall and precision values as well as the corresponding  $F_1$ -scores are provided in Tables 2 and 3. Accordingly, mean class recall values of 83.53% and 81.78% are obtained for the Paris-rue-Madame database and the Paris-rue-Cassette database, respectively. Finally, a visual impression on the quality of the derived results for individual point classification is depicted in Figs. 6 and 7.

### 6.3. Computational effort

The experiments have been conducted on an Intel Core i7-3820 with 3.6 GHz and 64GB RAM. We use the proposed slicing approach,



**Table 1**

Number of points in the most dominant classes. These classes cover 99.81% of the Paris-rue-Madame database and 99.56% of the Paris-rue-Cassette database.

Class	Paris-rue-Madame database [24]	Paris-rue-Cassette database [25]
Façade	9,978,435	7,026,016
Ground	8,024,295	4,228,639
Cars	1,835,383	367,271
Motorcycles/ 2 wheelers	98,867	39,331
Traffic signs/ road inventory	15,480	45,105
Pedestrians	10,048	22,999
Vegetation	–	211,131

**Table 2**

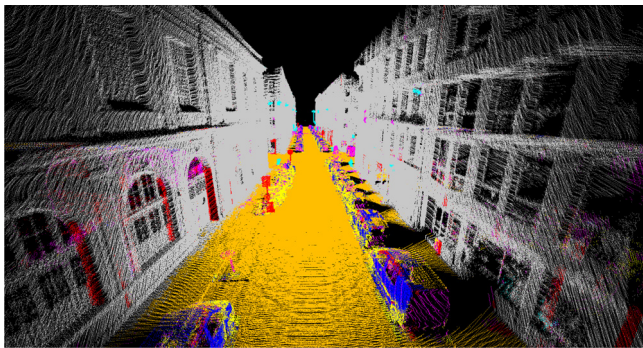
Recall, precision and  $F_1$ -scores for the Paris-rue-Madame database.

Paris-rue-Madame	Recall	Precision	$F_1$
Façade	0.9527	0.9620	0.9573
Ground	0.8650	0.9782	0.9182
Cars	0.6476	0.7948	0.7137
Motorcycles	0.7198	0.0980	0.1725
Traffic signs	0.9485	0.0491	0.0934
Pedestrians	0.8780	0.0163	0.0320

**Table 3**

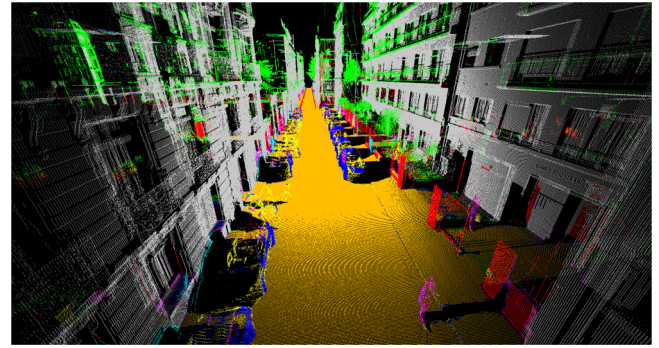
Recall, precision and  $F_1$ -scores for the Paris-rue-Cassette database.

Paris-rue-Cassette	Recall	Precision	$F_1$
Façade	0.8721	0.9928	0.9285
Ground	0.9646	0.9924	0.9783
Cars	0.6112	0.6767	0.6423
2 wheelers	0.8285	0.1774	0.2923
Road inventory	0.7657	0.1495	0.2501
Pedestrians	0.8225	0.0924	0.1661
Vegetation	0.8602	0.2566	0.3953



**Fig. 6.** Paris-rue-Madame database: classified point cloud with assigned semantic labels (façade: gray, ground: brown, cars: blue, motorcycles: yellow, traffic signs: red, pedestrians: pink). The points represented in cyan are those points which are not considered as the respective classes are not covered representatively. The noisy appearance results from individual point classification. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

where the slices have a width of 10 m and the padding has a width of 0.50 m. Whereas the prototype released with [28] is based on a full and straightforward Matlab implementation, our investigations revealed that a significant speedup can be achieved in two ways. Firstly, for the considered small point sets formed by up to  $k_2 = 100$  neighboring 3D points, a considerable speedup in the calculation of the respective 3D covariance matrices results from simply replacing



**Fig. 7.** Paris-rue-Cassette database: classified point cloud with assigned semantic labels (façade: gray, ground: brown, cars: blue, 2 wheelers: yellow, road inventory: red, pedestrians: pink, vegetation: green). The points represented in cyan are those points which are not considered as the respective classes are not covered representatively. The noisy appearance results from individual point classification. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

**Table 4**

Computational effort for processing the Paris-rue-Cassette database: the required processing times  $t_1$  for optimal neighborhood size selection,  $t_2$  for feature extraction,  $t_3$  for training on the small training set and  $t_4$  for testing on the respective test set are listed for different approaches. Note that  $t_1$  and  $t_2$  correspond to a successive processing of all slices, and that  $t_3$  and  $t_4$  do not change since they are not affected by our optimization.

Time	Prototype [28]	Optimized Matlab version	Proposed approach
$t_1$	27.45 h	10.90 h	2.11 h
$t_2$	11.84 h	10.75 h	4.28 h
$t_3$	~1–2 s	~1–2 s	~1–2 s
$t_4$	~90 s	~90 s	~90 s

the internal Matlab function `cov` with the respective vectorized straightforward implementation. Secondly, our new optimized approach can be used which has been implemented in C++ and, for comparison, we used the respective binaries in the test environment in Matlab. We consider both approaches. For the example involving the Paris-rue-Cassette database, the respective speedup achieved with the latter implementation clearly becomes visible in Table 4, where the processing times for the different subtasks are listed. Since the training phase will not change with larger datasets, the respective classification will only remain a question of computational and not human effort.

Taking a tile of defined size as a reference area would even allow us to extrapolate the resulting computational effort for data processing (which also accounts for those points in the padding). Thus, we would even be able to extrapolate the computational effort to full cities.

## 7. Discussion

In the experiments, it becomes apparent that particularly for the smaller classes a decrease in performance can be observed (Tables 2 and 3). This might indicate that those classes are still not covered representatively for the complexity of urban 3D scenes. However, the derived mean class recall values for both datasets indicate that completeness/quantity across all classes is relatively high compared to other approaches focusing on individual point classification [27,28]. Thus, the methodology is less prone to overfitting. Consequently, the decrease in performance might mainly arise from the similarity of local 3D structures belonging to respective classes. Additionally, for the Paris-rue-Cassette database, we can observe that vegetation is detected at the balconies



which is in contradiction to the reference labels, but maybe not always in contradiction to the real scene (Fig. 7). Note that the Paris-rue-Madame database contains more noise than the Paris-rue-Cassette database (Figs. 4 and 5) which might be a further reason for relatively low precision values obtained for the respective smaller classes (Table 2). In order to increase the precision values, introducing further features and/or multi-scale considerations seem to be necessary. Furthermore, due to the individual point classification, a noisy labeling can be expected which indeed can be observed in Figs. 6 and 7.

Concerning the computational complexity, we may consider the computation times required for processing different numbers of points. In Figs. 8 and 9, this is done for the separate slices (with the respective padding) as well as separately for optimal neighborhood size selection and feature extraction. It becomes apparent that optimal neighborhood size selection has a linear complexity for increasing numbers of considered 3D points, whereas feature

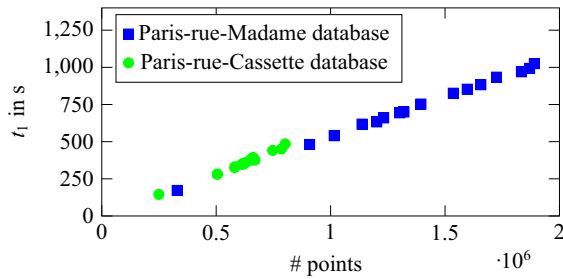


Fig. 8. Required computation times per slice for optimal neighborhood size selection: a linear behavior can be observed for increasing numbers of points.

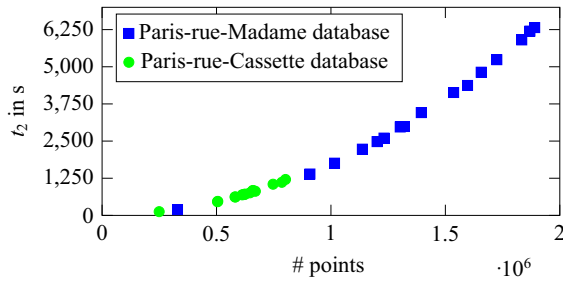


Fig. 9. Required computation times per slice for feature extraction: a non-linear behavior can be observed for increasing numbers of points.

extraction shows a non-linear dependency. In this case, we have a superposition of (i) a linear behavior for calculating 3D features in terms of basic geometric properties or eigenvalue-based features, (ii) a linear behavior for 2D features in terms of basic geometric properties or eigenvalue-based features and (iii) a non-linear behavior for 2D features based on the accumulation map.

The most crucial issue of the whole methodology remains an appropriate selection of the scale parameter  $k$ . The motivation of the applied approach is to avoid the use of empiric or heuristic knowledge on the scene with respect to neighborhood size and to obtain an automated, appropriate selection instead. Consequently, the approach is generally applicable and not tailored to a specific dataset. The consideration of individual neighborhoods even accounts for the idea that an optimal neighborhood size depends on the respective 3D structure and thus varies within a dataset. In order to provide further insights in addition to the clear trend of the scale parameter towards smaller values (Figs. 2 and 3), the behavior for the different classes is visualized in Fig. 10. Even though the analysis per class reveals a slight difference between the different classes, there is no clear indication of a characteristic which is specific for a certain class. Furthermore, we may state that the behavior of individual neighborhoods across a dataset indicates the quality of a dataset, since a much smoother behavior can be observed for the dataset with less noise (Figs. 4 and 5).

## 8. Conclusions

In this paper, we have presented a methodology for automated 3D scene analysis and its extension towards huge point clouds. The methodology generally requires a higher computational effort due to the consideration of individual 3D neighborhoods of optimal size which, in turn, is justified as it significantly improves the classification results in comparison to state-of-the-art approaches [28] and furthermore avoids human interaction guided by empiric or heuristic knowledge. Specifically, involving such optimal neighborhoods for feature extraction results in distinctive low-level geometric 3D and 2D features as important prerequisites for obtaining appropriate classification results. The further extension towards data-intensive processing via scene partitioning overcomes the limitation with respect to the computational burden and also allows large-scale 3D scene analysis. For two recently published point cloud datasets captured in urban areas, the derived results clearly reveal the potential of our methodology. For future work, we plan to involve spatial smoothing techniques,

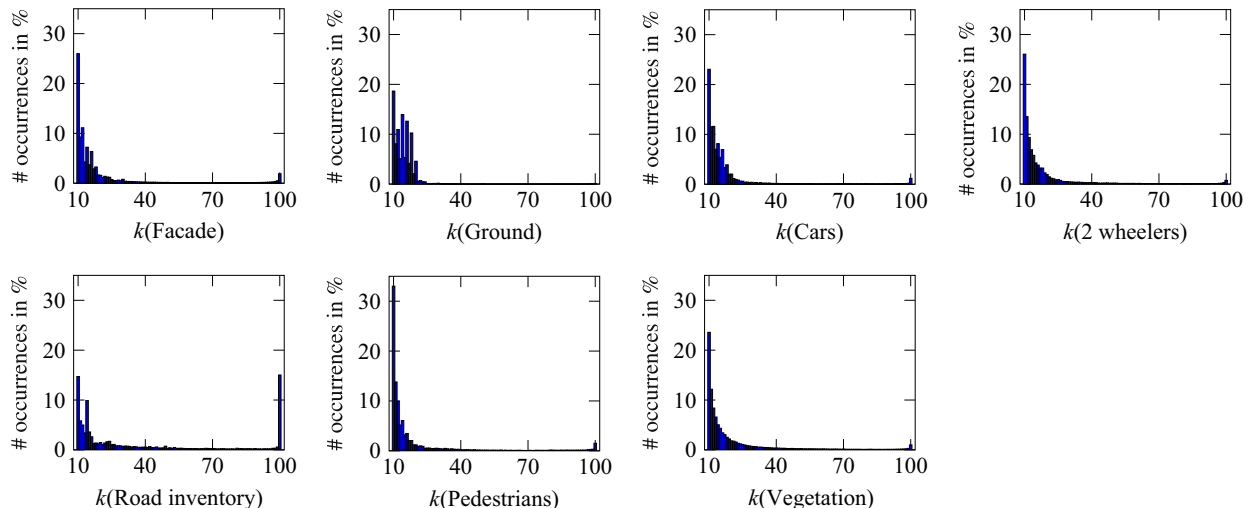


Fig. 10. Distribution of the assigned optimal neighborhood size  $k$  for different classes of the Paris-rue-Cassette database.

since neighboring 3D points tend to have correlated labels. Furthermore, an extended analysis up to object level would be desirable.

## Acknowledgements

The project was partially supported by KIT-GRACE, the Graduate School for Climate and Environment at the Karlsruhe Institute of Technology (KIT), and by the FP7 project IQmulus (FP7-ICT-2011-318787). Furthermore, a funding for a stay abroad of the first author was provided by the Karlsruhe House of Young Scientists (KHYS) at KIT in order to support the collaboration.

## References

- [1] Musialski P, Wonka P, Aliaga DG, Wimmer M, van Gool L, Purgathofer W. A survey of urban reconstruction. *Comput Graph Forum* 2013;32(6):146–77.
- [2] Zhou QY, Neumann U. Complete residential urban area reconstruction from dense aerial lidar point clouds. *Graph Model* 2013;75(3):118–25.
- [3] Lafarge F, Alliez P. Surface reconstruction through point set structuring. *Comput Graph Forum* 2013;32(2):225–34.
- [4] Zheng Q, Sharf A, Wan G, Li Y, Mitra NJ, Cohen-Or D, et al. Non-local scan consolidation for 3d urban scenes. *ACM Trans Graph* 2010;29(4):1–9 [Article no. 94].
- [5] Ceylan D, Mitra NJ, Zheng Y, Pauly M. Coupled structure-from-motion and 3d symmetry detection for urban facades. *ACM Trans Graph* 2014;33(1):2:1–15.
- [6] Pu S, Rutzinger M, Vosselman G, Oude Elberink S. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS J Photogramm Remote Sens* 2011;66(6):S28–39.
- [7] Velizhev A, Shapovalov R, Schindler K. Implicit shape models for object detection in 3d point clouds. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2012;1:3:179–84.
- [8] Bremer M, Wichmann V, Rutzinger M. Eigenvalue and graph-based object extraction from mobile laser scanning point clouds. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2013;11:5/W2:55–60.
- [9] Serna A, Marcotegui B. Detection, segmentation and classification of 3d urban objects using mathematical morphology and supervised learning. *ISPRS J Photogramm Remote Sens* 2014;93:243–55.
- [10] Boyko A, Funkhouser T. Extracting roads from dense point clouds in large scale urban environment. *ISPRS J Photogramm Remote Sens* 2011;66(6):S02–12.
- [11] Zhou L, Vosselman G. Mapping curbstones in airborne and mobile laser scanning data. *Int J Appl Earth Obs Geoinf* 2012;18(1):293–304.
- [12] Guan H, Li J, Yu Y, Wang C, Chapman M, Yang B. Using mobile laser scanning data for automated extraction of road markings. *ISPRS J Photogramm Remote Sens* 2014;87:93–107.
- [13] Serna A, Marcotegui B. Urban accessibility diagnosis from mobile laser scanning data. *ISPRS J Photogramm Remote Sens* 2013;84:23–32.
- [14] Kim HB, Sohn G. Random forests based multiple classifier system for power-line scene classification. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2011; XXXVIII-5/W12:253–8.
- [15] Vanegas CA, Aliaga DG, Benes B. Automatic extraction of Manhattan-world building masses from 3d laser range scans. *IEEE Trans Vis Comput Graph* 2012;18(10):1627–37.
- [16] Wurm KM, Kretschmar H, Kümmerle R, Stachniss C, Burgard W. Identifying vegetation from laser data in structured outdoor environments. *Robot Auton Syst* 2014;62(5):675–84.
- [17] Lafarge F, Mallet C. Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. *Int J Comput Vis* 2012;99(1):69–85.
- [18] Hebert M, Bagnell JA, Bajracharya M, Daniilidis K, Matthies LH, Mianzo L, et al. Semantic perception for ground robotics. In: *Proceedings of SPIE 8387, unmanned systems technology XIV*; 2012, p. 83870Y: 1–12.
- [19] Boulch A, Houllier S, Marlet R, Tournaire O. Semantizing complex 3d scenes using constrained attribute grammars. *Comput Graph Forum* 2013;32(5):33–42.
- [20] Hu H, Munoz D, Bagnell JA, Hebert M. Efficient 3-d scene analysis from streaming data. In: *Proceedings of the IEEE international conference on robotics and automation*; 2013, p. 2297–304.
- [21] Niemeyer J, Rottensteiner F, Soergel U. Contextual classification of lidar data and building object detection in urban areas. *ISPRS J Photogramm Remote Sens* 2014;87:152–65.
- [22] Nüchter A, Lingemann K. Robotic 3D scan repository. Jacobs University Bremen gGmbH and University of Osnabrück; 2011. <http://kos.informatik.uni-osnabrueck.de/3Dscans/> (last access: 30 August 2014).
- [23] Munoz D, Bagnell JA, Vandapel N, Hebert M. Contextual classification with functional max-margin Markov networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2009, p. 975–82.
- [24] Serna A, Marcotegui B, Goulette F, Deschaud JE. Paris-rue-Madame database: a 3d mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods. In: *Proceedings of the international conference on pattern recognition applications and methods*; 2014, p. 819–24.
- [25] Paparoditis N, Vallet B, Marcotegui B, Serna A. IQmulus & TerraMobilita contest – analysis of mobile laser scans (MLS) in dense urban environments. MATIS Laboratory, French National Mapping Agency (IGN) and Center for Mathematical Morphology (CMM), MINES ParisTech; 2014. <http://data.ign.fr/benchmarks/UrbanAnalysis/> (last access: 30 August 2014).
- [26] Weinmann M, Jutzi B, Mallet C. Describing Paris: automated 3d scene analysis via distinctive low-level geometric features. In: *Proceedings of the IQmulus workshop on processing large geospatial data*; 2014a, p. 1–8.
- [27] Weinmann M, Jutzi B, Mallet C. Feature relevance assessment for the semantic interpretation of 3d point cloud data. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2013;11:5/W2:313–8.
- [28] Weinmann M, Jutzi B, Mallet C. Semantic 3d scene interpretation: a framework combining optimal neighborhood size selection with relevant features. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2014;11:3:181–8.
- [29] Chehata N, Guo L, Mallet C. Airborne lidar feature selection for urban classification using random forests. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2009;XXXVIII-3/W8:207–12.
- [30] Mallet C, Bretar F, Roux M, Soergel U, Heipke C. Relevance assessment of full-waveform lidar data for urban area classification. *ISPRS J Photogramm Remote Sens* 2011;66(6):S71–84.
- [31] Tombari F, Salti S, Di Stefano L. Unique signatures of histograms for local surface description. In: *Proceedings of the European conference on computer vision*; 2010, p. 356–69.
- [32] Johnson AE, Hebert M. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans Pattern Anal Mach Intell* 1999;21(5):433–49.
- [33] Rusu RB, Marton ZC, Blodow N, Beetz M. Persistent point feature histograms for 3d point clouds. In: *Proceedings of the international conference on intelligent autonomous systems*; 2008, p. 119–28.
- [34] Rusu RB, Blodow N, Beetz M. Fast Point Feature Histograms (FPFH) for 3d registration. In: *Proceedings of the IEEE international conference on robotics and automation*; 2009, p. 3212–17.
- [35] Osada R, Funkhouser T, Chazelle B, Dobkin D. Shape distributions. *ACM Trans Graph* 2002;21(4):807–32.
- [36] Blomley R, Weinmann M, Leitloff J, Jutzi B. Shape distribution features for point cloud analysis – a geometric histogram approach on multiple scales. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2014;11:3:9–16.
- [37] Behley J, Steinhage V, Cremers AB. Performance of histogram descriptors for the classification of 3d laser range data in urban environments. In: *Proceedings of the IEEE international conference on robotics and automation*; 2012, p. 4391–8.
- [38] Jutzi B, Gross H. Nearest neighbour classification on laser point clouds to gain object structures from buildings. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2009;XXXVIII-1-4-7/W5.
- [39] West KF, Webb BN, Lersch JR, Pothier S, Triscari JM, Iverson AE. Context-driven automated target detection in 3-d data. In: *Proceedings of SPIE 5426*; 2004, p. 133–43.
- [40] Lalonde JF, Unnikrishnan R, Vandapel N, Hebert M. Scale selection for classification of point-sampled 3d surfaces. In: *Proceedings of the international conference on 3-D digital imaging and modeling*; 2005, p. 285–92.
- [41] Demantké J, Mallet C, David N, Vallet B. Dimensionality based scale selection in 3d lidar point clouds. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2011; XXXVIII-5/W12:97–102.
- [42] Guo B, Huang X, Zhang F, Sohn G. Classification of airborne laser scanning data using JointBoost. *ISPRS J Photogramm Remote Sens* 2014;92:124–36.
- [43] Waldhauser C, Hochreiter R, Otepka J, Pfeifer N, Ghuffar S, Korzeniowska K, et al. Automated classification of airborne laser scanning point clouds. In: *Kozziel S, Leifsson L, Yang XS, editors. Solving computationally expensive engineering problems*. New York, USA: Springer; 2014, p. 269–92.
- [44] Fehr D, Cherian A, Sivalingam R, Nickolay S, Morellas V, Papanikolopoulos N. Compact covariance descriptors in 3d point clouds for object recognition. In: *Proceedings of the IEEE international conference on robotics and automation*; 2012, p. 1793–8.
- [45] Khoshelham K, Oude Elberink SJ. Role of dimensionality reduction in segment-based classification of damaged building roofs in airborne laser scanning data. In: *Proceedings of the international conference on geographic object based image analysis*; 2012, p. 372–7.
- [46] Niemeyer J, Rottensteiner F, Soergel U. Conditional random fields for lidar point cloud classification in complex urban areas. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2012;1:3:263–8.
- [47] Pauly M, Keiser R, Gross M. Multi-scale feature extraction on point-sampled surfaces. *Comput Graph Forum* 2003;22(3):281–9.
- [48] Belton D, Lichti DD. Classification and segmentation of terrestrial laser scanner point clouds using local variance information. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2006;XXXVI-5:44–9.
- [49] Mitra NJ, Nguyen A. Estimating surface normals in noisy point cloud data. In: *Proceedings of the annual symposium on computational geometry*; 2003, p. 322–8.
- [50] Brodu N, Lague D. 3d terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: applications in geomorphology. *ISPRS J Photogramm Remote Sens* 2012;68:121–34.
- [51] Xiong X, Munoz D, Bagnell JA, Hebert M. 3-d scene analysis via sequenced predictions over points and regions. In: *Proceedings of the IEEE international conference on robotics and automation*; 2011, p. 2609–16.

- [52] Xu S, Vosselman G, Oude Elberink S. Multiple-entity based classification of airborne laser scanning data in urban areas. *ISPRS J Photogramm Remote Sens* 2014;88:1–15.
- [53] Secord J, Zakhor A. Tree detection in urban regions using aerial lidar and image data. *IEEE Geosci Remote Sens Lett* 2007;4(2):196–200.
- [54] Lodha SK, Fitzpatrick DM, Helmbold DP. Aerial lidar data classification using AdaBoost. In: *Proceedings of the international conference on 3-D digital imaging and modeling*; 2007. p. 435–42.
- [55] Shapovalov R, Velizhev A, Barinova O. Non-associative Markov networks for 3d point cloud classification. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2010;XXXVIII-3A:103–8.
- [56] Najafi M, Taghavi Namin S, Salzmann M, Petersson L. Non-associative higher-order Markov networks for point cloud classification. In: *Proceedings of the European conference on computer vision*; 2014. p. 500–15.
- [57] Shapovalov R, Vetrov D, Kohli P. Spatial inference machines. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2013. p. 2985–92.
- [58] Schindler K. An overview and comparison of smooth labeling methods for land-cover classification. *IEEE Trans Geosci Remote Sens* 2012;50(11):4534–45.
- [59] Lee I, Schenk T. Perceptual organization of 3d surface points. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2002;XXXIV-3A:193–8.
- [60] Filin S, Pfeifer N. Neighborhood systems for airborne laser data. *Photogramm Eng Remote Sens* 2005;71(6):743–55.
- [61] Linsen L, Prautzsch H. Local versus global triangulations. In: *Proceedings of eurographics*; 2001. p. 257–63.
- [62] Friedman JH, Bentley JL, Finkel RA. An algorithm for finding best matches in logarithmic expected time. *ACM Trans Math Softw* 1977;3(3):209–26.
- [63] Vosselman G, Klein R. Visualisation and structuring of point clouds. In: Vosselman G, Maas HG, editors. *Airborne and terrestrial laser scanning*. Dunbeath, UK: Whittles Publishing; 2010. p. 45–81.
- [64] Arya S, Mount DM, Netanyahu NS, Silverman R, Wu AY. An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. *J ACM* 1998;45(6):891–923.
- [65] Samet H. *Foundations of multidimensional and metric data structures*. St. Louis, USA: Morgan Kaufmann; 2006.
- [66] Muja M, Lowe DG. Fast approximate nearest neighbors with automatic algorithm configuration. In: *Proceedings of the international conference on computer vision theory and applications*, vol. 1; 2009. p. 331–40.
- [67] Shannon CE. A mathematical theory of communication. *Bell Syst Tech J* 1948;27(3):379–423.
- [68] Monnier F, Vallet B, Soheilian B. Trees detection from laser point clouds acquired in dense urban areas by a mobile mapping system. *ISPRS Ann Photogramm Remote Sens Spat Inf Sci* 2012;1-3:245–50.
- [69] Breiman L. Bagging predictors. *Mach Learn* 1996;24(2):123–40.
- [70] Efron B. Bootstrap methods: another look at the jackknife. *Ann Stat* 1979;7(1):1–26.
- [71] Breiman L. Random forests. *Mach Learn* 2001;45(1):5–32.
- [72] Criminisi A, Shotton J. *Decision forests for computer vision and medical image analysis*. *Advances in computer vision and pattern recognition*. London, UK: Springer; 2013.
- [73] Zhou QY, Neumann U. A streaming framework for seamless building reconstruction from large-scale aerial lidar data. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2009. p. 2759–66.
- [74] Goulette F, Nashashibi F, Abuhadrous I, Ammoun S, Laugeau C. An integrated on-board laser range sensing system for on-the-way city and road modelling. *Int Arch Photogramm Remote Sens Spat Inf Sci* 2006;XXXVI-1.
- [75] Paparoditis N, Papelard JP, Cannelle B, Devaux A, Soheilian B, David N, et al. Stereopolis II: a multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology. *Rev Fr Photogramm Télédélect* 2012;200:69–79.
- [76] Brédif M, Vallet B, Serna A, Marcotegui B, Paparoditis N. TerraMobilita/IQmulus urban point cloud classification benchmark. In: *Proceedings of the IQmulus workshop on processing large geospatial data*; 2014. p. 1–6.