

Generation Of Training Data For 3D Point Cloud Classification By CNN

Eike BARNEFSKE and Harald STERNBERG, Germany

Key words: CNN, 3D point clouds, training data, ground truth, web application

SUMMARY

Unstructured 3D point clouds are common (intermediate) results of surveys. They are the basis for modeling complex objects or geometric comparisons. For these complex analysis steps, a segmentation and/or classification of the original point cloud is necessary to process single objects or object parts of the scan. This segmentation and classification is nowadays more complex than data acquisition (approx. factor 1:8). In order to efficiently process mass data from laser scans, (partial) automation is necessary. Convolutional neural networks (CNN) are one method for efficient classification. To use CNN, a large amount of classified point clouds are necessary to train the parameters of the network. With a well-trained network reliable an unclassified point can reliable assigned to a certain class.

A method to generate a large amount of training data is the classification tool presented in this paper. Using our method, the point cloud is projected into the 2D space and images are calculated. In these images segments are generated, which are simply classified by a user. Based on the classified images, the classified 3D point cloud is created. First results of the classification tool are introduced and discussed in this work.

Generation Of Training Data For 3D-Point Cloud Classification By CNN

Eike BARNEFSKE and Harald STERNBERG, Germany

1. Introduction

Laser scanning is today a standard procedure in surveying, which is used for the fast, detailed and accurate acquisition of large objects. The measurement result is usually an unstructured point cloud of the object. This is used as a data basis for further evaluation and processing steps. Common applications of point clouds as data basis are the creation of surface models, (CAD-) models, Building Information Modeling or geometry comparisons. For all these applications, a variety of specialized algorithms have been developed to obtain additional information from the point cloud. In many cases it is attempted to organize the measurement data into (semantic) classes, with the aim of carrying out section-by-section or sub-object-oriented evaluations. A seemingly simple task for humans is the high-resolution classification of terrain by road, sidewalk or vegetation in order to create a surface plan based on the data of a point cloud. To solve this task automatically, an algorithm must detect features in the point cloud which reliably describe the desired classes. Once these features are detected, they must be found in the point cloud and a point can be assigned to a class. This task is very challenging when only a limited number of features of the point and adjacent points is available. Another field of application where a classification of point clouds is critical important is the reliable detection of wrong measured points. These false points need to be eliminated in the measurement data. KNOSPE & RICHTER (2018) present a street monitoring system in which all streets in the city of Essen are measured on a regular basis. The analysis of the data and the detection of damaged areas in the street are no longer possible by a human due to the amount of data. Here, algorithms must be able to detect areas of interest fast and reliably.

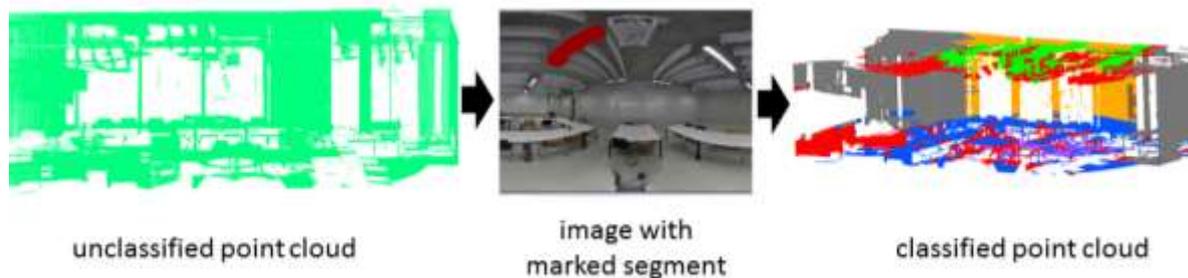


Figure 1: Overview of the classification tool: Having an unclassified point cloud (left) and projecting it into the 2D space to generate images. In these images segments with same features are calculated and classified by a user (middle). The classified images are used to calculate the classified 3D point cloud (right).

An evaluation of large laser scan data is no longer possible without automatic (pre-) evaluation. One method for this evaluation is to use artificial neural networks or convolutional neural networks (CNN). These CNN learn the optimal relationship between an unclassified point and a class on the basis of previously classified points. Here, the algorithm learns/determines the features whose points have to be assigned to a particular class. In order

to learn this relationship reliably, large amounts of training data (often manually classified point clouds) are necessary, which are given into the network. To efficiently generate a large and diverse number of training data for real laser scanner measurements, a classification tool has been developed that projects a colorized laser scan into a panoramic image space, forms a segment based on an edge detection algorithm, and calculates a classified point cloud (Figure 1).

This paper gives a brief overview of current approaches to classify point clouds with CNN. The classification tool for generating training data is presented. First training data is presented, discussed and further development steps are outlined

2. State of the art

The acquisition of single objects or extended areas as discrete points, point clouds or images is standard in many surveying projects. The necessary surveying systems, like LIDAR scanners, tachymeters or camera UAVs, have reached a high level of development. The geometry of real objects can be surveyed efficiently and reliably with high resolution and accuracy. Large amounts of data describing the object are generated in this way. In order to use these mass data to create models or do further analyses, it is necessary to organize these data in meaningful groups. This should be done by grouping points or pixels as segments and assign them into classes with similar characteristics. First approaches with learning algorithms to identify areas with the same geomantic properties in 3D point clouds (with different point densities) are presented in HACKEL ET AL. (2016). Precise segmentations and classifications of objects with CNN in images have been successfully performed by GIRSKICK (2015), REDMON ET AL. (2016) and many other computer vision scientists in the last few years. Classifications based on CNN are in some cases even superior to human classifications. Applying 2D CNN to unsorted 3D point clouds (or 3D objects in general) throws up a number of computing time, segmentation and classification challenges.

The success of CNN in 2D space led to attempts to use CNN for 3D data as well. Some of the first approaches project the 3D data into the 2D space and apply the established 2D CNN architectures. An approach where a 3D point cloud is projected into several 2D perspectives is presented in BOCHLCH ET AL. (2017). Here the pixel wise sematic classification is based on images that are combined and transformed back into 3D space. This method shows weaknesses for applications in which the entire 3D point cloud must remain geometrically unchanged. Through the transformation a separation of objects lying behind each other is often not possible and a generalization of the data is inevitable.

Many data preparation methods for 3D CNN use voxel structures. In those cases the voxels are arranged in an even grid similar to pixels in a digital photo. Adjacent information are combined by the voxels and the entire object space is represented by an even voxel grid. Using the *VoxNet* CNN as an example, it can be shown that areas in which no data must be represented by the grid as well (MATURANA & SCHERER, 2015). This leads to an increased computation effort. By determining the unoccupied voxels, this can be significantly reduced. Applications in which only certain objects, such as vehicles, pedestrians or bicyclists, shall be detected (e.g. for autonomous navigation) benefit from such a procedure that use only

occupied voxels, as described in ENGELCKE ET AL. (2016). The *PIXOR* CNN is based on the same ideas. It uses bird's-eye views to identify objects basis of the geometric information in the point cloud (YANG ET AL., 2018). Based on *VoxNet*, HACKEL ET AL. (2017), use a combination of voxel grids of different sizes to classify terrestrial laser scan data.

Besides these classification approaches in which 3D data (and in particular 3D point clouds) are adjusted to the existing architectures of 2D-CNN, there are algorithms in which the points of the 3D point cloud are transferred directly into CNN. Such a CNN is *PointNet* (QI ET AL., 2017), that assigns a semantic label to a set of points given to network (input block). Here, the input points can represent a single object or a part of a complete scene. This approach uses 3D data recorded and processed by the RGB-D measurement system (semi-synthetic data). The classification of complex 3D point scenes requires further improvements. The block wise point input into the network causes false classifications. An approach that has the objective of improving the scenes segmentation, determines characteristics of the input blocks. These characteristics are passed back to the next input blocks and improve the global classification (ENGELMANN ET AL., 2017). This effective CNN enhancement can be analyzed with synthetic indoor and outdoor scenes.

The CNN based classification methods outlined above use different types of measured (real) and synthetic 3D clouds for training and validation. Most of the current developments are based on synthetic data, because for one thing synthetic 3D data can be generated faster from models and secondly, reliable ground truth data is automatically available for training and evaluation. *ShapeNet* (CHANGE ET AL., 2015) is a big model database with more than 3 million classified and unclassified individual models that is used by many of the CNN presented above. *SUNCG* (SONG ET AL., 2016) is a 3D data set with more than 45 000 synthetic indoor scenes that is used for complex tasks. The data sets *KITTI* (GEIGER ET AL., 2012; GEIGER ET AL., 2013) and *vKITTI* (GAIDON ET AL., 2016) are some of the most popular data sets consisting of synthetic and measured data. Classified ground truth data is available for these data. The main applications of these data sets are mobile measured data, optimized for the investigations in research fields of autonomous driving and robotic. These data can also be used for applications in surveying, like mobile mapping. The *Sematic3D.Net* data set (HACKEL ET AL., 2017) consists of 31 high-quality and classified terrestrial panoramic laser scans. This data set consists of eight classes and contains only outdoor scenes. To the best of our knowledge, Set *Sematic3D.Net* is the data set that uses similar raw data as our tool. In order to have a high-quality and diverse data basis for CNN tests and optimizations, especially for surveying applications, the method presented in this paper is aimed at supplementing the available data sets.

3. Training data classification tool: Concept and implementation

A precise and reliable point cloud classification with CNN can be done if the network connections have been trained on a large and various set of training data. In addition to the CNN network architecture the training data is crucial for the classification performance. Conventional classification methods, where classification is done by manually selecting regions and assigning them to a specific class, are very time-consuming and depend on individual user skills. To generate training data efficiently and with a homogeneous quality,

we develop a web-based classification tool. For this purpose, colored laser scans are projected into two 2D images. In the web application images with highlighted segments of pixels with similar characteristic are displayed. A user will assign the segment to a certain class. In the final step, the classification results and the geometric data of the point clouds are merged and stored as a classified point clouds.

3.1. Preprocessing module

In the first module of the classification tool the 3D points, as polar coordinates, will be projected into the 2D space. Based on points in the 2D space two panoramic images with equidistant pixels are generated. For further data processing, the image will be considered as a data array and any pixel as an array cell. Some measured points fall into the same array cell because the point cloud density is higher than the pixel size is large. For these cells, a new color value is calculated from the mean value of all color values of the points. Depending on the grid resolution, some cells contain no points. In this case, the color value will be interpolated by the adjacent pixel. Our tool can also use scans where no color information is available. Here we use the ranges of the polar coordinates and calculated a depths image. The distances will be normalized and assigned to a gray value spectrum. The cell numbers (pixel coordinates) are stored with the polar coordinates in a database (basic data of each scan) and will be used for the calculation of the classified scans, later (Figure).

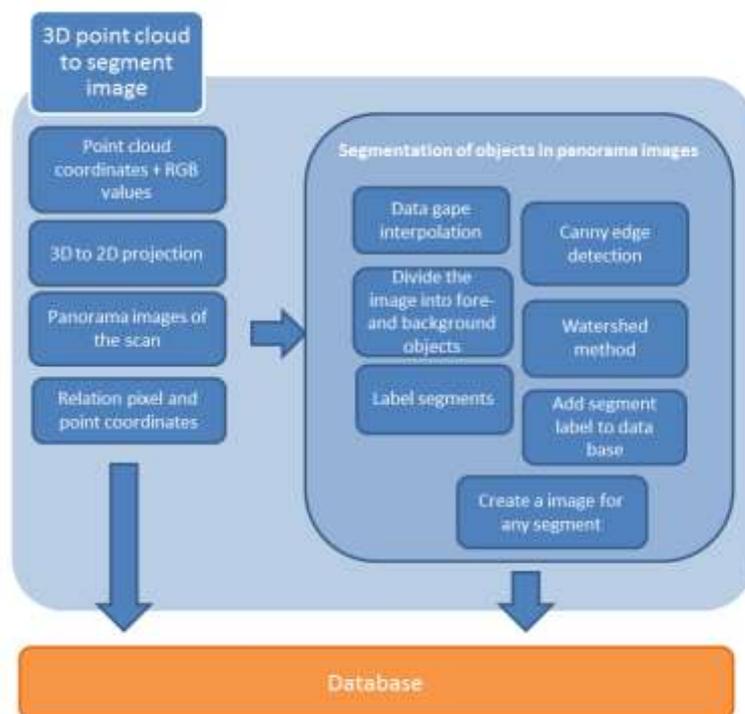


Figure 2: Workflow for colorized 3D point cloud to images with highlighted segments. On the left side the projection step is shown. On the right side an edge detection and watershed algorithm to calculate image segments is outlined.

The data arrays are used to calculate two panoramic images. Each image displays half of the scan. In the following, a variety of image processing operators can be applied to these images

in order to segment sections with the same semantic characteristics. For a first approach we used a simple edge detection algorithm. In the first image processing of this step a linear interpolation is preformatted to assign gray values of the next pixels to pixels that were not assigned to any gray values before. This calculation has no influence on the basic data and only serves for a more efficient use of the image processing algorithms. The interpolation leads to a more homogeneous image. In addition the calculation avoids a large number of small segments caused noise in the image.

Segments, which contain one object or an object part, are determined with the edge detector *auto_canny* by ROSEBROCK (2015A). This variant of the *canny* algorithm allows the automatic adjustment of the parameters for the detection of edges based on the gray value distribution in the current image. Especially with the wide variety of different scenes that should be processed in this application, this adjustment of the parameters is a special important for a target oriented use. In the next step, a background and a large number of foreground objects are created using the detected edges, following the algorithm of ROSEBROCK (2015B). The background object represents the boundaries between the individual foreground objects. For this application as few pixels as possible should be used as background, since these cannot be assigned to a particular segment. Each foreground object can now be assigned to a unique segment label and grouped as a closed segment with certain pixels (*OpenCV: components-and watershed-algorithm*). The segments are stored in a multi-dimensional array.

In the next step, the array with the stored segments is used to highlight the pixels belonging to a segment in one panorama image. In addition, a buffer of ten pixels is placed around the segments. These way pixels from background object can be assigned to segments as well. In the basic database the relation between point, pixel and segment label will be established. The images with the marked segments will be provided to the classification web page.

3.2. Classification module

The classification module is a webpage displaying the images with the marked segments. Each image is randomly loaded from the database to avoid tiredness from identical images during classification and to ensure that the entire data set is processed evenly. On the webpage, the marked segment in the image will be assigned to one out of 18 classes (Table 1). The segment is stored in a results database and a new image can be requested (Figure 3).

Table 1: Overview of the classes that can selected with the classification tool.

Code	Class	Code	Class
1	Table	10	Door
2	Wall	11	Wall decoration
3	Floor	12	Tree
4	Ceiling	13	Sign
5	Chair	14	Floor vegetation
6	Window	15	Building
7	Lamella	16	Car
8	Lamp	17	Pathway
9	Bin	18	Street

Currently, the webpage is in an extended test phase and can only be accessed from a university network. For a public launch of the web tool further quality checks have to be

applied. Harmful or accidental misclassifications are the greatest risks for a large and reliable dataset.

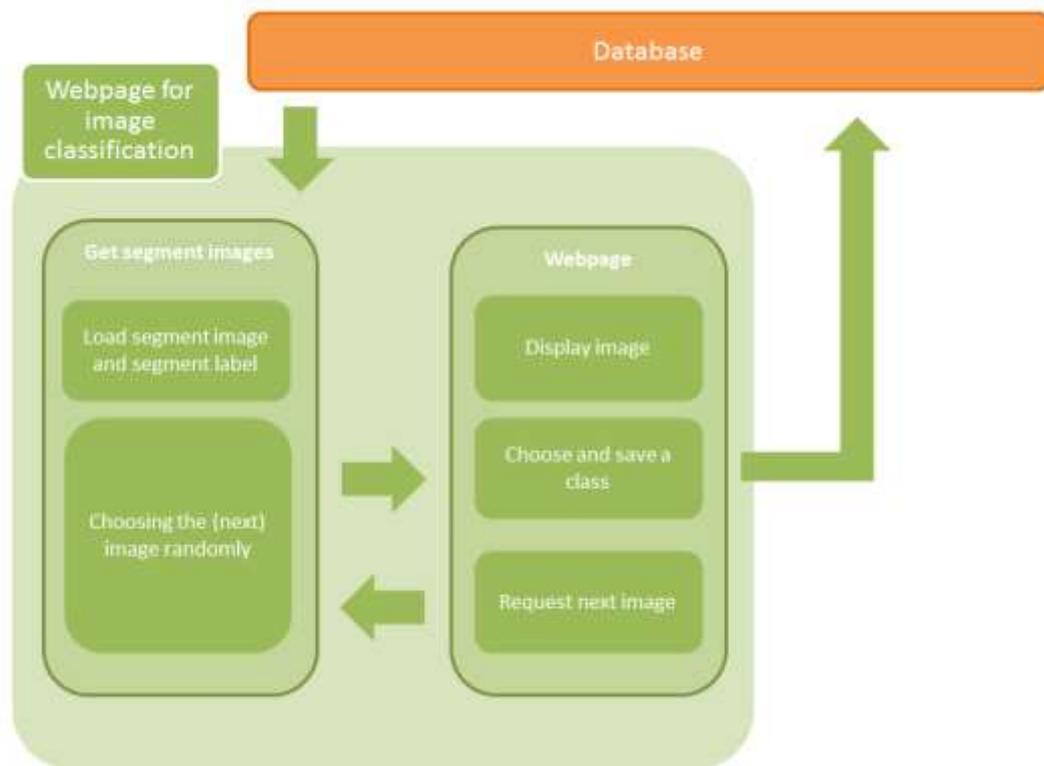


Figure 3: Workflow for image classification. The segment image is loaded to the webpage. A user is selecting the class of the marked segment and the section stored.

3.3. Classified point cloud generation module

The last module of the classification tool combines the classification from the image space with the coordinates of the points. This operation takes place for each scan individually after all data for the selected scan have been loaded from the basic database. All classifications for the marked segments are loaded from the result database. Usually there are several classifications for each segment. Due to errors or different interpretations for the marked segments, the classes are not always identical. In these cases, the segment is assigned the class that was selected most. Segments for which no class exists yet are assigned to the class 0. The classes are always assigned on the basis of the latest status of the result data base. Thus, this status is improved or changed with the continued use of the classification tool (Figure 4).

After all segments of the selected scan are assigned to an object class, the two databases are joined. Via the segment label the corresponding class is assigned to the points of the scan to the basic database.

The further visualization and analysis of the point cloud should be done using cartesian coordinates. Therefore the polar coordinates are transformed into cartesian coordinates. The origin of the coordinates remains in the center of the recording position. For further use, the

point cloud is stored as an entire point cloud in one file. Additionally, all points will be stored in separate files for each class. This storing is done in the ASCII format, so that point clouds can be loaded into many other programs without further conversions (Figure 4).

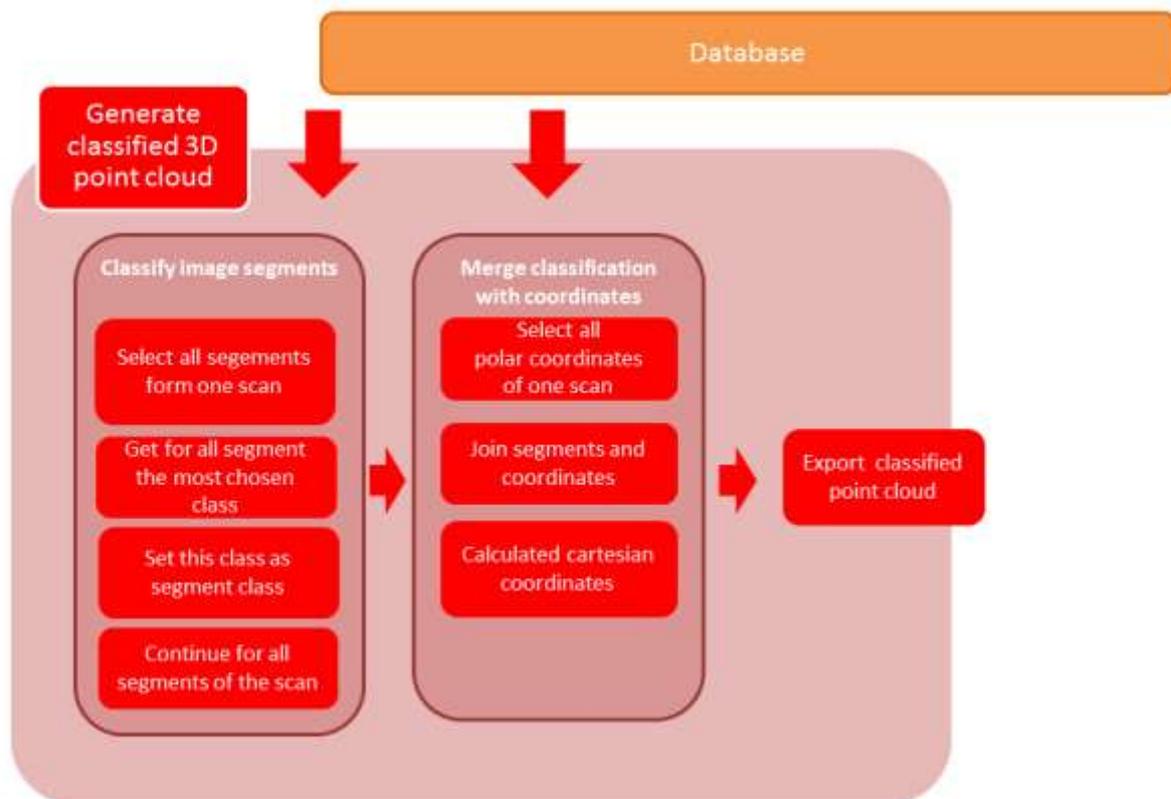


Figure 4: Workflow for creating a classified 3D point cloud using the classified segment images. The classified 3D point cloud is stored as complete point cloud and as individual point clouds by classes.

4. Results and Discussion

The function of the classification tool in the described configuration (chapter 3) is verified with a basic data set of 18 color scans. The focus of evaluation is on the interaction of the modules (data pre-processing, classification webpage, data post-processing) as well as on the central database. For the evaluation 6 187 images with marked segments were calculated. On average, 340 segments are calculated per scan and highlighted in the panorama images. The segmentation based on the edge detector algorithm is automatically performed by the software without setting scan-specific parameters. The classification was carried out in six hours. Three different human operators classified approximately 12 000 images for the first tests.

Furthermore, the performance of edge-based segmentation algorithms was briefly investigated. In the investigations images with a low resolution are used to create large and homogeneous segments in the images. The number of images is kept as low as possible so that a larger amount of data can be efficiently processed with the classification tool. This has the disadvantage that the segmentation of detailed objects is not possible. Objects that lie behind each other in the images are grouped together frequently. This becomes obvious by the

example of objects that lie behind trees or other fragile objects. These objects are only described by a few points. The image classification assigns the objects behind to the front object class (Figure 5). An individual adjustment of the parameters for the projection (resolution) and the edge detection (gray value differences) due to the scanned scene can result in satisfactory results with this segmentation method. This involves a large amount of manual work, so that for a detailed segmentation of many data sets different algorithms have to be applied.

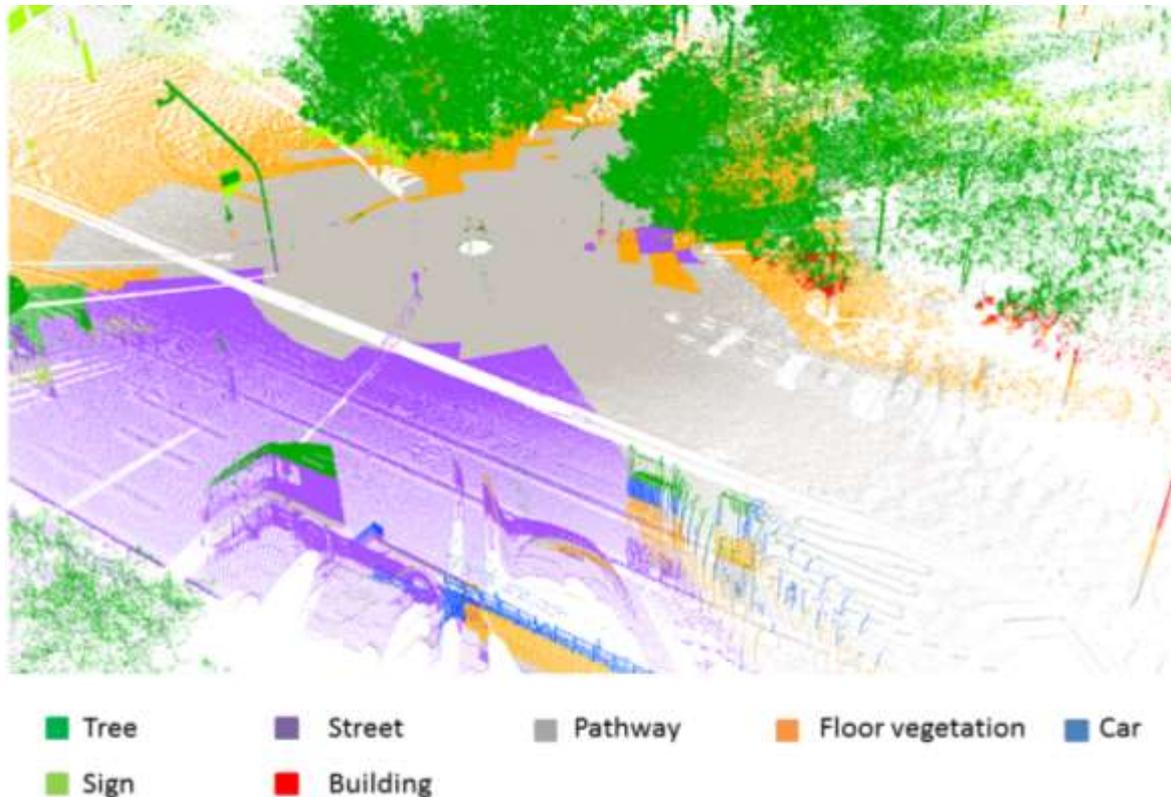


Figure 5: Classified 3D point cloud of an outdoor scene. The scene is showing a public place and streets in the HafenCity in Hamburg.

Figures 5 and 6 show the results of the classification tool. A separation of different bottom object classes like street, pathway or ground vegetation can be achieved. Small errors are caused by the low image resolution. The bottom object classes can be separated from the classes in the top scan zone. Trees and higher parts of buildings can be classified unambiguously. An insufficient classification performance is to be determined with objects in the middle of the outside scenes. These are usually false assigned to a bottom or top object classes. In this middle of the scene there are much smaller objects like signs. Small and medium-sized objects (e.g. lanterns or cars) that are more than 30 m away from the scanner are often assigned to the class of the surrounding object (e.g. streets or trees). Currently, a classification can only be made in the close-up range. For applications with indoor scenes, the 30 m range is not passed in our data sets (Figure 6). Here a more homogeneous and accurate classification is possible. Incorrect classifications occur when objects with a similar color

value lie behind each other. This is the case with white tables in front of white walls. A strict separation of small objects (e.g. tables and chairs) is often impossible.

Two alternative segmentation methods that improve the performance of the classification tool will be implemented in the next version and examined with high-resolution images. Graph-based segmentation of the images like FELZENSZWALB & HUTTENLOCHER (2004) and GAO (2016) can be effectively implemented into the existing structure of the classification tool. Advantages of graph-based segmentation are no training data is required or complex parameters are not to be selected. Studies of graph-based algorithms and other methods have shown (STUTZ 2015) that these are suitable for 2D object segmentation. In addition to the color values, the distance from object to scanner can be used as an additional feature. The second method to evaluate will be a pre-trained R-CNN. Adaptations for our problem must be implemented. Here CNN should not perform the classification yet, but determine similar features for segmentation.



Figure 6: Classified 3D point cloud of an indoor scene. Lecture hall with randomly arranged furniture.

5. Conclusion and Outlook

In cases if a processing of 3D point clouds is performed, the importance for an efficient and automatic evaluation strategy grows with the increase of data size. An evaluation with CNN could be a possible solution for this challenge. A further objective in point cloud analysis is to investigate whether data driven methods can achieve better results than model-based methods. To carry out investigations in this field of research, it is necessary to develop methods for the generation of training data in addition to improve the CNN. A classification tool for training data is presented in this paper. The results of the tool and possible approaches for further developments with graph- and CNN-based segmentation methods have been outlined and briefly discussed.

REFERENCES

- CHANG, C. X.; FUNKHOUSER, T.; GUIBAS, L.; HANRAHAN, P.; HUANG, Q.; LI, Z.; SAVARESE, S.; SAVVA, M.; SONG, S.; SU, H.; XIAO, J.; YI, L. & YU, F. (2015): Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.
- ENGELMANN, F., KONTOGIANNIA, T., HERMANS, A. & LEIBE, B. (2017): Exploring Spatial Context for 3D Semantic Segmentation of Point Clouds. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 716-124.
- FELZENSZWALB, P.F. & HUTTENLOCHER, D.P. (2004): Efficient graph-based image segmentation. *International journal of computer vision*, 59, 2, 167–181.
- GAIDON, A., WANG, Q., CABON, Y. & VIG, E. (2016): Virtual Worlds as Proxy for Multi-Object Tracking Analysis. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 4340-4349.
- GAO, L.; SONG, J.; NIE, F.; ZOU, F.; SEBE, N. & SHEN, H. T. (2016): Graph-without-cut: An Ideal Graph Learning for Image Segmentation. *Conference on Artificial Intelligence (AAAI)*, 2, 6.
- GEIGER, A., LENZ, P. & URTASUN, R. (2012): Are we ready for Autonomous Driving? The KITTI and Vision Benchmark and Suite. *Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 3354–3361.
- GEIGER, A., LENZ, P., STILLER, C. & URTASUN, R. (2013): Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*.
- HACKEL, T., WEGNER, J.D., & SCHINDLER, K. (2016): Fast Semantic Segmentation of 3D Point Clouds with Strongly Varying Densit. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences* 3, 3.
- HACKEL, T., SAVINOV, N., LADICKY, L., WEGNER, J.D., SCHINDLER, K. & POLLEFEYS, M. (2017): SEMANTIC3D.NET: A new large-scale point cloud classification benchmark. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 91–98.
- KNOSPE, F. & RICHTER, R. (2018): Erfassung, Analyse und Auswertung mobile Multisensorik im Straßenraum: Ein Erfahrungsbericht. In: *MST 2018 – Multisensortechnologie: Low-cost im Verbund, Beiträge zum 176. DVW-Seminar am 13. und 14. September in Hamburg, Wißner Verlag, German DVW Conference on multisensory technologies*.
- QI, C.R., SU, H., MO, K., & GUIBAS, L.J. (2017): PointNet: Deep learning on point sets for 3d classification and segmentation. *Computer Vision and Pattern Recognition (CVPR)*, IEEE, 77-85.

ROSEBROCK, A (2015A): Zero-parameter, automatic Canny edge detection with Python and OpenCV Tutorials, <https://www.pyimagesearch.com/2015/04/06/zero-parameter-automatic-canny-edge-detection-with-python-and-opencv/>, visited 15.01.2019.

ROSEBROCK, A. (2015B): Watershed OpenCV in Image Processing, Tutorials, <https://www.pyimagesearch.com/2015/11/02/watershed-opencv/>, visited 15.01.2019.

SONG, S., YU, F., ZENG, A., CHANG, A.X., SAVVA, M., & FUNKHOUSER, T. (2017): Semantic scene completion from a single depth image. *Conference on Computer Vision and Pattern Recognition (CVPR), IEEE*, 190–198.

STUTZ, D. 2015. Superpixel Segmentation: An Evaluation. *Pattern Recognition*, Springer International Publishing, 555–562.

YANG, B., LUO, W., & URTASUN, R. (2018): PIXOR: Real-Time 3D Object Detection From Point Clouds. *Computer Vision and Pattern Recognition (CVPR), IEEE*, 7652–7660.

CONTACTS

M.Sc. Eike Barnefske
Prof. Dr.-Ing. Harald Sternberg
HafenCity Universität Hamburg
Überseeallee 16
20457 Hamburg
GERMANY
Email: fistname.lastname@hcu-hamburg.de
Web site: www.hcu-hamburg.de