

Method for clustering and identification of objects in laser scanning point clouds using dynamic logic

Yevgeny Milanov

Peter the Great St. Petersburg Polytechnic University

Vladimir Badenko (✉ vbadenko@gmail.com)

Peter the Great St. Petersburg Polytechnic University

Vladimir Yadykin

Peter the Great St. Petersburg Polytechnic University

Leonid Perlovsky

Northeastern University

Research Article

Keywords: 3D laser scanning, clustering, LiDAR, data processing, heterogeneous digital data, mobile laser scanning data, point clouds, data mining, neural modeling fields, dynamic logic

Posted Date: March 19th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-248471/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Abstract

Today there is a gap between a presence of various new equipment on the market which provides streams of various digital data about the environment, in particular in the form of laser scanning point clouds, and the lack of adequate efficient methods and software for information extraction from such data. A solution to the problem of bridging this gap is proposed on the basis of neural modeling field theory and dynamic logic (DL). We present a DL-based method of extracting and analyzing information from hybrid point clouds, which include not only spatial coordinates and intensity, but also the color of each point, and can be from multiple sources including terrestrial, mobile and airborne laser scanning data. The proposed method is significant for creating a fundamental theoretical basis for new application algorithms and software for many new applications, including building information modeling, “smart city” environment, etc. The proposed method is fairly new to solving various problems related to extracting semantically rich information from a nontraditional type of digital data, especially hybrid point clouds created from laser scanning. This method will allow to significantly expand the existing boundaries of knowledge in the field of extraction and analysis of information from various digital data, because neural modeling field theory and DL can improve the performance of relevant calculations and close the existing gap in analysis of digital images.

1. State Of The Art

Currently, many modern remote sensing technologies, such as laser scanning technologies, generate three-dimensional point clouds [1–3]. These point clouds are a specific kind of digital imagery. In general, 3D digital image processing is a resource-intensive task that requires use of specially developed algorithms. The existing proposed solutions are related, among other things, to cognition modeling and artificial intelligence (AI) [4–6]. Algorithms for processing 3D point clouds often run into combinatorial complexity, which greatly complicates the task of classifying and dividing such an image into separate objects and assigning attributes to them [7]. Traditional algorithms based on mathematical logic for solving such a recognition problem use enumeration and comparison of all known test images (3D point clouds) with the investigated image [8]. It is easy to understand that, using even a small set of options for comparison, the number of operations would be gigantic. However, it should be noted that there have been successes in using such algorithms in combination with each other. For example, methods such as convolutional neural networks, K-means, decision trees, latent Dirichlet distribution, support vector machines, and deep learning are among them [9–13].

The search for possible innovations in 3D image processing showed that methods based on cognition modeling have been successful [14–16]. An original approach to cognition modeling based on the theory of neural modeling fields (NMF) and dynamic logic (DL) was developed by L.I. Perlovsky [17–19, 55]. This approach differs from the usual Aristotelian logic and is intended to overcome the shortcomings of formal logic associated with the combinatorial complexity (CC) of the corresponding algorithms. Instead of well-defined concepts such as “this is a window”, DL is a process “from vague to crisp”, realizing the causal principle of cognition, for example, in object recognition, when the current state of the NMF is

determined by the previous state [20]. NMF and DL have been developed based on data from psychology, philosophy and science about the main mechanisms of cognition, as well as on the analysis of modeling of cognitive functions in AI [17–20, 55]. As a result, Perlovsky's approaches make it possible to obtain new results on the formalization of the cognition process, for example, in [21]; a generalization of the theory of NMF and DL in the form of DL phenomena and cognition was obtained. The logical systems used in this case formulate the use of general terms, such as relations of generality, uncertainty, simplicity, teaching methods, the problem of maximum similarity with empirical content. It should be noted that currently available neural networks are too simple to explain human cognition. At the same time NMF is making a step towards explaining cognition and not just striving to improve methods of solving practical problems [22]. Whereas Aristotle's logic is a tool for formalizing previously known solutions, it is not suitable for solving new problems related to cognition. This is related to the lack of adaptability of the standard logic and the combinatorial complexity of the corresponding computations. In this regard, the structure of the NMF is built in accordance with current knowledge and ideas about the organization of the human brain. The universality of cognition and the extraordinary capabilities of the human brain are partially explained by dynamic logic. This paper presents the results of applying these approaches to clustering and classifying digital data in the form of point clouds that are formed as a result of 3D laser scanning.

No publications directly related to the use of NMF and DL methods for clustering and classifying point clouds of laser scanning were found. Therefore, the following is an overview of current research in the field of intelligent methods of extracting information from various digital data that may be relevant to the problem under consideration. The prospects of the NMF and DL method for the problem of clustering and classifying point clouds are also determined.

In [23–25] various methods used in biometrics are considered, indicating the methods, used functions and data sources. Open problems are the mathematical analysis of “soft” biometrics and statistical modeling, including processing of big data [23]. For example, the Indian project UIDAI had more than 1.2 billion subjects, this data required improved accuracy and performance of the biometrics algorithm. The NMF/DL method allows one to create statistical models and quickly work with big data without loss of accuracy, while simultaneously solving problems identified in [24], such as: what to combine, how to combine and when to combine.

The problem of road sign recognition is part of one of the most important challenges of the 21st century: autonomous driving [26–31]. Article [28] presents the authors' own methods and an analysis of recent publications with an indication of the methods used and an assessment of recognition success, while the authors use a test database of 11,000 characters. Recognition of road signs is important for solving the problems of inventorying signs on the country's roads and tracking their condition [29, 30], as well as for diagnosing the accessibility of buildings [31]. The main obstacles to effective recognition, as in the case of biometrics, are inhomogeneity in the signs themselves, lighting, and orientation. The NMF/DL method can help solve the processing problem in an acceptable time frame and will improve accuracy along with performance when using such large road sign databases.

Adequate formation of instructions for autonomous driving requires simultaneous detection in real time of all the most important elements of the environment. After recognizing road signs, traffic lights, other vehicles, pedestrians, etc., it is necessary to track their changes and combine them with information from all sensors of the vehicle to form control commands [32, 33]. The main difficulties are associated with adequate integration of such a huge volume of the data and their analysis in limited time. The “associative” aspect of the NMF/DL method holds great promise for solving this problem.

Controlling autonomous flight of UAVs (unmanned aerial vehicles) is a technology close to the above mentioned autonomous driving. It is also important for an autonomous UAV to make quick decisions based on the recognition of landscape objects [34–36]. UAVs are often used to create 3D city models based on scanning data [37–39]. In [40], the authors consider the construction of three-dimensional city models based on the LIDAR (Light Identification Detection and Ranging) technology due to the possibility of obtaining digital images of high density and high accuracy. The development of sensor technologies and corresponding algorithms make it possible to efficiently and accurately construct 3D models in the form of point clouds using photogrammetry methods only on the basis of 2D images from cameras installed on the UAV [30, 41–43]. DL algorithms can work with both LIDAR data and 2D images from cameras. In any case, the ability to use DL with LIDAR will be a significant advantage despite the large amount of data, as it provides an easier way to build 3D models. UAV images are often fuzzy and poorly structured due to unstable UAV motion [40] and the “associative” aspect of the NMF/DL method can also effectively help in addressing this problem.

The NMF/DL method due to its performance, can potentially cope with the huge volume of unstructured heterogeneous digital image data from a plethora of cameras at airports and railway stations, which requires radically new approaches in order to process [44, 45].

Other tasks for which effective analysis of digital images is critical and for which NMF/DL can work effectively include flood impact analysis [46]. For this problem, destroyed objects, especially residential buildings, are important [47, 48]. Destroyed buildings can be determined by the presence or absence of walls, modeled by planes [47]. The development of robust and accurate methods for automatic building detection and regularization using multisource data has been discussed in [49].

As stated in all of the above mentioned publications, any digital image, including a 3D laser scanning point cloud, usually contains various types of noise and clutter, which are a significant obstacle to the operation of many algorithms. When scanning a real environment, these noises are caused by the presence of fog or rain droplets, birds and insects flying by, debris, dust and leaves carried by the wind. The NMF/DL method can include a specification of the noise / obstacle model that will exclude the corresponding noise points, so that the rest of the points can be “cleanly” related to the underlying models.

Experience in the development of methods and algorithms for processing hybrid data of laser scanning and photography has shown that the resulting point clouds without noise and clutter are of great practical importance [50–53]. The main difficulties when working with such clouds are associated with

the fact that the initial data stream is very heterogeneous, in three-dimensional space only the surfaces of objects are obtained and point clouds are huge. Thus, an urgent task is to develop fundamentally new methods of processing and analyzing images in the form of clouds of laser scanning points. The purpose of this article is to develop methods for this type of data based on the neural modeling fields proposed and constructed by L.I. Perlovsky. The article presents the results that, based on the interpretation of a number of features of thought processes [17, 54], demonstrate overcoming the “curse of dimensionality” in the problems of object recognition in the clouds of laser scanning points.

2. Materials And Methods

The method was developed in accordance with the dynamic logic (DL) approach proposed by L.I. Perlovsky [17-19, 55]. The clustering-specific version of DL is used as opposed to the general DL method. Let us unravel Perlovsky's theory. Onwards we discuss the equations and main assumptions of the DL method used to build the algorithm. We use the following Gaussian likelihood measure:

$$\ell(n|m) = \frac{e^{-(X(n)-M_m)^T C_m^{-1} (X(n)-M_m)/2}}{\sqrt{(2\pi)^d |C_m|}} \quad (1)$$

where $X(n)$ is the input signal at data point n of N , C_m are the covariance matrices for models $m = 1, \dots, M$ and M_m are the coordinates for the center of the m 'th model (i.e. cluster) expressed in the same coordinates as $X(n)$. Gaussian likelihood is used because it is usually well fit with many naturally occurring densities. Generally, the data point $X(n)$ is in the form of a vector consisting of 3D spatial coordinates plus, optionally, intensity (I) or color (RGB), or both, or some features.

The proposed method (in its basic form) does not use any prior knowledge of the point cloud, therefore the model centers M_m are initialized randomly and so as to fall within the point cloud's boundary, yet not too close to it. Also, covariances are initialized randomly and so as to be large – on the order of the cloud's size.

Association variables, model rates (priors) and similarity measure are given, respectively, as follows:

$$f(m|n) = r_m \ell(n|m) \div \sum_{m' \in M} r_{m'} \ell(n|m') \quad (2)$$

$$r_m = \frac{1}{N} \sum_{n \in N} f(m|n) \quad (3)$$

$$L = \prod_{n \in N} \sum_{m \in M} r_m \ell(n|m) \quad (4)$$

The general DL equations from [17-19, 55] for models and parameters are replaced by more convenient and more suitable ones for identification and clustering tasks from [55]:

$$\mathbf{M}_m = \frac{1}{N_m} \sum_{n \in N} f(m|n) \mathbf{X}(n) \quad (5)$$

$$\mathbf{C}_m = \frac{1}{N_m} \sum_{n \in N} f(m|n) (\mathbf{X}(n) - \mathbf{M}_m)(\mathbf{X}(n) - \mathbf{M}_m)^T \quad (6)$$

Diagram 1 (see Diagram 1 in the Supplementary Files) below presents the iterative nature of the algorithm, meaning each successive iteration uses values calculated in the previous iteration. This justifies the cyclical interdependency of the above equations.

An outline is proposed in [55] for a mechanism for putting superfluous or redundant models to sleep and waking up or creating new models when more models are needed. This mechanism is under development and has not so far given entirely satisfactory results and is currently not used in our work. Its development and implementation would be useful in the future, since it would allow to automate the estimation of the number of models M , instead of having to preset this number manually upon initialization. For now we manually set this number at 25 (plus 1 cluster for clutter and noise). This number is chosen intuitively. Its value determines the resolution and size of the clusters; hence we choose it with regards to the order of magnitude of the physical size of objects that we want the algorithm to detect.

Several important tricks are used to significantly improve speed and precision of the algorithm, such as using normalized log-likelihood, but they are outside the scope of this article, since we do not intend to describe the algorithm in every detail, but only give its key mechanisms.

At first we used the algorithm in the obvious, natural way by running it once for some fixed amount of models, say $M=25$. We later discovered a technique that allowed for both better results and faster computation time while allowing for much larger data sets due to less memory usage: in the 1st run of the algorithm the entire point cloud is broken down into 5 clusters (plus noise), then, in the 2nd run, each of the 5 non-noise clusters are further broken down into 5 each, resulting in 25 non-noise clusters in total. This technique, which we named “5x5”, can handle much larger data sets with many more models M , because, first-of-all, during both runs the algorithm has 5x less models to handle, and, second-of-all, during the 2nd run it also uses 5x less input data in each computational branch of the algorithm. Aside from a faster calculation time, this also means we use 5x less computer memory (comparing peak load) with the 5x5 technique.

3. Results And Discussion

Point clouds from several free sources were used. Outside laser scanning point clouds of a European village (Bildstein, Austria) were obtained from the database Semantic3D (bildstein_station1_xyz_intensity_rgb). Another point cloud (an inside scan of an office space) was obtained from ISPRS's case study #5. Our own scans of geometric blocks were used as well.

Computations were carried out in MatLab R2018a on a 64-bit Windows 10 Pro with 64 GB RAM, an 8-core Intel Xeon E5 CPU and an NVIDIA Quadro P6000 GPU, with the latter relied on for most performance-heavy calculations.

Though LiDAR scans provide XYZIRGB coordinates, we decided to use only the 3 spatial coordinates (XYZ) because color and intensity interfere too heavily with clustering, as many changes in color due to shadows, different materials and angle of incidence do not correspond to meaningful cluster differentiation. We demonstrate this in Fig. 1, where in the original shots the color scale corresponds to intensity (I), while the coloring in the resulting shot corresponds to clustering done over all 7 of the XYZIRGB coordinates.

As can be discovered in Fig. 1, intensity of a point depends a lot on the angle of incidence of the laser beam, which is useful for clustering and discerning a roof from a façade (as seen by the booth), but when a façade wall is round, this is not useful. The same is true in the case of the back side of the car on the left of the image (Fig. 1).

Below in Fig. 2 we present the results of clustering the Bildstein scan into 25 clusters + 1 noise cluster using both techniques from “Materials and methods”: 5x5 and “in one go”. Note that light gray represents the noise cluster in all figures.

Instance 1/2 is the intermediate result, and we must look at instance 2/2 of the 5x5 technique, which shows that the 5x5 technique results in slightly better clustering (and much faster), when compared to the “in one go” method. Still, neither method is ideal: for example, parts of trees and buildings cluster together, while those objects themselves are split into different clusters. It must be pointed out that the algorithm proposed is useful only for clustering and not classification, and hence, in its basic form, is not intelligent enough to distinguish objects.

Furthermore, the positioning of the laser scanner has an enormous effect on the clustering. Local density of points is orders of magnitude higher at distances close to the scanner, and this has a large effect on the final result of the algorithm, since the weight of each cluster strongly depends on the relative number of points in it. In Fig. 2 we can easily identify where the scanner was placed, as it has many small-volume high-density clusters around it, and a result of the radial dispersion of laser rays from that center to the edges of the image can be seen.

A density readjustment through downsampling, voxelizing or other methods could be performed to resolve this issue. Having multiple scans available from different points of view could also improve the result.

We tried downsampling, as well as removing the ground, both separately and together, though to no avail, which is a big surprise.

Additionally, we suspected that since the algorithm involves gradient ascent, it may stop at some local maximum and not a global one. In many cases this is generally avoided due to the vague-to-crisp aspect of the DL approach [17–19, 55]. Nonetheless, this phenomenon was tested by running the algorithm 50

times, each time using randomly initiated cluster centers, and comparing final likelihood function values. We did not notice any strong difference in the results, which actually illustrates a robustness of our algorithm.

We also tested our DL-based algorithm on a subset (1 room) of an ISPRS point cloud (Case Study #5). Below in Fig. 3 we demonstrate the results using the 5x5 technique. Noise, as previously, is colored gray.

The most interesting and unexpected thing about these results is how well the tabletops and several other entities are clustered and separated from other objects, though sometimes the tabletops belong to several different clusters. This can be seen more clearly from different points of view in Fig. 4, where, additionally, the tabletops are manually gathered into one cluster and recolored in blue for ease of visualization. The divider between tables is visualized in similar fashion and is colored in white. All other entities, which were clustered mediocrely, are removed from the visualization for clarity.

A comparison of clustering results of the outside (Bildstein) and indoor (ISPRS) scans shows that the DL algorithm works better on indoor data, where objects are separated better. There is less of chaotic splitting and overlapping, e.g., when half a tree and half a building are clustered together while the remainders of these objects are in other cluster. Splitting, however, is better than overlapping, as this can be corrected by merging clusters, as was done with the tabletop and presented above.

Relative to other methods, quick computation time of the proposed algorithm is one of the main advantages of using DL. For example, in clustering the 29.7 million points of the Bildstein point cloud over the 3 spatial dimensions, if the number of models (clusters) is preset to $M = 25 + 1$, then 1 iteration of the main loop of the algorithm is computed in just 34 seconds on our computer configuration. Moreover, setting the maximum main loop iterations to 10–20 generally achieves a decent result, but based on our experience we recommend 40–50. Halving the number of iterations from 40 to 20 reduces computation time by 2x gives a loss of the shifted log-likelihood of around 4%

In the same way, the 5x5 technique (restricted to 40 iterations per branch/instance) takes 512 seconds on our computer. Dividing this by 40, we obtain just 12.8 seconds, which is 2.7x faster than the 34 seconds of the first technique.

According to the DL theory, the computation time for a DL-based algorithm should be linear in the number of points in the cloud, as well as in the number of parameters and clusters. However, our test computations seem to give better results: a sublinear time in all three numbers. This may be explained by use of a graphics processing unit (GPU) on our computer, without which our computation times were linear in the number of points, clusters and parameters. It is possible that computation time is in fact sublinear with a GPU up to a certain data and model size, beyond which the computation time tends towards an asymptote, i.e. becomes linear. But this fact is difficult to verify rigorously, because the GPU would need to be utilized in several capacities.

4. Comparison With Other Methods

It is important to compare our results to other algorithms with similar goals, e.g. K-means clustering and MeanShift. We used MatLab's internal K-means algorithm, while for MeanShift we used an algorithm from the internet by Bryan Feldman from 2006, which, notably, could be improved significantly, as the code is suboptimal. In addition, transferring computation of both these algorithms to the GPU would speed up calculations significantly. Nonetheless, for an initial comparison we used these algorithms as-is. We use the \approx 30-million-point Bildstein outside scan with just the XYZ coordinates.

K-means was tested in both "5x5" and "in one go" techniques, analogously to how we tested our algorithm. The tests' runtimes were 7.3 and 11.8 minutes, respectively. Both time results are similar to those of the analogous DL computations. However, in both cases the computations failed to converge in 100 iterations, which not only implies that the result will be subpar, but that such an algorithm is not suited for this kind of input data. Indeed, a plot of the results shows that the resulting clustering is absolutely unfitting, because the border lines are too straight and the clusters do not at all correspond to real-life objects: (see Figure 5)

Repeating the test with RGB color and intensity did not give better results.

For MeanShift we set bandwidth $h = 12$, which gives 35 clusters. Runtime was 9.5 minutes, which is very favorable (comparable with DL), considering the suboptimal code. However, the results are useless: (see Figure 6)

Another disadvantage of the aforementioned methods is that they do not, to our knowledge, allow for a noise/clutter cluster. Regardless, allowance for such a cluster would not noticeably improve results.

5. Conclusions

The results of applying the DL-based clustering method and algorithm we developed to laser scanning point clouds are described. Two examples were used for testing: scans of a village street and inside an office space. Testing of the algorithm showed that it is able to work quickly with clouds of millions of points in size. The best results were obtained by taking into account only the 3 spatial coordinates, i.e. ignoring the color and intensity of reflected laser beams. However, these parameters could be useful in other cases, for example, for scans of historical heritage sites and agricultural fields.

Two clustering approaches have been tested. In the first approach, segmentation is carried out simultaneously for 25 models (plus noise) in 1 run of the algorithm (the "in one go" approach). In the second approach we use the so-called 5x5 technique, where a point cloud is also clustered into 25 models + noise. However, in the second approach 2 consecutive runs of the algorithm are needed. During the first run an initial segmentation into 5 clusters + noise is performed. During the second run each of the 5 initial non-noise clusters is divided into a further 5. Testing with the original Bildstein point cloud as input shows that the 5x5 technique allows clustering to be performed 2.7 times faster than with the "in one go" approach while preserving the quality of the result (or even beating it). The 5x5 DL technique clusters the 29.7 million points of the Bildstein scan in only about 8 minutes.

The above-adequate segmentation of the dividers and countertop in the ISPRS office space scan stands out particularly, giving a better result than with Bildstein.

DL-based clustering results in point clouds being divided into clusters that make sense and those that don't. To overcome this disadvantage and to improve the result, additional approaches have been tested: downsampling, removing the ground, as well as a combination of the two. Surprisingly, this shows no improvement. The possibility that during gradient ascent a non-global local maximum is reached has also been tested, yet our experiments refute these doubts with high certainty: the algorithm was run 50 times with arbitrary initializations, with approximately the same final likelihood resulting each time.

Declarations

Funding

The research is partially funded by the Ministry of Science and Higher Education of the Russian Federation as part of World-class Research Center program: Advanced Digital Technologies (contract No. 075-15-2020-934 dated 17.11.2020)

Author information

Affiliations

Yevgeny Milanov, Vladimir Badenko, Vladimir Yadykin: Peter the Great St.Petersburg Polytechnic University, Polytechnicheskaya 29, 195251 St.Petersburg, Russia; e-mails: calc881@hotmail.com, vbadenko@gmail.com, v.yadikin@gmail.com, Leonid Perlovsky: Northeastern University, Boston, Massachusetts, United States; e-mail: lperl@rcn.com

Contributions

Yevgeny Milanov: algorithm&software development and computer experiments, draft version editing; Vladimir Badenko: research supervising, final editing and testing; Vladimir Yadykin: data obtaining for computer experiments, interpretation of results, algorithm development; Leonid Perlovsky: mail idea, key algorithm, final editing. All authors have read the article, partially edited it and agree with its contentio

Corresponding author: Vladimir Badenko

Correspondence to: vbadenko@gmail.com

Ethics declarations

Ethics approval

We further confirm that any aspect of the work covered in this manuscript that has involved either experimental animals or human patients has been conducted with the ethical approval of all relevant

bodies and that such approvals are acknowledged within the manuscript.

Consent to participate

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

Consent for publication

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing, we confirm that we have followed the regulations of our institutions concerning intellectual property.

Conflict of interest

The authors declare no competing interests.

Data and code availability

The results in this work is applicable. The original experiment data are not provided.

References

1. Kim Y, Nguyen CHP, Choi Y (2020) Automatic pipe and elbow recognition from three-dimensional point cloud model of industrial plant piping system using convolutional neural network-based primitive classification. *Automation in Construction*, 116: 103236.
<https://doi.org/10.1016/j.autcon.2020.103236>
2. Xia S, Chen D, Wang R, Li J., Zhang X (2020) Geometric primitives in LiDAR point clouds: A review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13: 685-707.
<https://doi.org/10.1109/JSTARS.2020.2969119>
3. Trujillo-Jiménez MA, Navarro P, Pazos B, Morales L, Ramallo V, Paschetta C, De Azevedo S, Ruderman A, Pérez O., Delrieux C, Gonzalez-José R (2020) body2vec: 3D Point Cloud Reconstruction for Precise Anthropometry with Handheld Devices. *Journal of Imaging*, 6(9): 94.
<https://doi.org/10.3390/jimaging6090094>
4. Volk R, Luu TH, Mueller-Roemer JS, Sevilmis N, Schultmann F (2018) Deconstruction project planning of existing buildings based on automated acquisition and reconstruction of building information. *Automation in construction*, 91: 226-245. <https://doi.org/10.1016/j.autcon.2018.03.017>
5. Hoiem D, Savarese S (2011) Representations and techniques for 3D object recognition and scene interpretation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(5): 1-169.
<https://doi.org/10.2200/S00370ED1V01Y201107AIM015>

6. Opitz R, Limp WF (2015) Recent developments in high-density survey and measurement (HDSM) for archaeology: implications for practice and theory. *Annual Review of Anthropology*, 44: 347-364. <https://doi.org/10.1146/annurev-anthro-102214-013845>
7. Chen X, Kundu K, Zhu Y, Ma H, Fidler S, Urtasun R (2017) 3d object proposals using stereo imagery for accurate object class detection. *IEEE transactions on pattern analysis and machine intelligence*, 40(5): 1259-1272. <https://doi.org/10.1109/TPAMI.2017.2706685>
8. Zhang Z, Li J, Guo Y, Li X, Lin Y, Xiao G, Wang C (2019) Robust procedural model fitting with a new geometric similarity estimator. *Pattern Recognition*, 85: 120-131. <https://doi.org/10.1016/j.patcog.2018.07.027>
9. Liu W, Sun J, Li W, Hu T, Wang P (2019) Deep learning on point clouds and its application: A survey. *Sensors*, 19(19): 4188. <https://doi.org/10.3390/s19194188>
10. Hashimoto M, Akizuki S, Takei S (2016) A survey and technology trends of 3D features for object recognition. *IEEJ Transactions on Electronics, Information and Systems*, 136(8): 1038-1046. <https://doi.org/10.1541/ieejeiss.136.1038>
11. Santoso F, Garratt MA, Pickering MR, Asikuzzaman M (2015) 3D mapping for visualization of rigid structures: A review and comparative study. *IEEE Sensors Journal*, 16(6): 1484-1507. <https://doi.org/10.1109/JSEN.2015.2498940>
12. Reiterer A, Wäschle K, Störk D, Leydecker A, Gitzen N (2020) Fully Automated Segmentation of 2D and 3D Mobile Mapping Data for Reliable Modeling of Surface Structures Using Deep Learning. *Remote Sensing*, 12(16): 2530. <https://doi.org/10.3390/RS12162530>
13. Chen S, Tan D (2018) A SA-ANN-based modeling method for human cognition mechanism and the PSACO cognition algorithm. *Complexity*, 2018: 6264124. <https://doi.org/10.1155/2018/6264124>
14. Gregory Trafton J, Harrison AM (2011) Embodied spatial cognition. *Topics in cognitive science*, 3(4): 686-706. <https://doi.org/10.1111/j.1756-8765.2011.01158.x>
15. Qi C, Li M, Wang Q, Zhang H, Xing J, Gao Z, Zhang H (2018) Facial expressions recognition based on cognition and mapped binary patterns. *IEEE Access*, 6: 18795-18803. <https://doi.org/10.1109/ACCESS.2018.2816044>
16. Cui C, Lin P, Nie X, Yin Y, Zhu Q (2017) Hybrid textual-visual relevance learning for content-based image retrieval. *Journal of Visual Communication and Image Representation*, 48: 367-374. <https://doi.org/10.1016/j.jvcir.2017.03.011>
17. Perlovsky LI (2016) Physics of the mind. *Frontiers in systems neuroscience*, 10: 84. <https://doi.org/10.3389/fnsys.2016.00084>
18. Perlovsky LI (2006) Toward physics of the mind: Concepts, emotions, consciousness, and symbols. *Physics of Life Reviews*, 3(1): 23-55. <https://doi.org/10.1016/j.plrev.2005.11.003>
19. Perlovsky LI (2007) Neural networks, fuzzy models and dynamic logic. *Studies in Fuzziness and Soft Computing*. 209(3): 363-386. https://doi.org/10.1007/978-3-540-37522-7_17
20. Fontanari JF, Tikhanoff V, Cangelosi A, Ilin R, Perlovsky LI (2009) Cross-situational learning of object-word mapping using Neural Modeling Fields. *Neural Networks*, 22(5-6): 579-585.

<https://doi.org/10.1016/j.neunet.2009.06.010>

21. Kovalerchuk B, Perlovsky L (2008) Dynamic logic of phenomena and cognition. In 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence) (pp. 3530-3537). IEEE. <https://doi.org/10.1109/IJCNN.2008.4634302>
22. Schoeller F, Perlovsky L, Arseniev D (2018) Physics of mind: experimental confirmations of theoretical predictions. *Physics of Life Reviews*, 25: 45-68.
<https://doi.org/10.1016/j.plrev.2017.11.021>
23. Dantcheva A, Elia P, Ross A (2015) What else does your biometric data reveal? A survey on soft biometrics. *IEEE Transactions on Information Forensics and Security*, 11(3): 441-467.
<https://doi.org/10.1109/TIFS.2015.2480381>
24. Singh M, Singh R, Ross A (2019) A comprehensive overview of biometric fusion. *Information Fusion*, 52: 187-205. <https://doi.org/10.1016/j.inffus.2018.12.003>
25. Drozdowski P, Rathgeb C, Busch C (2019) Computational workload in biometric identification systems: an overview. *IET Biometrics*, 8(6): 351-368. <https://doi.org/10.1049/iet-bmt.2019.0076>
26. Hu Z (2013) Intelligent road sign inventory (IRSI) with image recognition and attribute computation from video log. *Computer-Aided Civil and Infrastructure Engineering* 28(2): 130-145.
<https://doi.org/10.1111/j.1467-8667.2012.00768.x>
27. Li F, Lehtomäki M, Elberink SO, Vosselman G, Kukko A, Puttonen E, Chen Y, Hyppä J (2019) Semantic segmentation of road furniture in mobile laser scanning data. *ISPRS journal of photogrammetry and remote sensing*, 154: 98-113. <https://doi.org/10.1016/j.isprsjprs.2019.06.001>
28. Balali V, Golparvar-Fard M (2016) Evaluation of multiclass traffic sign detection and classification methods for US roadway asset inventory management. *Journal of Computing in Civil Engineering* 30(2): 04015022. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000491](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000491).
29. Spencer JrBF, Hoskere V, Narazaki Y (2019) Advances in computer vision-based civil infrastructure inspection and monitoring. *Engineering*, 5, 199–222. <https://doi.org/10.1016/j.eng.2018.11.030>
30. Che E, Jung J, Olsen MJ (2019) Object recognition, segmentation, and classification of mobile laser scanning point clouds: A state of the art review. *Sensors*, 19(4): 810.
<https://doi.org/10.3390/s19040810>
31. Balado J, Díaz-Vilariño L, Arias P, Soilán M (2017) Automatic building accessibility diagnosis from point clouds. *Automation in construction*, 82: 103-111. <https://doi.org/10.1016/j.autcon.2017.06.026>
32. Hussain R, Zeadally S (2018) Autonomous cars: Research results, issues, and future challenges. *IEEE Communications Surveys & Tutorials*, 21(2): 1275-1313.
<https://doi.org/10.1109/COMST.2018.2869360>
33. Xue JR., Fang JW., Zhang P (2018) A survey of scene understanding by event reasoning in autonomous driving. *International Journal of Automation and Computing*, 15(3): 249-266.
<https://doi.org/10.1007/s11633-018-1126-y>
34. Belmonte LM, Morales R., Fernández-Caballero A (2019) Computer Vision in Autonomous Unmanned Aerial Vehicles—A Systematic Mapping Study. *Applied Sciences*, 9(15): 3196.

<https://doi.org/10.3390/app9153196>

35. Al-Kaff A, Martin D, Garcia F, de la Escalera A, Armingol JM (2018) Survey of computer vision algorithms and applications for unmanned aerial vehicles. *Expert Systems with Applications*, 92: 447-463. <https://doi.org/10.1016/j.eswa.2017.09.033>
36. Loquercio A, Maqueda AI, Del-Blanco CR, Scaramuzza D (2018) Dronet: Learning to fly by driving. *IEEE Robotics and Automation Letters*, 3(2): 1088-1095. <https://doi.org/10.1109/LRA.2018.2795643>
37. Rebelo C, Rodrigues AM, Tenedório JA, Goncalves JA, Marnoto J (2015) Building 3D city models: Testing and comparing Laser scanning and low-cost UAV data using FOSS technologies. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9157: 367-379. https://doi.org/10.1007/978-3-319-21470-2_26
38. Gruen A, Huang X, Qin R, Du T, Fang W, Boavida J, Oliveira A (2013) Joint processing of UAV imagery and terrestrial mobile mapping system data for very high resolution city modeling. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 40(1W2): 175-182. <https://doi.org/10.5194/isprsarchives-XL-1-W2-175-2013>
39. Nex F, Remondino F (2014) UAV for 3D mapping applications: a review. *Applied geomatics*, 6(1): 1-15. <https://doi.org/10.1007/s12518-013-0120-x>
40. Rothermel M, Haala N, Wenzel K, Bulatov D (2014) Fast and robust generation of semantic urban terrain models from UAV video streams. In 2014 22nd International Conference on Pattern Recognition (pp. 592-597). IEEE. <https://doi.org/10.1109/ICPR.2014.112>
41. Yan WY, Shaker A, El-Ashmawy N (2015) Urban land cover classification using airborne LiDAR data: A review. *Remote Sensing of Environment*, 158: 295-310. <https://doi.org/10.1016/j.rse.2014.11.001>
42. Guan H, Li J, Cao S, Yu Y (2016) Use of mobile LiDAR in road information inventory: A review. *International Journal of Image and Data Fusion*, 7(3): 219-242. <https://doi.org/10.1080/19479832.2016.1188860>
43. Wang Y, Chen Q, Zhu Q, Liu L, Li C, Zheng D (2019) A Survey of Mobile Laser Scanning Applications and Key Techniques over Urban Areas. *Remote Sensing*, 11(13): 1540. <https://doi.org/10.3390/rs11131540>
44. Gharbi H, Bahroun S, Zagrouba E (2019) Key frame extraction for video summarization using local description and repeatability graph clustering. *Signal, Image and Video Processing*, 13(3): 507-515. <https://doi.org/10.1007/s11760-018-1376-8>.
45. Kong X, Ma K, Hou S, Shang D, Xia F (2019) Human Interactive Behavior: A Bibliographic Review. *IEEE Access*, 7: 4611-4628. <https://doi.org/10.1109/ACCESS.2018.2887341>
46. Macchione F, Costabile P, Costanzo C, De Lorenzo G (2019) *Extracting quantitative data from non-conventional information for the hydraulic reconstruction of past urban flood events. A case study.* *Journal of Hydrology*, 576: 443-465. <https://doi.org/10.1016/j.jhydrol.2019.06.031>
47. Zhang G, Vela PA, Karasev P, Brilakis I (2015) A sparsity-inducing optimization-based algorithm for planar patches extraction from noisy point-cloud data. *Computer-Aided Civil and Infrastructure Engineering*, 30(2): 85-102. <https://doi.org/10.1111/mice.12063>

48. Nguyen HL, Belton D, Helmholz P (2019) Planar surface detection for sparse and heterogeneous mobile laser scanning point clouds. *ISPRS journal of photogrammetry and remote sensing*, 151: 141-161. <https://doi.org/10.1016/j.isprsjprs.2019.03.006>
49. Gilani SAN, Awrangjeb M, Lu G (2016) An automatic building extraction and regularisation technique using lidar point cloud data and orthoimage. *Remote Sensing*, 8(3): 258. <https://doi.org/10.3390/rs8030258>
50. Badenko V, Fedotov A, Vinogradov K (2019) Hybrid Algorithms of Laser Scanning Point Cloud for Topological Analysis. *Advances in Intelligent Systems and Computing*, 797: 223-234. https://doi.org/10.1007/978-981-13-1165-9_20
51. Badenko V, Zotov D, Fedotov A (2018) Hybrid processing of laser scanning data. *E3S Web of Conferences*, 33: 01047. <https://doi.org/10.1051/e3sconf/20183301047>
52. Badenko V, Fedotov A, Vinogradov K (2018) Algorithms of laser scanner data processing for ground surface reconstruction. *Lecture Notes in Computer Science*, 10961: 397-411. https://doi.org/10.1007/978-3-319-95165-2_28
53. Badenko V, Fedotov A, Zotov D, Lytkin S, Volgin D, Garg RD, Min L (2019) Scan-to-BIM methodology adapted for different application. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(5/W2): 1-7. <https://doi.org/10.5194/isprs-archives-XLII-5-W2-1-2019>
54. Perlovsky LI (2017) Physics of the mind, dynamic logic, and monotone Boolean functions. *Studies in Computational Intelligence*, 683: 193-231. https://doi.org/10.1007/978-3-319-51052-1_13
55. Perlovsky LI, Deming R, Ilin R (2011) Emotional Cognitive Neural Algorithms with Engineering Applications. Springer-Verlag Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-22830-8>

Figures

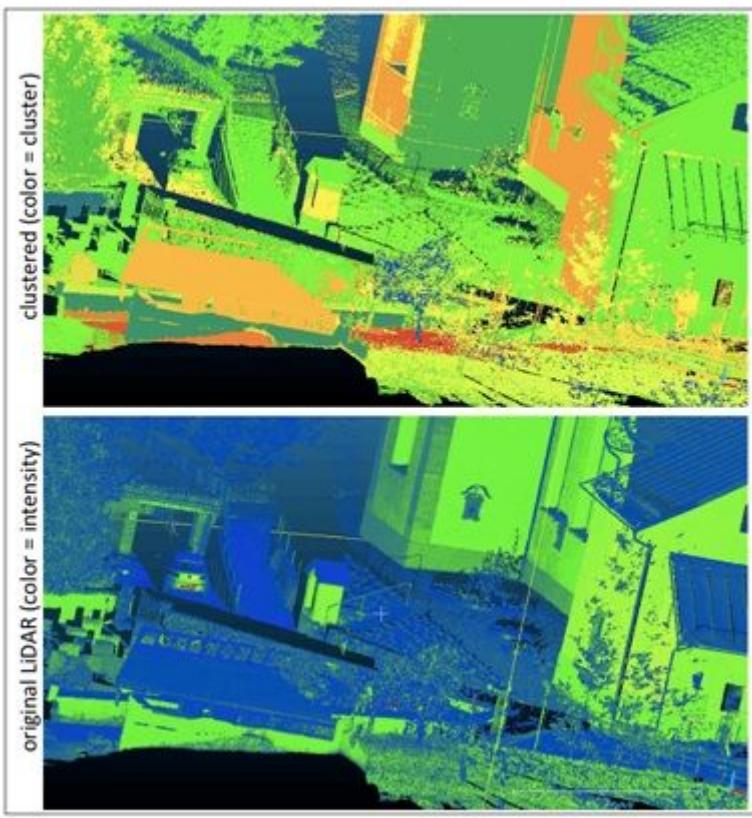


Figure 1

Comparison of input data and its clustered result using XYZIRGB: coordinates (XYZ), intensity (I) and color (RGB). Because of unjustified influence of IRGB parameters on clustering results, they are ignored henceforth.

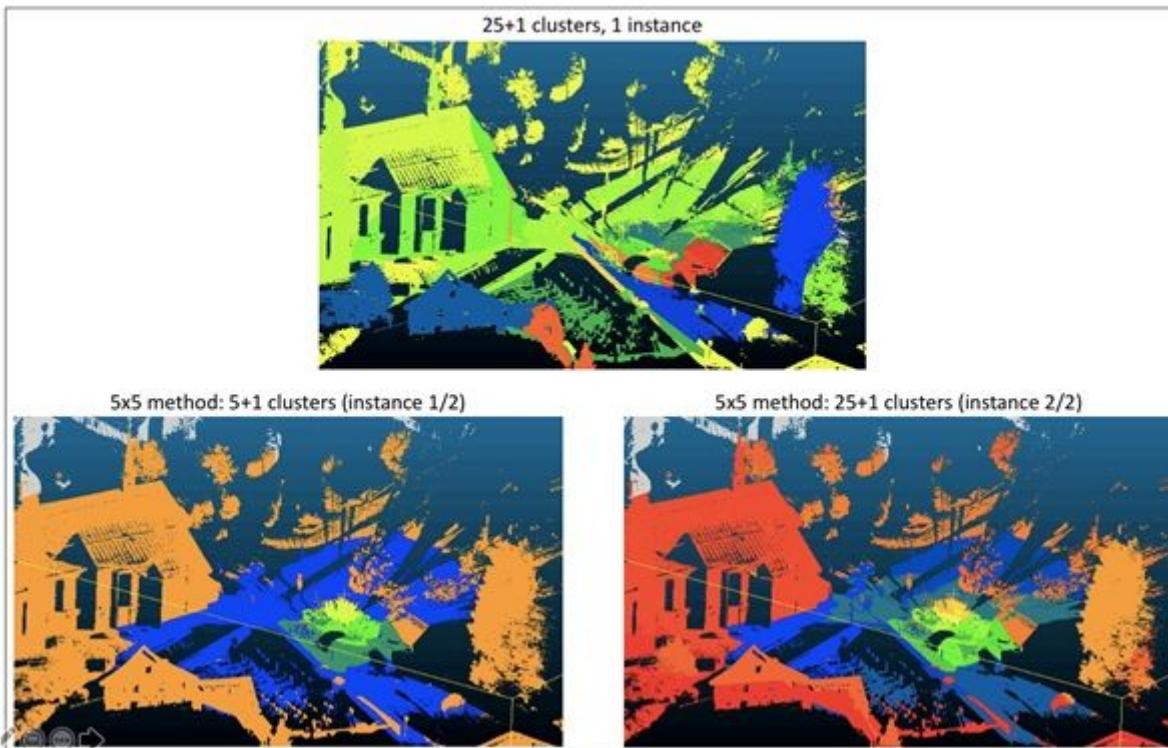


Figure 2

Results of clustering the European village scan from Semantic3D with 2 different methods.

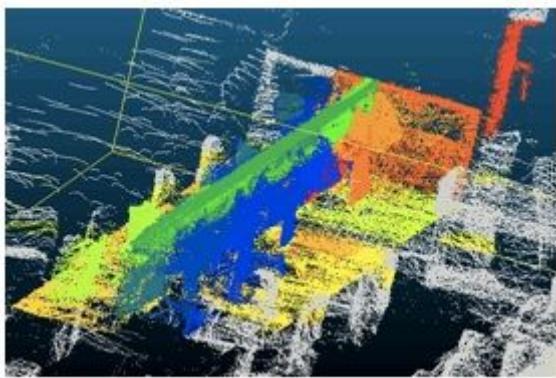


Figure 3

Result of clustering the indoor ISPRS data.

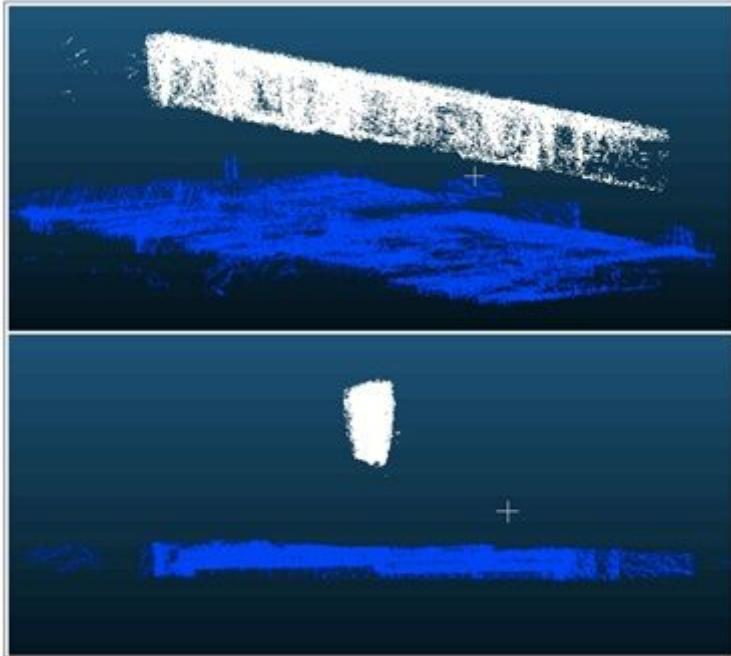


Figure 4

Partial breakdown of clustered ISPRS data.

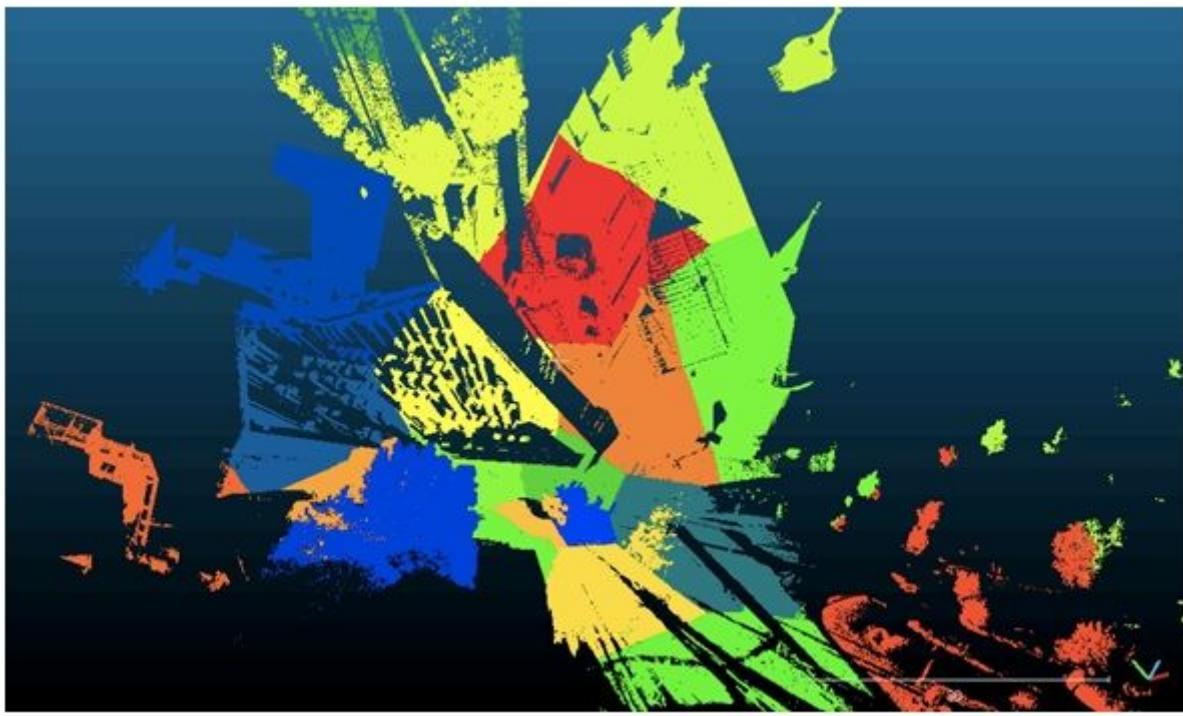


Figure 5

Result of K-means clustering using the “in one go” approach.

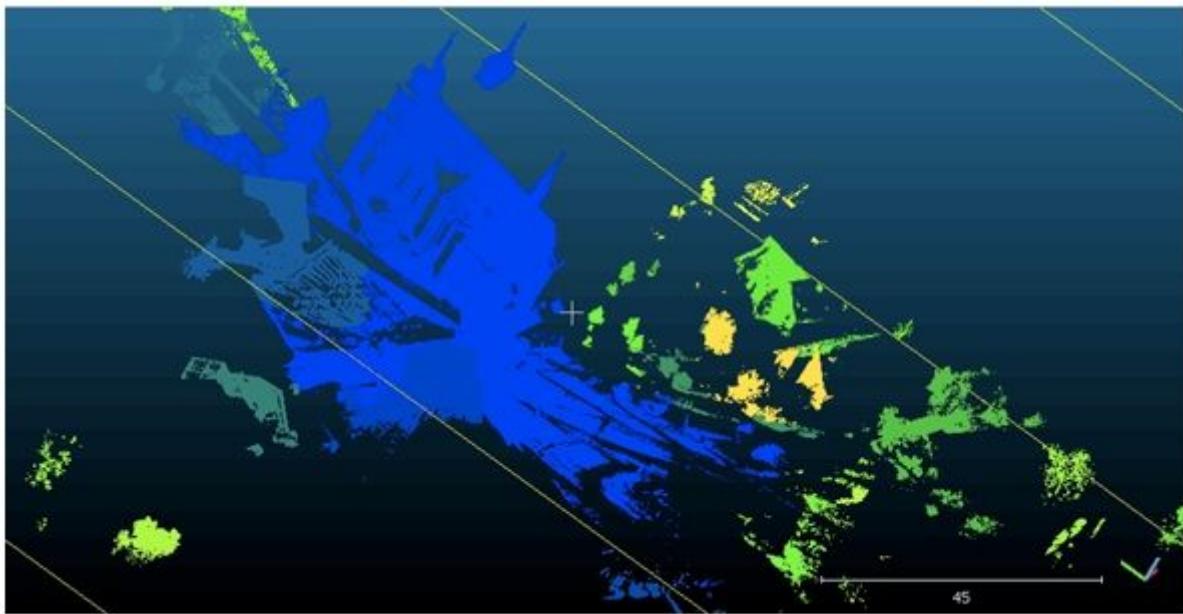


Figure 6

Result of MeanShift clustering with $h=12$ (35 clusters).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Diagram1.pdf](#)