TerraMobilita/iQmulus Urban Point Cloud Analysis Benchmark

Bruno Vallet[a,*], Mathieu Brédif[a], Andres Serna[b], Beatriz Marcotegui[b], Nicolas Paparoditis[a]

[a]*Université Paris-Est, IGN Recherche, SRIG, MATIS*
73 avenue de Paris, 94160 Saint Mandé, France
[b]*Centre de Morphologie Mathmatique (CMM)*
35 rue Saint Honor, 77305 Fontainebleau, France

Abstract

The object of the TerraMobilita/iQmulus 3D urban analysis benchmark is to evaluate the current state of the art in urban scene analysis from mobile laser scanning (MLS) at large scale. A very detailed semantic tree for urban scenes is proposed. We call analysis the capacity of a method to separate the points of the scene into these categories (classification), and to separate the different objects of the same type for object classes (detection). A very large ground truth is produced manually in two steps using advanced editing tools developed especially for this benchmark. Based on this ground truth, the benchmark aims at evaluating both the classification, detection and segmentation quality of the submitted results.

Keywords: Benchmark, urban scene, laser scanning, mobile mapping, classification, segmentation, detection, analysis

1. Introduction

Nowadays, LiDAR technology (Light Detection And Ranging) has been prospering in the remote sensing community. We can find several developments such as: Aerial Laser Scanning (ALS), useful for large scale buildings survey, roads and forests; Terrestrial Laser Scanning (TLS), for more detailed but slower urban surveys in outdoor and indoor environments; Mobile Laser Scanning (MLS), less precise than TLS but much more productive since the sensors are mounted on a vehicle; and more recently, "stop and go" systems, easily transportable TLS systems making a trade off between precision and productivity.

Since all these technologies, the amount of available 3D geographical data and processing techniques has bloomed in recent years. Many semi-automatic

---

[*]Corresponding author
 Email address: `bruno.vallet@ign.fr` (Bruno Vallet)
 URL: `http://recherche.ign.fr/labos/matis/~vallet` (Bruno Vallet)

and automatic methods aiming at analysing 3D urban point clouds can be found in the literature. It is an active research area. However, there is not a general consensus about the best detection, segmentation and classification methods. This choice is application dependent. An additional stake is that mobile mapping systems produce a very large amount of data (a few TeraBytes per day), and handling such massive amounts is an additional stake for research. One of the main locks to research in this direction is the lack of large publicly available datasets and benchmarks.

This paper presents such a benchmark, along with a large dataset, that aims at stimulating researchers from different fields such as Computer Vision, Computer Graphics, Geomatics and Remote Sensing, working on the common goal of processing 3D MLS data, and benchmarking segmentation and classification methods for this data. This will provide a ground for cross-fertilization and discussions on the future challenges in this important research area. Moreover, the benchmark focuses on processing large datasets. More information about this benchmark are available on the webpage: `http://data.ign.fr/benchmarks/UrbanAnalysis/index.html`

In the literature, most available urban data consist in close-range, aerial and satellite images but few MLS datasets. Moreover, manual annotations and algorithm outputs are rarely found in available 3D repositories. The annotated 3D MLS datasets publicly available are:

- The `Oakland 3D` point cloud dataset [4]: it consists of 1.6 million points with only x,y,z coordinates and a label, separated in a training, validation and testing dataset.

- The `RSE-RSS` dataset [10] was collected from a velodyne scanner on the ground with a 360 field of view. It contains 10 scans of approximately 65,000 points each.

- The `Paris-rue-Madame` dataset [9] was produced by the StereopolisII mobile mapping system. The test zone contains approximatively a 160 m long street section between rue Mzires and rue Vaugirard and consists of 20 million points.

Note that these are labelled datasets (which is already very useful), not benchmarks. The only related benchmark that we are aware of is an `EuroSDR` benchmark from 2012 [3] which scope was limited to facade classification and tree trunk and poles detection. The trajectory length was 1.7km and the sensor used was a FARO Photon 80 laser scanner, but the number of points is not given in the report.

Compared to the aforementioned datasets, our dataset is roughly between 10 (`Paris-rue-Madame`, `EuroSDR`) and 100 (`Oakland 3D`, `RSE-RSS`) times larger. This represents both a challenge for the efficiency of the processing, but also regarding the diversity of the scene. Moreover, it has much broader objectives than the `EuroSDR` benchmark (cf Section3).

The paper is organized as follows: the dataset used in the benchmark is presented in Section 2, the problem that participants have to solve is described

2

Figure 1: 3D View of the dataset.

in Section 3 and Section 4 explains how the ground truth was produced. The evaluation protocol is presented in Section 5, the participants are listed in Section 6 and their results are evaluated in Section 7. Finally some conclusions are drawn in Section 8.

2. Experimental Dataset

The dataset consists of 3D MLS data from a dense urban environment in Paris (France). It has been acquired by Stereopolis II [5], a MLS system developed at the French National Mapping Agency (IGN) in the framework of the iQmulus and TerraMobilita projects.

The acquisition covers approximately 10 km of streets within a square km of the 6th district of Paris and consists of 300 million points acquired in January 2013.. Most streets are covered in this square kilometer area so the dataset is representative of this part of Paris. All coordinates are geo-referenced (E,N,U) in Lambert 93 and altitude IGN1969 (grid RAF09) reference system, reflectance is the laser intensity. An offset has been subtracted from the XY coordinates with the aim of increasing data precision: X0 = 649000 m and Y0 = 6840000 m. The accuracy on point positions relative to the vehicle is subcentimetric, so we can expect a local noise below 1cm. The drift between the real trajectory and the one produced by the georeferencing system is very slow (below 1mm per meter) so it can be neglected. Each vertex contains the attributes presented in table 1. The data is split into ten separate zones (each of which was annotated by a different person) of exactly a tenth of the scan (in number of points).

In this dataset, the entire 3D point cloud is segmented and classified, i.e. each point contains a label and a class (cf Section 4). Thus, point-wise evaluation of detection-segmentation-classification methods becomes possible. The datasets and their processing results must be presented in PLY format with little endian encoding.

3

Type	Properties	Description
`float32`	x,y,z	Measured position
`float32`	x0,y0,z0	Sensor position
`float32`	reflectance	backscattered intensity corrected for distance
`uint8`	num_echo	number of the echo (to handle multiple echoes)
`uint32`	id	object identifier in the segmentation
`uint32`	class	class label assigned to its segmented object.

Table 1: Vertex Properties. Positions are expressed in the Lambert 93 system. Two points having the same id must have the same class. Since each point of the dataset contains an id and a class, the evaluation will be carried out in a point-wise way.

3. Analysis Problem Statement

The problem addressed by this benchmark is to perform a point-wise segmentation and classification, which we call analysis but is also called scene parsing or contextual classification by some authors. Each processed file, provided by each participant, must be a PLY file containing the original points (in the original order), their original attributes and two additional attributes: id and class. All the 3D points belonging to the same object will have the same object identifier (id). Thus, the number of different ids in the 3D point cloud corresponds to the number of objects. In the classification step, a category is assigned to each segmented object. Each class represents a semantic urban entity, as defined in the following subsection. If two points have the same label, they must have the same class.

Figure 2 shows an example of labels and classes on a 3D point cloud. In the left image, note that ground, facades, pedestrians and cars are represented with different colors because they are different objects and have different labels. In the right image, the colors represent the object classes: ground in gray, facades in blue, cars in yellow, pedestrians in skin color, and furniture in red and cyan.

3.1. Classification Ontology

In this benchmark, a hierarchy of semantic classes has been defined. The class tree is downloadable as an xml file from `http://data.ign.fr/benchmarks/UrbanAnalysis/download/classes.xml` and its first levels are composed as follows:

Surface Surface of unbounded or very large objects

> **Ground** Ground surface
>
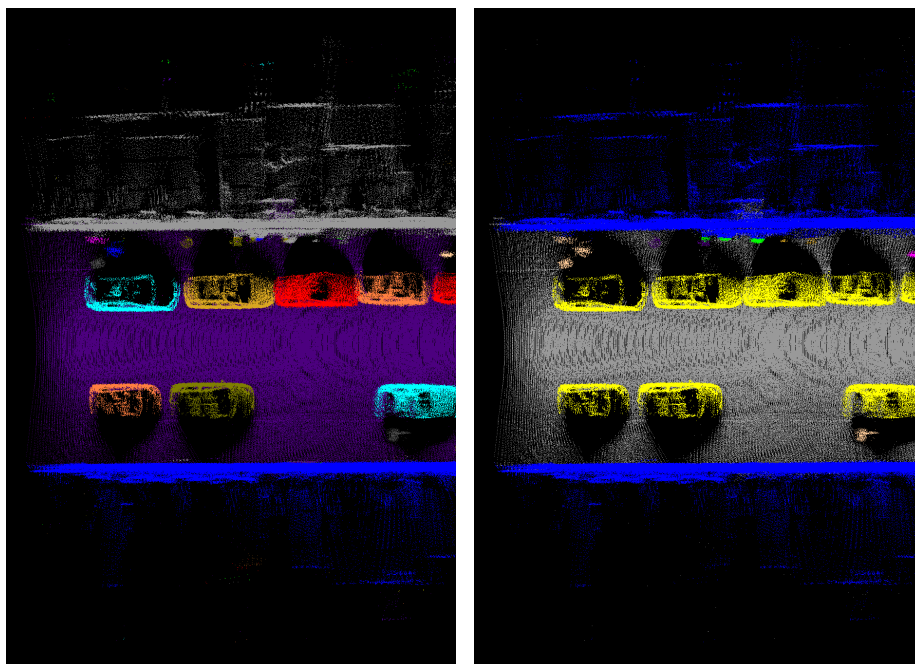> **Building** All points lying on the outside surface of a building

4

Figure 2: (left) Ids. (right) Classes.

  Other Surface Surface of unbounded objects that does not fit in one of the following categories

Object All semantic objects

  Static Objects that are not made to be easily moved

  Dynamic Individuals or objects that can move easily

  Natural Natural objects or vegetation

  Other Object Objects that do not fit in the other categories

Other Undefined points, outliers, points inside buildings...

Unclassified Not classified yet

The tree is voluntarily very detailed as we aim at producing a ground truth that can be useful to a wide range of methods. Participants can choose the classes that they want in this tree and the evaluation will be performed accordingly. The **Unclassified** label is used to only focus the evaluation on the portion of the point cloud that has been classified only. **Other X** labels are scattered throughout the hierarchy so that the classifications may differentiate between a classification that did not try to distinguish among the more specialized child labels of an inner node of the classification hierarchy, and a classification that expresses that the relevant label is not part of the classification hierarchy.

3.2. Previous works

This joint segmentation/classification problem has been scarcely studied in the litterature, where it is sometimes called scene understanding, parsing or recognition. A major previous work [1] relies on four steps:

1. Localization: Potential objects locations are found based on point density
2. Segmentation: Objects are separated from their background.
3. Feature extration: a set of features is computed for each objects
4. Classification: each object is labelled according to its features.

Most other related work perform directly a point cloud classification, usually by adapting techniques from aerial lidar scanning to the terrestrial perspective and relying on local shape descriptors extracted from the point cloud. Section 7 will compare an object based approach similar to [1] with one pure classification approach. An intermediate approach is proposed by [4] using Functional Max-Margin Markov Networks to perform a classification that is context aware (thus not limited to the neighborhood analysis of each point).

4. Semi-Automatic Ground Truth Production

As soon as any form of algorithm is used to produce the Ground Truth that will be used to evaluate a classification result, this particular algorithm will bias the results to favour similar approaches. That is why we have aimed to propose an annotation method that is as manual as possible, thereby reducing algorithmic bias, while providing an editing tool that enables an efficient segmentation and classification of the benchmark dataset. Therefore, we have set up a semi-automatic approach where the user has a full control over the resulting segmentation and classification.

4.1. Segmentation

Navigating and selecting through a point cloud is a counter-intuitive task, due to the absence of proper occlusion and its sparse nature. As the segmentation was to be performed at the point level rather than alternative representations such as bounding boxes, an efficient browsing and segmentation of the point cloud was a key issue. We tackled this problem by proposing an interface that shows the point cloud in sensor space (figure 3). This was made possible and convenient by the geometry of acquisition and the availability of its two parameters in the raw dataset : the constant time step dt_{pulse} between two consecutive emitted pulse and the constant time step dt_{turn} between each rotation of the lidar. Please note that we do not require that each turn is composed of an integral number of pulses, which yield overall shear in figure 3. This parameter allows to recover a regular topology out of the point cloud stream : the pulse neighbours are the immediately preceding and succeeding pulses and the closest ones on the preceding and succeeding rotations. The recovery of such a 2D topology with 4-connectivity enables the use of traditional image processing techniques. A particularity is on its topological boundary : it is composed of
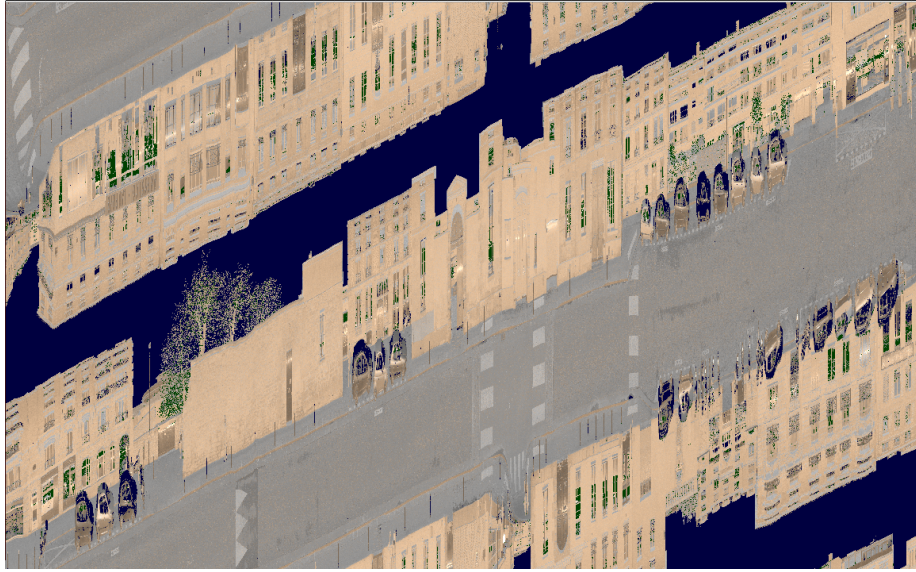
Figure 3: Lidar point cloud viewed in sensor-space : horizontal axis corresponds to time and vertical axis corresponds to rotation angle.

two topological circles corresponding to the pulses of the first and last rotations. This is due to the continuous sampling of the particular scanner used to acquire this dataset : the last pulse of a rotation is connected to the first pulse of the following one.

In this 2D sensor-space, it is then trivial to create, maintain a 2D segmentation by providing to the user a graph edition tool : node creation and deletion, edge creation between existing nodes, poly-line input, insertion of a node by splitting an edge... In order for this graph to define a partition of the sensor measurements, nodes are placed at pixel corners and edges are Manhattan curves along pixel boundaries. The user experience is improved by allowing zooming, panning, and snapping.

The only automatic task used in this editing tool applies when a user adds an edge between two nodes or when one is modified by a moving end node. Two modes are available to rasterize a path between the two end nodes of the path on the pixel boundaries :

Sensor-space Line This is a simple rasterization of the sensor-space line segment as a Manhattan curve passing through pixel corners.

Optimal Path This mode performs an $A*$ search to provide the rasterized path which is optimal with respect to the sum of weights of the traversed pixel boundaries. This *cost* is expressed in terms of the absolute differences of the measured quantities $\Delta_\alpha, \Delta_d, \Delta_r$, where α is the angle between the normal and the vertical direction, d is the measured distance and r is the
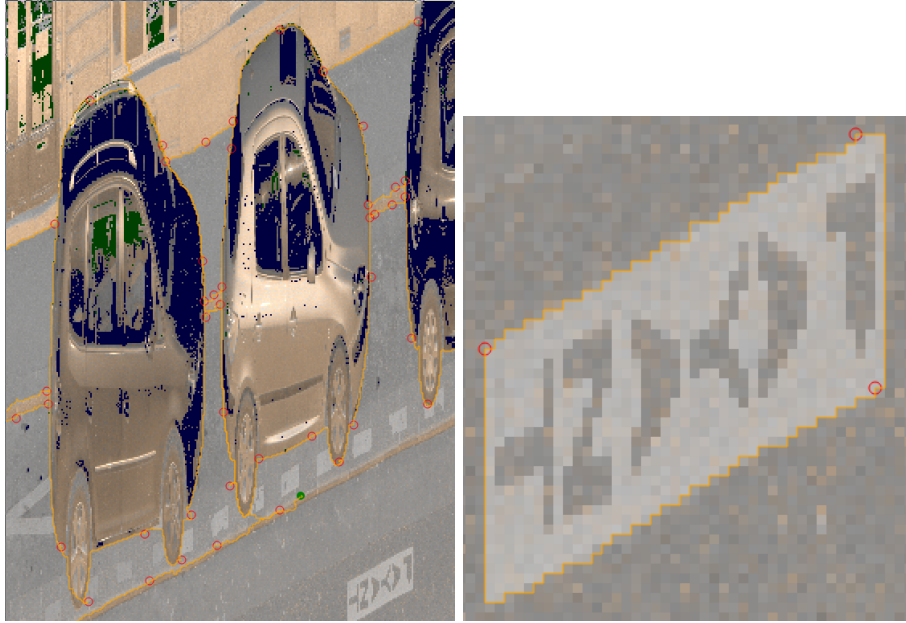
Figure 4: Segmentation-inducing graph.

measured reflectance, and $\lambda, \alpha_0, d_0, r_0$ are empirical weights :

$$cost = \max\left(1, \lambda\left(1 - \max\left(\frac{\Delta_\alpha}{\alpha_0}, \frac{\Delta_d}{d_0}, \frac{\Delta_r}{r_0}\right)\right)\right) \tag{1}$$

The user experience is improved by preprocessing the shortest path from the selected node as a background process, so that the optimal path may be updated efficiently as the mouse moves. The user click then only validates the already optimized and displayed path, increasing the productivity.

4.2. Classification

As the number of objects is limited compared to the massive size of the point cloud, assignation of class labels to an already segmented lidar point cloud is performed by a paint-bucket-like interface : the user selects a class label from a drop-down list of available labels and then clicks on the point cloud to assign this label to the segmented object id. The display mode may be either set to segmentation mode that applies a random palette according to the segment ids or a classification mode where the visualization depends on the class label. To further help this process, points to be rendered may be filtered base on there class labels.

8

5. Submission Evaluation

The proposed semantic tree is very detailed and probably no existing method treats the whole problem. This is why, the participants to the benchmark can choose to analyse the scene using any subset of labels from the semantic tree, and a label "other" to the classes they do not detect. The evaluation will be performed accordingly and only the relevant metrics will be given.

The benchmark does not aim at ranking the participants but at providing insights on the strengths and weaknesses of each methods. We consider that the quality of a method is subjective and application dependent, and the results of this benchmark should only help a user choosing the approach that best fits its own specific requirements. Results will be evaluated at three levels: as a classification, as a detection and as a segmentation. Details of the evaluation metrics used are given in the following subsection.

5.1. Classification quality

The classification quality will be evaluated point-wise. The results of the evaluation will be a confusion matrix for each node of the tree that the evaluated method handles. Rows and columns will be the subclass labels from the ground truth and the evaluated method respectively, and matrix values are the number of points with the corresponding labels. For easier reading, these numbers are given as a percentage of the total number of points. Also for binary classification (only 2 classes), only the classification accuracy will be given. The classification accuracy is the trace of the confusion matrix and can be simply interpreted as the percentage of points well classified by the algorithm. All nodes from the semantic tree have an "other" class, so participants can classify into less classes than what is given in the tree.

5.2. Detection quality

The detection quality work measures the capacity of the method to detect the objects present in the scene. Thus it requires to choose a criterion to decide if an object from the ground truth is detected or not. This biases the evaluation as this choice will impact the result. The solution that we propose is to give the evaluation result for a varying threshold m on the minimum object overlap. In this benchmark, an object is defined by the subset of points with the same object identifier. For a such subsets S^{GT} of the Ground Truth and S^{AR} of the evaluated Algorithm Result, we will validate S^{AR} as a correct detection of S^{GT} (a match) iff:

$$\frac{|S^{GT} \cap S^{AR}|}{|S^{GT}|} > m \text{ and } \frac{|S^{GT} \cap S^{AR}|}{|S^{AR}|} > m \qquad (2)$$

where $|\cdot|$ denotes the cardinal (number of points in our case) of a set. Note that the intersection of a GT and AR object is a set of points with the same id

9

in both the GT and the AR. The standard precision/recall are then functions of m:

$$precision(m) = \frac{\text{number of detected objects matched}}{\text{number of detected objects}}$$

$$recall(m) = \frac{\text{number of detected objects matched}}{\text{number of ground truth objects}}$$

Precision/Recall will be evaluated for each object types at each level of the semantic tree that the participants have handled and results will be presented as two curves. Precision/Recall are decreasing in m and this decay indicates the geometric quality of the detection (good geometry implies slower decay).

Precision/Recall does not depend on object size so it gives the same weight to small and large objects. On the opposite classification metrics are point based so they favour large objects composed of many points on them. This is why we give both metrics: depending on the application, one or the other might be more appropriate.

5.3. Segmentation quality

When the threshold m is below 0.5, the criterion (2) does not guarantee that objects are uniquely matched. When $m < 1/n$, n objects from the Ground Truth (GT) can be matched to a single object of the Algorithm Result (AR), or the opposite. Thus for $m < 0.5$ we will also give the curves of over-segmentation (1-to-n) and under-segmentation (n-to-1) by averaging n over the matches defined by (2). Note that $m = 0.5$ is often used, which we consider too strict as it penalizes over-segmentation too much (an object of the GT can only match a single object of the AR and vice versa), which was our initial motivation for a varying threshold.

The three metrics defined above also indirectly measure robustness as our scene is very large an present major challenges (occlusions, outliers, ...) so the classification detection and segmentation quality will be higher for a method robust to these problems.

5.4. Efficiency

Requiring for the participants to provide standalone executables is makes it really hard for them because research protoypes usually have manual steps which automation might be very time consuming, which is why we asked for the participants to download the data and upload their results. Because of this choice, we cannot benchmark the timings objectively (in the same conditions). Thus the timings will be given based on declarations of the participants.

6. Participants

For the moment, two participants have shown interest, but the benchmark will stay open at least until the end of 2016.

6.1. IPF - KIT

The institute of Photogrammetry and Remote Sensing (IPF) from the Karlsruhe Institute of Technology (KIT) will participate with a method that they present as follows:

We propose a new methodology for large-scale urban 3D scene analysis in terms of automatically assigning 3D points respective semantic labels. The methodology focuses on simplicity and reproducibility of the involved components as well as performance in terms of accuracy and computational effort. Exploiting a variety of low-level geometric features and considering recent advancements in order to improve their distinctiveness, the methodology is in principal designed to process point clouds with a few millions of 3D points. For analysing huge 3D point clouds with possibly billions of points for a whole city like Paris, however, an adaptation has to be introduced. For this purpose, we propose an adaptation which is based on a tiling of the scene and thus allows a successive processing in reasonable time without affecting the quality of the classification results. We demonstrate the performance of our methodology on two adequate, labelled 3D point cloud datasets with respect to robustness, efficiency and scalability.

The method relies on random forest classification based on a very large set of features, including some with automatic scale selection, and a thorough feature selection framework. For further details, the reader is encouraged to review [11].

6.2. CMM - MINES ParisTech

The Centre de Morphologie Mathématiques (CMM) from MINES ParisTech will participate with a method that they present as follows:

Our method is based on elevation images and it uses image processing techniques, specially Mathematical Morphology, and machine learning techniques. Our general workflow is presented in Fig. 5 and it consists in the following steps: i) the 3D point cloud is projected onto an elevation image; ii) ground is segmented using the λ–flat zones labelling algorithm, curbs are segmented and their accessibility is analysed; iii) facades are segmented as the highest vertical structures in the urban scenario; iv) objects are detected as discontinuities on the ground; v) small and isolated structures are eliminated and connected objects are separated using a constrained watershed; vi) objects are classified in several categories using a SVM approach with geometrical and contextual features; and, vii) the segmented and classified images can be reprojected to the 3D point cloud for visualization purposes.

For further details and complete analyses in each step, the reader is encouraged to review [2, 6, 7, 8].

The main difference with the approach of [1] is that the ground and facade are extracted prior to object individualization.

7. Results

The results of the benchmark were presented at the IQmulus workshop taking place on July 8th, 2014 in Cardiff (UK), in conjunction with SGP'14. For
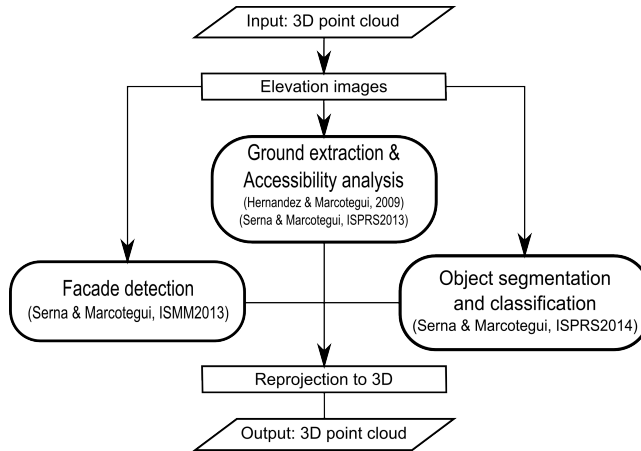
Figure 5: Work-flow of our proposed segmentation methodology.

GT/AR	0	2	4
0	4.06%	0.4%	0.08%
2	0.40%	8.72%	1.02%
4	10.43%	19.173%	55.70%

Table 2: KIT 0- (pedestrian) vs 2- (bicycles, moped, motorbike,...) vs 4-wheelers (cars) confusion matrix for mobile object points

practical reasons, the benchmark only consisted in one of the ten proposed zones (results on the ten zones will be provided on the website when the participants finish processing them). A part of this benchmark zone is shown on Figure 6.

7.1. IPF - KIT

The IPF-KIT participated only to the classification part of the benchmark, so it is only evaluated based on the classification metrics. The classification accuracy is:

- 87.8% Surface vs Object classes

- 93.7% Ground vs Building surface (for surface points)

- 91.5% Static vs Mobile objects (for object points)

- 68.5% Pedestrian vs 2 vs 4 wheelers (for mobile points) as detailed in Table 2

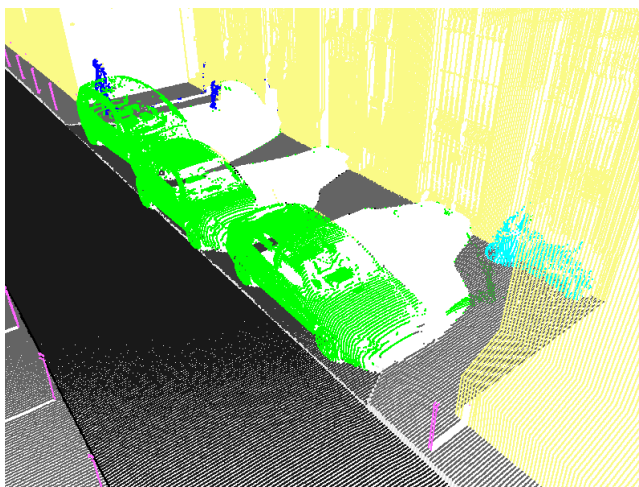A classification result of the algorithm is displayed in Figure 7.

12

Figure 6: Ground truth on a part of the benchmark area. Main classes are road (dark gray), sidewalk (gray), curb (light gray), building (yellow), car (green), pedestrians (blue), 2-wheelers (cyan), punctual objects (magenta)

GT/AR	Horz	Vert	GT/AR	0	2	4
Road	71.8%	0.68%	0	1.63%	0.00%	0.12%
Sidewalk	25.7%	0.68%	2	0.39%	0.65%	0.00%
Curb	0.18%	0.89%	4	0.11%	0.03%	97.06%

Table 3: Left: CMM horizontal (road, sidewalk, islands) vs vertical (curb) confusion matrix for ground surface points. Right: CMM 0- (pedestrian) vs 2- (bicycles, moped, motorbike,...) vs 4-wheelers (cars) confusion matrix for mobile object points

7.2. CMM - MINES ParisTech

The CMM - MINES ParisTech participated only to both the classification and segmentation parts of the benchmark. The classification result of the algorithm is displayed in Figure 8. The classification accuracy is:

- 92.6% Surface vs object classes

- 98.3% Ground vs Building surface (for surface points)

- 98.4% Curb vs road/sidewalk (for ground surface points) as detailed in Table 3 (left)

- 91.8% Static vs Mobile objects (for object points).

- 99.3% Pedestrian vs 2 vs 4 wheelers (for mobile points) as detailed in Table 3 (right)

The detection quality is given by precision/recall and topologic quality (1-n = oversegmentation, m-1 = undersegmentation) in Figure 9 and 10. They are
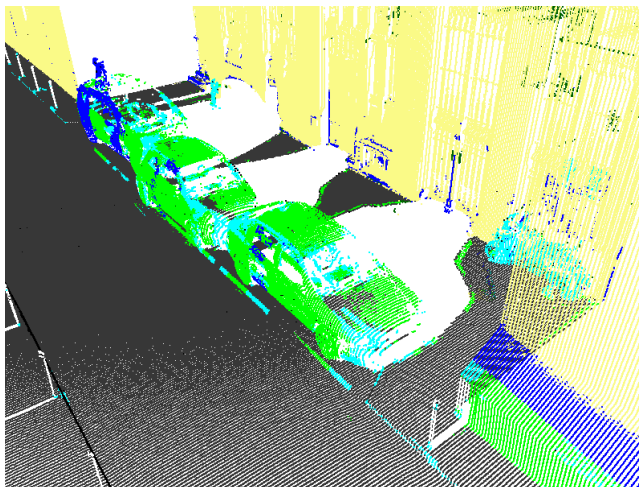
Figure 7: IPF-KIT result on a part of the benchmark area.

shown as curves depending on the threshold used for matching as explained in Section 5. They show a 100% recall for sufficiently low thresholds and a precision around 80% for both static and mobile objects demonstrating a tendency to over-detection. Low thresholds induce topological errors (both under and over-segmentation). A threshold of 0.5 is a good compromise for this method (precision/recalls stay high while topological errors decrease significantly).

7.3. Discussion

It is quite obvious from the results that the CMM method outperforms the KIT method in all aspects. The explanation is quite simple: the KIT method is a point based classification only using a local information (neighbourhood analysis) to make its decision. As many classes are composed of objects, a neighbourhood base method fails to classify all the objects points in the correct object class. This issue is discussed in [10]. This would tend to show that the combined segmentation/classification problem should be solved by first segmenting, then classifying the results. A more local classification as that of KIT might be useful as a pre-processing step to help the segmentation and the object classification, but should probably restrict to less semantised (and more geometric) classes (plane, corner, round object, scattered). Finally, we see an interesting common point between the CMM method and [1]: they separate the background (surface classes, mainly ground and faades) from the foreground (objects). This is very coherent with our semantic tree where this distinction is the highest level one. This would tend to show that generic point cloud segmentation tools should be adapted to urban scenes to take this particularity into account. It would thus be very informative to have [1] participating to our benchmark to find out whether it is better to perform the background separation before or after object localization.
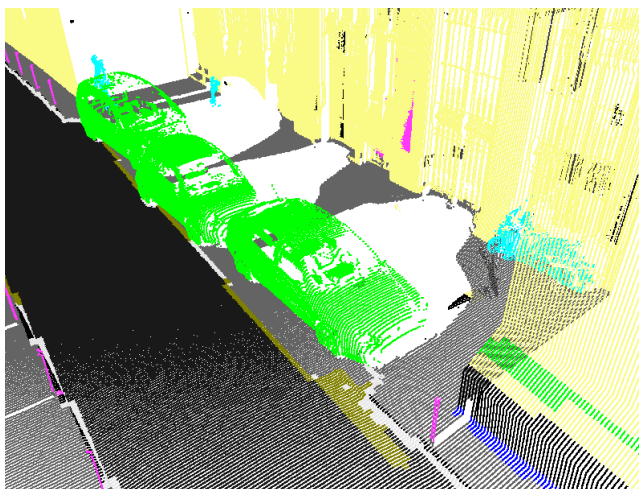
Figure 8: CMM result on a part of the benchmark area.

The classification accuracies are quite high, but we have to keep in mind that the numbers are computed only on the points that were classified in the same class of the higher level (for instance, for mobile vs static object, we only count the point classified as objects in the Ground Truth and the Algorithm Result).

8. Conclusions

Analysing an urban scene from a Mobile Laser Scanning is an extremely challenging task because of the variety of objects, incomplete scanning (occlusions) but also for methodologic reasons. Analysing a scene relies on both classifying and segmenting the scene, and both problems are tightly connected. The benchmark presented in this paper aims at triggering interest on this difficult subject, provide data to people wishing to address it, and to give objective measures of the qualities of the proposed methods in order to help advance the state of the art. Thus this benchmark will stay open for at least two years during which any submission will be evaluated and the results published on our webpage.

Acknowledgements

[1] Golovinskiy, A., Kim, V.G., Funkhouser, T.. Shape-based recognition of 3D point clouds in urban environments International Conference on Computer Vision, ICCV (2009)
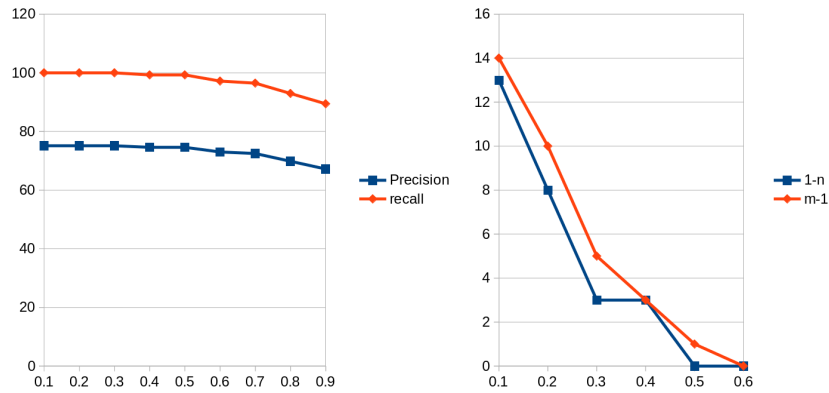
Figure 9: CMM Precision/Recall (left) and topologic (right) quality for all objects.
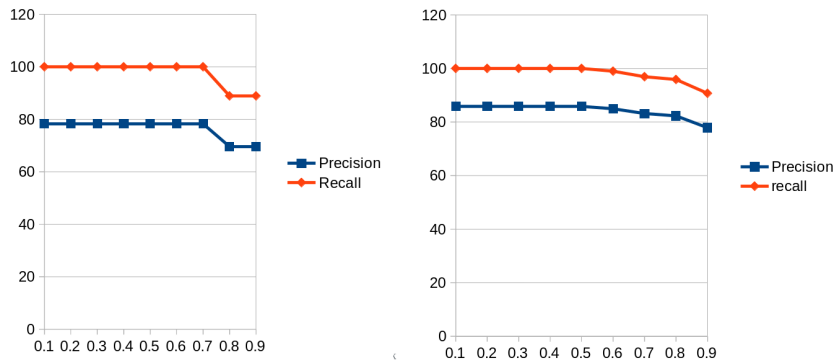

Figure 10: CMM Precision/Recall for static (left) and mobile (right) objects.

[2] Hernández, J., Marcotegui, B.. Filtering of artifacts and pavement segmentation from mobile LiDAR data. In: Bretar, F., Pierrot-Deseilligny, M., Vosselman, M.G., editors. ISPRS workshop Laser scanning '09; vol. XXXVIII-3/W8 of *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Paris, France; 2009, p. 329–333.

[3] Kaartinen, H., Kukko, A. , Hyypp, J., Lehtomki, M. EuroSDR benchmarking of Mobile Mapping Algorithms and Systems. EuroSDR report, 2012

[4] Munoz, D., Bagnell, J. A., Vandapel, N. and Hebert, M.. Contextual Classification with Functional Max-Margin Markov Networks. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), June, 2009.

[5] Paparoditis N., Papelard, J.-P., Cannelle, B., Devaux, A., Soheilian, B.,

David, N., Houzay, E. Stereopolis II: A multi-purpose and multi-sensor 3D mobile mapping system for street visualisation and 3D metrology. Revue Franaise de Photogrammtrie et de Tldtection 200: 69-79, October 2012.

[6] Serna, A., Marcotegui, B.. Attribute controlled reconstruction and adaptive mathematical morphology. In: 11th International Symposium on Mathematical Morphology. Uppsala, Sweden; 2013, p. 205–216.

[7] Serna, A., Marcotegui, B.. Urban accessibility diagnosis from mobile laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing 2013;84:23–32.

[8] Serna, A., Marcotegui, B.. Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. ISPRS Journal of Photogrammetry and Remote Sensing 2014;93:243–255.

[9] Serna, A., Marcotegui, B., Goulette, F., Deschaud, J.-E. Paris-rue-Madame database: a 3D mobile laser scanner dataset for benchmarking urban detection, segmentation and classification methods. ICPRAM 2014. `http://cmm.ensmp.fr/~serna/rueMadameDataset.html`

[10] Shapovalov, R., Velizhev, A. and Barinova, O. Non-associative Markov networks for 3D point cloud classification. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII, Part 3A, pp. 103?108, 2010.

[11] Weinmann, M., Jutzi, B., Mallet, C. Feature relevance assessment for the semantic interpretation of 3D point cloud data. ISPRS Workshop on Laser Scanning, Antalya, 2013.