# SZTAKIBudapest: a multimodal Lidar benchmark for autonomous vehicles

**August 16, 2021**

Örkény Zováthi
research associate
SZTAKI

Balázs Nagy
research fellow
SZTAKI

Csaba Benedek
research advisor
SZTAKI

# Contents

# 1 Introduction

In this report we present our collected multimodal *SZTAKIBudapest* Benchmark, a benchmark to carry both onboard Lidar vehicle measurement (OBM) data and mobile laser scanning (MLS) high density 3D city maps at the same places for the purpose of evaluating Lidar based cross-source registration and change detection algorithms in urban environments.

# 2 Data acquisition

The onboard Lidar measurement sequences of the Benchmark have been captured by SZTAKICar, a test car of our research institute in various main city roads of Budapest Hungary. The measurement platform has been equipped with a Velodyne HDL64E[1] 64-beam Rotating Multi-beam (RMB) Lidar scanner and an external GPS receiver fixed on the roof top which provided rough global position estimations of the recorded point clouds.

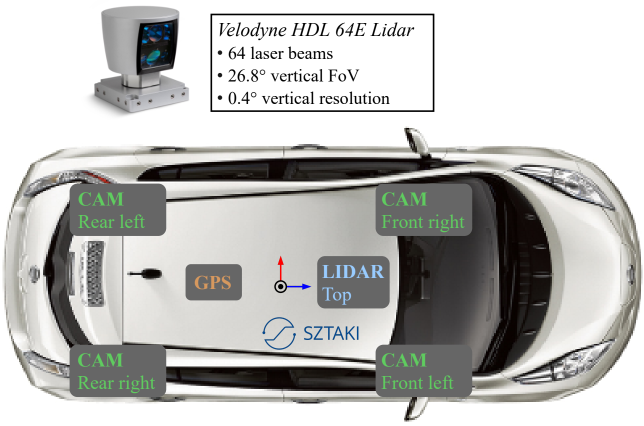



Figure 1: SZTAKICar: The measurement platform.

The Velodyne HDL-64E sensor was originally designed to help real-time perception of autonomous robots and vehicles. It provides a stream of relatively sparse (6-10 $\times 10^4$ points/frame) point clouds with a temporal frequency of 15 fps. The spatial accuracy is around 1-2 cm in the sensor's own coordinate system, but the point density quickly decreases as a function of the distance from the sensor and it shows typical ring patterns.

The Mobile Laser Scanning (MLS) measurements have been recorded with a Riegl VMX450[2] mobile mapping system by the Budapest Road Management company (Budapest Közút Zrt.). The Riegl VMX450 MLS system is highly appropriate for city mapping, urban planning and road surveillance applications. It integrates two Riegl laser scanners, a well-designed, calibrated camera platform and a high performance Global

[1] `http://velodyneLidar.com/`
[2] `http://www.riegl.com/`

Navigation Satellite System (GNSS), providing extremely dense, accurate (up to global accuracy of a few centimetres) and feature rich data with a quite uniform point distribution.

# 3 Real-time global localization and change detection (SZTAKI-CityCDLoc)

In this section, we describe the point cloud registration and change detection task on our Benchmark, and we provide a dataset for evaluating similar algorithms.

**Introduction** SZTAKI-CityCDLoc has been created for the purpose of evaluating multimodal 3D semantic point cloud registration and change detection algorithms in urban environments, based on mobile laser scanning (MLS) data of a Riegl VMX-450 mobile mapping system and onboard vehicle measurements captured by a Velodyne HDL 64E rotating multi-beam Lidar sensor.

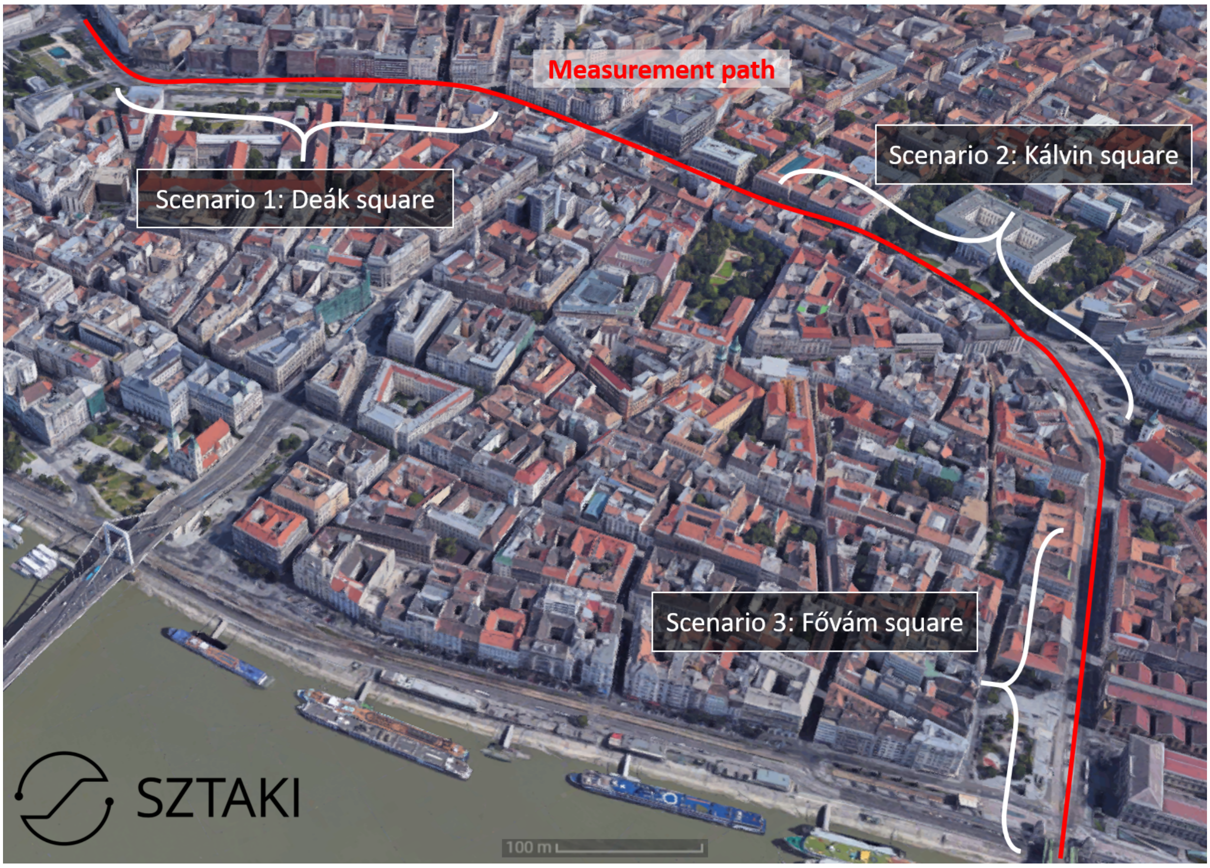



Figure 2: The data acquisition path and the three test scenarios

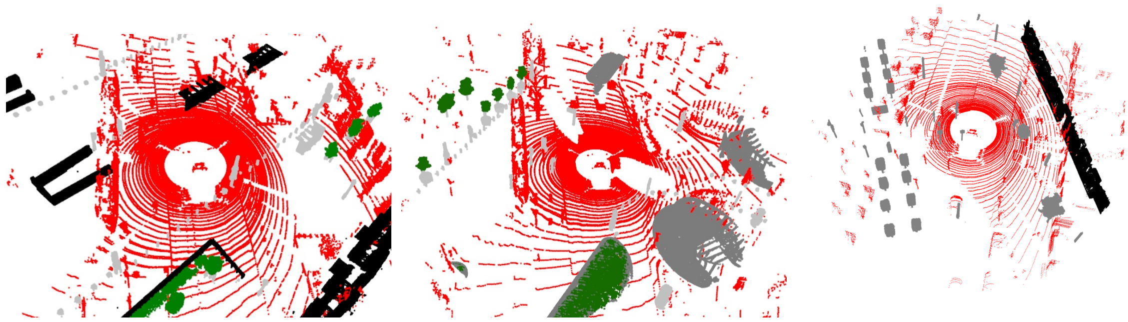
| MLS scenario | Format | Start frame | End frame | Length [m] | Length [s] |
|---|---|---|---|---|---|
| Fővám square | #frame.pcd | 8400 | 8600 | 300 | 13.33 |
| Kálvin square | #frame.pcd | 9500 | 9700 | 300 | 13.33 |
| Deák square | #frame.pcd | 11650 | 11730 | 200 | 5.33 |

Table 1: Lidar measurement sequences

**Motivation** In dense urban environment, we should expect that the initial position estimation of a vehicle might be notably inaccurate and the global positioning error of the vehicles may reach several meters in city regions with poor GPS signal coverage (Figure 3). Assuming that using an efficient segmentation [1] of the available raw MLS data, we can construct a subset of the segmented MLS point cloud, which contains static classes (tall column, street furniture, facade) and represents empty street segments. Therefore, we can consider it as highly detailed reference models for the vehicles' onboard Lidar measurements. In this context, accurate *global localization* of the vehicle's Lidar measurements in the MLS data, and the *detection of relevant changes* in the vehicle's environment based on the static MLS data may appear as challenging tasks.

**Technical details** We provide three test scenarios from the downtown of Budapest (Figure 2, Table 1). Each scenario contains the following data:

- a **geo-referred MLS point cloud** (*#scenario_MLS.pcd*) that is automatically segmented by [1] and it contains class labels as colors regarding the *facade*, *tall column* and *street furniture* semantic classes.
- **RMB Lidar measurement sequences**. Each frame of the measurement is stored in a separate .pcd file (*#scenario_framenum_timestamp.pcd*). The coordinates of each frame are local to the sensor's center position. Beside the local XYZ coordinates, each point contains an intensity attribute as well.



(a) Scenario 1, Fővám square (b) Scenario 2, Kálvin square (c) Scenario 3, Deák square

Figure 3: GPS based positioning of the vehicles' Lidar measurements in the geo-referenced MLS data's coordinate system.

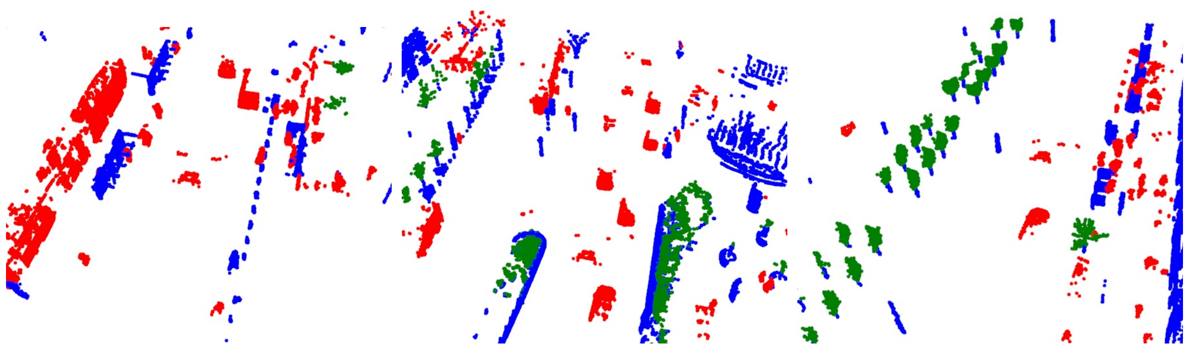
- a separate **GPS metadata** (*#scenario_ GPS.csv*), which contains timestamp-GPS coordinate pairs during the *whole measurement path* for positioning a Lidar frame in the geo-referred MLS data.

For the *point cloud registration* task, we provide ground truth information (stored in *#scenario_ GT.csv*) by manually aligning uniformly sampled point cloud frames to the global MLS point cloud. On the other hand the manual alignment definitely has some uncertainty, as there is no determinable *„best"* alignment between the significantly different point sets. Therefore, we suggest to use other error metrics as well to measure the efficiency of your alignment algorithms, such as the Modified Hausdorff Distance (MHD) or Median Point Distance (MPD).

For the *change detection* task, we annotated uniformly sampled Lidar point cloud measurement frames from each test scenario, and labelled the ground truth information in a semi-automatic manner. First, we performed an approximate offline registration between the i3D and MLS frames using the Iterative Closest Point (ICP) [2] algorithm, then we applied an automated nearest neighbor search based classification with a small distance threshold (5 cm) as an initial segmentation result. Thereafter, the labeling of the different change regions (especially on the region borders) in the sampled Lidar frames was manually revised using our previously mentioned 3D point cloud annotator tool [1]. At the annotation, we distinguished four change classes (Figure 4) by GT labeling:

- *Dynamic changes* that refer either to moving street objects such as traffic participants, to temporarily available objects such as barriers, or to changes in static scene elements such as a re-located bus station or kiosk. These regions are not presented in the MLS data. At GT labeling, these points are marked with **red** ($r = 255$, $g = 0$, $b = 0$).

- *Seasonal changes*, which regions are typical for vegetation areas. These regions are segmented as vegetation in the MLS data, and may have modified appearance



(a) Scenario 1, Fővám square (b) Scenario 2, Kálvin square (c) Scenario 3, Deák square

Figure 4: Change detection GT labeling samples from each test scenario. Color codes: dynamic change, seasonal change, no change.

during the different time periods/seasons. At GT labeling, these points are marked with **dark green** ($r = 0$, $g = 128$, $b = 0$).

- *Unchanged regions*, which contain static environment parts. These regions are also present in the MLS data. At GT labeling, these points are marked with **blue** ($r = 0$, $g = 0$, $b = 255$).

The annotations are stored in a separate folder in .pcd files with the same name as the original measurement frame, where the given colors code the corresponding class for each point.

**Access to the data** The dataset is publicly available, and can be downloaded from out GitHub site: `https://github.com/sztaki-geocomp/Lidar-SCU#dataset`

**Inspection/ Visualization inferface** There are a few MATLAB scripts attached to the dataset serving the purpose of data visualization and inspection.

- Run *RMBstreamer.m* to play a full RMB Lidar measurement sequence. Options: Deák, Fővám or Kálvin square.
- Run *frameToGlobal.m* to display an RMB Lidar frame in the global MLS map's coordinate system. Options: Ground Truth or GPS-based positioning.
- Run *displayChangeDetection.m* to display a sample RMB Lidar frame annotated for change detection.

# References

[1] B. Nagy and C. Benedek. 3D CNN-based semantic labeling approach for mobile laser scanning data. *IEEE Sensors Journal*, 19(21):10034–10045, Nov 2019.

[2] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.