

Real-time point cloud alignment for vehicle localization in a high resolution 3D map

Balázs Nagy^{1,2[0000-0003-0338-863X]} and Csaba Benedek^{1,2[0000-0003-3203-0741]}

¹ Machine Perception Research Laboratory Institute for Computer Science and Control, H-1111 Kende u. 13-17 Budapest, Hungary
`firstname.lastname@sztaki.mta.hu`
<http://mplab.sztaki.hu/geocomp/>

² Pázmány Péter Catholic University Faculty of Information Technology and Bionics, H-1083, Práter utca 50/A, Budapest, Hungary

Abstract. In this paper we introduce a Lidar based real time and accurate self localization approach for self driving vehicles (SDV) in high resolution 3D point cloud maps of the environment obtained through Mobile Laser Scanning (MLS). Our solution is able to robustly register the sparse point clouds of the SDVs to the dense MLS point cloud data, starting from a GPS based initial position estimation of the vehicle. The main steps of the method are robust object extraction and transformation estimation based on multiple keypoints extracted from the objects, and additional semantic information derived from the MLS based map. We tested our approach on roads with heavy traffic in the downtown of a large city with large GPS positioning errors, and showed that the proposed method enhances the matching accuracy with an order of magnitude. Comparative tests are provided with various keypoint selection strategies, and against a state-of-the-art technique.

Keywords: Lidar, Point Cloud, Registration, Scene Understanding

1 Introduction

Self driving vehicles (SDV) offer several benefits for the society ensuring for example a decreased number of road accidents, and more effective traffic distribution on heavy roads. Since these vehicles are equipped with various sensors, apart from their original transportation functionality, they can also contribute to solving environment monitoring, mapping, surveillance, and change detection tasks [1], [7]. Taking the advantage of these massive, moving sensor parks on the roads, algorithms can forecast traffic jams, and they can automatically notify the community about particular or unusual events such as accidents, or police actions.

Accurate and robust localization and environment mapping are key challenges in autonomous driving. Although the GPS-based position information is usually suitable for helping human drivers, its accuracy is not sufficient for navigating a self driving vehicle. Instead, the accurate position and orientation of

the SDV should be calculated by registering the measurements of its onboard visual or range sensors to available 3D high definition (HD) city maps [17, 9].

Rotating Multi-beam (RMB) Lidar laser scanners [2, 3] mounted on vehicle tops are efficient candidates to ensure robust positioning of SDVs. RMB Lidars measure directly the range information, and as active light based sensors, they efficiently perform under different illumination and weather conditions, offering accurate point cloud data with a large field of view in real-time. Although RMB Lidars provide strong geometric features about the environment, the captured point cloud data is quite sparse and inhomogeneous. In addition due to nature of the scanning process, objects are effected by occlusions and motion artifacts, thus robust object detection and classification on such measurements are not straightforward [2]. The point clouds of RMB Lidars are originally obtained in the sensor's local coordinate system, which can be shifted with the actually measured GPS position of the SDV. However in urban environments, due to lack of accurate navigation signals the error of GPS-based position estimation may be often around 2-10 meters.

We can also witness a quick improvement of up to date HD map generation technologies. Mobile Laser Scanning (MLS) can rapidly provide very dense and detailed 3D maps of the cities, however economic (*offline*) filtering and semantic labeling, and *real time* exploitation of the raw MLS data are currently a highly challenging problems [18].

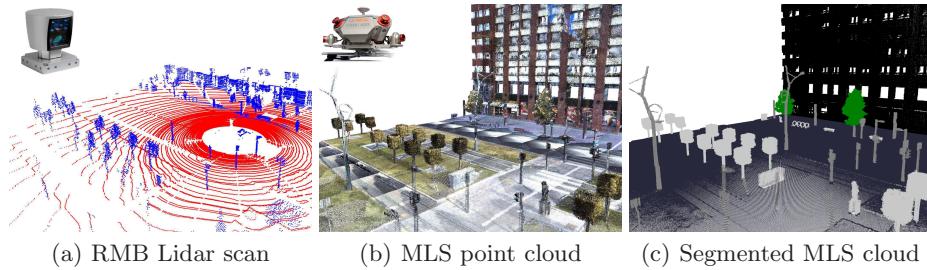


Fig. 1. Point cloud scenes captured at a downtown area using a Velodyne HDL64E Rotating Multi-Beam (RMB) Lidar sensor and a Riegl VMX450 Mobile Laser Scanning (MLS) system. Class color codes in the segmented cloud; black: *facade*, dark gray: *ground*, mid gray: *tall column*, bright gray: *street furniture*, green: *tree crowns*

In this paper, we utilize the measurements of the Velodyne HDL64E RMB Lidar sensor and the Riegl VMX450 MLS system. The Velodyne sensor was originally designed to help real time perception of autonomous vehicles or robots. It provides a stream of relatively sparse ($60-100 \times 10^3$ points/frame) point clouds with a temporal frequency of 10-15 fps. The spatial accuracy is around 1-2cm in the sensor's own coordinate system, but the point density quickly decreases as a

function of the distance from the sensor. As a result of the rotating multi-beam scanning approach, the point clouds show typical ring patterns (Fig. 1(a)).

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 Navigation Satellite System (GNSS). It provides extremely dense, accurate (up to global accuracy of a few centimetres) and feature rich data with relatively uniform point distribution as shown in Fig. 1(b).

For self localization of the SDV, we have to register the sparse Velodyne data with GPS based coarse initial position estimation to the dense and accurate MLS point cloud, used as HD map. While point cloud registration is a deeply explored topic, matching 3D measurements with such different point density and characteristics is a highly challenging task. In our approach, we propose an accurate and fast object based alignment algorithm between the RMB Lidar point clouds and the MLS HD map.

2 Previous work on point cloud registration

Normal Distribution Transform (NDT) [8] and Iterative Closest Point (ICP) [19] algorithms are among the most cited methods in the field of point cloud registration. ICP has several different versions with various improvements: [6] extend the ICP with geometric constraints derived from local point neighborhoods, [10] use color information and [5] make improvements via tracking. However all of these methods are quite sensitive to the different density characteristics of the point clouds, particularly the typical ring pattern of the Velodyne sensor may mislead the registration process. Moreover all of the mentioned methods have a critical precondition: ICP based methods locally minimize the error, so they need a sufficiently accurate pre-alignment between the point clouds. In practice, the GPS based initial alignment with an error of several metres does not prove sufficiently good enough for this purpose. Other techniques focus on applications with larger displacement between the point clouds: [15] and [11] extract local feature descriptors to find global correspondences, which approach can also be used to find an initial pre-alignment before the ICP process. However, as bottleneck, these algorithms have large computational cost even working with smaller point cloud parts, thus in real-time mapping, SLAM and localization applications they are not efficient. A technique has been introduced for scan alignment based on ICP [3], which solves data mapping at the object level by explicitly matching segments across scans rather than using standard point-to-point type of search. This method proved to be efficient for matching Velodyne frames, however the computational time remained 3-15 seconds per scan pair. Registering point clouds with different modalities and density characteristics has a limited bibliography. A sequential technique for cross modal point cloud alignment has been proposed in [4], which extracts first abstract object patches in both point clouds, then it calculates a coarse alignment between the frames purely based on the estimated object centers, finally an NDT based point level

refinement process is applied. As drawbacks, the object level matching may fail in case of several diverse object types, and the additional NDT steps induces significant computational overload.

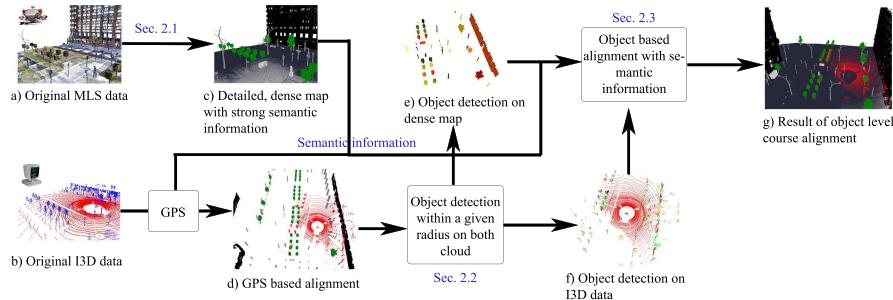


Fig. 2. The workflow of the proposed algorithm.

3 Proposed approach

We propose a real-time, robust object based alignment technique between sparse RMB Lidar measurements and dense MLS point clouds. The workflow of the new approach is shown in Fig. 2. As a preliminary step, we perform a semantic segmentation of the MLS data, and jointly exploit the raw point cloud and the extracted labels as a High Definition map, to support the registration process. For estimating the optimal scan matching transform, we adopt the fingerprint minutia matching algorithm [4, 14], which is able to find a robust transformation between two point sets even if the size of the point sets are different. However, as a novel contribution, instead of considering the object centers only which only enable coarse frame matching [4], we recommend a new key point selection technique that yields an accurate alignment without the computationally expensive point-level refinement step, and also provides more stable results in scenes with several - often only partially extracted - objects. In addition, we also apply semantic constraints for matching the object candidates obtained by a quick segmentation and patch analysis of the actual RMB Lidar frame, and the preliminary extracted segmentation labels of the HD map.

3.1 Creating the reference High Definition map

The Riegl VMX450 system rapidly provides detailed and very dense MLS point cloud data from large-scale outdoor environments. The captured data is geo-referenced, therefore following semantic segmentation it can be directly used as

a 3D High Definition (HD) map. Although manual labeling of billions of points is a highly resource intensive task, deep learning based point cloud classification approaches such as the PointNet++ [13] offer promising way to automate the process. However, dealing with urban MLS data a number of particular challenges appear - such as the phantom effect caused independent object motions [12] - , which are not handled by general point clouds segmentation algorithms. For this reason, we extended here a 3D convolutional neural network based technique [12] developed particularly for MLS data filtering, by accommodating it to separate various urban classes such as *ground*, *facade*, *phantom*, *vehicle*, *pedestrian*, *vegetation* (bushes and tree crowns), *tall column* (including traffic sign holders and tree trunks) and *street furniture* (various further street objects such as benches, dustbins, short columns). Fig. 1(c) demonstrates the result of the labeling process. Object separation in the segmented MLS HD map is quite straightforward: object samples, such as a particular traffic sign, are obtained from the corresponding segmented class regions by Euclidean clustering [16] in an offline way. For the subsequent scene registration process, we will use the objects of the MLS based map as landmarks in background model, therefore will ignore all dynamic (*phantom*, *vehicle*, *pedestrian*) or time-varying objects (*vegetation*), as well as classes of large regions (i.e. *ground* and *facade*). As a consequence, for scene matching we will rely on the extracted object samples of the *tall column*, and *street furniture* classes, which all have a static appearance, compact shape thus can be used as landmarks.

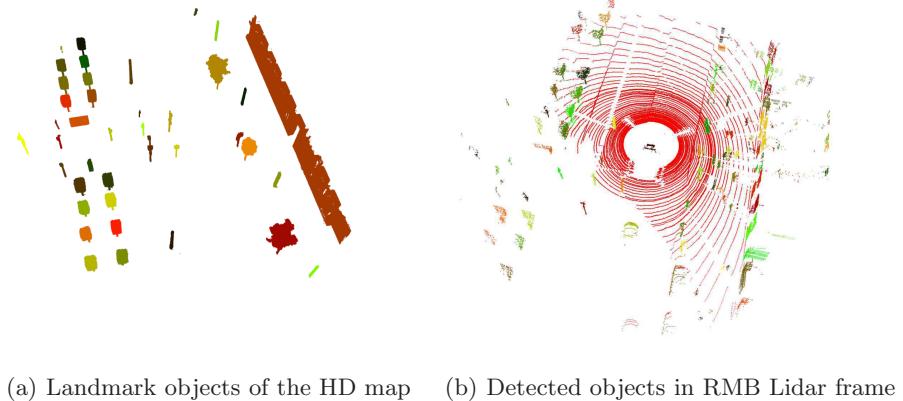


Fig. 3. Extracted objects used for the alignment calculation. Color codes are the following (a) different landmark objects of the HD map are displayed with different color (b) red: ground/road, other colors: different detected objects in the RMB Lidar frame

3.2 Real time object detection in the RMB Lidar point clouds

Since we are dealing with object based point cloud alignment, accurate and robust object detection is also essential in the RMB Lidar frames. On the other hand, the task is highly challenging here, due to the low and inhomogeneous point density, several partially scanned object shapes, occlusions and the real time requirement of the process. First, we only keep the points within a $r = 30\text{m}$ radius region around the sensor's rotation axis (parameter r was optimized for Velodyne HDL64), as the distant regions are too sparse for reliable scene analysis.

The next step is ground removal. On one hand, the typical ring patterns of RMB Lidars particularly affect the ground regions, which phenomena can mislead the registration process. Furthermore, separation of field objects is facilitated by eliminating the ground which connects the object candidates in the raw frames. Planar ground models are often used in the literature based on robust plane estimation methods such as RANSAC, however, they are less efficient in cases of significant elevation differences within the observed terrain parts (e.g. uphill and downhill roads). Instead, we apply a cell based locally adaptive terrain modeling approach based on [2]. First, we fit a regular 2-D grid with 0.2m rectangle width (i.e. grid distance) - optimized to urban environment according to [2] - onto the P_z horizontal plane of the RMB Lidar point cloud's local Euclidean coordinate system. We assign each point to the corresponding cell, which contains its projection to P_z . We mark the cells as ground (road) candidate cells where the differences of the observed maximal and minimal point elevation values are lower than 10cm , which condition admits up to 26° ground slope within a cell. Next, for obtaining a local elevation map, we calculate for the previously marked ground candidate cells the average of the included point elevation coordinates. To eliminate outlier values in the elevation map, resulted by e.g. flat car roofs, we apply a median filter considering neighboring ground cells. For the remaining non-ground cells - which presumptively contain the field objects - the local ground elevation value z_0 is interpolated using the neighboring ground cells, and all included points with elevation z_p are denoted as non-ground points, where $z_p - z_0 > \tau$ (used $\tau = 10\text{cm}$).

After ground removal, we cluster corresponding *non-ground* points to separate individual object candidates. This process is implemented in the 2D cell map with a region growing algorithm, where empty cells act as stopping criterion. Although in this way, some adjacent objects may be merged together due to the limited resolution of the grid, this 2D object detection approach proved to be faster with two orders of magnitude than traditional kd-tree based 3D Euclidean clustering algorithms. Let us note again, that unlike by offline HD map generation, here the processing speed should fulfill the real time requirements.

Fig. 3(b) allows us a qualitative analysis of the object detection step. Field object such as vehicles, columns or tree samples usually appear as separated blobs, while large facade regions are separated into smaller wall segments. Nevertheless, considering the above detailed limitations of the RMB Lidar point clouds, we do not perform a strict classification of the extracted object blobs, and will use all of them in the subsequent scene matching process.

3.3 Object based alignment

In this section, we aim to estimate the optimal geometric transform for registering the sparse *observation frame* recorded by the RMB Lidar to the MLS based HD map data. First, we use the GPS-based coarse position estimation of the vehicle (p_0) for an initial positioning of the *observation frame*'s center in the HD map's global coordinate system. To make the bounding area of the two adjustable point clouds equal, we also cut a 30m radius region from the MLS cloud around the current p_0 position.

Exploiting that the Lidar sensors provide direct measurements in the 3D Euclidean space up to cm accuracy, the estimated spatial transform between the two frames can be represented as a rigid similarity transform with a translation and a rotation component. On one hand, we search for a 3D translation vector (dx , dy and dz), which is equal to the originally unknown position error of the GPS sensor. On the other hand, we have experienced that assuming a locally planar road segment within the search region, the road's local normal vector can be fairly estimated in an analytical way from the MLS point cloud, thus only the rotation component α around the vehicle's up vector should be estimated via the registration step. In summary, we model the optimal transform between the two frames by the following homogeneous matrix:

$$T_{dx,dy,dz,\alpha} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & dx \\ -\sin \alpha & \cos \alpha & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

For limiting the parameter space, we allow a maximum 45° degree rotation (α) in both directions, since from the GPS data, we already know an approximate driving direction. For parameters dx and dy we allow ± 12 m offsets, while for the vertical translation ± 2 m.

We continue with the description of the transformation estimation algorithm. Instead of aligning the raw point clouds, our proposed registration technique matches various keypoints extracted from the *landmark objects* of the HD map (Sec. 3.1), and the *observed object candidates* in the RMB Lidar frame (Sec. 3.2). In addition, exploiting the semantic information stored in the HD, we only attempt to match keypoints which correspond to *compatible* objects. Therefore the remaining part of the algorithm consists of three steps, presented in the following subsections: i) *keypoint selection*, ii) defining *compatibility constrains between observed and landmark objects*, iii) *optimal transform estimation* based on compatible pairs of keypoints.

Keypoint selection A critical step of the proposed approach is keypoint extraction from the observed and landmark objects. A straightforward choice [4] is extracting a single keypoint from each object, taken as the center of mass of the object's blob (see Fig. 4(a) and (b)). However, as discussed earlier, the observed RMB Lidar point clouds contain several partially scanned objects, thus the shape

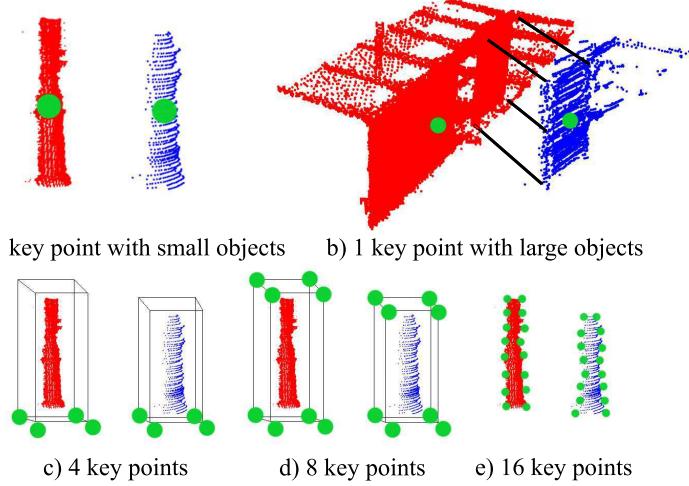


Fig. 4. Choosing key points for registration.

of their point cloud blobs may be significantly different from the more complete MLS point cloud segments of the same object, yielding that the extracted center points are often very different.

For the above reasons, we have implemented various multiple keypoint selection strategies. Beyond the single keypoint based registration approach, we tested the algorithm's performance using 4, 8 and 16 keypoints, whose alignment is demonstrated in Fig. 4(c),(d) and (e). As shown there, using the 4- and 8-keypoint strategies, the feature points are derived as corner points of the 3D bounding boxes of the *observed* and *landmark* objects. For the 16-keypoint case, we divide the 3D bounding box of the object into $2 \times 2 \times 4$ equal cuboid regions, and in each region we select the mass center of the object boundary points as keypoint.

Our expectation is here, that using several keypoints we can obtain correct matches even from partially extracted objects, if certain corners of the (incomplete) bounding box are appropriately detected. On the other hand using a larger number of keypoints induces some computational overload, while due to the increased number of possible point-to-point matching options, it may also cause a false optimum of the estimated transform.

Compatibility constrains between observed and landmark objects As discussed earlier, we estimate the optimal transform between two frames via sets of keypoints. Since we implement an object based alignment process, our approach allows us to filter out several false keypoint matches based on object level knowledge. More specifically, we will only match point pairs extracted from *compatible* objects of the scene. According to the HD map generation process

(Sec. 3.1), we can distinguish *tall column* and *street furniture* samples among the landmark objects, thus all landmark keypoints are derived from samples of the above two object types. Although such detailed object classification was not feasible in the RMB Lidar frame, we prescribe the following compatibility constraints:

- a *tall column* MLS landmark object is compatible with RMB Lidar blobs, which have a column shaped bounding box, i.e. its height is at least twice longer than its width and depth.
- the ratio of the bounding volumes of compatible RMB Lidar objects and *street furniture* MLS landmark objects must be between 0.75 and 1.25.

Applying the above pre-defined constraint we increase of the evidence of a given transformation only if the objects pairs show similar structures, moreover in this way by skipping many transformation calculations we increase the speed of the algorithm.

Note that the RMB Lidar point clouds may contain various dynamic objects such as pedestrians and vehicles, which fulfill the above matching criteria with certain landmark objects of the MLS based map. These objects will generate outlier matches during the transformation estimation step, thus their effect should be eliminated at higher level.

Optimal transform estimation Let us denote the sets of all *observed* and *landmark* objects by \mathcal{O}_o and \mathcal{O}_l respectively.

Using the 3D extension of the Hough transform based schema [14], we search for the best transformation between the two object keypoint sets by a voting process (Fig. 5). We discretize the transformation space between the minimal and maximal allowed values of each parameter, using 0.2m discretization steps for the translation components and 0.25° steps for rotation.

Next we allocate a four dimensional array to summarize the votes for each possible (dx, dy, dz, α) discrete quadruple, describing a given transformation. We set zero initial values of all elements of this array.

During the voting process, we visit all the possible $\mathcal{O}_o, \mathcal{O}_l$ pairs of *compatible* objects from $\mathcal{O}_o \times \mathcal{O}_l$. Then, we attempt to match each keypoint of \mathcal{O}_o to the corresponding keypoint in \mathcal{O}_l , so that for such a keypoint pair o_o, o_l , we add a vote for every possible $T_{dx, dy, dz, \alpha}$ transform, which maps o_o to o_l . Here we iterate over all the discrete $\alpha^* \in [-45^\circ, +45^\circ]$ values one by one, and for each α^* we rotate o_o with the actual α^* , and calculate a corresponding translation vector $[dx^*, dy^*, dz^*]^T$ as follows:

$$\begin{bmatrix} dx^* \\ dy^* \\ dz^* \end{bmatrix} = o_l - \begin{bmatrix} \cos \alpha^* & \sin \alpha^* & 0 \\ -\sin \alpha^* & \cos \alpha^* & 0 \\ 0 & 0 & 1 \end{bmatrix} o_o$$

Thereafter we increase the number of votes given for the $T_{dx^*, dy^*, dz^*, \alpha^*}$ transform candidate. Finally at the end of the iterative voting process, we find the maximum value of the 4-D vote array, whose $(\alpha, dx, dy$ and $dz)$ parameters represent the optimal transform between the two object sets.

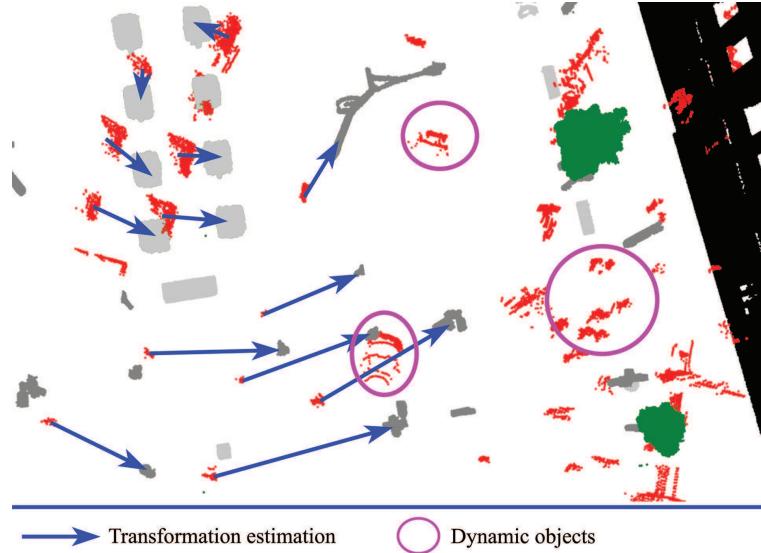


Fig. 5. Illustration of the output of the proposed object matching algorithm based on the *fingerprint minutiae* approach [14]. Red points mark the objects observed in the RMB Lidar frame.

4 Evaluation

We evaluated the proposed method on different scenarios from dense city areas, some qualitative results are shown in Fig. 6. During quantitative evaluation we compared the different keypoint selection strategies using our proposed model, and also compared our approach to the state-of-the-art technique [4]. As evaluation metrics we used the average distance of the keypoints after the optimal point cloud alignment, since following our subjective visual verification this metrics proved to be relevant for numerical comparison of different matches.

Fig. 7 demonstrates the result of the method comparison in 25 different RMB Lidar frames, where we displayed the calculated transformation scores in a logarithmic scale. Regarding the keypoint selection, Fig. 7 shows that the optimal strategy proved to be the 8-keypoint approach (shown in Fig. 4(d)), with an average error between 0.15 and 0.5 meters for the different frames. We can observe that using 1 or 4 keypoints, the resulting error is slightly higher than with applying the 8-point version. On the other hand, 16 keypoint selection suffers from overfitting problems, since it yields large errors in some of the frames.

By comparing the proposed method to [4], we can experience the clear superiority of our new technique with any keypoint selection variants. On one hand, the reference technique [4] only used the 2D object centers from a top-view projection to find the optimal transform, which solution was only appropriate to find a coarse matching between the two point cloud frames. On the other hand,

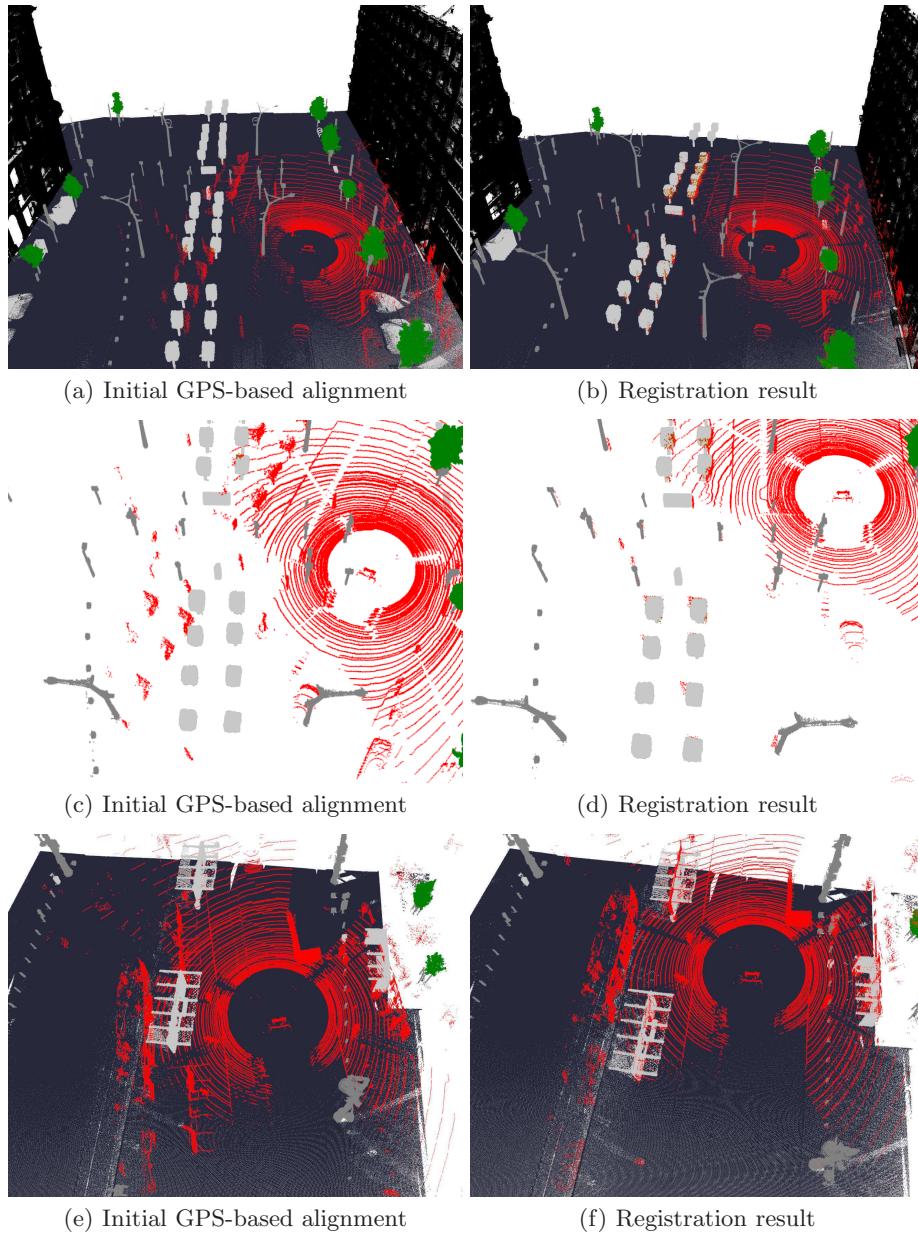


Fig. 6. Results of the proposed registration approach with 8 keypoint selection strategy. RMB Lidar point clouds are displayed with red, while the MLS data is shown with multiple colors depending on the segmentation class. First two rows correspond to the same scene, just ground and facades are not displayed in the 2nd row for better visualization of the object based alignment

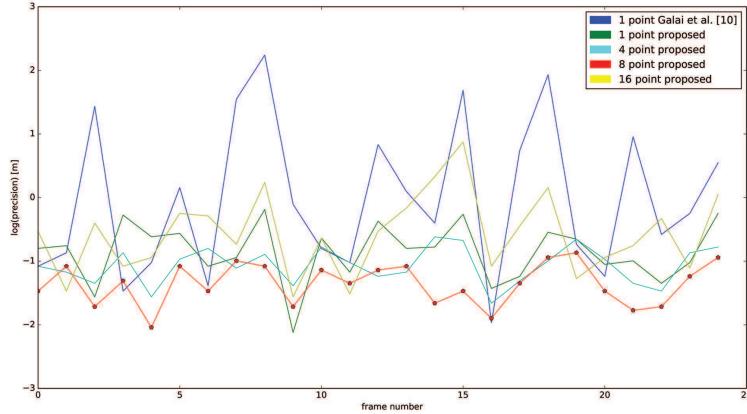


Fig. 7. Evaluation of the proposed approach with various keypoint selection strategies and comparison to [4]

[4] did not use any object specific knowledge from the HD map, that highly contributed to eliminate false matches in our present model. In addition, since our method does not use the computationally expensive point level NDT refinement step, it is able to run with 10 frame/seconds (fps) on a desktop computer, in contrast to the 0.5 fps speed of [4].

5 Conclusions

We have proposed an object based algorithm for accurate localization of self driving vehicles (SDV) equipped with a RMB Lidar sensor. Assuming that a High Definition (HD) point cloud map is available from the environment obtained by Mobile Laser Scanning technology, the problem is to solve the registration of point clouds with significantly different density characteristics. Apart from exploiting semantic information from the HD map, various keypoint selection strategies have been proposed and compared. We have experienced that the 8-keypoint approach yields a highly efficient solution for the problem, which is superior over other keypoint selection strategies and also over a state-of-the-art method.

Acknowledgment

This work was supported by the National Research, Development and Innovation Fund (grants NKFIA K-120233 and KH-125681), and by the Széchenyi 2020 Program (grants EFOP-3.6.2-16-2017-00013 and 3.6.3-VEKOP-16- 2017-00002). Cs. Benedek also acknowledges the support of the Janos Bolyai Research Scholarship of the Hungarian Academy of Sciences.

References

1. Bayat, B., Crasta, N., Crespi, A., Pascoal, A.M., Ijspeert, A.: Environmental monitoring using autonomous vehicles: a survey of recent searching techniques. *Current Opinion in Biotechnology* **45**, 76 – 84 (2017). <https://doi.org/https://doi.org/10.1016/j.copbio.2017.01.009>, <http://www.sciencedirect.com/science/article/pii/S0958166916302312>
2. Börcs, A., Nagy, B., Benedek, C.: Fast 3-D urban object detection on streaming point clouds. In: *Workshop on Computer Vision for Road Scene Understanding and Autonomous Driving at ECCV'14*, LNCS, vol. 8926, pp. 628–639. Springer, Zürich, Switzerland (2015)
3. Douillard, B., Quadros, A., Morton, P., Underwood, J.P., Deuge, M.D., Hugosson, S., Hallström, M., Bailey, T.: Scan segments matching for pairwise 3D alignment. In: *IEEE International Conference on Robotics and Automation (ICRA)*. pp. 3033–3040. St. Paul, MN, USA (May 2012). <https://doi.org/10.1109/ICRA.2012.6224788>
4. Gálai, B., Nagy, B., Benedek, C.: Crossmodal point cloud registration in the Hough space for mobile laser scanning data. In: *International Conference on Pattern Recognition (ICPR)*. pp. 3374–3379. IEEE, Cancun, Mexico (2016)
5. Gressin, A., Cannelle, B., Mallet, C., Papelard, J.P.: Trajectory-Based Registration of 3D LIDAR Point Clouds Acquired with a Mobile Mapping System. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* pp. 117–122 (Jul 2012). <https://doi.org/10.5194/isprsannals-I-3-117-2012>
6. Gressin, A., Mallet, C., David, N.: Improving 3D LIDAR Point Cloud Registration Using Optimal Neighborhood Knowledge. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* pp. 111–116 (Jul 2012). <https://doi.org/10.5194/isprsannals-I-3-111-2012>
7. Kang, M., Hur, S., Jeong, W., Park, Y.: Map building based on sensor fusion for autonomous vehicle. In: *International Conference on Information Technology: New Generations*. pp. 490–495. Las Vegas, NV, USA (April 2014). <https://doi.org/10.1109/ITNG.2014.30>
8. Magnusson, M.: The Three-Dimensional Normal-Distributions Transform – an Efficient Representation for Registration, Surface Analysis, and Loop Detection. Ph.D. thesis, Örebro University (December 2009)
9. Matthaei, R., Bagschik, G., Maurer, M.: Map-relative localization in lane-level maps for ADAS and autonomous driving. In: *IEEE Intelligent Vehicles Symposium Proceedings*. pp. 49–55. Dearborn, MI, USA. (June 2014). <https://doi.org/10.1109/IVS.2014.6856428>
10. Men, H., Gebre, B., Pochiraju, K.: Color point cloud registration with 4D ICP algorithm. In: *IEEE International Conference on Robotics and Automation (ICRA)*. pp. 1511–1516. Shanghai, China (May 2011). <https://doi.org/10.1109/ICRA.2011.5980407>
11. Mian, A., Bennamoun, M., Owens, R.: On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *International Journal of Computer Vision* **89**(2), 348–361 (Sep 2010). <https://doi.org/10.1007/s11263-009-0296-z>
12. Nagy, B., Benedek, C.: 3D CNN based phantom object removing from mobile laser scanning data. In: *International Joint Conference on Neural Networks (IJCNN)*. pp. 4429–4435. Anchorage, AK, USA (May 2017)

13. Qi, C., Yi, L., Su, H., Guibas, L.: PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In: Conference on Neural Information Processing Systems (NIPS). pp. 5105–5114. Long Beach, CA, USA (2017)
14. Ratha, N.K., Karu, K., Chen, S., Jain, A.K.: A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18**(8), 799–813 (Aug 1996). <https://doi.org/10.1109/34.531800>
15. Rusu, R.B., Blodow, N., Beetz, M.: Fast Point Feature Histograms (FPFH) for 3D registration. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 3212–3217. Kobe, Japan (May 2009). <https://doi.org/10.1109/ROBOT.2009.5152473>
16. Rusu, R.B., Cousins, S.: 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA). pp. 1–4. Shanghai, China (May 2011). <https://doi.org/10.1109/ICRA.2011.5980567>
17. Seif, H.G., Hu, X.: Autonomous Driving in the iCityHD Maps as a Key Challenge of the Automotive Industry. *Engineering* **2**(2), 159 – 162 (2016). <https://doi.org/https://doi.org/10.1016/J.ENG.2016.02.010>, <http://www.sciencedirect.com/science/article/pii/S2095809916309432>
18. Yu, Y., Li, J., Guan, H., Wang, C.: Automated detection of three-dimensional cars in mobile laser scanning point clouds using DBM-Hough-Forests. *IEEE Trans. Geosci. and Rem. Sens.* **54**(7), 4130–4142 (2016). <https://doi.org/10.1109/TGRS.2016.2537830>
19. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision* **13**(2), 119–152 (1994)