HIGH-ACCURACY VEHICLE LOCALIZATION USING A PRE-BUILT PROBABILITY MAP

Yunming Shao, PhD Candidate
Charles Toth, Professor
Dorota Grejner-Brzezinska, Lowber B. Strange Endowed Professor and Chair
Department of Civil, Environmental and Geodetic Engineering
The Ohio State University
2070 Neil Avenue, Columbus, OH 43210 USA
(shao.209, toth.2, grejner-brzezinska.1) @osu.edu

ABSTRACT

One of the significant challenges to achieving fully autonomous driving is reliable and accurate vehicle localization, based on multiple sensors on board. However, the accuracy and robustness required by autonomous vehicles cannot be fulfilled with traditional GPS(GNSS)-Inertial Navigation Systems (GPS/INS), especially in large-scale urban settings. Researchers have investigated this problem and proposed different solutions. In this paper, we propose a high-accuracy vehicle localization framework using a pre-built probability map. We firstly obtain an optimized pose (position and orientation) by combining GPS/INS solution, LiDAR scan matching and visual odometry. Then, the pose and the LiDAR scan measurements are utilized to build a two-dimensional (2D) probability map. This is a grid map, in which two values are calculated to represent each cell based on all of the LiDAR points enclosed in that cell: the mean and the variance of the reflectance intensity values. Finally, we implement a histogram filter method to localize the vehicle using the pre-built map. Experiments on the Oxford Robot Car datasets demonstrate that our framework have obtained competitive accuracy.

KEYWORDS: autonomous driving, mapping, vehicle localization, histogram filter

INTRODUCTION

Since the DARPA Grand Challenges (Thrun et al., 2006), Autonomous Vehicles have grown into a reality supported by the progress in mapping and localization research, smarter fusion of various sensors, and employment of the Artificial Intelligence techniques. Autonomous Vehicles are already on the road (Kelly, 2013) and will gradually become available for consumers. Most traditional automobile companies and some of the leading information technology companies have announced their timeline to make autonomous vehicles commercially available, such as Ford, Volvo and Uber.

Due to the high safety and robustness requirements, the state-of-the-art localization methods (Levinson et al., 2007, 2010) and (Wolcott and Eustice, 2015) mostly rely on a pre-built map of the environment. By manual or semi-automatic annotating of the static objects, such as traffic signs and traffic lights, in the map, we transform the difficult perception problem into a localization problem. For example, once we localized the vehicle in the pre-built map with traffic signs annotated, we obtained the relative position between the vehicle and the traffic signs, thus avoiding online detection the position of the traffic signs. The reflectance measurements from LiDAR contains rich information about the target, and ground plane, i.e., road surface, is a desired mapping choice because it is relatively stable over time. Therefore, LiDAR reflectance is often used to create the map of ground plane.

Note that a high cost three-dimensional (3D) LiDAR was exclusively used in the previous publications on the subject matter (Levinson et al., 2007; Fu et al., 2016). Recently, The Oxford RobotCar Dataset (Maddern et al., 2016) was publicly released, which contains various sensor data collected during repeated travels on the same route for more than 100 times. The Oxford RobotCar is equipped with one stereo camera, three monocular cameras, 2D and 3D LiDAR scanners, and high-end GPS/INS system. Please reference to Figure 1 for sensor configuration.

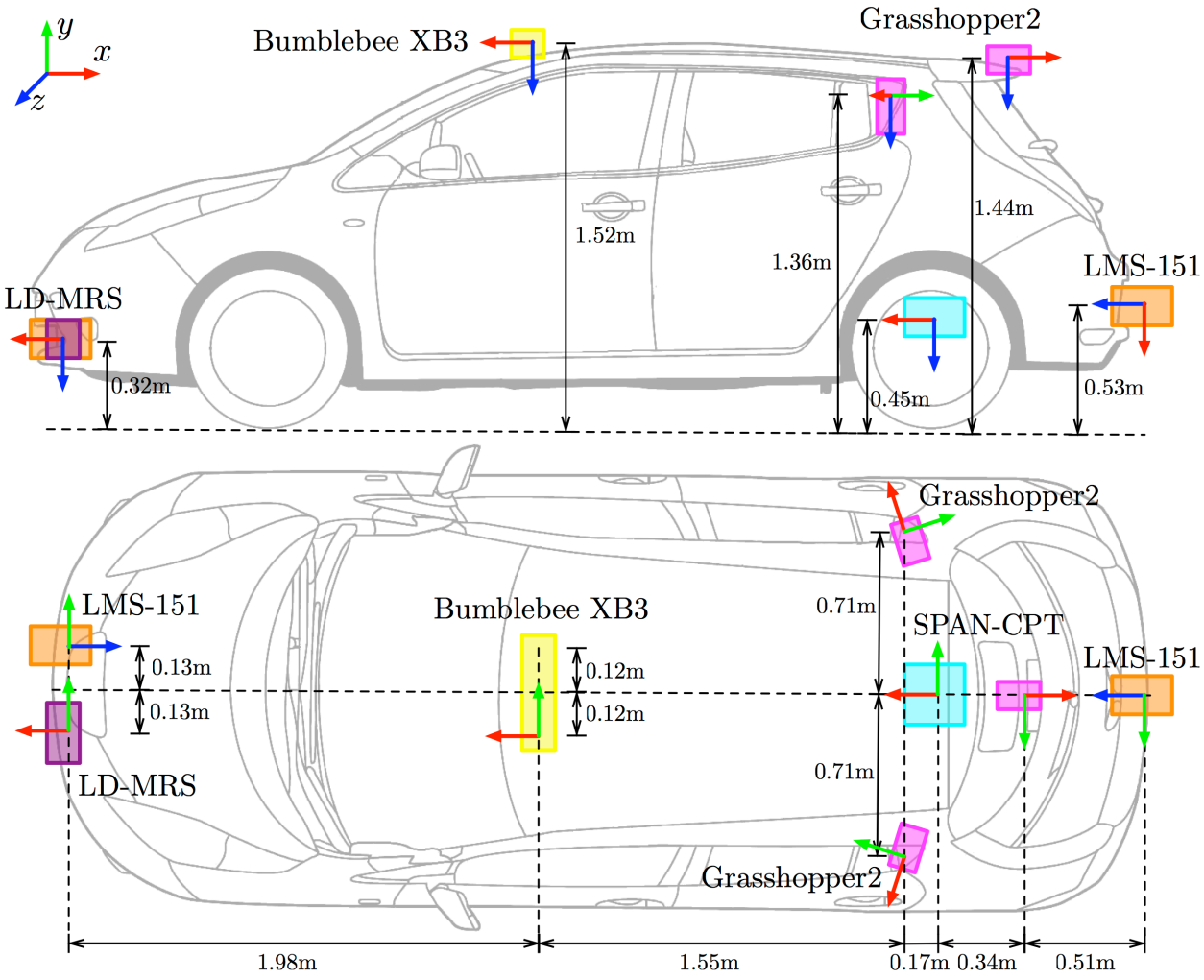

Figure 1. Sensor position on vehicles. Coordinate frames use the convention x = forward (red), y = right (green), z = down (blue). (From http://robotcar-dataset.robots.ox.ac.uk/documentation/)

Among these sensors, what interest us is two SICK 2D LMS-151 LiDAR scanners pointing downwards to the ground. The price of a 2D LiDAR scanner is only a fraction of that of a 3D LiDAR, therefore it is more affordable. In this paper, we intend to leverage these 2D LiDAR scanners to build a reflectance probability map, and utilize the measurements from the same 2D LiDAR to localize the vehicle. However, we also use other sensors on board, which will be explained later in this paper.

Figure 2 describes the framework of our localization system. From inside to outside, there are three modules: pose optimization, 2D grid map building, and grid localization. The innermost black dashed box contains the post optimization module, which takes in LiDAR scan matching and stereo visual odometry relative poses, and the absolute poses from GPS/INS system (hereafter we will refer to GPS/INS pose). Its output is the 6-degree of freedom (6DOF) optimized poses. Then, the 2D grid map building module utilizing the optimized poses and 2D LiDAR scans to produce a 2D grid map, is included in outer red solid box. These two modules are executed offline and prior to the online grid localization. Finally, the grid localization module in the outermost black dashed box localizes the vehicle by computing the posterior probability of the offsets from positioning results of GPS/INS system using a 2D histogram filter. This is a high-level summary of our framework, and readers should refer to the following sections for a comprehensive understanding.

The following section discusses related work. Next, we introduce our probability map building system and localization system. Then, experiments and results are presented. Finally, we offer summary and conclusions.

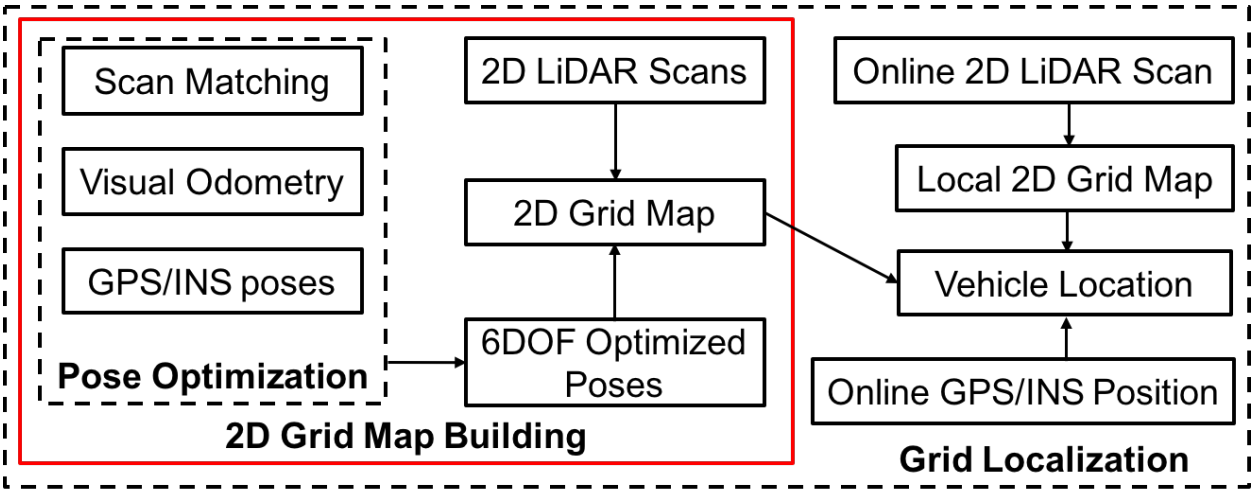

Figure 2. Workflow of the proposed vehicle localization system. From inside to outside, there are three modules: pose optimization (contained in innermost black dashed box), 2D grid map building (contained in outer red solid box), and grid localization (contained in outermost black dashed box).

RELATED WORK

To realize level 3 or above autonomous driving defined by Society of Automotive Engineers (SAE, 2014), the autonomous vehicles must be able to localize itself relative to the nearby environment. One of the most promising mapping characteristics is the ground plane reflectance intensity, which can capture lane markings, pavement edges or even braking marks. All these features can help localize the vehicles.

(Levinson et al., 2007) is the first paper that proposed and implemented this idea to their autonomous vehicle. In their work, a 3D LiDAR scanner was used to observe the ground plane reflectivity, with which they built a map of ground plane reflectivity. Within this map, they then localized their autonomous vehicles using incoming 3D LiDAR scans and an IMU. To increase robustness to environment change, they extended their map to a probability grid map whereby each cell is represented as its own Gaussian distribution over reflectivity values (Levinson et al., 2010), but still relied on a 3D LiDAR scanner during the mapping and localization stages.

In an effort to lower the cost, (Wolcott and Eustice, 2015) proposed a visual localization within a pre-built 3D LiDAR map by maximizing the Normalized Mutual Information (NMI) using images from significantly cheaper camera. However, it still relied on the map created from 3D LiDAR scanner, and the light condition influences the performance of the camera leading to poor localization performance. Another work of (Baldwin and Newman, 2012) employed one 2D LiDAR scanner to build 3D swathes as the vehicle traversed the route. Our work is similar to this approach, but we employ a different mapping and localization approaches, as explained in the sequel.

The localization is equivalent to registering a locally observed point clouds to some prior representation of the environment. Some of the popular methods including Iterative Closest Point (ICP) (Besl and McKay, 1992), Generalized Iterative Closest Point (GICP) (Segal et al., 2009), and Normal Distribution Transform (NDT) (Magnusson, 2009) are used to register an observed point clouds to another point clouds. But these methods are highly dependent on the initial position and are prone to falling into local minimums. In our implementation, instead of directly registering the point clouds, we transform the point clouds to 2D grid map and then estimate the position of the vehicle in the grid map using a 2D histogram filter. The advantage of a 2D grid map lies in its simple map representation, low storing consuming of memory, and thus more proper for real-time localization application.

PRE-BUILT PRABALIBITY MAP

As described in the introduction, our framework consists of three primary modules: pose optimization, 2D grid map building, and grid localization. The purpose of the first two modules is to build a 2D grid map as the representation of ground plane features. Each cell in the grid map not only contains the mean but also the variance of the reflectance, which make it a probability map because each cell maintains a Gaussian distribution defined by the mean and variance of the enclosed reflectivity values. This probability map is more robust compared with fixed reflectance value map by considering the variance of the reflectance over the environments. The map is generated in an offline manner, and the subsequent grid localization step is conducted using this pre-built map. In the next section, we will explain it in details.

Pose Optimization

An optimized pose is used to assemble the original 2D LiDAR scans. To demonstrate that GPS/INS poses are not sufficient to build continuous point clouds, we produced a local 3D point cloud by assembling the 2D LiDAR scans using GPS/INS pose. Then, the point cloud was projected onto the images, as shown in Figure 3. As can be observed in the figure, the point cloud generated from GPS/INS pose does not match with the image, while point cloud from visual odometry pose match well with the image. This indicates that GPS/INS pose is not sufficient to build a LiDAR based map.

We cannot rely only on visual odometry or scan matching relative pose to build the map, because both poses will drift over larger distance because the lack of absolute reference and the accumulative error (Maddern et al., 2016). To take advantage of the high-accuracy relative pose of scan matching and visual odometry, we combine it with GPS/INS pose to produce an optimized pose. The state-of-the-art nonlinear least-square pose-graph (Kümmerle et al., 2011; Dellaert, 2012; Kaess et al., 2012) approach is designed to solve this kind of full Simultaneous Localization And Mapping (SLAM) problem.

We first construct a pose-graph to solve the full SLAM problem, as shown in Figure 4. In this pose graph, each node represents a pose and each edge represents a constraint. To fully take advantage of the Oxford Robot Car Datasets, three kinds of constrains are applied in our implementation: GPS a priori constrain over each node (G), stereo visual odometry constrain (V), and 3D LiDAR scan matching constrain (S). The odometry constrain is from the stereo images using the visual odometry system described in (Churchill, 2012), which provide the 6DOF relative pose estimates between two nodes. We processed the 3D LiDAR scanner data using the GICP methods in this paper (Segal et al., 2009), resulting in the scan matching constrain between two nodes. All these constrains are modeled as Gaussian random variables. After the pose graph is constructed, we optimize the pose graph with the GTSAM (Dellaert, 2012) algorithm.

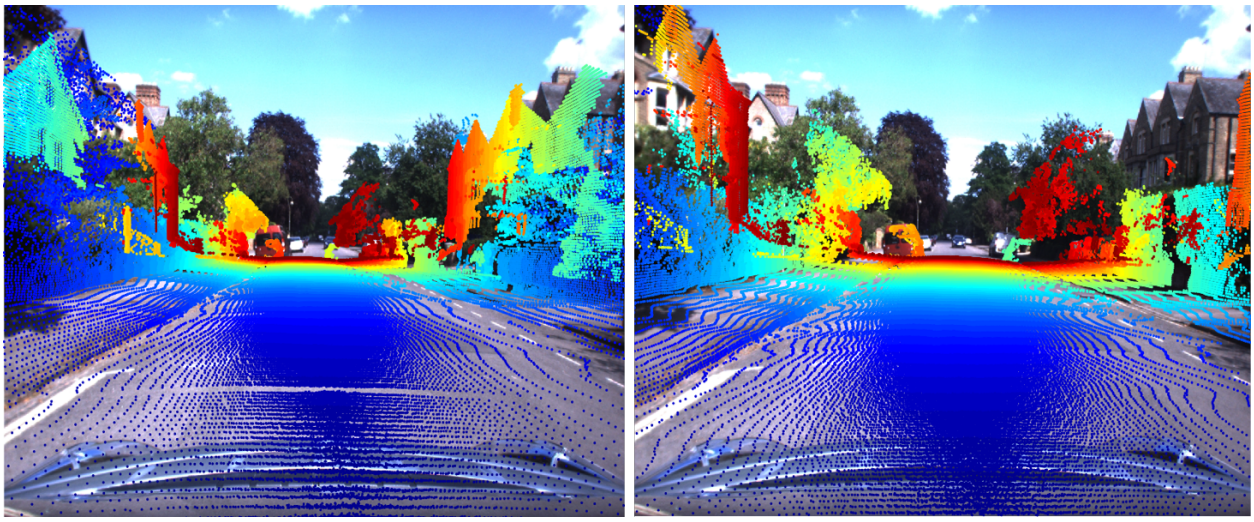

Figure 3. 3D point cloud created using GPS/INS poses (left) and visual odometry poses (right). GPS/INS poses is from the GPS/INS system onboard, and visual odometry poses is obtained from is from the stereo images using the visual odometry system described in (Churchill, 2012). Left: with poor GPS/INS poses, for example due to loss of satellite signals, the generated point cloud is corrupted and does not match the image. Right: visual odometry pose generated point cloud matches the image much better.

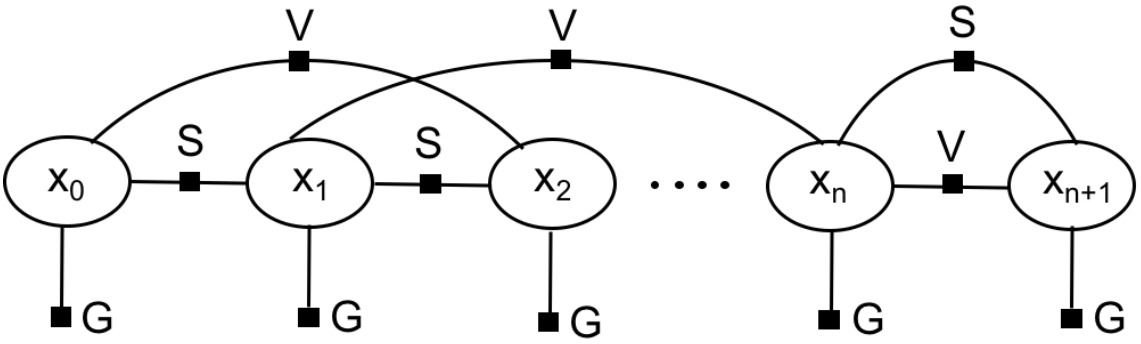

Figure 4. Factor Graph constructed to solve for the pose-graph SLAM problem in the pose optimization module. Here, X_i represents vehicle pose state, i.e., position and orientation. G, S and V represent three types of constrains: G for GPS/INS priori constrains, S for Scan Matching constrains, and V for and Visual Odometry constrains (V). All constraints are modeled as Gaussian random variables

2D Grid Map Building

Based on the optimized 6DOF pose, the measurements from two 2D LiDAR scanners are then assembled to generate point clouds. The map is initialized as a 2D grid of 20 cm spacing, and then the point clouds are projected on the initial map. For all points in each cell, two values are calculated: the mean and variance of the reflectance intensity.

It is worth noting that only points on or near the ground plane is retained in the process of mapping. Because 3D point clouds are obtained from 6DOF optimized trajectory and 2D LiDAR scans, we only project points near the road surface to the map based on the z-height value. In this way, we remove the potential dynamic objects on the road, for instance cars driving by. The resulting map is a 2D grid probability map, and each grid has two values. Figure 5 is the visualization of the results. We summarized our mapping procedure in Algorithm 1.

Algorithm 1 Probability grid map builder

Input: Optimized 6DOF poses $X = \{x_1, x_2, x_3, \ldots x_{n-1}, x_n\}$, 2D LiDAR scans $S = \{s_1, s_2, s_3, \ldots s_{n-1}, s_n\}$
Output: 2D grid map, *map*

Initialize 20 cm sparse grid, *grids*
Combine 2D LiDAR scan S and poses X to get 3D point clouds $P = \{p_1, p_2, p_3, \ldots p_{m-1}, p_m\}$
for p_i in P **do**
    check z-height to determine if p_i is near the ground plane
    if yes
        project p_i to *grid j*
        update *mean* and *variance* for this *grid j*
    end if
end for
discard grid without point or with small amount of points projected on

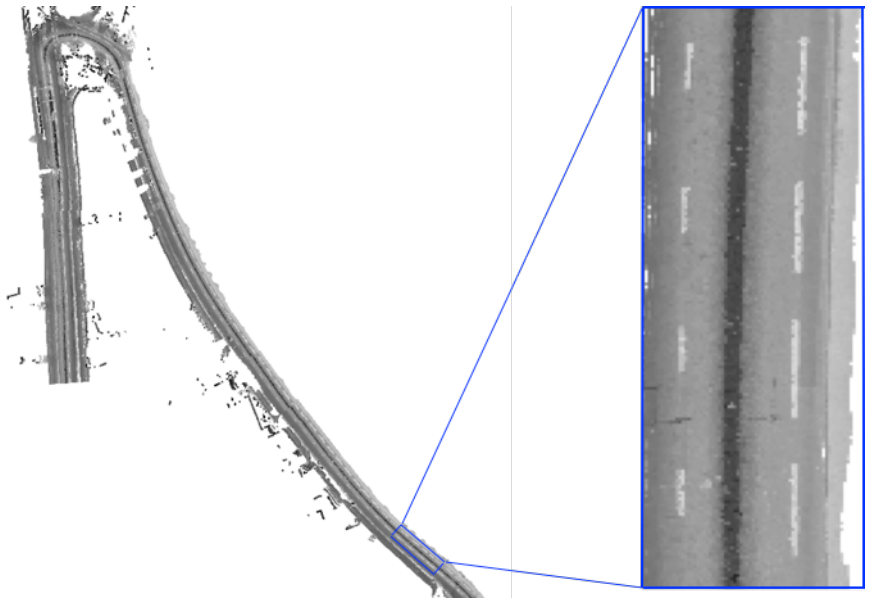

Figure 5. The 2D grid map shows the mean reflectivity of 20x20cm cell, which is built from 2D LiDAR scans and optimized poses. Left: Overhead view of the 2D grid map, with length about 3 minutes driving. Right: partial zoom in to the ground plane, and the highly-reflective road markings are clearly visible.

GRID LOCALIZATON

A smooth horizontal vehicle location can be obtained by integrating the horizontal velocity from the GPS/INS system, but this vehicle location would diverge from the truth, resulting in an offset (x, y). Recovering these offsets is equivalent to estimating the vehicle location. Thus, it is this offset that our localization module is trying to estimate.

To fully exploit the potential of the probability map, we choose grid localization to localize our vehicle in real time. As explained in (Thrun et al., 2005), grid localization approximates the posterior probability using a histogram filter over a grid decomposition of the pose space. Histogram filter is a nonparametric filter, which does not make any assumptions of the probability distribution. However, as a filter, it is also comprised of two steps: the motion update and measurement update. Next, we will briefly describe our implementation of these two steps based on the algorithm presented in (Levinson et al., 2007).

Motion Update

While the drift can be modeled with a Gaussian noise variable with zero-mean, the motion update can be modeled as:

$$\bar{P}(x,y) = \eta \cdot \sum_{i,j} P(i,j) \cdot \exp\left(\frac{(i-x)^2(j-y)^2}{-2\sigma^2}\right) \quad (1)$$

where $\bar{P}(x,y)$ is the posterior probability that the offset is located in cell (x,y) after motion update. (i,j) is the index of surrounding cells of cell (x,y), which can be as small as several cells away from cell (x,y) because of the high update rate and low drift rate. η is the normalizing constant. σ is the drift rate.

Measurement Update

The measurement update step uses the incoming LiDAR scan to refine the estimate of the offsets from the motion update. The way we process the incoming 2D LiDAR scan is the same as the mapping procedure described above in

"Algorithm 1", except that the input is only one scan, so no pose is needed. After this process, the LiDAR scan measurements are transformed to a local 2D grid map, which is represented by m. We also represent the pre-built grid map by M. According the Bayes' Rule:

$$P(x, y|m, M) = \eta \cdot P(m|x, y, M) \cdot P(x, y) \tag{2}$$

$P(x, y)$ can be modeled as the product of GPS Gaussian with variance σ_{GPS}^2 and the posterior probability after motion update:

$$P(x, y) = \eta \cdot \exp\left(\frac{x^2+y^2}{-2\sigma_{GPS}^2}\right) \cdot \bar{P}(x, y) \tag{3}$$

Here comes the core of histogram filter, which is to represent the cumulative posterior for each region by a single probability value. In our case, the probability of a local grid map m given an offset (x, y) and pre-built map M can be expressed in the following way:

$$P(m|x, y, M) = \prod_{i,j} \exp\left(\frac{-(M^m_{(i-x,j-y)} - m^m_{(i,j)})^2}{2(M^\sigma_{(i-x,j-y)} + m^\sigma_{(i,j)})^2}\right)^\alpha \tag{4}$$

where M^m and M^σ is the mean and standard deviation of the reflectance intensity in certain cell on pre-built map M; m^m and m^σ is the mean and standard deviation of the reflectance intensity in certain cell on local map m. As explained above, local map m is from the online 2D LiDAR scan, and pre-built map is a previous built 2D grid map M. $\alpha < 1$ is used to account for the fact that the 2D LiDAR scans are not entirely independent, for example due to systematic calibration error.

Combine (2)(3)(4), we have:

$$P(x, y|m, M) = \eta^2 \cdot \prod_{i,j} \exp\left(\frac{-(M^m_{(i-x,j-y)} - m^m_{(i,j)})^2}{2(M^\sigma_{(i-x,j-y)} + m^\sigma_{(i,j)})^2}\right)^\alpha \cdot \exp\left(\frac{x^2+y^2}{-2\sigma_{GPS}^2}\right) \cdot \bar{P}(x, y) \tag{5}$$

Because the localization accuracy of our GPS/INS system on board alone can reach several meters, only $P(x, y|m, M)$ of cells within this range are computed, which also reduces our computational burden. For situation that GPS signal is not available, we must increase this range to account for increasing drift of the INS system.

Offset Estimate

The final estimate of the offset (x, y) is:

$$X = \frac{\sum_{x,y} P(x,y)^\beta \cdot x}{\sum_{x,y} P(x,y)^\beta} \qquad Y = \frac{\sum_{x,y} P(x,y)^\beta \cdot y}{\sum_{x,y} P(x,y)^\beta} \tag{6}$$

where $P(x, y)$ is the abbreviation of $P(x, y|m, M)$, which is the posterior probability that the offset is located in cell (x, y) given the pre-built map M and local map m. x and y is offsets in lateral and longitudinal direction in a

certain cell. $\beta > 1$ is a constant parameter. These are all known. X and Y is the unknown offsets, which is our goal to estimate.

Even though, the natural choice of offset is the offset (x, y) that can maximize the posterior probability $P(x, y)$, it is not appropriate for autonomous vehicle application because the maximum of a multimodal distribution can vary significantly, leading very unstable offsets. In our implementation, we use the above formula (6) to estimate the offsets. If there is no β in this equation, the estimate is the weighted mean of all offsets. $\beta > 1$ is used to prevent the estimated offset (x, y) biased too much to the center of posterior probability. It turns out to be a very stable and accurate estimate.

EXPERIMENTS AND RESULTS

To complete a quantitative analysis of the above framework, we validated and tested it on the Oxford Robot Car Datasets. We have described this dataset and the sensor configuration in the introduction part of this paper. As a preliminary experiment, we selected about 3 minutes of a driving route, and then picked one driving-through of that route completed in June, 2014. Based on the sensor data collected during this driving-through, we built a 2D probability grid map of the route using the mapping algorithms described above. Figure 5 shows part of the route. Once the 2D grid map was created, sensor data collected during a survey of June, 2014 was used to simulate the process of online localization, by localizing the vehicle in the pre-built map. We also optimized a trajectory using the pose graph optimization algorithm from scan matching, visual odometry and GPS/INS, as the ground truth.

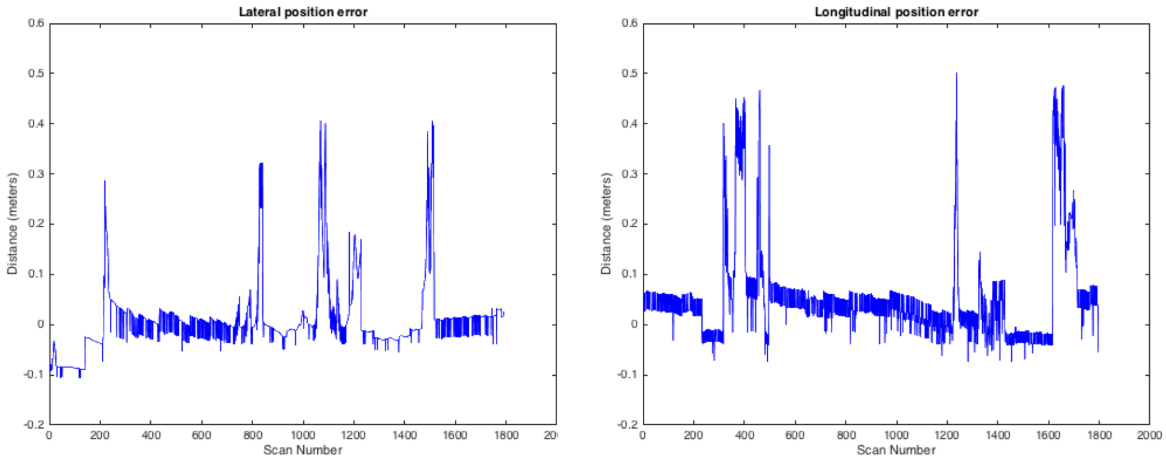

Figure 6. Vehicle localization lateral error (left) and longitudinal error (right) of our localization algorithm within the pre-built 2D grid probability map. The pose-graph optimized trajectory is deemed as the ground-truth.

Figure 6 left shows the lateral error, which is the lateral direction difference between our localization results and the ground truth during the about 3 minutes' drive. As the figure illustrates, the resulting error is much lower than several meters, which is the error level of GPS/INS only system. Figure 6 right shows the longitudinal error. From the example shown in the figures, we found that the longitudinal error is relatively larger than the lateral error, with longitudinal error RMS of 13cm and lateral error RMS of 11cm. The reason is that the number of features, such as lane markers, in longitudinal direction is much less than that of the lateral direction.

CONCLUSIONS

In this paper, we proposed a vehicle localization framework in a pre-built grid map. Different from the existing published approach of either using expensive 3D LIDAR for mapping and localization, or using a much less expensive

camera for localization but suffering from high requirement of lighting condition, our framework utilizes two 2D LIDAR to build a 2D probability grid map of the ground plane and localize the vehicle in it. The price of 2D LIDAR is only a fraction of that of a 3D LIDAR. Preliminary experiments on the Oxford Robot Car Datasets demonstrate that our framework has the potential of localizing the vehicle in an accuracy of decimeter.

REFERENCES

Baldwin, I. and Newman, P., 2012, May. Road vehicle localization with 2d push-broom lidar and 3d priors. *In Robotics and automation (ICRA), 2012 IEEE international conference* on (pp. 2611-2617). IEEE.

Besl, P.J. and McKay, N.D., 1992, April. Method for registration of 3-D shapes. *In Robotics-DL tentative (pp. 586-606). International Society for Optics and Photonics.*

Churchill, W.S., 2012. Experience based navigation: Theory, practice and implementation (*Doctoral dissertation, University of Oxford*).

Dellaert, F., 2012. Factor graphs and GTSAM: A hands-on introduction. *Georgia Institute of Technology*.

Fu, H., Ye, L., Yu, R. and Wu, T., 2016, August. An efficient scan-to-map matching approach for autonomous driving. *In Mechatronics and Automation (ICMA), 2016 IEEE International Conference* on (pp. 1649-1654). IEEE.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J.J. and Dellaert, F., 2012. iSAM2: Incremental smoothing and mapping using the Bayes tree. *The International Journal of Robotics Research*, 31(2), pp.216-235.

Kelly, H., "Self-driving cars now legal in California". *CNN*. Retrieved 11 October 2013.

Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K. and Burgard, W., 2011, May. g 2 o: A general framework for graph optimization. *In Robotics and Automation (ICRA), 2011 IEEE International Conference* on (pp. 3607-3613). IEEE.

Levinson, J., Montemerlo, M. and Thrun, S., 2007, Map-Based Precision Vehicle Localization in Urban Environments. *In Robotics: Science and Systems* (Vol. 4, p. 1).

Levinson, J. and Thrun, S., 2010, Robust vehicle localization in urban environments using probabilistic maps. *In Robotics and Automation (ICRA), 2010 IEEE International Conference* on (pp. 4372-4378). IEEE.

Maddern, W., Pascoe, G., Linegar, C. and Newman, P., 2016. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*, p.0278364916679498.

Magnusson, M., 2009. The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection (*Doctoral dissertation, Örebro universitet*).

SAE, On-road Automated Vehicle Standards Committee. 2014, SAE J3016: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. *SAE International*.

Segal, A., Haehnel, D. and Thrun, S., 2009, June. Generalized-ICP. *In Robotics: science and systems* (Vol. 2, No. 4).

Thrun, S., Burgard, W. and Fox, D., 2005. Probabilistic robotics. *MIT press*.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G. and Lau, K., 2006. Stanley: The robot that won the DARPA Grand Challenge. *Journal of field Robotics*, 23(9):661-692.

Wolcott, R.W. and Eustice, R.M., 2014, September. Visual localization within lidar maps for automated urban driving. *In Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference* on (pp. 176-183). IEEE.

Wolcott, R.W. and Eustice, R.M., 2015, Fast LIDAR localization using multiresolution Gaussian mixture maps. *In Robotics and Automation (ICRA), 2015 IEEE International Conference* on (pp. 2814-2821). IEEE