Regressing Autonomous Guided Vehicle Localization from Non-Visual Sensor Data

Bruno C. da Silva
http://www.di.ubi.pt/
Luís A. Alexandre
http://www.di.ubi.pt/~lfbaa/

NOVA LINCS and Departamento de Informática
Universidade da Beira Interior
6201-001 Covilhã, Portugal

Abstract

To navigate efficiently, a robot needs to have effective strategies regarding its navigation stages: perception, mapping, localization and path planning. In the localization aspect, a robot estimates its current location in an environment. The more precise this estimation is, the more accurate will be the map of the environment and the robot's ability to create a more precise trajectory of the path. In this paper we study different approaches to obtain an estimate of an autonomous guided vehicle localization, built from non-visual sensor data. We compare results from different regressions methods, namely ridge, lasso, elastic net and support vector regression, with data from individual sensors and two standard fusion approaches, Adaptive Monte Carlo Localization and the Extended Kalman Filter. We concluded that the elastic net regression is a viable method for fusion information from multiple sources (sensors and prediction algorithms) to improve the localization accuracy.

1 Introduction

Localization is a task which involves the robot to use its sensors to retrieve data from the environment and estimate its position. The measurements of its sensors are fundamental to help robots to perceive its surroundings and thus, perform the localization task.

Besides the level of sensor technology, the noise in these sensors must be taken into account to understand the difference between their measurement and the real world. Sensors such as inertial measurement units (IMU) and odometry can accumulate drift errors over time [2], and global positioning system (GPS) can suffer from signal propagation errors, dilution of precision and delays provided by earth layers [4]. Hence, the literature shows that the accuracy of the robot's position in an environment can be improved by integrating different sensor information.

This strategy has been adopted by several works throughout the years. Recently, Sarker et. al. [3] introduced a Bayesian filtering based data reconstruction scheme to increase the reliability of autonomous navigation of mobile robots. The authors transform the prediction step and propose an Imputation step of Extended and Cubature Kalman Filter models to work toward missing data estimation. In their evaluation, they compare the performance of the two Kalman-based methods using a baseline model data stream that uses only the unfiltered sensor data. They used localization coordinates calculated using data from a LIDAR, GPS module and orientation information provided by a gyroscope. Their method worked well for estimating unfiltered and corrupted data. In the work of [1], the authors propose a lightweight algorithm which creates a virtual-IMU to store data from multiple IMU sensors. Their method fuses these information with exteroceptive sensors, achieving better localization accuracy compared to methods that fuses sensors with a single IMU and can be integrated with filter-based algorithms, as well as optimization-based filter algorithms. The simulated tests are performed using nine IMUs and a monocular camera, using poses and sensor measurement based on real-world data and the results show that the localization error can be improved. For the real-world tests, the authors recorded poses from a GPS-RTK module and used them as ground truth data creating three indoor and three outdoor datasets. Their tests regarding localization precision and improvement of the inertial odometry algorithm were successful.

Adaptive Monte Carlo Localization (AMCL) and Extended Kalman filter (EKF) are considered standard approaches in fusing sensors to achieve robot's localization. Although sensor-fusing techniques are popular for robot localization, distinct scenarios require the use of distinct approaches. In this paper we analyze different regression approaches, namely ridge, lasso, elastic net and support vector regression (SVR) to understand if these methods can be used to improve the robot's localization accuracy by fusing data from sensors compared with AMCL and EKF.

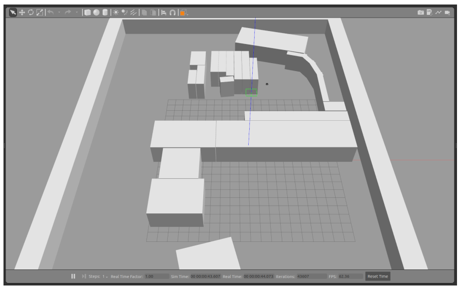

Figure 1: Simulated environment of a part of the Stellantis Factory in Mangualde, Portugal. This section represents the trajectory of where the AGV will navigate.

2 Our proposal

2.1 Simulated Environment

The evaluation was executed inside a simulated environment of the Stellantis factory in Mangualde, Portugal. The environment was created using Gazebo sim and ROS. Accordingly to the factory management, one way length of the trajectory is roughly 370m. With a speed around 0.2 m/s, it means that the AGV will take around 60 minutes do complete the path. The simulation was build in a scale of 10 times smaller.

An autonomous guided vehicle (AGV) inside a factory is used to transport loads without an onboard driver and its navigation aspects are software-sensor defined. For this work, the simulated autonomous capabilities of the AGV were configured using ROS packages such as the navigation package, which includes parameters to configure the move_base node and the Dynamic Window Approach algorithm to allow the robot to navigate autonomously. Since the data from the sensors and algorithms are in Cartesian format, the GPS latitude and longitude values were converted using ROS navsat_transform package.

The ground truth values were established by adopting the almost perfect odometry data from the Gazebo plugin. We compare values from localization-related sensors and fusion algorithms to the ground truth value and analyze the precision of each component. There are different sensors that can be used to measure data related to the robot position, the ones used for this work were divided in two parts: individual measurements and fusion-based measurements. For the individual measures, an odometry sensor and a GPS sensor were used. For the fusion-based measures, the AMCL algorithm was configured with odometry and laser sensors, and the EKF was configured with an IMU, odometry, and GPS sensors.

2.2 Data Extraction

The extraction of data from the sensors was done by creating a method to retrieve 30 Cartesian points per second from ROS topics. The Gazebo odometry, which has perfect location information, was used as our ground truth, an odometry configured with noise to simulate the output of real odometry sensors, a GPS sensor, the AMCL and EKF algorithms.

2.3 Fusing data with Regression Methods

After extracting data from the odometry with noise, GPS, AMCL and EKF we fused it using several regression methods: ridge, lasso, elastic net and support vector regression. The main goal is to investigate if these regression approaches could achieve more accurate results than the sensors and algorithms individually, therefore improving the robot localization.

First, we calculate the euclidean distance between the locations given by the ground truth and all the sensors and algorithms, and then we calculate the mean and the standard deviation of these results, which appear in the second columns of Table 1.

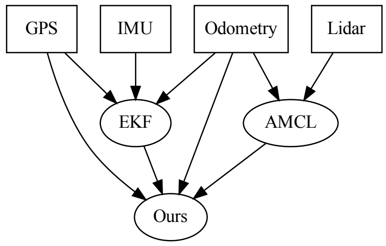

Figure 2: We propose to fuse localization information from two sensors (noisy odometry and GPS) and two prediction methods (AMCL and EKF). Rectangles represent sensors and ovals represent prediction methods. 'Ours' represents the regression approaches evaluated in this work.

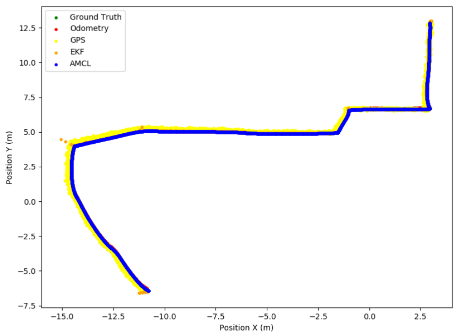

Figure 3: Comparison between the data from sensors/algorithms and ground truth values throughout the path navigated by the AGV.

With the extracted data, we created a data set containing 4989 localization points from the odometry (noisy and noise-free that is used as ground truth), GPS, AMCL and EKF. Figure 3 contains a visual comparison between the localization provided by the sensors, the algorithms and the ground truth values throughout the path in the simulated environment.

With these points, we performed tests with the regression models using the first 60% of the data for the training set, the following 10% for validation and remaining 30% for test. We did not shuffle the data such that no nearby points from one set appear in any of the others. As the regression models require a penalty term (C for SVR and λ for the others) to reduce bias and overcome overfitting, we needed to find the best value for these terms. We created a function where we tested a range of values for C and λ from the following set, $\{0.01, \ldots, 1\}$, to see which one would give us the best score for each regression model. This parameter optimization was performed using only the training and validation data. After that, we tested the the models using the test data set containing the last 30% of the data extracted, with the following configurations: ridge, λ=0.2, *max_iter=None*, *tol*=0.001; lasso, λ=0.01, *max_iter*=1000000, *tol*=0.001; elastic net, λ=0.01, *max_iter*=10000, *tol*=0.001; and support vector regression, C=0.7, *max_iter*=1000.

The values for C and λ indicate the regularization penalty to improve the model estimate capacity by reducing its variance. The *max_iter* values are the number of iterations of the solver in the algorithm. Finally, the *tol* value is the precision of the solution, where a tolerance criteria for stoppage is established. The results of the optimization process are shown in Fig. 4. We can see that SVR is only affected by the C term for values below 0.07 and ridge regression is not affected by the particular value of λ on the tested range, while for the other two approaches, the λ that maximized the score was chosen. The values of *max_iter* were obtained in a similar manner starting from 1 and increasing tenfold until convergence was obtained. Finally, we used the default value for *tol*.

In Table 1, one can visualize the Root Mean Squared Error of the x (RMSE_x) and y (RMSE_y) coordinates and the Mean Absolute Error of x (MAE_x) and y (MAE_y) coordinates produced by the two sensors, the two estimation algorithms and the regression methods, on the test data.

The results show that fusing data from GPS, Odometry, EKF and AMCL using regression methods, can generate a more precise localization than when only using the original four localization sources separately. We also tested fusing only sensor data and only algorithm data to determine if the accuracy of fusing raw and transformed sets of data separately could surpass the proposed model. Despite the fusion between AMCL

Table 1: Values for the mean and standard deviation of the euclidean distance between ground truth points and the sensors and algorithms points, root mean square error of x and y points, mean absolute error of x and y coordinates of the extracted data from the AGV's sensors and algorithms evaluated on the test data set.

	Mean (Stddev)	RMSE_x	RMSE_y	MAE_x	MAE_y
AMCL	0.235 (0.007)	0.094	0.216	0.092	0.216
Odometry	0.058 (0.006)	0.046	0.036	0.046	0.035
GPS	0.294 (0.051)	0.213	0.210	0.203	0.202
EKF	0.423 (0.290)	0.320	0.401	0.233	0.346
Ridge (all sources)	0.045 (0.008)	0.028	0.037	0.027	0.034
Ridge (GPS+Odom)	0.051 (0.008)	0.037	0.036	0.037	0.034
Ridge (AMCL+EKF)	0.076 (0.052)	0.022	0.089	**0.018**	0.071
Lasso (all sources)	0.052 (0.027)	**0.021**	0.054	**0.018**	0.045
Lasso (GPS+Odom)	0.066 (0.025)	0.061	0.035	0.055	0.033
Lasso (AMCL+EKF)	0.079 (0.051)	**0.021**	0.092	**0.018**	0.074
ElasticNet (all sources)	**0.043** (0.017)	**0.021**	0.041	**0.018**	0.037
ElasticNet (GPS+Odom)	0.065 (0.025)	0.062	0.032	0.055	0.029
ElasticNet (AMCL+EKF)	0.079 (0.051)	**0.021**	0.092	**0.018**	0.074
SVR (all sources)	0.047 (0.023)	0.045	**0.026**	0.040	**0.020**
SVR (GPS+Odom)	0.067 (0.023)	0.036	0.061	0.029	0.056
SVR (AMCL+EKF)	0.135 (0.024)	0.117	0.073	0.116	0.061

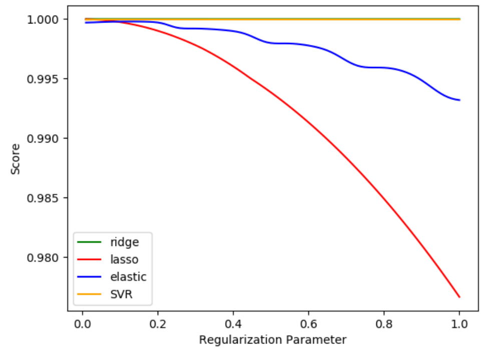

Figure 4: Scores while optimizing λ and C on validation data.

and EKF data presenting quite accurate localization values, when fusing all the sources, elastic net presented the best overall results, beating all other methods with smaller mean distance error and obtaining good results on the corresponding RMSE and MAE metrics w.r.t the X axis.

3 Conclusions

In this paper we explored the use of regression approaches to improve the localization accuracy of a simulated AGV. Our experiments showed that the elastic net regression method can be used as fusion method that can improve the localization quality of an AGV. This is somewhat expected as elastic net is useful for problems with multiple correlated features, which is the case here. Future work will explore the use of neural-based approaches to this problem.

Acknowledgments

This work was supported by NOVA LINCS (UIDB/04516/2020) with the financial support of FCT- Fundação para a Ciência e a Tecnologia, through national funds, and partially supported by project 026653 (POCI-01- 0247-FEDER-026653) INDTECH 4.0 – New technologies for smart manufacturing, cofinanced by the Portugal 2020 Program (PT 2020), Compete 2020 Program and the European Union through the European Regional Development Fund (ERDF).

References

[1] Mingyang Li, Ming Zhang, Yiming Chen, and Xiangyu Xu. A lightweight and accurate localization algorithm using multiple inertial measurement units. *IEEE Robotics and Automation Letters*, 2020.

[2] Prabin Kumar Panigrahi and Sukant Kishoro Bisoy. Localization strategies for autonomous mobile robots: A review. *Journal of King Saud University - Computer and Information Sciences*, 2021.

[3] Victor Kathan Sarker, Prateeti Mukherjee, and Tomi Westerlund. Enhanced reliability of mobile robots with sensor data estimation at edge. *IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT)*, 2020.

[4] Junjie Shen, Jun Won, Zeyuan Chen, and Qi Alfred Chen. Drift with devil: Security of multi-sensor fusion based localization in high-level autonomous driving under gps spoofing. *USENIX Symposium*, 2020.