Project supported by:

EasyLid

Using Machine Learning to detect corrupted data coming from a LiDAR on an autonomous car

Team

Camille Puech

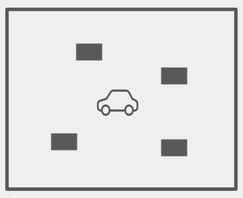
Antoine Mureddu

Anass Sadik

Valentine Bellet

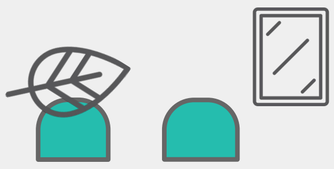
Gabriel Nussli

Objectives



Make the car moves autonomously in an indoor environment

Design a structured and reusable database



Detect anomaly from the LiDAR

Notify operator in real time

Accomplishments

Autonomous behavior of the car

- 6 ultrasonic sensors detect obstacles
- Obstacles: walls or objects larger than 50cmx50cm
- Indoor environment: empty room with 4 walls

No obstacle :
→ move straight
Front obstacle < 1m :
→ turn right
Front < 1m && Right < 70cm obstacle :
→ turn left
Obstacle < 30 cm :
→ emergency stop

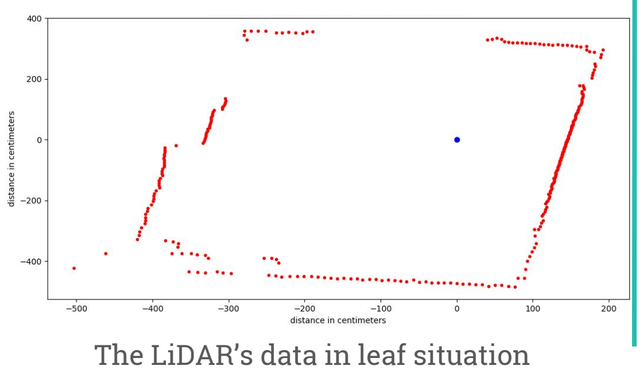
Overview of the algorithm

LiDAR anomaly detection

- Using Convolutional Neural Network Machine Learning to detect and classify anomalies: a leaf on the LiDAR or a reflective glass.
- Collecting more than 40,000 measurements for training purposes
- Installing Machine Learning algorithm directly on the Raspberry Pi
- Libraries: Front-end Keras, Back-end TensorFlow

Leaf situation

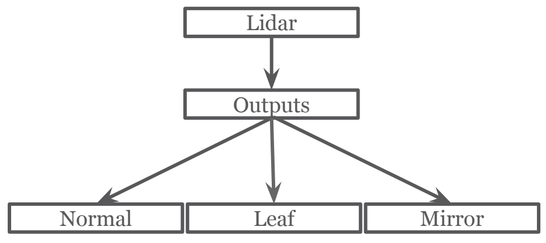
Mirror situation



The LiDAR's data in leaf situation

Database

- The LiDAR takes 360 points with each revolution, saved as one measurement.
- Points: distance in millimeters from LiDAR sensor to its surroundings
- 50 measurements stored in one file named yyyy mm dd hh mm ss
- Distribution in 3 folders: Normal, Leaf and Mirror



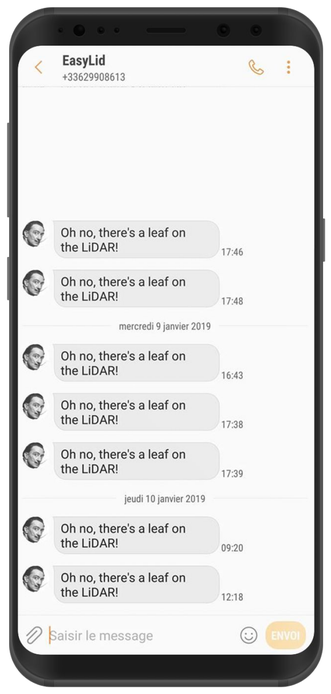
Classification system

2790.25, 2849.0, 2881.25, 2915.5, 2935.5, 2979.75, 3010.75, 3041.75, 3120.0, 3163.25, 3207.5, 3260.75, 3301.0, 3344.5, 3404.0, 3521.5, 3563.25, 3621.75, 3680.0, 3760.5, 3843.5, 3971.5, 4082.0, 4141.25, 4260.0, 4389.75, 4316.5, 4269.0, 4176.0, 4130.25, 4112.0, 3913.75, 3877.75, 3896.25, 3969.25, 3924.75, 3902.75, 3883.25, 3869.0, 3838.25, 3826.75, 3780.0, 3757.5, 3735.25, 3718.5, 3704.5, 3682.0, 3658.75, 3651.25, 3635.0, 3636.0, 3627.25, 3619.75, 3620.75, 3610.5, 3600.25, 3599.25, 3600.25, 3598.25, 3595.5, 3588.0, 3597.5, 3600.25, 3596.5, 3604.0, 3611.25, 3623.5, 3619.75, 3631.25, 3643.5, 3657.0, 3650.25, 3676.25, 3702.5, 3723.5, 3746.5, 3760.5, 3794.25, 3829.75, 3843.5, 3872.25, 3909.25, 3921.25, 3941.25, 3988.5, 4010.5, 4025.5, 3958.0, 3834.0, 3635.0, 3554.25, 3453.75, 3373.25, 3287.25, 3228.75, 3154.0, 3022.5, 2963.75, 2905.5, 2852.25, 2797.75, 2749.75, 2706.75, 2656.75, 2578.25, 2534.25, 2508.25, 2462.5, 2437.25, 2403.75, 2364.25, 2318.0, 2287.75, 2261.75, 2234.0, 2208.25, 2193.5, 2169.25, 2129.0, 2108.5, 2100.0, 2074.25, 2054.75, 2043.5, 2030.75, 1996.0, 1988.0, 1975.75, 1964.75, 1958.5, 1945.0, 1936.75, 1927.75, 1912.25, 1898.0, 1899.0, 1882.5, 1880.0, 1876.5, 1872.25, 1864.25, 1852.5, 1851.0, 1850.75, 1847.25, 1842.25, 1847.25, 1841.25, 1841.75, 1840.75, 1840.5, 1848.0, 1849.0, 1852.0, 1852.25, 1856.25, 1866.25, 1872.75, 1877.5, 1878.5, 1881.25, 1893.5, 1904.25, 1911.25, 1920.5, 1927.25, 1942.0, 1950.25, 1954.25, 1974.5, 1994.75, 2010.0, 2018.5, 2030.75, 2053.75, 2060.0, 2079.5, 2113.75, 2144.0, 2154.25, 2179.25, 2195.5, 2243.25, 2270.75, 2289.5, 2319.5, 2344.75, 2366.0, 2439.25, 2465.0, 2509.75, 2534.0, 2577.75, 2624.75, 2653.75, 2738.0, 2787.5, 2842.0, 2892.5, 2953.75, 3010.25, 3131.75, 3191.0, 3273.5, 3339.75, 3418.0, 3517.25, 3597.5, 3799.5, 3924.75, 3938.0, 4166.0, 4281.0, 4109.5, 4104.75, 4123.0, 4152.25, 4164.75, 4156.0, 4179.75, 4178.5, 4211.25, 4295.25, 4298.0, 4262.5, 4256.0, 4244.5, 4224.0, 4212.5, 4210.0, 4179.75, 4174.75, 4173.5, 4172.25, 4179.75, 4146.25, 4142.5, 4166.0, 4257.5, 4243.25, 4251.0, 4254.75

20181221134159 : 21/12/2018 13:41:59

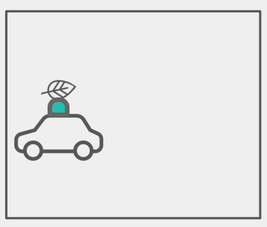
SMS operator notification

- SMS sent to an operator when anomalies are detected
- SMS module connected to the Raspberry Pi
- Iwow SMS module with Bouygues Sim Card



Results

No wall or obstacle collision (95% accuracy)



Stop in less than 1 sec when an unusual situation is detected

SMS received in less than 5 min

- 100 *epochs*
- Training time: 7min
- Training accuracy: 99.95%
- Prediction time: 50ms
- Testing accuracy (in a perfect environment) : 94%

Final video of our project:

Improvements

More efficient autonomous behavior in an outdoor environment

Big-Data architecture for measurement storage

New situations : fog, sun glare, rain and dust

Redundancy of real time notification

GitHub

5th year SIEC projects

EasyLid website