Machine Learning Engineer Nanodegree

Capstone Proposal

Hoang Phuoc Le

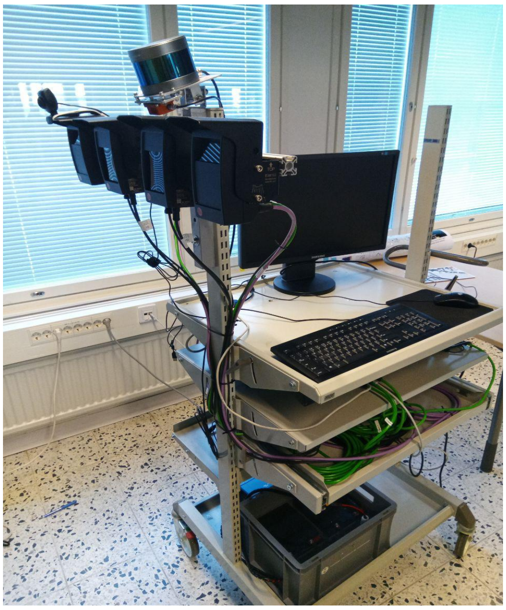
July 4th, 2020

Proposal: Detecting floor surface for an autonomous Robot Indoor Navigation

Domain Background

Autonomous robots, just like humans, should have the ability to make their own decisions and then act accordingly. An autonomous robot is one that can perceive its environment, make decisions based on what it perceives, and then actuate a movement. For an autonomous robot to navigate a task or go on a journey, it will need the input of its surrounding environment. One of the traditional problems in robotic navigation is to recognize the surface it passed by.

The project is inspired by the Kaggle CareerCon 2019 - Help Navigate Robots. There are several other papers [1], [2], which you can find in the reference list.



Problem Statement

In this project, a robot is arranged to travel through different floor surfaces. It is attached with an Inertial Measurement Units (IMU sensors) for data collection. The aim of our task is to build a classification model and predict the type of surface based on the new input sensor data. The output will be uploaded to Kaggle for the scoring feedback.

Datasets and Inputs

The dataset locates in the Kaggle platform, which has been collected by researchers from Tampere University, Finland.

The IMU sensor data is the combination of accelerometer data, gyroscope data (angular rate) and internally estimated orientation. In particular:

```
- Orientation: encode the current angles how the robot is oriented as a
quaternion, including 4 attitude quaternion channels, 3 for vector part and
one for scalar part;
- Angular rate: describes the angle and speed of motion, including 3
channels, corresponding to the 3 orthogonal IMU coordinate axes X, Y, and
Z;
- Acceleration: describe how the speed is changing at different times,
including 3 channels, specific force corresponding to 3 orthogonal IMU
coordinate axes X, Y, and Z.
```

The input data, covering 10 sensor channels and 128 measurements per time series plus three ID columns:

```
- row_id: The ID for this row.
- series_id: ID number for the measurement series. Foreign key to
y_train/sample_submission.
- measurement_number: Measurement number within the series.
```

Solution Statement

We plan to build several methods and compared with each other to select the best approach for the problem. One of the traditional approach is to apply feature engineering and create a set of features (statistics and Fast Fourier Transform) for each series (10 sensor channels and 128 measurements). Afterwards, we can apply algorithms like Random Forest or Gradient Boosting for building a classification model.

Another approach will be to apply Recurrent Neural Network with LSTM for timeseries data or Conv-1d to train the raw data and build a deep learning model for the task.

Benchmark Model

The paper of F. Lomio et al.(2019)[3] gives us the baseline, which achieves an accuracy of over 68% with the nine-category dataset. The high result of Kaggle competition output (more than 95%) was due to the data leak of Orientation input, which will not be feasible in practice. Therefore we will only utilize the right features to build the model and compare with the formal benchmark.

Evaluation Metrics

The Multiclass Accuracy, which is the average number of observations with the correct label for all the categories, will be used as the evaluation metrics. However, we will also compute the satisficing metric of Area Under the ROC Curve (AUC) to further understand the problem.

Project Design

Machine learning project is quite highly iterative in general; as we progress through the ML lifecycle, we'll need to iterate on a section until reaching a satisfactory level of performance, then proceeding forward to the next task (which may be circling back to an even earlier step). However, we will approach the project using the pipeline as follows:

```
- Gathering data (Kaggle platform)
- Data pre-processing (cleaning and featuring engineering)
- Researching the model that will be best for the type of data (Random
Forest, Boosting or Deep Learning models)
- Training and testing the model
- Evaluation
```

To improve the model we might also tune the hyper-parameters of the model and maximize the accuracy.

References:

[1] Lauro Ojeda, Johann Borenstein, Gary Witus, and Robert Karlsen. Terrain characterization and classification with a mobile robot. Journal of Field Robotics, 23(2):103–122, 2006

[2] Terrain traversability analysis methods for unmanned ground vehicles: A survey P Papadakis - Engineering Applications of Artificial Intelligence, 2013 - Elsevier

[3] Surface Type Classification for Autonomous Robot Indoor Navigation Francesco Lomio, et al. - arXiv, 2019 - https://arxiv.org/abs/1905.00252

[4] Kaggle - CareerCon 2019 - Help Navigate Robots https://www.kaggle.com/c/career-con-2019/