TRAFFIC SENSORS ALGORITHMS PROJECT LOG

Fangyu Wu

August 2, 2015

Abstract

This is a short summary of algorithms used, or to be used for vehicle detection and speed estimation on the experimental traffic sensors pack. I would like to keep the log updated as the project moved forward. **So far**, I have implemented the overall architecture of the sensors pack software, and one vehicle detection algorithm. **Next**, I plan to complete implementing existing vehicle detection algorithms, and one speed estimation algorithm by mid of August.

1 Vehicle Detection Algorithms

1.1 Adaptive Thresholding Model

This model is intended for processing the ultrasonic sensor readings, due to its simplicity and verified effectiveness. The model is adaptive in the sense that it can reset its threshold upon exception. It is therefore a series of static thresholding models discretely combined in time domain. The algorithm is implemented as follows:

1. Accumulate a certain period of sensors readings into a list.
2. Use the data to find distinct states of the traffic, i.e., non-occupied, first-lane occupied, second-lane occupied, etc.
3. Determine the static upper bound and lower bound for each traffic states.
4. Make prediction based on new observation.
5. Evaluate the prediction outcomes. If they are reasonable, then output to I/O devices and loop back to step 4. If they are not valid in any case, then rebuild the model based on current observations based on steps 1, 2, 3.

1.2 Gaussian Mixture Model

This model is intended for processing PIR sensors readings, due to the nature of PIR signal being noisy and less deterministic. The major steps of building such a model are as follows:

1. Accumulate a certain period of sensors readings into a list.
2. Use the data to find distinct states of the traffic, i.e., ambient environment, first-lane vehicle passing, second-lane vehicle passing, etc.
3. Build statistic models that reflect the number of distinct states of the traffic.
4. Update the model and make prediction based on new observation.
5. Evaluate the prediction outcomes. If they are reasonable, then output to I/O devices and loop back to step 4. If they are not valid in any case, then rebuild the model based on current observations based on steps 1, 2, 3.

2 Speed Estimation Algorithms

There is not any formalization of speed estimation algorithms currently, as the priority is currently given to develop reliable vehicle detection algorithms. I am planning to start speed estimation algorithms after August 8th.

3 Selecting and Combining Models

It is important to select the most appropriate models for the current traffic status. Using filter techniques to combine predictions from different sensor data is also recommended to improve the overall system reliability.

4 Architecture

The software architecture notes should go here.