CarND-Path-Planning-Project

Self-Driving Car Engineer Nanodegree Program

Prediction:

Prediction uses to sensor fusion data from the simulator, which provides car's localization

```
double car_x = j[1]["x"];
double car_y = j[1]["y"];
double car_s = j[1]["s"];
double car_d = j[1]["d"];
double car_yaw = j[1]["yaw"];
```

Using velocity information, s, distance formula and speed projected the car x value. And from this reference point predicted

1. If there is a car in front of our car which is close to maintain current speed
2. If there is a car either on left or right lane which is not safe to change lane

Above explanation covered in source code from Line 266 – 317

Behavior planning:

Here we use Predicted information and decide

1. Increase Acceleration smoothly +5 m/s in case no cars ahead of our car to make sure no jerk
2. If there is car in ahead of our car which is too close deaccelerate smoothly, -5m/s
3. Change either to left or right lane if is safer to do so
4. Change back to center line if is it is safer to do so

This is covered in source code from Line 319 - 349

Trajectory Generation.

Goal here is to define a path made up of(x,y) points that the car will visit sequentially every .02 seconds. And this achieved in following steps

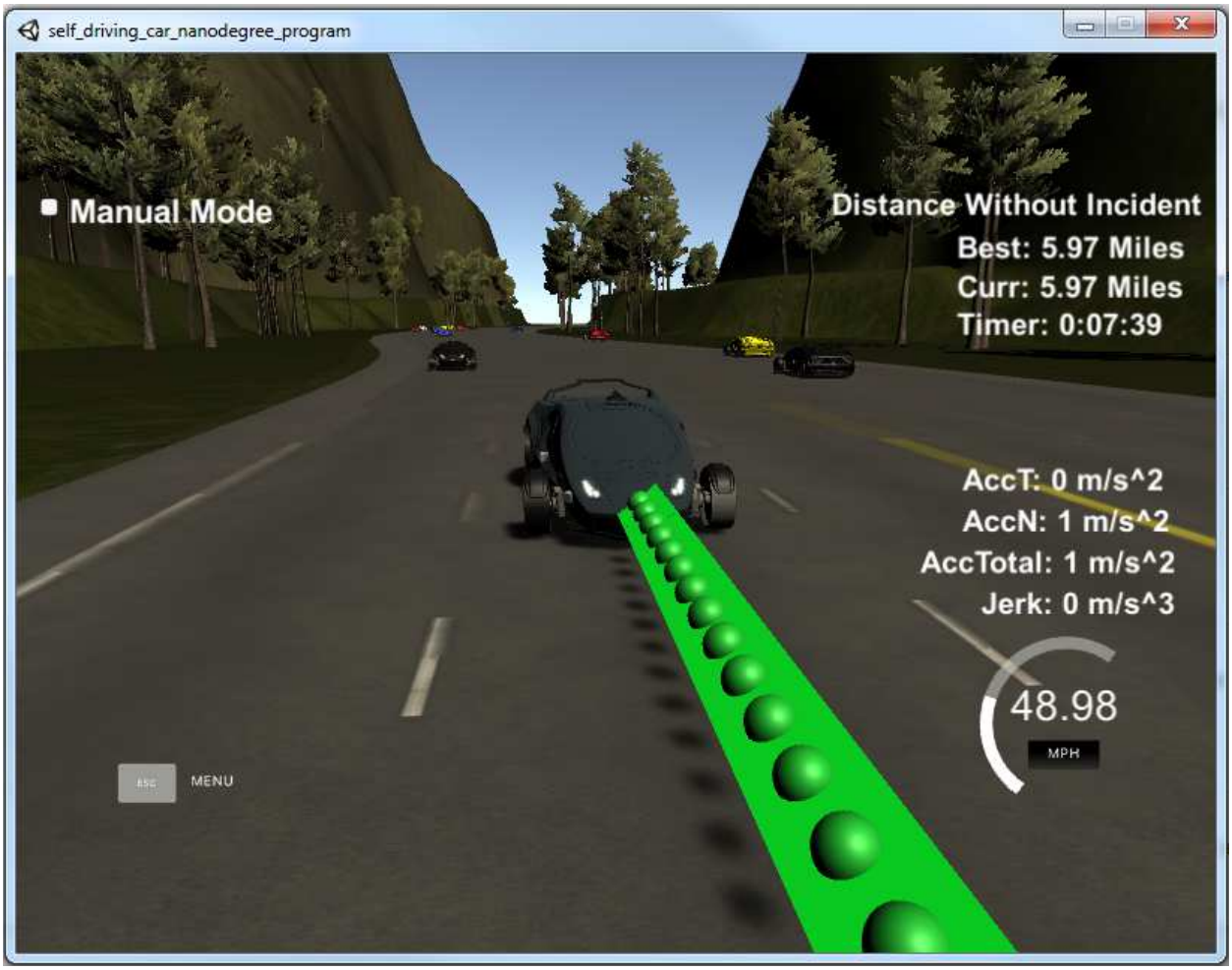
1. Use the previous path end points or car reference as a starting point where the car is, in case previous path size with points is almost zero.
   Which makes path tangent to the previous path's end or tangent to the car direction
2. Use Frenet coordinates and Add three 30m space points ahead of the starting reference
   Note: here used current lane to have a smooth lane change transition
3. Add 5 points to the points list
4. Transform from global to local car coordinates to simplify mathematics

5. Create spline using spline header file and set previously added points to the spline
6. Now start creating Horizon path points, we use horizon length as 30 meters and get corresponding x and y value from the spline and calculate distance to the target
   By using formula Sqrt(x2+y2)
7. In above step we calculated how we could break up spline points so the we can travel at desired reference velocity
8. We need 50 points, so fill up the rest of the points from the spline
   Note, it is 50 – previous path points used already and points are divided based on spline is divided at N from x point
9. Convert back from local to global coordinates system
10. And Finally Pass the generated points to the simulation

Above steps covered by source code from Line 355 - 519

Result

- The car is able to drive at least 4.32 miles without incident



- The car able to change lane when necessary to pass the vehicle and to come back to the middle lane

It also fulfills other requirements of the project

- The car drive according to below speed limit and doesn't exceed the 50MPH
- The Car does not exceed the total acceleration of 10 10 m/s^2 and a jerk of 10 m/s^3.
- Car does not come into contact with any other cars on the road
- The car does not spend more than a 3-second length outside the lane lanes during changing lanes, and every other time the car stays inside one of the 3 lanes on the right hand side of the road.