ADAPT

Autonomous **D**riving for **A**dverse **P**erceived **T**errain

Project Test Plan

Team	Shasa Antao
	Bryson Jones
	Shaun Liu
	Evan Schindewolf
	Wesley Wang
Sponsors	Dr. Dimitrios Apostolopoulos
	Dr. John Dolan
Advisor	Dr. David Held

Team B

Carnegie Mellon University

Master of Science in Robotic Systems Development
Robotics Institute
School of Computer Science
Carnegie Mellon University

September 23, 2020

THE ROBOTICS INSTITUTE

Contents

1 Introduction

The autonomous vehicle industry, specifically with respect to passenger vehicle and trucking and freight applications, has quickly become a large focus in technology development. However, in general, most autonomy systems lack the ability to transition from operating in ideal road or terrain conditions to areas of wet or icy road conditions. Autonomous vehicles that enter these conditions could lose control and endanger passengers, nearby pedestrians, and vehicles.

Research suggests that thousands of people are killed and hundreds of thousands are injured every year due to weather related crashes, and up to $3.5 billion is lost every year by the trucking and freight industry during delays due to poor environmental conditions. Even in the event that a high level of autonomy is achieved for autonomous vehicles in dry and sunny conditions, it is unlikely to translate to any reduction in these numbers, since people will always need to travel and transport goods, no matter the road conditions.

The ADAPT (Autonomous Driving for Adverse Perceived Terrain) system is designed to combat this issue. The system serves to actively detect changes in road conditions and adjust the vehicle controls to account for these changes, making transitions into wet or otherwise unfavorable road conditions much safer. Current systems are almost exclusively reactive, meaning that they only modify the vehicle's actions after an incident has occurred, whereas the ADAPT system aims to predict future scenarios and maximize the ability to safely transport passengers and goods.

The ADAPT project test plan provides a detailed approach to the various tests that are going to be conducted during the Fall Validation Demonstration. These tests will highlight the various functionalities of the intended subsystems based on the updated performance and non-functional requirements listed in Appendix A. In addition, the plan has information on the validation criteria for each test based on which the test can be deemed successful or not after the demonstration.

2 Logistics

2.1 Location

Due to requiring a fair amount of space for testing, all tests that involve autonomous driving by the physical robot or are environment specific (as well as FVD) will be performed on the East Campus Garage. At this time we expect to be on the top level, but will survey the different levels at a later date. In the event that we cannot gain access to the East Campus Garage, our backup location is the bike track at Highland Park.

2.2 Personnel

Since Bryson will not be in Pittsburgh during this semester, he will not be able to physically be present at tests or FVD. He has been listed under 'Personnel' for some tests and for FVD as his work is relevant to those demonstrations and he will be available to answer questions live for them as well. It is not yet determined, but Bryson may travel back to Pittsburgh at some point this semester. In the event that he cannot complete quarantine before a test or demonstration we will make sure to follow social distancing practices.

2.3 Equipment

Some basic equipment is necessary for running outdoor testing as listed below.

- Vehicle
- Laptop
- Folding Table
- Portable Power Station
- Monitor, Keyboard, and Mouse

3 Schedule

Table 1: Milestones

Event	Milestone	Test/Demonstration
PR8 9/30/20	Window and lid fabricated and integrated	Test 1
	PDB fully integrated	Test 9
	ROS fully setup and tested	Test 8
	FLIR and ZED (with filters) physically integrated	Test 5
	Encoders calibrated	Test 1
	Baseline planning (Gen 2) operable with simulation in python – Given a set of waypoints offline generate a trajectory – Show the trajectory being followed in existing simulation	Test 2
PR9 10/14/20	Baseline Map Synthesis	Test 7
	Software-in-the-loop simulation setup	Test 11
	More representative perception data annotated	Test 6
	Baseline puddle segmentation functioning in real-time	Test 6
	Perception incorporates ZED with filters –Basic image segmentation using classical methods	Test 5
	Advanced data streaming available on GUI –Camera stream and segmented network feed both available –Path tracking feed performance feed –Plotting sensor/vehicle states available	Test 8
	Localization pipeline works within reasonable tolerances	Test 3
PR10 10/28/20	Motion planning pipeline operable on-board robot	Test 4
	Ability to track paths tracking with closed loop control in real world	Test 4
	Baseline puddle tracking and localization	Test 7
PR11 11/11/20	Map Synthesis integrated with Perception and Planning	Test 10
	Terrain Comprehension complete	FVD
FVD 11/18/20	Model learning tuned	FVD
FVD Encore 11/23/20	Full system testing	FVD

4 Tests

Test 1: Encoders & Hardware

Objective	
Demonstrate that encoders can provide accurate velocity information for both slow and high speed.	
Elements	**Location**
- Vehicle Subsystem - Localization Subsystem	East Campus Garage
Equipment	**Personnel**
- Tape Measure	Shaun Liu Evan Schindewolf
Procedure	
1. Move the car in a straight line and measure the distance traveled with tape measure. 2. Calculate the distance traveled using encoder information and compare with actual distance traveled. 3. Repeat 1 and 2 with different speeds.	
Verification Criteria	
- Verify that wheel speed on all four wheels are reasonable. - Verify that the actual distance traveled is the same as the distance calculated from encoder data.	

Test 2: Planning in Simulation

Objective	
Demonstrate that our planning pipeline is able to generate a trajectory that our software can use to traverse between waypoints in simulation	
Elements	**Location**
- Dynamics and Controls Subsystem - Motion Planning Subsystem - Localization Subsystem	- Remote
Equipment	**Personnel**
- MATLAB Simulation	- Bryson Jones - Evan Schindewolf
Procedure	
1. Run motion planning code to pre-generate path based on desired map input 2. Display planned path 3. Start MATLAB main simulation file 4. Show simulation running using planned trajectory 5. Show plots that display performance	
Verification Criteria	
- Show that the path generated is feasible and reasonable given the waypoints - Display that vehicle is able to track path in simulation, and stays within 0.75m of the path - Display graphs of simulated sensor data to show performance	

Test 3: Localization

Objective	
Demonstrate that the robot can determine its position in the world and the position of puddles relative to itself	
Elements	**Location**
- Vehicle Subsystem - Perception Subsystem - Localization Subsystem	- East Campus Garage
Equipment	**Personnel**
- GPS Base Station - Marking Tape - Tape Measure	- Bryson Jones - Shaun Liu - Evan Schindewolf
Procedure	
1. Mark out a start and finish location on the ground a set distance apart. 2. Command the robot to drive from one point to the other. 3. Compare the real world location of the robot to where it has localized itself to. 4. Set out two lines on the ground a set distance apart. 5. Place the robot to the far side of one line and command it to drive at a constant speed. 6. Record the difference in time between the two line crossings.	
Verification Criteria	
- Robot is able to localize itself within a few centimeters of ground truth. - Robot is able to estimate speed within a half meter per second.	

Test 4: Sensors through Path Tracking

Objective	
Demonstrate that the robot can autonomously follow a pre-provided series of waypoints.	
Elements	**Location**
- Vehicle Subsystem - Localization Subsubsystem - Planning Subsystem - Dynamics and Controls Subsystem	- East Campus Garage
Equipment	**Personnel**
- GPS Base Station - Marking Tape - Tape Measure	- Bryson Jones - Shaun Liu - Evan Schindewolf
Procedure	
1. Physically mark out the waypoints on the ground using tape. 2. Place the vehicle at the starting waypoint and turn the system on. 3. Use ROS to send the start command and information. 4. Observe whether the vehicle follows the waypoints within 3/4 of a meter. 5. Measure the displacement off the final waypoint when the vehicle comes to a stop and verify that this is less than the allowable threshold.	
Verification Criteria	
- Vehicle does not start moving until the command is given through the GUI. - Vehicle does not exceed 3/4 of a meter from the series of waypoints. - Final displacement is within the assigned threshold. - No remote control is necessary to achieve the motion.	

Test 5: Classical Segmentation

Objective	
To test the capability of the system to detect puddles using classical segmentation	
Elements	**Location**
- Vehicle Subsystem - Perception Subsystem	- East Campus Garage
Equipment	**Personnel**
- Vehicle Enclosure - Zed Stereo Camera - Jetson Xavier - Jugs of water - Laptop	- Shasa Antao - Shaun Liu - Wesley Wang
Procedure	
1. Pour water on asphalt to create puddles 2. Drive the vehicle around with puddles in view of the Zed stereo camera 3. Verify that the IoU values are above the threshold of the performance requirements	
Verification Criteria	
-Puddle segmentation IoU > 65% for > 60% of frames -Road segmentation IoU > 65% for> 60% of frames	

Test 6: Neural Network Based Segmentation

Objective	
Achieve adequate segmentation of puddles and road from the live camera feed from the FLIR Grasshopper3 camera.	
Elements	**Location**
- Perception Subsystem	- East Campus Garage
Equipment	**Personnel**
- FLIR Grasshopper3 Camera - Jugs of Water	- Shasa Antao - Shaun Liu - Wesley Wang
Procedure	
1. Pour water on asphalt to create puddles 2. Drive the vehicle around with puddles in view of the FLIR 3. Verify that the IoU values are above the threshold of the performance requirements	
Verification Criteria	
- Road segmentation IoU > 65% for > 60% of frames - Wet asphalt segmentation IoU > 65% for > 60% of frames - Puddle segmentation IoU > 65% for > 60% of frames	

Test 7: Online Map Synthesis

Objective	
Test online map synthesis.	
Elements	**Location**
- Planning Subsystem - Perception Subsystem	- East Campus Garage
Equipment	**Personnel**
- GPS Base Station	- Bryson Jones - Evan Schindewolf - Shaun Liu
Procedure	
1. Drive car around environment. 2. Observe map output on GUI.	
Verification Criteria	
- Map resembles environment - Map data derived from the perception subsystem is actionable for the localization subsystem	

Test 8: ROS and Interface Pipeline

Objective	
Demonstrate the capabilities of our ROS and Interface pipelines to store, communicate, and display useful data for both the robot and users	
Elements	**Location**
- ROS Pipeline - Interface Pipeline	- Remote/Video
Equipment	**Personnel**
- Onboard Sensors - Client (Control/User) PC - Server (Onboard) PC	- Bryson Jones - Evan Schindewolf
Procedure	
1. Launch GUI on Control PC 2. From GUI, spin up ROS processes for both the Control PC and the Onboard PC 3. From GUI start all ROS nodes 4. Initiate motion to engage sensor measurements and display outputs on GUI 5. Display ROS rqt graph running with all needed nodes for pipeline	
Verification Criteria	
- Show that ROS connections between Control and Onboard PCs is established - Display sensor and performance data on GUI, running on the control PC - Show that GUI display values successfully update at a reasonable frame rate	

Test 9: Electrical

Objective	
Test the capability and reliability of the Power Distribution Board and Switching Board	
Elements	**Location**
-Vehicle Subsystem	-East Campus Garage
Equipment	**Personnel**
- Power Distribution Board - Switching Board - System Batteries - Onboard Sensors - Onboard PC - Robot Enclosure - Multimeter	- Shasa Antao - Shaun Liu - Wesley Wang
Procedure	
1. System should be in a powered down mode and all wired connections should be checked. 2. Once the power is switched on, voltage should be checked at all test points with a multimeter. 3. After 5 minutes, recheck all test points with the multimeter and record voltage readings again.	
Verification Criteria	

- Sensors and PC are powered and working
- The required voltage is achieved at all the component connectors
  - Voltage at Jetson Xavier - 20V
  - Voltage at USB hub - 12V
  - Voltage at Arduino - 12V
  - Voltage at GPS - 20V
  - Voltage at IR Camera - 20V
  - Voltage at PWM board - 3.3V
  - Voltage at FLIR camera - 20V
- Traces on the board do not melt or deform due to high current
- LEDs glow
- Toggle switch on the Switching Board switches between autonomous and manual modes	

Test 10: Wet Full System Test

Objective	
Test the entirety of the ADAPT system on the physical robot with genuine puddles on the course.	
Elements	**Location**
- Vehicle Subsystem - Terrain Comprehension Subsystem - Planning Subsystem - Dynamics and Controls Subsystem - Interface Subsystem	- East Campus Garage
Equipment	**Personnel**
- GPS Base Station - Marking Tape - Tape Measure - Jugs of Water	- Shasa Antao - Bryson Jones - Shaun Liu - Evan Schindewolf - Wesley Wang
Procedure	
1. Provide the robot with a predetermined set of waypoints. 2. Command the robot to follow the waypoints in series. 3. Observe run and wait for it to complete. 4. Pour water to generate puddles among the waypoints. 5. Reset the robot location and command the robot to follow the same set of waypoints as before. 6. Observe run, specifically the differences between the two runs relative to the puddles.	
Verification Criteria	
- In both runs, the robot stays within 3/4 of a meter from the series of waypoints at all times. - In the first, run the robot slows down for tight corners and the final waypoint. - In the second run, the robot slows down for puddles that span the road. - In the second run, the robot deviates from the path it took in the first run to avoid a puddle.	

Test 11: Software in the Loop

Objective	
To demonstrate the ability to run and test all of the ADAPT onboard software system in a custom built MATLAB simulator.	
Elements	**Location**
- Planning Subsystem - Dynamics and Controls Subsystem - Interface Subsystem - ROS Pipeline	- Remote/Video Recording
Equipment	**Personnel**
- MATLAB Simulator	- Bryson Jones - Evan Schindewolf
Procedure	
1. Launch python main script to initialize onboard software 2. The script will then launch the MATLAB simulator 3. View simulated path generation and tracking in the simulation environment 4. Display plots that show onboard software performance	
Verification Criteria	
- Show that onboard code written in Python/C++ can be run within MATLAB Simulator - Display plots that can be used to debug and test new algorithms	

4.0.1 Fall Validation Demonstration

All demonstrations for Fall Validation Demonstration (FVD) will be conducted at the East Campus Garage. Shown in Tables 2, 3, and 4 are the various FVDs.

Table 2: Fall Validation Demonstration #1

Equipment	Demonstration	Success Criteria
Vehicle Water	1. Splash vehicle with water 2. Show Interior is dry	**M.N.4.** Interior is dry to touch

Table 3: Fall Validation Demonstration #2

Equipment	Demonstration	Success Criteria
Vehicle Remote control Power Laptop Water	1. Give robot cyclic waypoints 2. Run robot 3. Add water to circuit 4. Show segmentation feed 5. Show robot tracking waypoints 6. Show data analysis once run is complete	**M.P.4.** planning horizon > 8m **M.P.5.** real-time motion controls (>= 55 Hz) **M.P.6.** Less than 0.75m away from waypoints at all times **M.N.1.** max achieved speed >= 30kph **M.N.2.** vehicle runs without an external cable **M.N.3.** video stream visible on laptop during run

Table 4: Fall Validation Demonstration #3

Equipment	Demonstration	Success Criteria
Vehicle Remote control Power Laptop Water	1. Wet a patch of asphalt such that a > 1ft diameter puddle and > 1 ft diameter patch of wet asphalt forms 2. Move the camera(s) such that wet and dry road is visible 3. Capture image of the patch using onboard cameras 4. Display semantically segmented version of the image 5. Show segmented video	**M.P.1.** Road segmentation IoU > 65% for > 65% of frames **M.P.2.** Wet asphalt segmentation IoU > 60% for > 60% of frames **M.P.3.** Puddle segmentation IoU > 65% for > 65% of frames

5 Appendix A: Updated Requirements

5.1 Mandatory Performance Requirements

The system will:

M.P.1. Achieve road segmentation IoU greater than 65% for at least 65% of frames.

M.P.2. Achieve wet asphalt segmentation IoU greater than 60% for at least 60% of frames.

M.P.3. Achieve puddle segmentation IoU greater than 65% for at least 65% of frames.

M.P.4. Have planning horizon always greater than the braking distance (8m at 30kph).

M.P.5. Have real-time motion controls (at least 55 Hz).

M.P.6. Stay within 0.75m of waypoints at all times.

There were changes to the performance requirements since the Critical design review. M.P.1., M.P.2., and M.P.3. were dropped by 20 percentage points to better align with accuracy rates for more relevant works. Previously the values had been based on offline methods. The original versions have been added to Desirable Performance Requirements as D.P.4., D.P.5., and D.P.6., respectively. M.P.4. and M.P.7. have been relegated to the Desirable Performance Requirements list as well as items D.P.7. and D.P.8., respectively. This is change as well as the following changes are in recognition of additional time limitations and logistic challenges that have come about due to COVID. M.N.2. and M.N.3. have been relegated to the Desirable Non-functional Requirements and items D.N.6. and D.N.7., respectively.

5.2 Mandatory Non-functional Requirements

The system shall:

M.N.1. Be able to reach speed that is scaled equivalent to typical highway passenger vehicle speeds (30 km/hr).

M.N.2. Use self-contained energy sources.

M.N.3. Stream video and classification feed.

M.N.4. Have IP64 or better Ingress Protection rating.

5.3 Desirable Performance Requirements

The system will:

D.P.1. Achieve wet leaf segmentation IoU greater than 80% for at least 80% of frames.

D.P.2. Achieve pothole segmentation IoU greater than 80% for at least 80% of frames.

D.P.3. Achieve sand segmentation IoU greater than 80% for at least 80% of frames.

D.P.4. Achieve road segmentation IoU greater than 85% for at least 85% of frames.

D.P.5. Achieve wet asphalt segmentation IoU greater than 80% for at least 85% of frames.

D.P.6. Achieve puddle segmentation IoU greater than 85% for at least 85% of frames.

D.P.7. Have just-in-time tire traction prediction.

D.P.8. Have online prediction model improvement.

5.4 Desirable Non-functional Requirements

The system shall:

D.N.1. Be able to function under adverse weather conditions.

D.N.2. Cost less than the average cost of a tractor-trailer incident.

D.N.3. Generate terrain map for user.

D.N.4. Provide warnings of incoming adverse terrain to user.

D.N.5. Use reinforcement learning to improve vehicle dynamics.

D.N.6. Maintain less than 1.47 lateral grms at all times.

D.N.7. Have representative suspension ride frequency and damping characteristics within 25% of typical passenger vehicles.