Vehicle Intersection Control

McMaster University

Verification and Validation

SE 4G06

Group 6

Alex Jackson	-	1302526
Jean Lucas Ferreira	-	1152120
Justin Kapinski	-	1305257
Mathew Hobers	-	1228607
Radhika Sharma	-	1150430
Zachary Bazen	-	1200979

Table of Contents

List of Tables

1 Revisions

Table 1: VIC Table of Revisions

Date	Revision Number	Authors	Comments
April 17, 2017	Revision 1	Alex Jackson Jean Lucas Ferreira Justin Kapinski Mathew Hobers Radhika Sharma Zachary Bazen	- Completed and updated test cases - Updated traceability matrix
April 10, 2017	Revision 1	Alex Jackson Jean Lucas Ferreira Justin Kapinski Mathew Hobers Radhika Sharma Zachary Bazen	- Increased table line width to reduce invisible table lines when zoomed. Printing unaffected. - Updated intersection controller communication
February 27, 2017	Revision 0	Alex Jackson Jean Lucas Ferreira Justin Kapinski Mathew Hobers Radhika Sharma Zachary Bazen	N/A

2 Purpose

The purpose of this document is to examine the previous project goals and requirements, and to see how the end system complies with these requirements. Through validation, it will determined if the project goals were met. Verification will allow for the detection of errors, and build a level of confidence in the system.

This document will include a traceability matrix to map the test cases to the functional requirements.

The intended audience for this document consists of Dr. Alan Wassyng and the course's teaching assistants.

3 Validation

3.1 Project Goals and Functional Validation

The system consists of two main components: Vehicle Controller and Intersection Controller. Each component of the system has its own specific goals. By integrating these component-specific goals with our functional requirements, the entire project goals are realized. Therefore, if we can validate that our system components meet our functional requirements, we can assert that they fulfill our project goals.

Table 2: Vehicle Controller Component Goals

Component Name	**Goals**	**Functional Requirement**
Image Processing	- Detect lanes, obstacles, and the intersection	V2, V3, V5, V6
Vehicle Navigation	- Guide the vehicle movements on the track - Stop the vehicle when necessary	V2, V4, V5, V6
Communication	- Send request messages to the Intersection Controller - Receive response messages from the Intersection Controller	V1
Servo Motor	- Set the desired angle of the wheels	V2, V5, V6
Speed Controller	- Set the desired speed of the vehicle	V2, V4, V5, V6
Ultrasonic Controller	- Detect obstacles in front of the vehicle within a certain distance	V6

Table 3: Intersection Controller Component Goals

Component Name	**Goals**	**Functional Requirement**
Vehicle Detection	- Detect location of cars and obstacles present at the intersection	IC1, IC2
Communication	- Receive request messages from the Vehicle Controller - Send response messages to the Vehicle Controller	IC4, IC5
Intersection Management	- Control the traffic flow of the intersection	IC3, IC6

4 Traceability Matrix

Identifier	Reqs Tested	V1	V2	V3	V4	V5	V6	IC1	IC2	IC3	IC4	IC5	IC6
Test Cases	102	13	19	2	9	9	13	3	5	11	8	6	4
HC1.1	3		X			X	X						
HC1.2	3		X			X	X						
HC1.3	3		X			X	X						
HC2.1	4		X		X	X	X						
HC2.2	4		X		X	X	X						
HC2.3	4		X		X	X	X						
HC3.1	1						X						
HC3.2	1						X						
HC3.3	1						X						
VC1.1	1	X											
VC1.2	1	X											
VC1.3	1	X											
VC1.4	1	X											
VC2.1	1		X										
VC2.2	1		X										
VC2.3	1		X										
VC2.4	2		X	X									
VC2.5	2		X			X							
VC2.6	1		X										
VC2.7	1		X										
VC2.8	1		X										
VC2.9	1		X										
VC2.10	3		X	X	X								
VC3.1	1		X										
VC3.2	1		X										
VC3.3	1		X										
VC3.4	1				X								
VC3.5	1				X								
VC3.6	1					X							
VC3.7	1						X						
VC3.8	1						X						
VC3.9	1						X						
IC1.1	2	X									X		
IC1.2	3	X								X	X		
IC1.3	3	X								X	X		
IC1.4	3	X								X	X		
IC2.1	2	X										X	

Identifier	Reqs Tested	V1	V2	V3	V4	V5	V6	IC1	IC2	IC3	IC4	IC5	IC6
IC2.2	3	X								X		X	
IC2.3	3	X								X		X	
IC2.4	3	X								X		X	
IC2.5	3	X								X		X	
IC3.1	1										X		
IC4.1	1							X					
IC4.2	1							X					
IC4.3	1								X				
IC4.4	1								X				
IC4.5	1								X				
IC4.6	1								X				
VIC.1	2									X			X
VIC.2	5				X		X			X	X		X
VIC.3	4				X					X	X		X
VIC.4	2					X							X
VIC.5	5				X			X	X	X	X	X	

5 Verification

5.1 Hardware Verification

5.1.1 Individual Vehicle Hardware Component Verification

4.1.1.1 Servo Verification

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
HC1.1	V2, V5, V6	Tell the servo to go to a specified angle. The actual angle of the servo will be measured using a protractor and the actual angle needs to be within 3 degrees of the desired angle to be considered a pass.	Set servo to -30 degrees	Servo is between -33 and -27 degrees	-26 degrees	Fail
HC1.2			Set servo to 0 degrees	Servo is between -3 and 3 degrees	3 degrees	Pass
HC1.3			Set servo to 30 degrees	Servo is between 27 and 33 degrees	33 degrees	Pass

4.1.1.2 Speed Controller Verification

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
HC2.1	V2, V4, V5, V6	The speed controller will be tested by having the car start at a specified speed then changing it after 1 second.	Start at 10% of max then stop	Vehicle stops after 1 second	Stop after 1 second	Pass
HC2.2			Set speed to 10% of max then increase to 30%	Speed should increase after 1 second	Speed increases after 1 second	Pass
HC2.3			Set speed 30% of max then decrease to 10%	Speed decreases after 1 second	Speed decreases after 1 second	Pass

4.1.1.3 Ultrasonic Sensor Verification

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
HC3.1	V6	The ultrasonic sensor will be tested by placing obstacles at specific distances from the sensor and then checking what the sensor measures the distance as.	Place obstacle 10 cm from the sensor	Sensor measures between 8 and 12 cm	11 cm	Pass
HC3.2			Place obstacle 15 cm from the sensor	Sensor measures between 13 and 17 cm	16 cm	Pass
HC3.3			Place obstacle 20 cm from the sensor	Sensor measures between 18 and 22 cm	22 cm	Pass

5.2 Software Verification

Any necessary variables for describing the inputs/outputs will be defined prior to the respective verification test table.

5.2.1 Vehicle Software Verification

5.2.1.1 Vehicle Communication Verification

Variables used:

- string: request_msg = "carId_comingFrom_goingTo_listeningPort_timestamp" (example: '1_N_S_3000_1488233082083')
- integer: response_msg $\in [0, 1, 2]$
  Where 0 = Proceed, 1 = Stop until further notice, 2 = stop car and shut off all services.

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
VSC1.1	V1	An intersection has been detected, and a request has **not** been sent to the Intersection Controller	request_msg	All bytes of the message successfully sent over the socket. (ie: 28 bytes sent)	28 bytes delivered	Pass
VSC1.2	V1	An intersection has been detected, a request has already been sent to the Intersection Controller, and another car approaching the intersection	response_msg	response_msg = 1 car does not leave intersection	Car remains stopped at intersection	Pass
VSC1.3	V1	An intersection has been detected, a request has already been sent to Intersection Controller, and there is **no** other car approaching the intersection	response_msg	response_msg = 0 car proceeds through intersection	car begins driving towards the dictated *goingTo* direction	Pass

VSC1.4	V1	An emergency stop signal brodcasted from the Intersection Controller has been received	response_msg	response_msg = 2, car stops and exits the main running thread safely.	Car stops, shutdown all sockets, uninitializes hardware modules, and frees dynamically allocated memory	Pass

5.2.1.2 Image Processing Verification

The following table are verifications tests for the lane detection and lane extraction from a real-world image scenario (VC2.1 to VC2.5). As well as the verification of the image processing logic and identification (VC2.6 to VC2.10).
Variable Definitions:

- imgData (defined by the given struct)

```
struct ImageData {
     double avg_slope;
     double left_line_length;
     double right_line_length;
     double intersection_distance;
     int intersection_detected;
     int intersection_stop;
     int intersection_type;
}
```

Explanation of ImageData through Figure 1.
On the left side, we have a bare image with a yellow grid.
The yellow grid represents the area that the car tries to detect an intersection (ie. the red markers on the ground).
After this step, we binarize the image to get just the lanes of the track onto the image as seen on the image to the right.

White lines on the black image were detected via Canny edge detection algorithm.
Red lines are created through a Probabilistic Hough Transform.
The green lines connects the detected red lines to the center of the image, and allows the creation of two sets of lines: the left side lines and the right side lines.
Using the two sets of green lines, we can calculate the average left angle, average right angle, average left line length, and average right line length.

In addition, the *avg_slope* represents the average angle that all the red lines detected make with a vertical line.
Lastly, the *intersection_type*, there exists two types of intersections, since the track has an awkward parallelogram shape. One with a wide right turn and shallow left turn, the other a shallow right turn and a wide left turn.

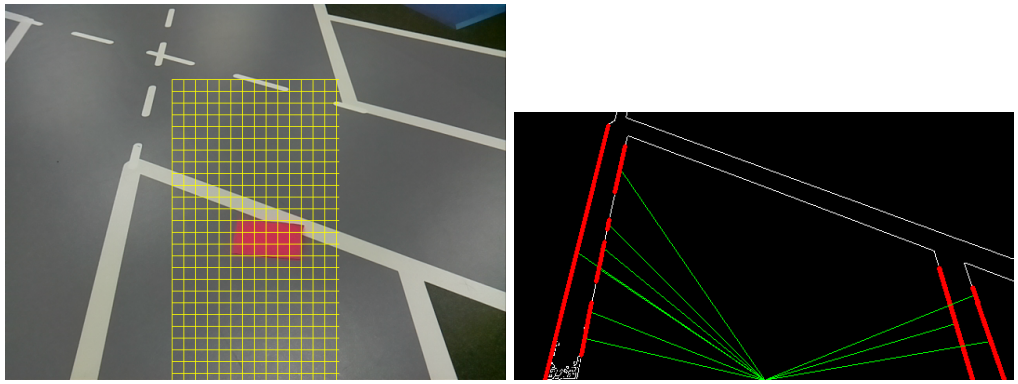

Figure 1: Image Data Explanation

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
VC2.1	V2	At any point of the track, the lanes must be detected and extracted from the image.	Car on straight track segment	A straight line detected on each side of the car	Figure 2	Pass
VC2.2	V2	"	Car on left turn segment	A small left turning curve on the left side of the screen. A large left turning curve on the right of the screen	Figure 3	Pass
VC2.3	V2	"	Car on right turn segment	A small right turning curve on the right side of the screen. A large right turning curve on the left of the screen	Figure 4	Pass
VC2.4	V2, V3	"	Car approaching intersection	Two vertical lines, one on each side of the car, terminating at a horizontal line, and a coloured marker on the ground.	Figure 5	Pass
VC2.5	V2, V5	"	Car navigating through intersection	Small vertical line segments on each side of the car	Figure 6	Pass

VC2.6	V2	The geometry of the captured image must be valid in order to correctly deduce logic and navigate the vehicle	Car on straight track segment and centered on its lane	avg_slope ≈ 0 leftLength $\approx$ rightLength intersectionDetect = 0	avg_slope = 2 leftLength = 358.31 rightLength = 361.45 intersectionDetect = 0	Pass
VC2.7	V2	"	Car on straight track segment and off-centered to the left	avg_slope ≈ 0 leftLength $\ll$ rightLength intersectionDetect = 0	avg_slope = -3 leftLength = 464.34 rightLength = 279.36 intersectionDetect = 0	Pass
VC2.8	V2	"	Car on a left turn and centered on its lane	-45 < avg_slope $\ll 0$ leftLength $\ll$ rightLength intersectionDetect = 0	avg_slope = - 18 leftLength = 335.14 rightLength = 408.33 intersectionDetect = 0	Pass
VC2.9	V2	"	Car on a right turn and centered on its lane	45 > avg_slope $\gg 0$ intersectionDetect = 0	avg_slope = 29 leftLength = 377.55 rightLength = 291.06 intersectionDetect = 0	Pass
VC2.10	V2, V3, V4	"	Car approaching intersection	avg_slope ≈ 0 leftLength $\approx$ rightLength intersectionDetect = 1 intersectionDistance $\gg$ 0 intersectionType = 1	avg_slope = 3 leftLength = 332.12 rightLength = 396.56 intersectionDetect = 1 intersectionDistance = 131 intersectionType = 1	Pass

5.2.1.3 Vehicle Navigation

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
VSC3.1	V2	A straight lane has been detected and the vehicle is travelling at standard operating speed	avg_slope ≈ 0 intersectionDetect = 0 obstacleDetect = 0 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle travels along middle of lane and stays within lane lines	Vehicle travels in a straight line	Pass

VSC3.2	V2	A left turn has been detected and the vehicle is travelling at standard operating speed	-45 $<$ avg_slope $\ll$ 0 intersectionDetect = 0 obstacleDetect = 0 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle follows left turn and stays within lane lines	Car follows curved path	Pass
VSC3.3	V2	A right turn has been detected and the vehicle is travelling at standard operating speed	0 $\gg$ avg_slope $>$ 45 intersectionDetect = 0 obstacleDetect = 0 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle follows right turn and stays within lane lines	Car follows curved path	Pass
VSC3.4	V4	An intersection has been detected and the vehicle is travelling at standard operating speed	avg_slope $\approx$ 0 intersectionDetect = 1 obstacleDetect = 0 distanceToIntersection $\geq$ 200 speed = 0.5 m/s	Vehicle reduces speed to 0.35 m/s	Vehicle slows in anticipation of stopping at intersection	Pass
VSC3.5	V4	Vehicle has detected and approached intersection	avg_slope $\approx$ 0 intersectionDetect = 1 obstacleDetect = 0 distanceToIntersection $<$ 200 0.35 m/s $\leq$ speed $\leq$ 0.5 m/s	Vehicle comes to a complete stop before intersection line	Vehicle stops before crossing intersection line	Pass
VSC3.6	V5	Vehicle has come to complete stop at intersection and has received permission to travel though intersection	avg_slope $\approx$ 0 intersectionDetect = 0 obstacleDetect = 0 distanceToIntersection = -1 speed = 0 m/s	Vehicle accelerates to 0.5 m/s and travels through intersection, staying in its lane	Vehicle cross intersection at regular speed and exits intersection in correct lane	Pass
VSC3.7	V6	An obstacle has been detected in a straight lane and the vehicle is travelling at standard operating speed	avg_slope $\approx$ 0 intersectionDetect = 0 obstacleDetect = 1 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle comes to a complete stop before making contact with obstacle	Vehicle does not stop	Fail
VSC3.8	V6	An obstacle has been detected in a left turn and the vehicle is travelling at standard operating speed	-45 $<$ avg_slope $\ll$ 0 intersectionDetect = 0 obstacleDetect = 1 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle comes to a complete stop before making contact with obstacle	Vehicle does not stop	Fail

VSC3.9	V6	An obstacle has been detected in a right turn and the vehicle is travelling at standard operating speed	$0 \gg$ avg_slope > 45 intersectionDetect = 0 obstacleDetect = 1 distanceToIntersection = -1 speed = 0.5 m/s	Vehicle comes to a complete stop before making contact with obstacle	Vehicle does not stop	Fail

5.2.2 Intersection Controller Software Verification

5.2.2.1 Communication - Receive

Car:

- Abstract data type that contains all pertinent car request information
- This object is created for each new vehicle request
- Fields: car_ID, destination_Port, destination_bluetooth_address, direction_from, direction_to, proceed_status and proceed_message and time_stamp

Arrival Queue:

- Queue that preserves the order of the intersection requests in the form of Car objects

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
IC1.1	V1, IC4	Intersection request has been sent to the intersection controller from a single vehicle	Single vehicle intersection request	Car object containing the request information pushed into the arrival queue	Car object with relevant information pushed to the arrival queue	Pass
IC1.2	V1, IC3, IC4	Two vehicles arrive at the intersection at the same time and request to proceed through the intersection.	More than one vehicle intersection request	Car objects containing the request information pushed to the arrival queue in the order that they were received	Car objects pushed to arrival queue in order they were received	Pass
IC1.3	V1, IC3, IC4	Vehicle intersection requests that are transmitted incompletely should be dropped by it intersection controller	Vehicle intersection request that is terminated prior to receive completion	Timeout/ disconnected receive error is caught and system becomes available for subsequent requests	System remained available for service	Pass

IC1.4	V1, IC3, IC4	The system should be able to handle large request volumes without losing requests.	Many vehicle requests	Car objects containing the request information pushed to the arrival queue in the order that they were received	Car objects pushed to arrival queue in order they were received	Pass

5.2.2.2 Communication - Send

Proceed Queue:

- Queue of car objects that preserves the order determined by the intersection controller that vehicles should proceed through the intersection.
- Proceed commands are sent to the vehicles based on the FIFO sequence of the proceed queue

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
IC2.1	V1, IC5	Vehicle proceed request has been determined safe and must be communicated to the vehicle	Car object in the proceed queue	Remove vehicle from proceed queue and send proceed message to vehicle	Vehicle commands were sent to respective waiting vehicle	Pass
IC2.2	V1, IC3, IC5	Multiple vehicle proceed requests have been determined safe and must be communicated to the vehicle	More than one car object in the proceed queue	Remove vehicles from departure queue and send proceed message to vehicles in FIFO order.	Vehicle proceed commands were processed and sent to respective vehicles in FIFO order	Pass
IC2.3	V1, IC3, IC5	The system should be able to handle large send volumes without dropping proceed commands	Many vehiles in the proceed queue	“”	“”	Pass
IC2.4	V1, IC3, IC5	System should make further attempts to transmit messages to an unresponsive connection up to a max limit	Car object in the proceed queue	Inactive connection error should be caught. The system should then attempt to send the message again (up to the maximum attempt limit)	System attempted to retransmit message to maximum limit of three	Pass

IC2.5	V1, IC3, IC5	“”	“”	Timeout/ disconnected send error should be caught. The system should then attempt to send the message again (up to the maximum attempt limit)	“”	Pass

5.2.2.3 Communication - Message Extraction

Test ID	Requirement	Description	Input	Expected Values	Actual Values	Pass/Fail
IC3.1	IC4	Vehicle message contents are separated by an agreed upon separator	2_N_1_timestamp	Car(2, bluetooth Address, 1, ’N’, ’S’, ‘’, ‘’, ’timestamp’)	Car(2, bluetooth Address, 1, ’N’, ’S’, ‘’, ‘’, ’timestamp’)	Pass

5.2.2.4 Intersection Image Processing

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
IC4.1	IC1	The IC must be able to detect when there are the intersection is occupied	Video frame of the intersection with no cars on the track	The module determines that there are no cars in the intersection	The module shows that there are no cars at the intersection	Pass
IC4.2	IC1	" "	Video frame of intersection with a car in the intersection	Module determines the intersection is occupied	Module determines the intersection is occupied	Pass
IC4.3	IC2	The intersection must determine which car has navigated through the intersection using directions	Video clip showing vehicle proceeding from North to South of the intersection	The module records the movement	The module records the movement	Pass
IC4.4	IC2	" "	Video clip showing vehicle proceeding from South to North of the intersection	The module records the movement	The module records the movement	Pass

IC4.5	IC2	" "	Video clip showing vehicle proceeding from East to West of the intersection	The module records the movement	The module records the movement	Pass
IC4.6	IC2	" "	Video clip showing vehicle proceeding from West to East of the intersection	The module records the movement	The module records the movement	Pass

5.2.3 Integrated Vehicle Intersection Software Verification

5.2.3.1 Intersection Controller

Test ID	Requirement	Description	Input	Expected Behaviour	Actual Behaviour	Pass/Fail
VIC.1	IC3, IC6	Vehicles are allowed to proceed directly through the intersection, without stopping if the intersection is deemed safe and clear	Single vehicle, clear intersection	Vehicle proceeds directly through the intersection	Vehicle stops very briefly at intersection before proceeding	Fail
VIC.2	V4,IC3, IC4, IC6	“”	Single vehicle, intersection controller system not working	Vehicle should stop at the intersection and wait	Vehicle stops at intersection and does not proceed	Pass
VIC.3	V5, IC6	Vehicles are allowed to proceed directly through the intersection, without stopping if the intersection is deemed safe and clear and the intended paths are parallel	Multiple vehicles, parallel directions, clear intersection	Vehicles proceed directly through the intersection	Vehicles travelling in parallel directions receive proceed signals from intersection controller (tested by simulating car requests - not yet tested with actual vehicles)	N/A
VIC.4	V4, IC1, IC2, IC3, IC4, IC5	Vehicles with crossing directions must stop at the intersection, the vehicle that arrived first proceeds through the intersection first	Vehicle A and Vehicle B, crossing directions, clear intersection, Vehicle A is first	Vehicles A proceeds directly through the intersection, Vehicle B stops at the intersection	Vehicle A receives a proceed signal, while vehicle B stops and waits for a proceed signal (tested by simulating car requests - not yet tested with actual vehicles)	N/A

VIC.5	V1, V5, V6, IC4, IC5	“”	Multiple vehicles, intersection controller not working	Vehicles should stop at the intersection and wait	N/A	N/A

6 Appendix

6.1 Image Processing Verification Results

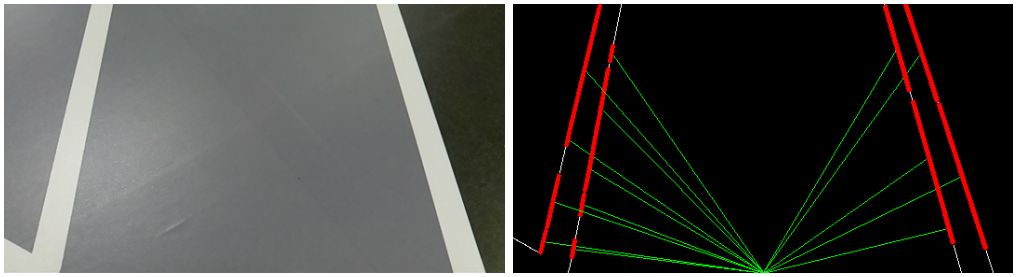

Figure 2: Straight line segment

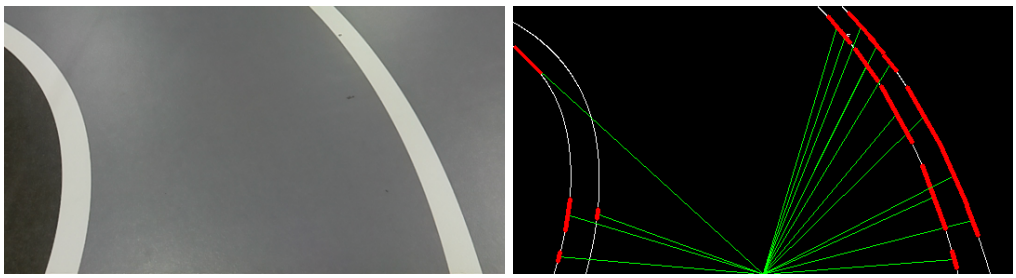

Figure 3: Left turn segment

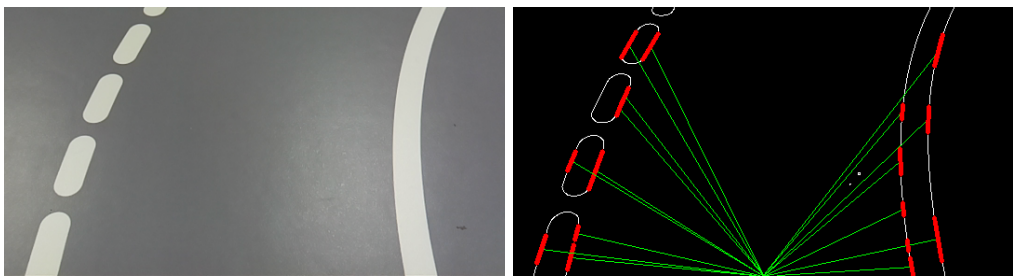

Figure 4: Right turn segment

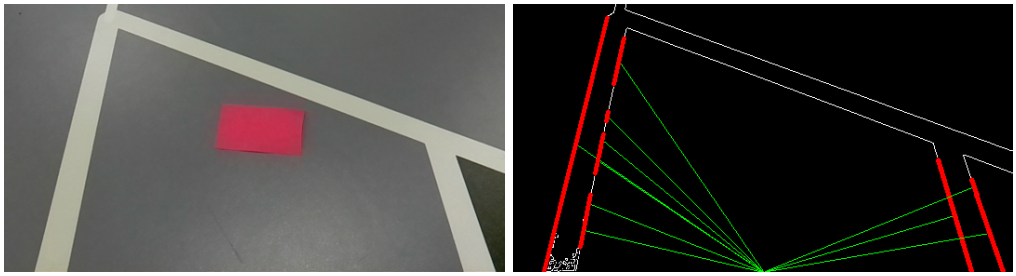

Figure 5: Approaching an intersection

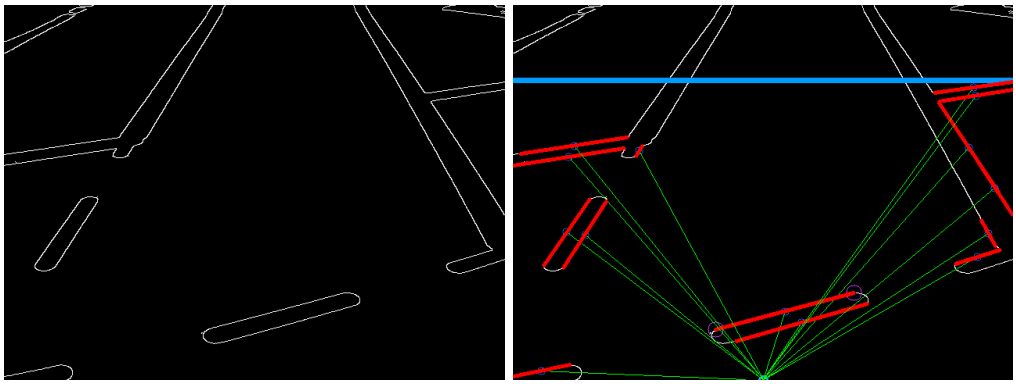

Figure 6: Navigating through an intersection