Intelligent Ground Vehicle Competition

IGVC - 2017

Indian Institute of Technology Madras, India

Team Abhiyaan

Faculty Advisor Statement:

I hereby certify that the development of vehicle, Kernel, described in this report has been equivalent to the work involved in a senior design course. This report has been prepared by the students of Team Abhiyaan under my guidance.

Nitin Chandrachoodan

Dr.Nitin Chandrachoodan
Associate Professor, Department of Electrical Engineering, IIT Madras

Team Members: Muhammedh Ajwahir, Sai Ramana Kiran, Kuldeep Purohit, Karteek Dhara, Abhijeet Ghawade, Mohammed Khandwawala, Siddharth Bhatia, Deepak Kumar, Viswajeet Anand, Favas M, Fakheem, Fariz, Utsav Tank, Atul, Vishnu Raj, Anand Rajasekar, Asit Tarsode, Milind Srivastava, Ajay Joshi, Narendiran CG, Yamini Singh

Contents

1 Introduction

The main functionality of this ground vehicle is to perceive the environment, localize itself in it and navigate outdoor while avoiding obstacles. We take sensor data from various sources and process it into a form that helps the bot map the environment and localize in it. This data is also processed by specific nodes to generate a costmap that effectively represents the obstacles in the environment and inflates them to approximately represent the configuration space of the bot.

We currently have the following 5 sensors on our bot:

- LiDAR (Light Detection and Ranging)
- GPS (Global Positioning System)
- IMU (Inertial Measurement Unit)
- Encoders
- Camera

LiDAR is the primary sensor used to detect obstacles. It scans the surroundings in a 2D plane and gives a polar profile that enables one to calculate the position of an obstacle within a certain range. GPS, IMU, Encoders are used for localization of the bot. We have also implemented visual odometry using a camera. An overview of the architecture can be seen in the flow chart 1.

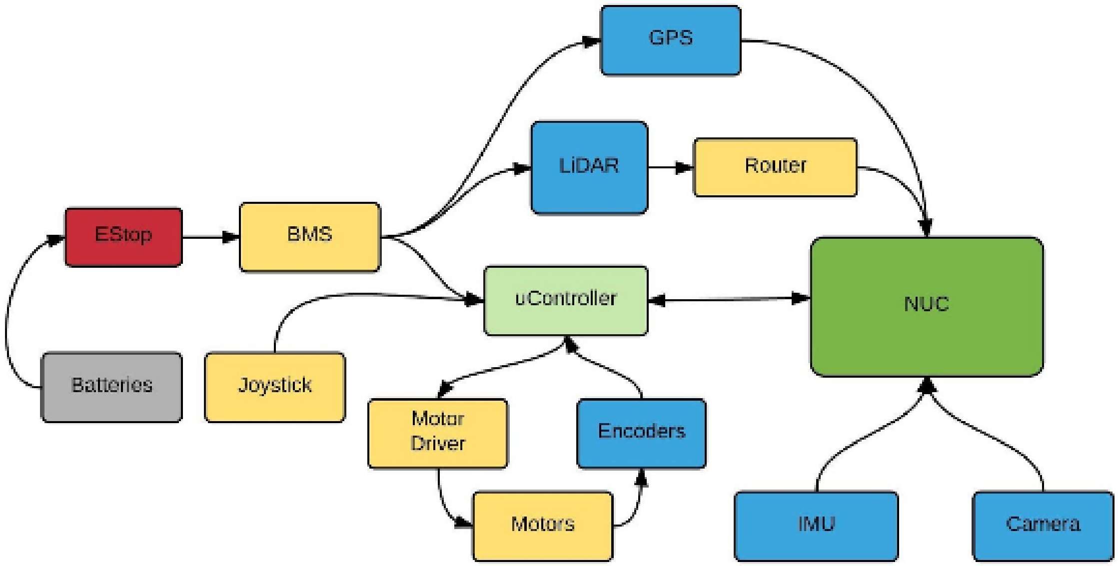


Figure 1: Architecture overview

2 Team organization and task split

The team consists of about 30 enthusiastic undergraduate as well as postgraduate students from various departments. The team is organized into 4 modules - Electrical, Mechanical, Navigation and Computer Vision. Bot Kernel is the second prototype of an autonomously navigating ground vehicle created by Team Abhiyaan. The shortcomings of the team during the previous years led to problems in the mechanical and electrical aspects of the bot. Efforts were made towards correcting these mistakes and to make the bot more robust.

3 Innovations in design

- **Modularity**: The bot is designed to be modular. It can be disassembled easily by unscrewing the screws holding the aluminium channels together. All sensors can also be removed easily from the mechanical frame of the bot. The acrylic sheets attached for rainproofing can also be simply unscrewed. This helps in easy transport of the bot.
- **Compartmentalization**: There are 3 levels of compartmentalization. The lower level has all the wiring, gears and motors. The middle level holds the circuitry and the BMS (Battery Management System). The upper level holds the components necessary for computation such as NUC and its battery, router as well as the IMU.
- **Bumper**: Bumpers are fixed onto the bot to protect the lower mechanical frame of the bot from serious damage. They are made from Fibre Reinforced Plastics(FRP).
- **3-D Printing**: We have 3D printed a custom made insulated casing for holding batteries and insulating PCB. It contains a LiPo bag that will contain any possible explosion and add another level of modularity that will be useful while disassembling and assembling the bot.

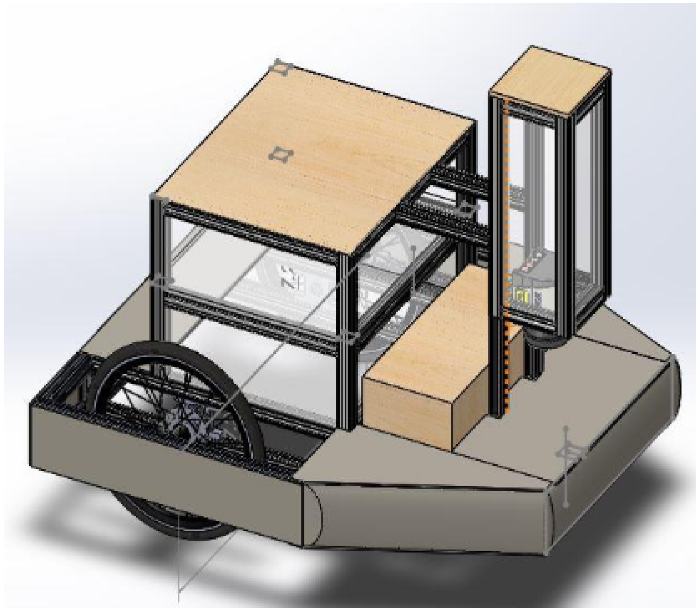


Figure 2: Initial Solidworks model of the bot

4 Mechanical Module

Weight and dimensions of payload are main considerations of design. The major factors considered to choose shape are efficient space use, low CG, stability, maneuverability.

4.1 Vehicle Structure

The vehicle2 can be subdivided into drive train, frame, steering and castor positioning.

1. Drive train:
   - Motors in central part
   - Drive shaft and motor connected using flange coupling arrangement
   - Ball bearings used to connect to frame to avoid friction and bending moments.
2. Frame:
   - Made using aluminium extrusions, ensuring easy assembly/disassembly and modularity. These channels are joined using angle brackets at right-angle joints
3. Steering:
   - 2 Differential drive motor powered wheels and 1 castor
   - Microcontroller controlled motor RPM
   - Internal encoders used for precise motor control and odometry calculations

4.2 Modularity

1. Modular system with independent floors, allowing independent operation
2. It came out to be extremely useful when individual changes were to be made in the electrical or mechanical aspect of machine in a short span of time.

4.3 Robustness

1. Motors are connected using flange coupling arrangement. The coupling arrangement is attached to the frame using ball bearings on either side of the wheel, which provide support for smoothness and help avoid friction and bending moments.
2. This coupling prevents transmission of high loads and shocks to the motor.
3. This arrangement helps in maximizing space utilization and increases the stability of the vehicle by having low centre of gravity.

4.4 Changes in the new design

1. New prototype has a higher LIDAR position and wider view. It gives a better identification of obstacle.
2. New prototype is rear wheel driven because of the following reasons.
3. Better acceleration:
   - When you accelerate quickly from a stop the weight of the bot transfers to the rear of the bot. Thus the normal force on the driving wheel is more while back wheel driven than while front wheel driven.
   - As normal force increases friction increases. Thus, we get more acceleration while back wheel driven.
4. Better Road Holding:
   - The better weight balance of rear wheel drive allows the bot to handle navigation better.
   - The more even weight allows the bot to drive neutrally through a corner. This means both the front and rear of the bot have near equal loads acting upon them.
   - In a front drive bot, the heavy front end causes the front end to have a higher load on it, causing the front tires to eventually lose grip.

4.5 Dimensions and specifications

Dimensions of the bot is given in the table 1

Table 1: Dimensions

Length x width x height	41x36x42 inches
Wheel diameter	15.7inches
Castor diameter	6 inches

Motor specifications: Maxon Brushless DC motors are used whose specifications2 were calculated using a max speed of 3.9 kmph and an average speed of 2.4 kmph

Gearbox specification:

The sepcifications of the gearbox design are given in the table 3.

Gear trains were first simulated 3a and then designed 3b accordingly.

Power train: Load on each motor (in terms of torque) while climbing up 15 ° Inclined track:

$$Load_T = 2 * m * g * R * \sin\theta = 2 * 60 * 9.81 * 0.2 * \sin 15 = 15.22Nm \quad (1)$$

Speed of the motor at the rated load:

$$\omega = V/R = 5 * 0.44704 * 3.5/0.2 = 39.11rad/s \quad (2)$$

$$Rpm of motor, N = 60 * \omega * 2 * \pi = 60 * 39.112 * \pi = 373.5rpm \quad (3)$$

Table 2: Motor sepecifications

No load speed	7580 rpm
No load current	15.7inches
Stall torque	2280 mNm
Max. Efficiency	91%
Operating Voltage	24V

Table 3: Gear sepecifications

Type of gear drive	Spur gear
Module	2 mm
Pitch diameter of Gear	126 mm
Pitch diameter of pinion	36 mm%
Gear Ratio	3.5
Pressure angle	20 $^{\circ}$

5 Electrical Module

5.1 Power System

Among all the sensors on the bot, LiDAR, GPS, motors and Intel NUC consume the most power. Owing to their high energy density mainly, the source of power on the bot are Lithium Polymer (LiPo) batteries. The power source comprising of three batteries power the who le bot 4.

5.2 Battery Management System

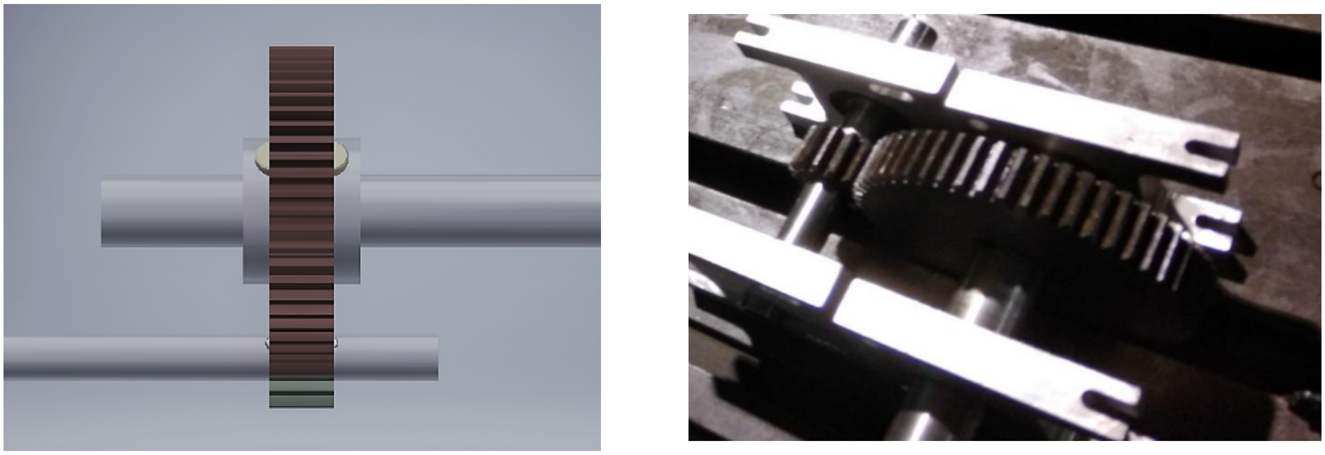
LiPo batteries are known to have a lot of energy cycles, but only if they are operated safely and within a certain voltage limit. As the battery discharges, its voltage also falls and for a LiPo battery when its voltage falls under a certain threshold, it becomes unusable and in some cases even unsafe. In order to deal with this issue our bot is equipped with a battery management circuit, which continuously samples each cell's individual voltage and when any cell's voltage falls below the threshold it immediately switches over to another battery of the same voltage.

5.3 Wireless control

- The Auxiliary power source is connected through the RF module on the bot. This RF Module helps to wirelessly control the circuit by switching it ON/OFF using a remote control. This serves as the Wireless E-stop, added to the **Mechanical E-stop** (Push-button) on the vehicle.
- Wireless Joystick: A wireless joystick has been designed for controlling the bot manually when it is not on 'self-driving' mode. The joystick is an analog 2-axis thumb device. NRF24L01 which wirelessly transfers the analog value to the bot, which has another transceiver NRF24L01.

5.4 Microcontroller unit

The Arduino Due was preferred as the MCU because of its QEI (Quadrature Encoder Interface). Two Arduino Due are used since any one of them individually cannot handle the encoder counts in the motors. UART serial communication interface is implemented between the 2 Arduinos which act as master and slave system to transfer data between them.5



(a) Simulated Gear

(b) Physical Gear

Figure 3: Gears

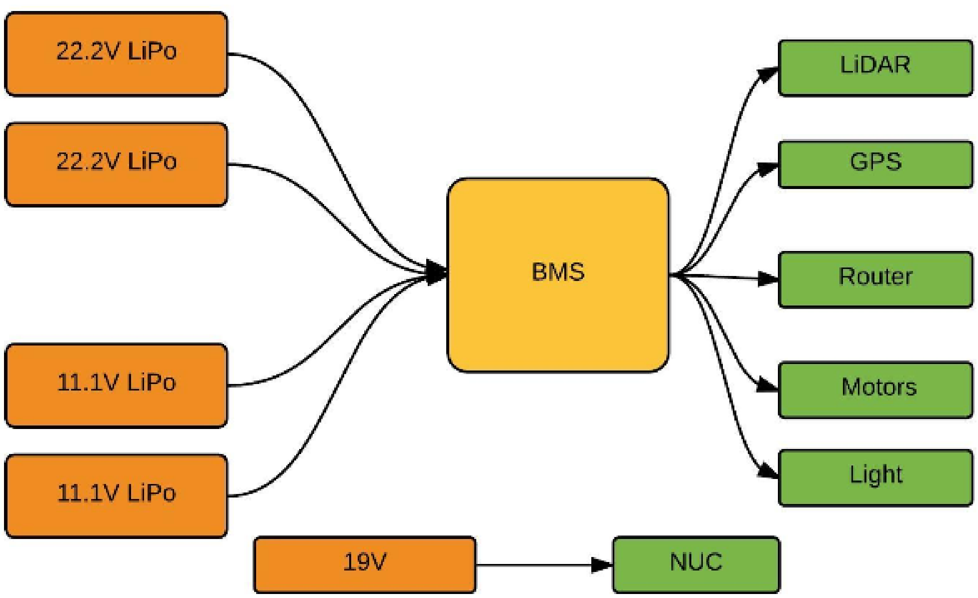


Figure 4: Power distribution

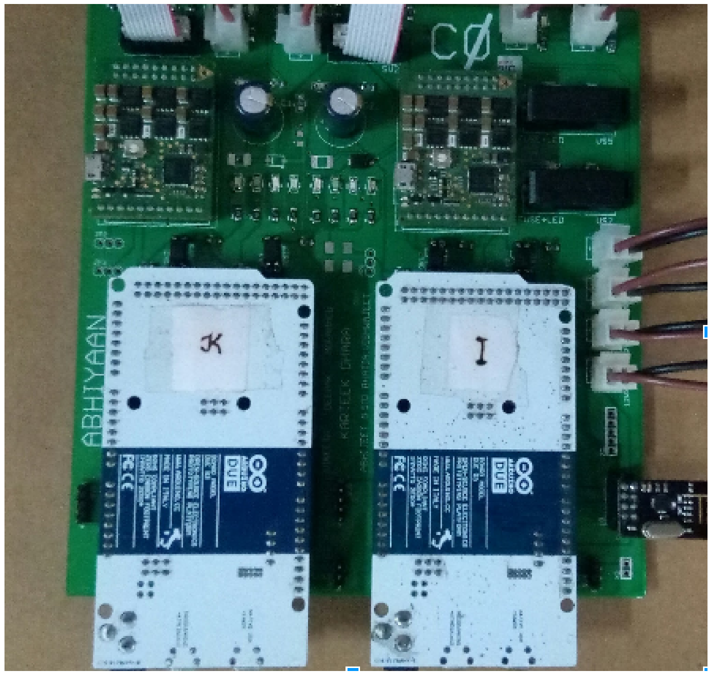


Figure 5: Main PCB

Since the digital I/O pins in the Arduino can only endure voltage upto 3.3V, there is a risk of burning the MCU while interfacing with sensors like the wheel encoders. To prevent this, optocouplers are used to step down the voltage signal of the encoder before providing it to the Due.

6 Software Module

We are currently using ROS (Robot Operating System) as the base framework for our bot. At the lowest level, ROS offers a message transmission interface that provides inter-process communication. The whole ROS architecture is divided into nodes that communicate with each other by means of topics. Nodes can subscribe to topics to access specific information and can publish to topics to disseminate information in the form of messages. The advantage of ROS is that it provides a modular base to work on, which effectively abstracts out the details of the working of one node from another. This helps in easy debugging.

The software consists of 2 main parts, one that deals with navigation part and the other that deals with computer vision.

6.1 Navigation

There are 3 main components that comprise the software. These are localization, mapping and path planning. We then use simulation to validate the robustness of these algorithms.

- **Localization** - Proprioceptive sensors used are IMUs and Motor Encoders. Exteroceptive sensors used are LiDAR and Camera. Environmental sensor are GPS. Data from proprioceptive and environmental sensor are fed into a EKF. The posterior from EKF and data from exteroceptive sensors is used as a prior in particle filter to improve the pose estimate of the robot.6

- **Mapping** - Occupancy grids are used to map data about environment as perceived by the sensors. Occupancy grids are filled with data from observation sources. Different occupancy grids

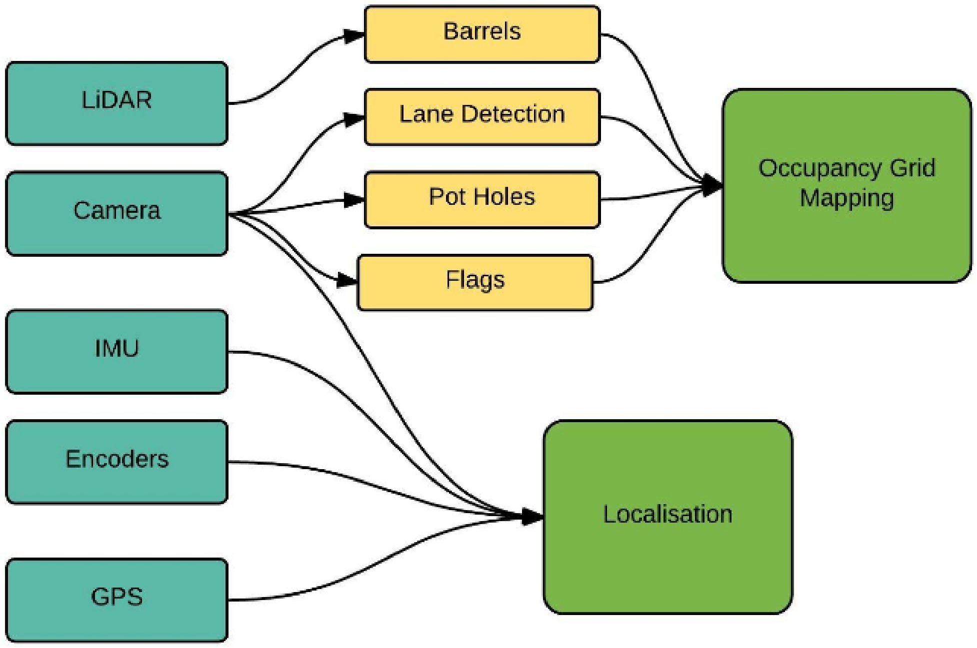


Figure 6: Localization and Mapping

are made to detect flags, lanes, barrels and they are combined. The resulting occupancy grid is used to create a costmap which approximates the robot configuration space. Obstacles/lanes and flags in the occupancy grid are inflated using a parameter with appropriate costs to create a costmap using data about the obstacles

- **Path Planning** - We plan paths in the costmap using the Teb Local Planner (Timed Elastic Planner). This planner exposes lot of parameters that allows one to customize its behaviour to a large extent. This planner also integrates a controller within it that provides command velocities that are fed to the motor drivers.

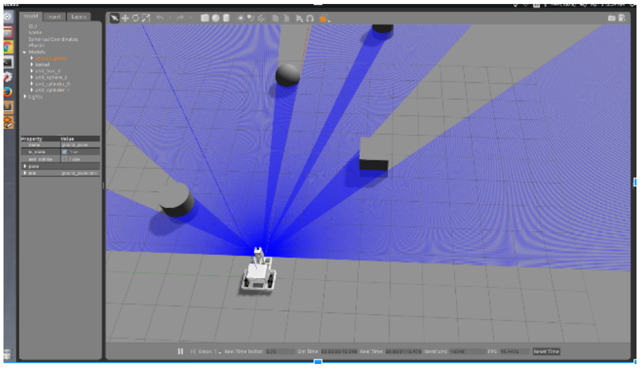
6.2 Simlulation

A robust physics engine is used to evaluate the model the efficiency of the algorithms and the performance of controller system and sensors. The simulation aids in reducing the stress on the battery system by limiting the testing of the bot only when it is necessary by acting as a point of validation. Screenshot of the same are in the fiugre 7

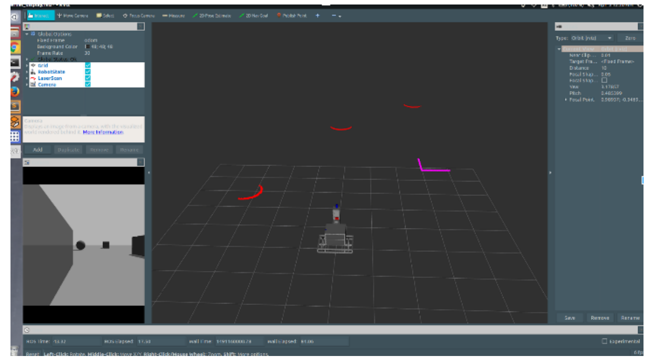
6.3 Computer Vision

We utilize the gradient and color information for lane detection which can then be utilized as a base map for generating a costmap. We use innovative algorithms in computer vision to achieve this task.

To avoid processing the weak edges resulting from the unnecessary texture within the grass regions, we utilize mean-shift filtering which causes edge-preserving smoothing of the image. This preserves and emphasizes the boundaries between the lane and grass regions.



(a) LiDAR detecting obstacles in the environment



(b) Visualization in RViz

Figure 7: Simulation of bot Kernel in Gazebo

Within the processed image, we detect lanes using a custom algorithm, namely the matched filter with first-order derivative of the Gaussian (MF-FDOG), as an extension and generalization of the MF. It has been successfully applied to nerve segmentation in retinal images. This method is computationally efficient with the simplicity of a convolutional operation using gaussian kernels and its derivatives, but leads to much higher accuracies than edge (or color) thresholding based methods. Considering that the cross section of a lane is a symmetric Gaussian function, we use a pair of filters, the zero-mean Gaussian filter (i.e. the MF) and the first-order derivative of the Gaussian (FDOG), to detect the vessels. For a true lane, it will have a strong response to the MF around its peak position, while the local mean of its response to the FDOG will be close to zero around the peak position. In contrast, for non-lane structures, for example the step edge, it will have high response to the MF but the local mean of its response to the FDOG will also be high. We apply a locally adaptive threshold to the response of matched filter using the local output value of FDOG filter.

This method is robust to shadows or uneven lighting conditions and works with lanes of any background. The working is shown in the figure 8 in sequence.

7 Modes of failure and remedies

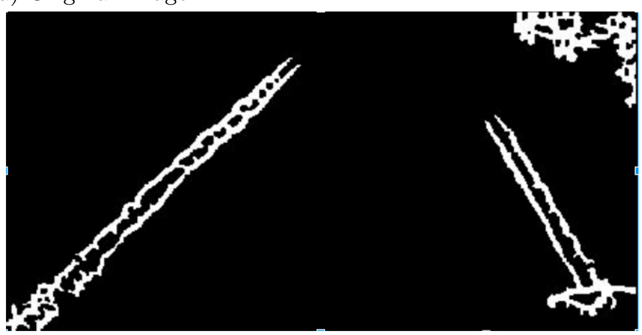
They are tablulated in table 4

We complement this approach with a color based method for image segmentation, namely the Graph-Cut algorithm. It minimizes the sum of a data cost and a smoothness cost calculated over all the pixels in the image. The data-cost is the cost of assigning a particular class label to any pixel based on its color. It decreases if the pixel is more white (lane-like). The smoothness cost is the cost of assigning a particular label to a pixel, given the labels assigned to its neighboring pixels. It decreases when all the nearby pixels have the same class label. Proceeding in this fashion, we obtain color based segmentation for every frame.

We take the intersection of the two results for every frame and align it with Laser-sensor based coordinates for path-planning. We then take homographic projection and combine it with the laserscan. The limitation of this algorithm is that due to high gradients in the grass region if they aren't smoothened properly they might also be detected as lanes.

(a) Original Image

(b) Mean-shifted Image



(c) Detected lanes

Figure 8: Lane detection using MF-DoG

Table 4: Falure modes and remedies

Failure Modes	Resolution/Remedy
Excess vibration that may distort sensor data	Air filled tyres with pressure that can be varied accordingly to dampen vibrations
Battery drain	BMS will change over to a spare battery and in worst case stops power supply if no spare battery charge is left
Battery explosion	Insulated casing with LiPo pack to contain the effects of battery explosion
Damage to wires	Running wires through grooves in aluminium extrusions to prevent external factors from damaging wires
No data from LiDAR/Overflow of encoder data	Stopping the vehicle in such extreme cases
False detection of lanes	Using meanshift algorithm to smoothen the image to avoid false detection

8 Cost Estimation

In the table 5, the second column indicates the cost born by this year's team. Certain sensor components were already bought by the previous year team.

Table 5: Cost

Components	**Actual Price (in USD)**	**Cost born by the team**
Sick LMS1xx	5,000$	0$
Hemisphere a101	3,000$	0$
Sparton IMU	1,500$	0$
Maxon Motors	770$	770$
Escon Motor Drivers	320$	160$
Microcontrollers	190$	190$
Batteries	500$	500$
Miscellaneous Electrical components	160$	160$
Frame and fasteners	300$	300$
Miscellaneous Mechanical components	100$	100$
Acrylic sheets	30$	30$
Total	13,860$	2,210$

9 Acknowledgment

The team would like to acknowledge IIT Madras for exposing us to such an amazing opportunity to participate in IGVC – 2017. Our faculty advisor, Prof Nitin Chandrachoodan, has always been extremely supportive of our project. We would like to thank our seniors and contributors especially Prathyusha, Arvind and Ashish who have been a pioneer in this field and guided us in the early stages. We would also like to thank the student body, Center For Innovation (CFI), of IIT Madras which has always stood by us in times of challenges and motivated us.