DEVELOPMENT OF SENSOR FUSION ALGORITHM FOR ENHANCED LOCALISATION AND NAVIGATION

Pavansudhan H, Naveen Kumar P, Roshan Kumar S, Dr. Arunachalam K

Department of Automobile Engineering, Madras Institute of Technology, Anna University, Chennai, India

**Abstract**

In Autonomous vehicles the main objective is to be able to navigate and drive to the desired location without any driver input. To be able to do this the vehicle must be able to have an answer to a very important question that is “where am I”. Only when a vehicle can identify its position, it will be able to navigate and drive to the desired location. So, to achieve a good navigation, the vehicle’s position and orientation in an environment must be clear. To achieve a good localization estimate, sensor fusion is done by using Kalman filters. In conventional systems, the functioning of a system is fully dependent on the data from a single sensor. We cannot fully rely on the data from a single sensor, if the functioning of the sensor gets affected by a weather condition like blurred images in camera during rain or some other external means or the sensor fails the whole system may fail or will not be able to function. When data from multiple sensors are fused, the reliability of the data and the accuracy of the estimate increases. Object detection and classification is done using YOLO (Convolutional Neural Network) due to its speed and accuracy. In our algorithm we are planning to fuse data from IMU, GPS, LIDAR, CAMERA. The combination of these four sensors will help to improve the reliability of the localization estimate. Where GPS and IMU are crucial for the position estimate, while LIDAR and CAMERA are useful to provide a better understanding of the external vehicle environment. The obtained localisation estimate will have better accuracy when compared with single GPS and IMU. It also provides promising results irrespective of the external conditions. Hence the sensor fusion method is an effective solution which can be used in autonomous vehicles.

*Keywords:* Sensor Fusion, Kalman Filter, Neural Network, localization

**1. Introduction**

In the era of intelligent systems and technological innovation, the improved precision in localization and navigation has become challenging in various fields, ranging from autonomous vehicles to robotics and augmented reality. The development of a sensor fusion algorithm plays an important role in achieving excellent accuracy and reliability in determining the spatial coordinates and orientation of objects in static and dynamic environments. The algorithm is designed for faster performance and improved accuracy to obtain enhanced localization and navigation. This research paper focuses in developing the algorithm for sensor fusion designed to synergize diverse sensor modalities for enhanced localization and navigation capabilities. The KITTI dataset [1] contains recorded sensor data obtained from a standard station wagon equipped with two high-resolution colour and grayscale video cameras, Lidar, IMU and GPS. Accurate ground truth is provided by a Velodyne laser scanner and a GPS localization system. The datasets were captured by driving around the mid-size city of Karlsruhe, in rural areas and on highways. Up to 15 cars and 30 pedestrians are visible per image. The recording platform is a Volkswagen Passat B6, which had been modified with actuators for the pedals (acceleration and brake) and the steering wheel. The data was recorded using an eight core i7 computer. This paper aims to solve the localisation and perception block of an autonomous vehicle [2,3]. The algorithm seeks to provide a more comprehensive and accurate understanding of the surroundings, contributing to reliable localization and improved navigation performance. By improving the precision of the real-world estimation, it improves the performance of the vehicle.

**2. Analyzing Data from Sensors**

There are multiple sensors available with their own advantages and limitation that are used in Autonomous vehicles to collect detailed information about the surroundings. The sensors used in this project are LiDAR (Light Detection and Ranging) for 3D point cloud generation, Cameras for visual information of the environment, GPS for global positioning, and IMUs (Inertial Measurement Units) for measuring accelerations and angular velocities to obtain the orientation of the vehicle and the estimate of the vehicle over a time interval. These data obtained from the sensors are fused which enables continuous update of the surroundings for safe and effective autonomous navigation. Analyzing camera data involves tasks such as object detection, recognition, and tracking, that results in a detailed understanding of the scene. Generated point clouds from LIDAR represents the three-dimensional structure of the surroundings. It provides the distance information of each obstacle from the vehicle all around.



Fig1a : Sample Image from KITTI Dataset



Fig1b: Sample Images from KITTI Dataset

*2.1. Calibration of Sensors*

Camera calibration includes Intrinsic Calibration and Extrinsic Calibration [4]. Intrinsic Calibration involves the parameters of camera such as focal length, principal point, and lens distortion. This step helps to improve the quality of the images from the camera by removing the distortions, blurred images etc. Extrinsic Calibration helps to establish the relative positions and orientations of the cameras providing a common reference frame. This develops a relation between the world coordinate and the camera coordinates. Lidar Point cloud is generated from the points obtained by the lidar sensor in the 3D space. Image from camera is placed in the 3D space to align the image with the lidar output. To convert a point from the LiDAR to the camera image space, we must perform the transformations on the point cloud. All points in the point cloud lying the region of interest must be transformed to Camera Image. These steps are done with the help of transformation matrices in which quick substitution would translate points to Camera Image [5]. The first transformation converts the point from the LiDAR to camera frame of reference via rigid body transformations. The camera frame of reference is a 3D point in the world with respect to the camera's location on the ego vehicle.

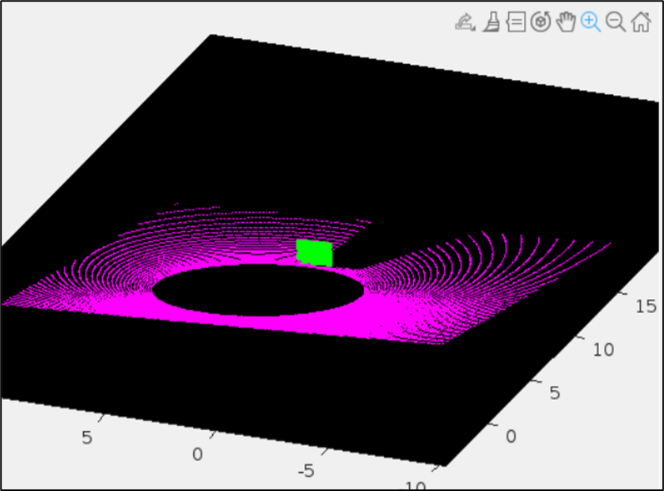


Fig2: Lidar to Camera Calibration

A rigid transformation is a rotation followed by a translation. A rigid transformation matrix is the combination of these two components. The r's in the matrix correspond to the 3x3 rotation matrix about the three axes x, y and z and the t's correspond to the 3x1 translation vector about three axes. This transformation matrix can be used to perform rigid transformations from Camera to Camera. The rectifying transformations are done to sync the lidar and camera data. This step is necessary since the stereo images are used that are referenced to the same y-axis. The object on left image has the same y-coordinate as the right image, but the x-coordinate can be different. When a pair of stereo images are referenced to the same image plane y-axis it is obtained as a rectified image. Finally, the camera projection transformation is applied. This gives us the location of a 3D point in 2D image space, which

we denote as (u,v) or (u,v,z) if depth info is available. Lidar to Camera calibration, lidar to IMU calibration and camera to camera calibrations are also done using the transformation matrices.

**3. Processing Data from Sensors**

*3.1. Lidar Point Cloud Generation*

Lidar point cloud was generated using the binary points with 3D coordinated obtained from the LIDAR sensors using MATLAB. The binary points were converted into x, y and z coordinates and arranged in the point sequence from 1 to 64 since the lidar used is Velodyne HDL 64E that sends 64 beams around 360 degrees of rotation continuously. This provides a base idea about the environment how the obstacles are located. The lidar point cloud can be viewed in the Lidar viewer app inbuilt in matlab by importing the lidar points sequence.



Fig3: Lidar Point Cloud

*3.2 Integration of Lidar and Camera*

As the cameras capture images of the vehicle's surroundings, the YOLO algorithm [6] processes these images, dividing them into a grid and predicting bounding boxes, class probabilities, and confidence scores for detected objects in a single pass through the neural network. The images from the camera are provided as input to the YOLO algorithm that process at high rate of 35 frames per second that depends on the processer specifications. It is known for its real-time efficiency, it is integrated with cameras, serving as a key component in the sensor suite. YOLOv5 is used due to improved capabilities to identify even smaller objects in the environment [7]. Classes such as bicycle, car, van, bus, person, traffic signal and miscellaneous are created and each class is provided 50 images taken at different angles, different models etc for the algorithm to train. Feature extraction layer is now generated in the neural network. Now for the input image each grid is compared with this layer and the mean average precision value (MAP) is measured. The class with the highest precision value is displayed on the detection boxes. YOLO was chosen over the other types of CNNs such as the Region-based CNN and Faster CNN due to its high speed, decent accuracy and precision values [8].

A street with signs and trees

Description automatically generated

Fig4: Object Detection using YOLO

*3.3. 2D Mapping of Vehicles and Obstacles*

2D Mapping involves the simultaneous creation of a map of the environment and the localization of the vehicle within that map [9]. This process is crucial for the autonomous vehicle to navigate and make informed decisions about its surroundings. The coordinates from GPS sensor provides the present location of the vehicle and the location of the obstacles are calculated using the distance parameters which are the x and y measurements from the lidar point cloud. Using the obstacle detection and classification algorithm the various obstacles and their respective distances were measured. The information is used to find the coordinate of each obstacle and they are plotted simultaneously in the maps.



Fig 5: Obstacle Mapping on 2D Image

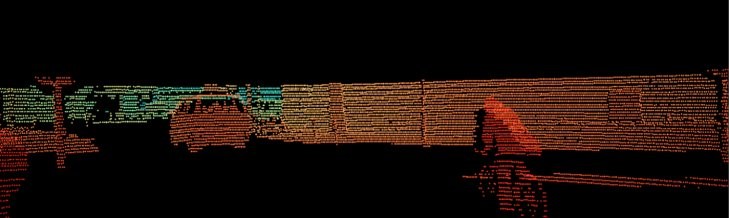
The green marker indicates the current position of the vehicle, and the red markers indicate the positions of the identified obstacles. The facing direction of the vehicle is known and the angle at which the obstacle located is obtained from the lidar sensor. This helps the vehicle to obtain a clear idea of the surroundings and can make a clear decision on overcoming the obstacles. The obstacle positioning can also be done on empty canvas or designed images that represent road conditions to convey the vehicle in an efficient way to understand the surroundings.

*3.4 Position and Orientation Estimate*

The imu unit contains three main components viz MAGNETOMETER, GYROSCOPE AND ACCELEROMETER. The linear acceleration of the vehicle is measured with the accelerometer, the angular velocity with gyroscope and the magnetic field with magnetometer [10]. The Magnetometer readings helps to identify the direction of tilt if the vehicle(orientation) through the direction of the magnetic field related to the local magnetic field direction (which points into the ground in the Northern hemisphere). These readings are essential in the orientation estimation of the vehicle. The filter used was AHRS filter which works on the principle of Kalman function as it had quicker response compared to another Complimentary filter (lower tuneable parameters). Using the angular velocities and acceleration the position of the vehicle at a time interval is estimated using the Kalman filters. It is thus continuously updated in the maps.

**4. Results and Discussion**

From the generated 3D lidar point cloud which is plotted over 360 degrees of rotation, the region of interest which is the part of the environment captured by the camera is selected to integrate with the image. Hence the point cloud from the start to end of the image pixel is cropped and stored to fuse with the camera image. In an empty canvas the point cloud with the distance information of each point is stored with varying colour range. The colour range that is used to plot the points on the image also indicates the distance from the vehicle. The colour range from violet to red indicates distant to nearly obstacle respectively.

  
Fig6: Lidar – Range of Interest

Now the cropped region from the point cloud is projected on the image to obtain both the visual information and the distance information in a single output. This simplifies the output for a better understanding by the vehicle.



Fig7: Lidar projection on Image

From the output of image detection and classification the bounding boxes are generated with width and height dimensions. The centroid of these boxes are calculated which is the half distance from the edges. The coordinate of the centroid is synced with the lidar output and the distance information of that coordinate is obtained and displayed on the detected boxes. The detailed information about the surrounding obstacles and their distances from the vehicle is obtained. Now mapping of these objects would improve the understanding of the vehicle.

  
Fig8: Obstacle Distance Measurement

The coordinates from GPS sensor provides the present location of the vehicle and the location of the obstacles are calculated using the distance parameters. The location of the obstacle is obtained by the summation of the distance from the present location and the angle at which the obstacle is located. This information can be accessed from the lidar output.

A map of a city

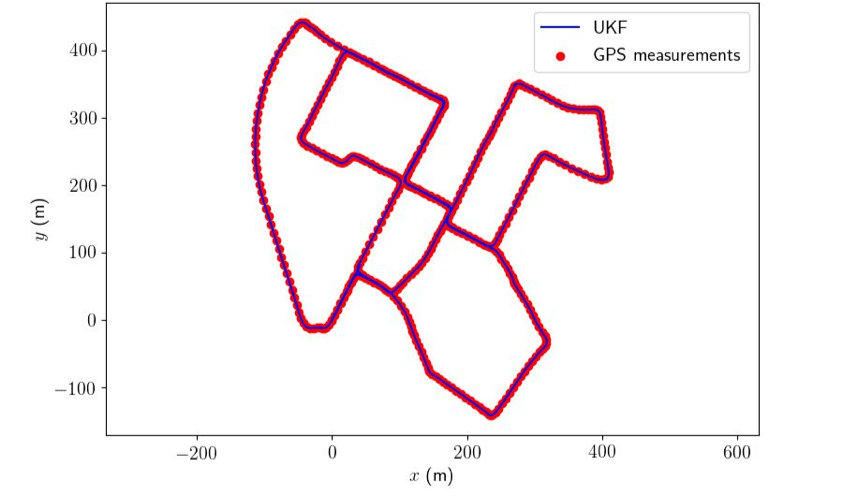
Description automatically generated

Fig9: 2D Simultaneous Mapping

The blue marker indicates the current position of the vehicle, and the red markers indicate the positions of the identified obstacles. The facing direction of the vehicle is known and the angle at which the obstacle located is obtained from the lidar sensor. This helps the vehicle to obtain a clear idea of the surroundings and can make a clear decision on overcoming the obstacles. Obstacle positioning can also be done on empty canvas or designed images that represent road conditions to convey the vehicle in an efficient way to understand the surroundings.

*5.5. IMU-GPS Integration*

An UKF (Unscented Kalman Filter) is used for sensor-fusion of an IMU with GPS, it is done to obtain an estimate of the vehicle’s position in terms of either xy coordinates or latitude and longitude. The UKF is a filter that estimates the states of a discrete-time nonlinear system. The Kalman filter works basically in with successive prediction and correction steps, first it predicts the state of the system with initial values or variables. At each time step, the Kalman filter predicts the current state of the system based on the previous state and a system model. This prediction is represented by two components: the predicted state estimate and the error covariance estimate. The Kalman filter then incorporates the current estimate to update the state estimate. This update is based on the difference between the predicted measurement and the actual measurement, along with the measurement matrix. The filter minimizes the error of the estimate by combining the prediction and measurements steps. It adapts to changes in the system dynamics and is particularly effective when dealing with systems subject to noise and uncertainties. In this application the state or position of the vehicle is continuously predicted by the Kalman filter using the variables from the IMU and GPS units (3d orientation, velocity, position, gyro bias, accelerometer bias, vehicle angular velocities, vehicle specific forces) and subsequent measurement updates. In the figure below we can see that the results are coherent with the GNSS.

  
Fig10: Position and Orientation estimates

**6. Conclusion**

In conclusion, this research has presented a method to integrate the sensors data obtained from various sensors positioned in an autonomous car to obtain the detail perception of the environment. The location and orientation of the vehicle is obtained through GNSS and the IMU (Inertial Measurement Unit). The environment study is done by Camera for visual perception and LIDAR for distance measurement. It also provides a comprehensive comparative analysis of two prominent object detection frameworks, YOLO (You Only Look Once) and R-CNN (Region-based Convolutional Neural Network), shedding light on their respective strengths, weaknesses, and performance characteristics. The YOLO framework, with its real-time processing capabilities and single-pass object detection, demonstrates remarkable efficiency, making it particularly well-suited for applications where low-latency responsiveness is critical. On the other hand, R-CNN, characterized by its region proposal mechanism and detailed feature extraction, excels in accuracy and precision, especially when confronted with complex scenes and small object instances. While YOLO emerges as a frontrunner in terms of speed and streamlined processing, R-CNN showcases superior performance in scenarios demanding heightened localization accuracy and robust object recognition. The choice between these frameworks becomes contingent on the specific requirements of the application at hand, balancing the need for real-time responsiveness against the quest for meticulous object delineation. Furthermore, the comparative analysis has underscored the evolving nature of object detection methodologies, with subsequent iterations of both YOLO and R-CNN, such as YOLOv4 and Faster R-CNN, pushing the boundaries of performance and versatility. As the field continues to advance, future research endeavours may explore hybrid approaches that leverage the strengths of both frameworks, seeking to capitalize on real-time efficiency without compromising on the intricacies of object representation.

**Acknowledgment:**

The authors acknowledge Mathworks for providing student accounts with MATLAB license which helped in research and development of the sensor fusion algorithm.

**References**

[1] Andreas Geiger, Philip Lenz, Christoph Stiller and Raquel Urtasun. Vision meets Robotics: The KITTI Dataset, International Journal of Robotics Research (IJRR), 2013

[2] Scott Drew Pendleton, Hans Andersen, Xinxin Du, Xiaotong Shen, Malika Meghjani, You Hong Eng, Daniela Rus, Marcelo H. Ang Jr. Perception, Planning, Control, and Coordination for Autonomous Vehicles, Singapore, 2017

[3] Jamil Fayyad, Mohammad A. Jaradat, Dominique Gruyer, Homayoun Najjaran. Deep Learning Sensor Fusion for Autonomous Vehicle Perception and Localization: A Review. 2020

[4] Ji Dong Choi, Min Young Kim. A sensor fusion system with thermal infrared camera and LiDAR for autonomous vehicles and deep learning based object detection, Republic of Korea, 2021

[5] Jun Zhang, Prarinya Siritanawan, Yufeng Yue, Chule Yang, Mingxing Wen, Danwei Wang. A Two-step Method for Extrinsic Calibration between a Sparse 3D LiDAR and a Thermal Camera, Singapore, 2018

[6] Naman Mittal, Akarsh Vaidya, Asst. Prof. Shreya Kapoor (2019), “Object Detection and Classification Using Yolo”, International Journal of Scientific Research & Engineering Trends - Volume 5, Issue 2, Mar-Apr-2019, ISSN (Online): 2395-566X

[7] Dr. Viraktamath. S. V, Madhuri Yavagal, Rachita Byahatti (2021), “Object Detection and Classification using YOLOv3”, International Journal of Engineering Research & Technology (IJERT) Vol. 10 Issue 02, February-2021.

[8] Muhamad Munawar, Yusro Rozniza Ali, Muhammad Suzuri Hitam. Comparison of Faster R-CNN and YOLOv5 for Overlapping Objects Recognition, Malaysia, 2022

[9] Johann Laconte, Abderrahim Kasmi, Romuald Aufrère, Maxime Vaidis, Roland Chapuis. A Survey of Localization Methods for Autonomous Vehicles in Highway Scenarios, Canada, 2021.

[10] Mahdi Abolfazli Esfahani, HanWang, Keyu Wu, Shenghai Yuan. OriNet: Robust 3-D Orientation Estimation With a Single Particular IMU. IEEE 2020