CSCI 491/591 - Embedded System - Project Proposal

Final project – CSCI491/591 Embedded System

Autonomous Robot Car

Subtitle: Real-time self-driving robot car implementing Extended Kalman Filter for noise reduction and efficient states for car's operations

Nhan Cao and Ma Vang

# Introduction

Autonomous robot cars have gained popularity in recent years due to advancements in robotics and artificial intelligence. These cars can sense their environment and navigating without human intervention. In this project, we propose to design and implement a low-cost, real-time autonomous robot car using a hardware PoC of Arduino Uno and different sensors (IMU ICM20948, ultrasonic HCSR04). In addition, we also apply a technique called Extended Kalman Filters on the sensors data for noise reduction and introduce a set of operation states to help the car work stably.

# Goal/Objective

The primary goal of this project is to design and implement a lightweight, low-cost, and real-time autonomous robot car using an Arduino Uno, Inertial Measurement Unit (IMU) sensor, and ultrasonic sensors for distance measurement. The car will be able to navigate and avoid obstacles using sensor fusion as an output from EKF algorithm.

The project will be divided into the following components:

* Hardware and sensor integration: Assemble the robot car and integrate the necessary sensors for navigation and obstacle detection.
* Software implementation: Develop the EKF algorithm and control logic for the car's movement.
* Testing and evaluation: Test the autonomous robot car in different environments and evaluate its performance.

# Controlling mechanism and Algorithm design

***Flow chart***

Diagram

Description automatically generated

***State machines***

Diagram, timeline

Description automatically generated

***What are the possible challenges that you may need to solve?***

* Sensor noise and inaccuracies: The IMU and ultrasonic sensors might provide noisy and inaccurate data, which can affect the car's navigation and obstacle detection capabilities.
* Real-time processing: Ensuring that the system can process sensor data and control the actuators in real-time, especially with limited computational resources on Arduino Uno.
* The lack of resources on Tiny embedded platform like Arduino UNO (only 32kB Flash and 2kB SRAM), since some opensource libraries for those kind of sensors like ICM20948, HC-SR04, VL053X, Servo/Motor shield as well as the TinyEKF usually require large footprint of memory (mainly affect the SRAM). W
* Path planning and navigation: Designing an algorithm that can efficiently navigate the car along a predefined path while avoiding obstacles and missing the waypoints.

***What are possible solutions to solve these problems?***

* Sensor noise and inaccuracies: Use the Extended Kalman Filter to reduce noise and improve the accuracy of IMU data. In addition, we can also try some other common lightweight filtering techniques on IMU data like complementary filter (quite similar to EKF) and lowpass filter. Regarding the distance data collected from Ultrasonic sensor, since it works based on a strict timing impulse mechanism, we need to replace the standard library of HC-SR04 implemented by Arduino (well known for the unstable operation) with an optimized version that interact directly with the registers.
* Real-time processing: Optimize the algorithms and code for efficient execution on the microcontroller, mainly adjust the level of computation from EKF to achieve real-time purpose.
* The lack of resources: Need to do some optimizations like storing some large size variable to Flashlash or EEPROM, change the variable definition to minimize the memory footprint. We also plan to build our own version of some sensors opensource libraries to reduce the code size and memory usage, have done with HC-SR04, still has one of ICM20948 (currently it takes over 600 bytes to just initialize the ICM20948 class on Arduino, we have checked their driver and there are many things that we do not need for this project).
* Path planning and navigation: Implement a reliable algorithm to keep the car move straight, can pass the obstacles, and do not miss any waypoint.

# Hardware

The hardware components of the autonomous robot car will include:

* Microcontroller Arduino Uno: ~50mA in normal condition
* IMU sensor (ICM20948): ~3mA max due to the datasheet
* 3 Ultrasonic sensors for distance measurement (HC-SR04): ~45mA (15 mA max each due to the datasheet)
* Arduino Motor Shield V2: around 300mA
* Battery for powering the system (4 AA batteries in series): each with a capacity of around 2000mAh (typical for alkaline AA batteries). Since the batteries are connected in series, the total capacity remains the same: 2000mAh.
* Battery life = Battery capacity / Total current consumption = 2000mAh / 398mA ≈ 5 hours in perfect condition when the car keep moving frequently