Czech University of Life Science Prague

Faculty of Economics and Management

Department of Information technologies

Diploma Thesis

Artificial Intelligence in Self-Driving Vehicles: mBot Robot

Feras Daoud

Czech University of Life Science

Faculty of Economics and Management

DIPLOMA THESIS ASSIGMINGT

BSc. Feras Daoud

System Engineering and Informatics

Thesis title:

Artificial Intelligence in Self-Driving Vehicles: mBot Robot

Objectives of thesis

The aim objective of the thesis is to implement a smart robot called mBot to be an example of autonomous vehicle (driverless) by using Artificial Intelligence, Internet of Things and Machine learning technologies.

Methodology

The methodology for this thesis is based on a study of the latest autonomous vehicle and artificial intelligence literature. According to these requirements, mBot and Arduino have chosen instead of Python after the supervisor acceptance. We will try to develop and implement comprehensive AV system based on AI and ML in realistic terms. Final suggestion will be made, and assumptions established by synthesizing and comparing observations in the literature review and the practical section.

The proposed extent of the thesis

60-80

Keywords

Recommended information sources

1- H. Khayyam, B. Javadi, M. Jalili, and R. N. Jazar, “Artificial Intelligence and Internet of Things for Autonomous Vehicles,” in *Nonlinear Approaches in Engineering Applications*, 2020.
2- J. Anderson, N. Kalra, K. Stanley, P. Sorensen, C. Samaras, and O. Oluwatola, *Autonomous Vehicle Technology: A Guide for Policymakers*. 2016.
3- “The 6 Levels of Vehicle Autonomy Explained | Synopsys Automotive.” https://www.synopsys.com/automotive/autonomous-driving-levels.html (accessed Oct. 12, 2020).

Expected date of the defense

2020/21 WS – FEM

The Diploma Thesis Supervisor

doc. Ing. Vojtech Merunka, Ph.D.

Supervising Department

Department of Information Technologies

Electronic approval: 31. 01. 2019

Ing. Martin Pelikán, Ph.D.

Head of department

Electronic approval: 31. 01.2019

Ing. Martin Pelikán, Ph.D

Dean

Prague on 05. 04. 2020

Declaration

I announce that I have been working on my diploma thesis named “A**rtificial Intelligence in Self-Driving Vehicles: mBot Robot**” by myself and have only used the sources stated at the end of the thesis. As the writer of the study, I announce that the work does not break any person's copyrights.

In Prague on 05. 04. 2020 ______________________________

Acknowledgement

I would like to thank my **family** and my supervisor **Ing. Vojtech Merunka, Ph.D** for the support and encouragement throughout this project.

Artificial Intelligence in Self-Driving Vehicles: mBot Robot

Abstract

The ability of autonomous vehicles (AVs) is to mold urban life and to change travel habits substantially. "Autonomous system" means a technology that can conduct a vehicle by a human driver without physical supervision or tracking.

With the arrival of autonomous vehicles humanity will have to face a new risk collection and some great benefits, which requires the opportunity for the first time to make difficult risk reduction decisions with socially integrated types of artificial intelligence: decisions which will inevitably have real lives and death-effects. Since the decision-making mechanisms of AI are fundamentally different from those of humans, it raises concerns about how AI judges' decisions, how are we to impact these decisions, what all these decisions means for the others. In the other hand, AVs give us many benefits and open many opportunities in the future, our life will be smart and easy.

We use mBot as a robot of our learning project in this thesis work. Next, we tried to build some programs, and we succeed in getting it to follow the black line in the form of a (∞) line. It depends mostly on sensors that are connected to the underside of the car to decide the path or the direction. Then we were looking for something more exciting. We have created a program which following the line and avoid the barriers, the robot can avoid any barrier in its way by using the ultrasonic sensor also following the black line by using the following line sensor and the decision is based on these sensors.

Finally, we have created more interesting programs like do not cross the line, random movement without making an accident and mBot dancing, all these programs were an experiment of making a good decision based on some rules and sensors.

Keywords: Autonomous Vehicle, Artificial Intelligence, Self-Driving, Internet of Things, mBot, Arduino, Machine learning, Deep learning.

Table of Contents

List of pictures

List of tables

List of abbreviations

- **Artificial Intelligence (AI)**
- **Autonomous vehicles (AVs)**
- **Autonomous vehicle (AV)**
- **Connected Vehicles (CVs)**
- **Machine Learning (ML)**
- **Wide Area Network (WAN)**
- **Wireless Sensor Network (WSN)**
- **Local Area Network (LAN)**
- **Advanced Driver Assistance Systems (ADAS)**
- **Defense Advanced Research Projects Agency (DARPA)**
- **National Highway Traffic Safety Administration (NHTSA)**
- **Dedicated Short-Range Communications (DSRC)**
- **Inertial Measuring Unit (IMU)**
- **Global Navigation Satellite System (GNSS)**
- **Kalman Filtering (KF)**
- **Weighted Averaging (WA**
- **Dempster-Shafer Evidential Reasoning (DSER**
- **Naive Bayesian Inference (NBI)**
- **Artificial Neural Networks (ANN)**
- **Fuzzy logic (FL)**
- **Decision Trees (DT)**
- **Support Vector Machines (SVM)**
- **Discriminant Analysis (DA)**
- **Density Estimation (DE)**
- **Electronic Stability Control (ESC)**
- **Airbag Control (AC)**
- **Traction Control Systems (TCS)**
- **Anti-lock Braking Systems (ABS)**
- **Perception, Localization, Planning, Control and Prediction (PLPCP)**
- **Neural Networks (NNs)**
- **Deep Learning (DL)**
- **Supervised learning (SL)**
- **Unsupervised learning (UL)**
- **Reinforcement learning (RL)**
- **Internet of Things (IoT)**
- **Machine-to-Machine (M2M)**

1 Introduction

Autonomous vehicles are cars, buses, motorcycles, and trucks in which, never human drivers need to take care of the car for safe running or controlling the vehicle. Often referred to as automated or "driverless" vehicles. Cameras, radar, GPS system, artificial intelligence (AI), sensors and software are combined to monitor, navigate, control, and drive the vehicle between destinations.[1]

Autonomous vehicle (AV) technology provides the prospect of transportation being radically modified. Autos and light vehicles are fitted. It would potentially minimize collisions, energy usage with this technology, both pollution — congestion are minimized.[2]

The aim of this thesis is to research how the recent AI and IoT approaches will support the AV. Human accidents have been shown to cause 90% of car collisions and also the safest drivers are ten times higher than the average [3]. Automated protection is substantial, and consumers require a reasonable degree of risk 1000 times smaller. Several of AVs' unbelievable advantages are: (1) enhancing safety of cars, (2) reducing collisions, (3) reducing fuel consumption, (4) freeing driver time and investment opportunities, (5) new markets for the industry and (6) emissions and pollution particulate matter minimized [4].

Due to the increasing usage of connected vehicles (CVs), this data is forecastle increased further. The AV 's growth gives industrial manufacturers and distributors new opportunities, which allows businesses to use AI to boost their client's value. When AI is processing these information \ data, Machine Learning (ML) algorithms are the most effective solution. The ML algorithms help to shape behavioral models for some driver profiles and also give vehicle owners with a corresponding application precisely what they want or need on the vehicle and via their smart phones. They achieve this by observing their actions and analyzing their driving experience as well as the scenario on the road [4].

Although AI can interact with the AV big data, certain additional data criteria, including traffic, experiences and pedestrians, need to be obtained across different IoT networks; For e.g., Wide Area Network (WAN), Wireless Sensor Network (WSN) or Local Area Network (LAN). This massive volume of information / data requires some resources, such as built-in electronic devices, sensors, buildings, applications and network access, to gather and distribute the data. These IoT-enabled AVs use a range of interconnected technologies to provide various real-time supports such as safety improvement, fuel consumption reduction, and security for the vehicle. IoT as well as the automobile industry 4.0 are being transformed to offer a great boost by minimizing machine failure, strengthening the quality control, increasing efficiency and simultaneously reducing costs. The promise of IoT technologies and the predictions are impressive [4], [5].

2 Objectives and Methodology

2.1 Objectives

This thesis main goals to firstly provide an overview of autonomous vehicles (AV) and its applications and how robots can implement the driverless by using Artificial Intelligence (AI) and Machine learning (ML). Many robots are working in the same field, but we will implement a special robot called mBot for our experiment.

Regarding to mBot module we will be using Arduino as an electronic prototyping framework for open-source users to create our electronic interactive project.

Furthermore, we want to cover the most important topics and give some basic information about AV, AI, ML and IoT in order to make readers more comfortable and better understand to the relevant technologies.

The last objective will be establishing some different driving modes for the robot to give a real example of AV. Moreover, we will make some recommendations based on our experiment and the practical part.

2.2 Methodology

The literature review would be the first part of this thesis, which defines some key concepts such as AV, AI, ML, DL, IoT, Arduino and mBot.

In addition, we want the reader to know what we are going to do with this hypothesis by literature review and to clarify the fundamental principles used more in this thesis.

After we get know the principles and the most important concepts, we will have an introduction to our mBot that will be used in the practical section. In addition, we can clarify a general description of how we believe AI and IoT should be used.

The practical section is the final part of this thesis which will be the implementation of AV by using mBot robot, and there will be 5 different driving modes to test AV by using Arduino framework.

3 Literature Review

The section of the literature review will cover the necessary details that will assist with the practical part. It will cover the foundation of autonomous vehicles, beginning with general details about this project, which will then cover other fields like what are AI, AV, IoT, Arduino and mBot , also how ML is working to improve AV for making a good decision. In addition, the introduction of ML and AI, like mBot, will be covered. This literature review covers the effects of AV in the industrial field and will help us appreciate the need or not for AV in new-generation vehicles.

3.1 Autonomous Vehicle

An autonomous vehicle is a vehicle that detects the environment and operates without human involvement. A human passenger is not necessary at any time to take control of the vehicle, nor is a human passenger expected to be present in the vehicle at all. An autonomous vehicle could go anywhere and could do whatever a successful human driver would do. [6]

The AV has grown into a kind of driverless vehicle which is the driving ability of computers for the future. AVs is aimed except that: (1) increased vehicle safety, (2) reduced crash uncertainty, (3) reduced fuel consumption, (4) freed driver time and business prospects, (5) new opportunities for the industry, and (6) decreased emissions and dust particles. Around 10 million AVs are scheduled to be in the roads by 2020 and it is predicted that AVs will generate $7 trillion annual income by 2050 [7].

In general, autonomous mobile navigation is needed to locate the vehicle autonomously, (1) to locate the vehicle, (2) to create the maps, (3) to design the path and (4) to follow the path. Additionally, AV obstacle avoidance is required via detection and classification [4].

Some of the major issues with AVs : (1) Software accuracy and anti-crash software is necessary to ensure that no problems occur, (2) Map completeness and correctness by enhancing map functionality with some additional knowledge, including recognition of nearby objects, creation of virtual maps to help AVs find the best path and look at dynamic obstacles. (3) Sensor fusion and calculations for sensing various unexpected calibration conditions are important in order to differentiate between very dangerous and less dangerous circumstances [4].

AV information / data used in which Advanced Driver Assistance Systems (ADAS) and entertainment is increasingly complex. Therefore, it is imperative to develop hardware and software requirements, which utilize sensors, actuator hardware and software, to compete for similar superhuman functions as targeted by AI. AV sensors and devices generate information like time , date, motion identification, navigation, fuel usage, voice recognition, accelerated speed, deceleration, accumulated mileage, voice search, recommendations engines, eye tracking, picture recognition, feeling analyzes, voiced and gestured recognition, and virtual support. Therefore, the cumulative data for 100,000 cars are 100 terabytes per year [8], [9].

3.1.1 Brief history of autonomous vehicles

The development of Autonomous Vehicles (AVs) originated in the 1930s, when science fiction authors visualized and created self-driving cars to be a new challenge for the entire automotive industry [4].

Between 2003 and 2007, Three 'Grand Competitions' were held by the United States Defense Advanced Research Projects Agency (DARPA), which greatly stimulated developments in AV technology. The first and second occurred in rural areas, the third occurred in an urban area. Each one facilitated technology development by university teams.[2]

Private companies have built more recently AVs. A fleet of vehicles has been built and tested by Google's Driverless Car project and promotions have been introduced to highlight the technology applications the vehicles of Google, operating completely autonomously, have driven more over 500,000 miles without an automation-related crash.

In the near future, AV can achieve excellent human efficiency by using sensing algorithms for compulsory driving skills. Intelligent perception is similar to executing human activities such as AV identification, localization, direction tracking. A recent study forecasts that AVs will be broadly adopted by 2020 and that the adoption of AV efficiency will not be limited to individual transportation [10]. Since 2016, some countries such as the USA (Nevada, Florida, Michigan, CA), Canada, France, UK, and Switzerland have approved several public road test legislation and regulations [4]. **Table 1** offers a quick history of AV.

Year	Companies/projects	Activity
1925	Houdina Radio Control	Demonstrates a radio-controlled ‘driverless’ car
1939	General Motors	Exhibit ‘Futurama’ model
1949	RCA	Begin the technical explorations
1950s	General Motors /RCA	Research collaborative a large project
1950s	General Motors	The concept car called Firebird II
1956	General Motors	The Firebird II exhibited is equipped with receivers for detector circuits in roadways
1958	Chrysler	The first car with cruise control called imperial
1960s	Kikuchi and Matsumoto	Wire following in Japan
1964	General Motors	Futurama II exhibit
1964	OSU	Research by Fenton
1970s	Tsugawa	Vision guidance in Japan
1979	Stanford Cart	Used a video processing to navigate a cluttered room without human input
1980s	Dickmanns	Vision guidance in Germany
1986	California PATH and PROMETHEUS	Programs start
1994	PROMETHEUS	Demo in Paris
1995	VaMP	Autonomous vehicle drivers (almost) completely autonomously for 2000 km
1995–1998	National AHS Consortium	Demo ‘97
2003	PATH	Automated bus and truck demos
2004–2007	DARPA	Grand challenges are founded to incentivize autonomous vehicle development
2009	Google	Self-driving car project begins
2015	Tesla	Release its Autopilot software update
2016	Google	Self-driving car has its accident
2017	General Motors	Plans to include autonomous controls in the Bolt and Super cruise in Cadillac Ct6
2017	Volvo	Plans to launch 100 self-driving vehicles to customers

Table 1 A short overview of autonomous driving by separate research and development projects, Taken from [4]

3.1.2 Autonomous vehicle levels

The technology of autonomous vehicles (AV) provides for fundamental transportation changes. It is possible that the use of this technology to equip cars and light vehicles will minimize collisions, energy, pollution and minimize congestion.

Nearly every car accident is caused by human error, which Advanced Driver Assistance Systems (ADAS) can prevent. ADAS 'job is to reduce collisions and casualties by reducing the number of incidents involving vehicles and the significant effects of incidents that cannot be prevented.[11]

The National Highway Traffic Safety Administration's proposed spectrum (NHTSA), with numerous technological advantages that have been realized at various automation levels:

Level 0: All car operations are entirely controlled by the human driver. The driver is always entirely and primarily responsible for controlling the roads and for the proper running of the vehicle.

Level 1: Automation of one function. At this step, the automation requires one or more basic control functions, which are performed independently of each other while many functions are automated, but still it requires the driver control and attention.

Level 2: This includes specialized driver management services or ADAS. The vehicle can control both the direction and distance. The automation here is less automated since a person is seated in the driver's seat and it is able to monitor the vehicle at any moment. At the same time, more than one function (e.g., steering and acceleration) is programmed, but the driver must still be cautious.

Level 3: Vehicles are capable of "environmental detection" and can make educated choices, such as accelerating a slow-moving car. However, human override is still required. If the machine cannot accomplish the mission, the driver must stay prepared and ready to control. The driving functions are adequate to enable the driver to perform other tasks in a safe manner.

Level 4: Vehicles can be operated in self-driving mode. Yet only a small region (usually at urban area where high speeds exceed 30 mph in average) can be achieved before law and technology develops. This is called geofencing. So, we can see in this level the vehicle has the ability to drive itself, but not in crowded areas.

Level 5: Vehicles do not need human attention; the vehicle will drive itself. Fully autonomous vehicles are being tested in several pockets around the entire world, but none of them are still open to the public. The vehicle is equipped to perform both safety-critical driving and highway conditions tracking functions for the whole ride.

The type and scale of possible advantages of AV technology would depend on the degree of automation. For example, functional specific automation (e.g. automated barking) may provide some safety advantages to AV technology. Although the advantages of land use and the environment are probably to be achieved only by full automation (Level 5).

Researchers are predicting that around 8 million independent or semi-autonomous cars are going to take to the road by the year 2025. Until entering highways, self-driving vehicles would have to move through six stages of technical improvements in driver assistance. [12]

Figure 1 is showing the driving automation levels with some details about each level.

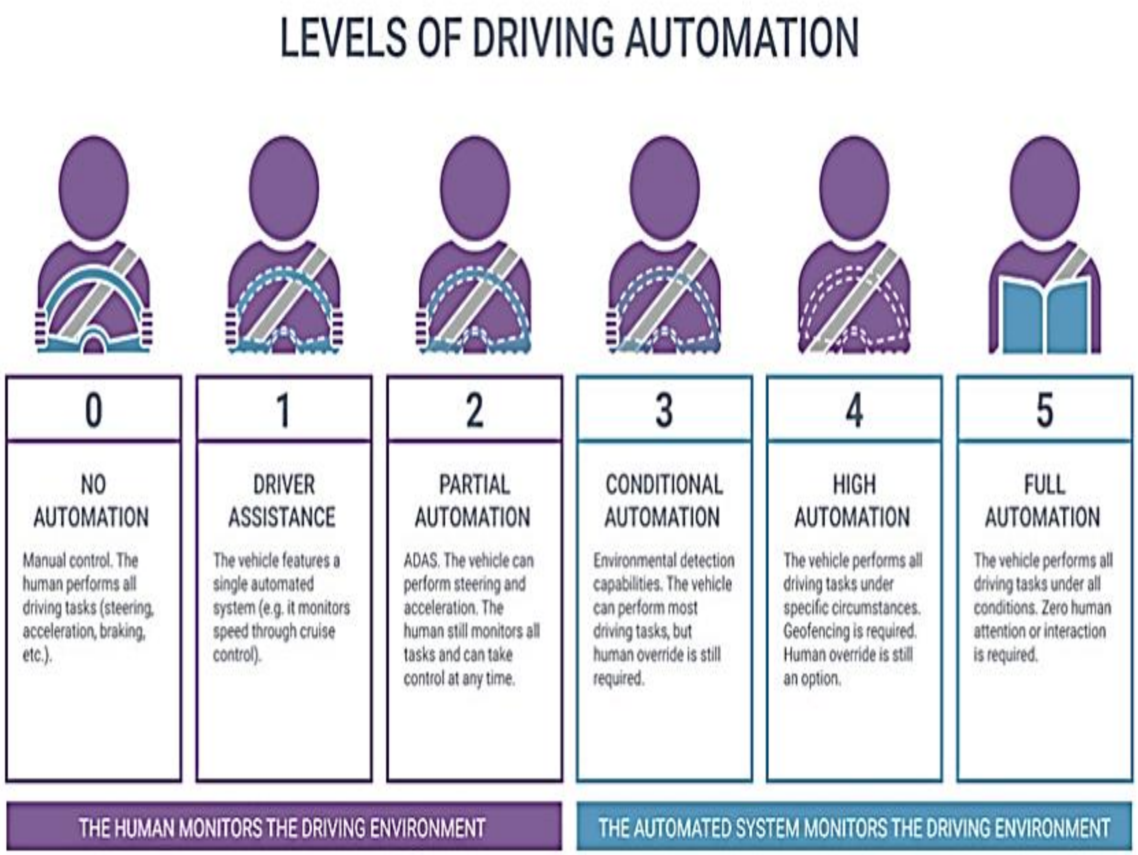

Figure 1 Autonomous Vehicle levels [12]

3.1.3 Sensors Used in Autonomous Vehicle Systems

For a vehicle, sensors and computer systems are expected to simulate its actual position, how the routes, location and dangers can be securely navigated between them. Among all the sensors available today, no single device is capable of decoding the environment in such a way as to provide accurate information for autonomous vehicles. [13] Therefore, engineers design a system that involves a combination of sensors, which leverages strategically the unique capacities of one system while adapting the shortcomings to others. In below, the most frequent sensors used in driver devices and autonomous systems are explained in **Figure 2**:

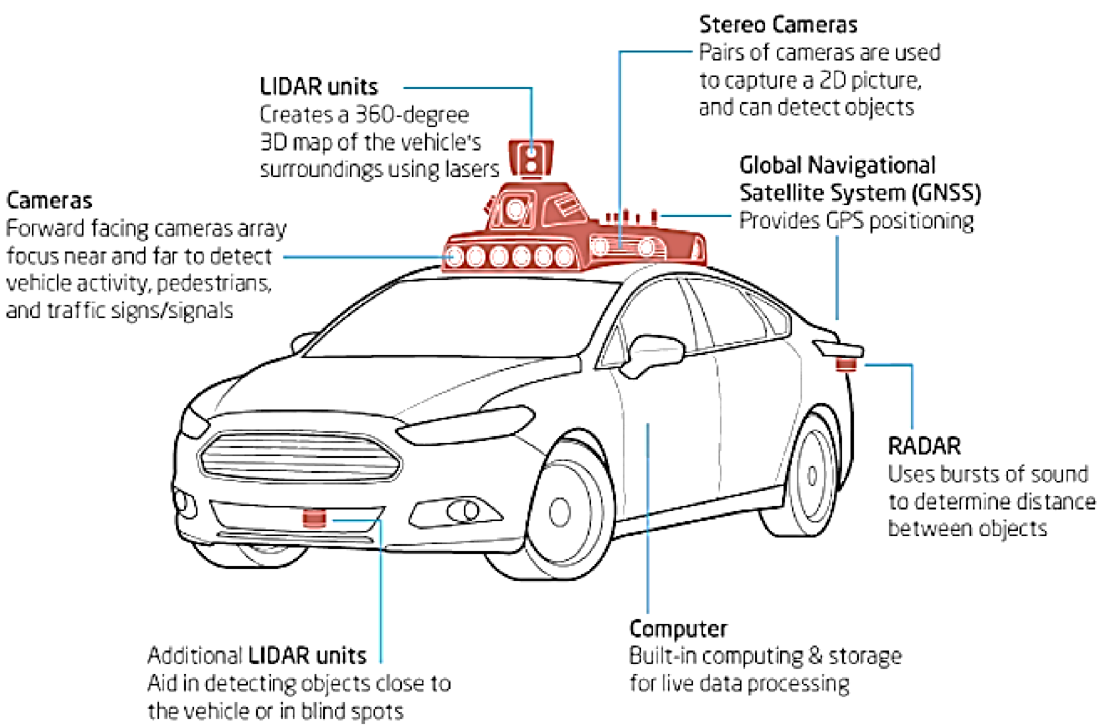

Figure 2 Vehicle sensors[34]

Radar: It works by using sound bursts to measure distance by tracking when the sound comes back to the sensor. Also, it is efficient for calculating relative speed. The sensor frees radio waves to detect short and long depth. The location of the vehicles nearby is tracked by radar sensors pointing around the car.

LIDAR: similarly, like radar, but it utilizes lasers to measure distance rather than sound with a range of up to 200 m. In addition, an exact 3D map of the vehicle area is generated. LiDAR sensors are used to sense road borders and distinguish lane markers through active light pulses from the environments in the car.

Video Cameras: Records several pictures (frames) to record a 3D dimensional image, provides color, comparison, and identification of optical character. Cameras recognize the real-time obstacle to allow lane departure and monitor roadway details. An image produced by the camera comprises a wide variety of individual pixel values, but these numbers are not important. The picture must also be understood by using computer vision algorithms to translate the low-level information to high-level image information.

Computer vision involves analyzing signals from: (a) cameras, (b) thermal sensors, (c) laser rangefinders, and (d) x-ray detectors. Computer vision is made up of three components: (1) Segmentation of real objects, (2) classification of objects, and (3) 3D reconstruction is estimation of 2D images [4].

Ultrasonic Range Sensors: It is a measurement device using ultrasonic waves to determine the distance between objects. An Ultrasonic Sensor uses an ultrasonic pulse to transmit and receive information about the proximity of an object. Also, pretty precise on short distances. [14]

Inertial Measuring Unit (IMU): It provides feedback on the real movement of the vehicle, intelligently merged to ensure a consistent location and distance calculation for the stability and navigation applications, the multi-axis sensors combine precision gyroscopes, accelerometers, and press sensors. [15]

Global Navigation Satellite System (GNSS): Used for autonomous geospatial positioning by satellites. A sub-set of GNSS is the Global Positioning System (GPS). Provides the positioning of unidentified vehicles. Positioning provides where there are no road markers or signals visible

Dedicated Short-Range Communications (DSRC): This is a one-way or two-ways short distance in wireless medium-range channels made specifically for vehicle use, along with a standardized set of standards and protocols. DSRC can use connectivity from Vehicle to Infrastructure (V2I), Vehicle-to-Vehicle (V2V), and Vehicle-to-Everything X (V2X) as 4 G, Wi-Fi, Bluetooth, etc. The required system is configured to have the lowest latency [4].

Global Positioning System (GPS): Global Positioning System Triangulates the traveling position of the receiver: the vehicle position with satellites can be measured for the latitude, longitude, altitude as well as speed and direction of motion. Present GPS is limited to some distance [4].

Accelerometers: A version used to calculate location shifts (x, y, z) and force F.

Wheel Odometry: It measures changes in the 2D direction (x, y, θ) of the steering angle and speed of the vehicle

Gyroscopes: Tests one, two, or three-degrees rotation. It measures (ϕ, θ, ψ) by summarizing rotations of the gyroscope.

Vehicle Dynamics: While position sensors to be built depend solely on vehicle motion kinematics, a dynamic model is needed to validate the efficiency of the vehicle [16], [17].

3.1.4 Data Fusion

As described above, the perception of autonomous vehicles (AV) in a regional situation largely depends on the extraction of information from a number of sensors (for example, camera, LiDAR, radar, ultrasonic) with their own operating conditions (for example, illumination, range, electric power). How to integrate locally sensed data to help a particular decision task such as vehicle detection has become one of the open challenges in the restoration and comprehension of the AV environment [18].

Sensor fusion is a software to merge multi-sensor data to better assess the system performance enhancement. The vehicle position will be calculated accurately and the direction integrating data from those separate sensors [19].

The strategies of data fusion can be defined as:

Estimation: The approach executes the evaluation task optimally with a precisely defined statistical framework [20]. like: (1) Kalman Filtering (KF) and (2) Weighted Averaging (WA) [21].

Classification: It may be used to resolve problems of classification. The challenge is to divide a multi-dimensional space between regions, each representing a particular class (group) like: (1) Decision Trees (DT) , (2) K-Nearest Neighbor, (3) Support Vector Machines (SVM), (4) Discriminant Analysis (DA), (5) Density Estimation (DE) [22].

Inference: This approach creates another type of probability theory-based fusion techniques like: (1) Dempster-Shafer Evidential Reasoning (DSER), (2) Naive Bayesian Inference (NBI) [22].

Artificial intelligence: This is based on heuristic methods like:(1) Artificial Neural Networks (ANN), (2) Fuzzy logic (FL) [23].

3.1.5 The new development of autonomous vehicles

New cars are using a wide range of sensing abilities. Moderately, a car actually has 70 sensors with atmospheric sensors, accelerometers, gyroscopes and humidity sensors in the USA cars are using a wide range of sensing abilities [24]. Vehicle sensors, which were developed before 2000, are not modern components; sensors like wheel orientation, acceleration and speed have been used for the internal condition of the vehicle. Vehicles already have a range of capabilities for integrating real-time sensing with vision and decision-making, like Electronic Stability Control (ESC), Airbag Control (AC), Traction Control Systems (TCS), and Anti-lock Braking Systems (ABS) [4].

In **Table 2** summarizes the new commercialized automated competency features.

Context	Automated functionality	Date
Parking	Intelligent Parking Assist System	Since 2003
Parking	Summon	Since 2016
Arterial & Highway	Lane departure system	Since 2004 in North America
Arterial & Highway	Adaptive cruise control	Since 2005 in North America
Highway	Blind spot monitoring	2007
Highway	Lane changing	2015

Table 2 Latest automated functions for AV [24]

Automated features support drivers or take fully specified steps to improve comfort and safety. Today's vehicles can operate adaptive cruise control mostly on highways, park their own, warn drivers on objects within blind spots and command themselves during traffic stop-and-go [24].

The automobile industry has been actively trying to improve AVs over the last few years. Approximately 46 private companies are active in AV automotive technology. A Gartner study predicted that approximately 250 million vehicles are likely to be connected to all vehicles by 2020 [25].

The following are a short list of some of the latest vehicle hardware and software developments [26]:

1- “The first autonomous, fully-electro-ship to be built on a public road in the US has been launched by Keolis and NAVYA (2017), in cooperation with the City of Las Vegas (2017).”

2- “Toyota (2018) launches the prototype car "e-Palette," a fully electric , AV that can be individualized by a partner to be used in applications like the distribution of food (Pizza Hut), the ride-sharing (Uber) or shopping fronts (Amazon)."

3- "Udelv, (2018) a tech company in the Bay Area, achieved the first shipment of goods by an autonomous vehicle, when it delivered grocery goods to San Mateo."

4- "Hyundai (2018) revealed that it had taken a fully effective autonomous trip from Seoul to PyeongChang from a fleet of its cell electric vehicles. This is the first time an electric fuel cell vehicle from level 4 has driven."

3.1.6 Artificial Intelligence in Autonomous Vehicle

An Autonomous Vehicle by Artificial Intelligence (AI) model consists of three steps: (1) data collection, (2) path planning, (3) act. The following are some description of these steps [4].

1- **Data Collection:** The AVs are fitted with multi-sensors like radars, cameras and connectivity and devices for generating big data from their vehicles and environments. These AV data contain the roads, road facilities, other vehicles and all other items placed on / close to the road, parking, traffic information, transmission and information statistics on the environment. These data would then be submitted as AV modified information to be analyzed. This is the first AV communication in particular situations in the vehicle and in the environment [4].

2- **Path Planning:** The massive data of AV system will be stored and added from each ride in a database named by Big Data with previous driving experiences. An AI agent also works on big data in order to make meaningful choices by strategy control. The control strategy for AV trajectory planning allows self-driving vehicles to determine the best, most efficient and most economically useful roads between point A and B, leveraging previous driving experience that will allow the AI agent to decide even more effectively in future [4].

3- **Act:** In this step basing the AI Agent's decisions, the AV can recognize road objects, road maneuvers, parking spaces, barriers, entertainments, traffic lights, motorcycles, walkers, offices, atmospheric conditions and other vehicles without interposing the driver and can be safely reached. Furthermore, AVs are fitted with AI-based control and operating systems, like steering controller, pedaling acceleration, voice and image recognition, brake pedal controller, object detection, protection systems , control of movements, cost-efficient fuel, some driving / monitoring systems and a lot more. These AV process loops can take place regularly including data collection, path planning and act [4].

3.1.7 challenges for Autonomous Vehicles

Autonomous vehicles (AVs), after years of intensive study and development, AVs are today's reality. Nevertheless, the entire architecture of self-driving vehicle systems needs for engineering, regulatory technology, traditional manufacturing technology and resources[4]. Significant challenges, such as lack and customers, confidence and acceptance, e.g. However, among those challenges, engineering technologies still need to be improved, especially in respect to data / information, Perception, Localization, Planning, Control and Prediction (PLPCP) for following conditions / zones [27]:

Road Conditions: Road conditions are highly evolving and differ from one stage to the next. Smooth and numbered large roads are available in some places. However, road conditions in some other places are heavily damaged — no markings in the street.

Weather Conditions: Another spoilsport is playing in atmospheric conditions. Sunny, windy, rainy, or snowy, stormy weather may occur. In all kinds of weather situations, the AVs should operate. The scope for failures or downtimes is completely lacking.

Traffic Conditions: In all kinds of traffic conditions, the AVs will have to go on the road. The vehicles would have to move on the road with the other AVs, because there would be several pedestrians at the same time. Wherever people are involved, there are multiple emotions. Transport can be strongly moderated and self-regulated. But still there are also situations in which road laws are broken by the people. An object may appear in unforeseen circumstances.

Accident Liability: Responsibility for accidents are the most significant part of AVs. Who is blamed for AV accidents? For AVs, the software is the primary driving power for the vehicle and makes all the critical decisions.

Radar Interference: For navigation, the AVs are using lasers and radar. The lasers are fixed to the roof, while the sensors are connected to the vehicle's frame. The radar theory operates by detecting radio wave-reflections from objects around it. Where the vehicle is on the road, radio frequency waves are constantly produced that mirror the vehicles and some other objects along the road. Reverse time is calculated to quantify the distance between the vehicle and the object. Effective action is then taken on the basis of radar readings. Will a vehicle be able to differentiate among its own signal as well as the signal of some other vehicles, as this technology is being used by thousands of road vehicles?

Big Data Analytics: Training and the decision-making mechanisms of the AV data volume must be applied in the real time. Innovative technologies can be significantly slowed down without effective data processing. Explore the four data aspects in the AV: (1) acquisition of data, (2) storage of data, (3) management of data and (4) labeling of data [28].

Vehicular Communication: AVs need an Internet connectivity network with big data / information’s in order to solve PLPCP: (1) energy-efficiency powertrain side. (2) environment side (3) vehicle side.

3.2 Artificial Intelligence and Machine learning

Artificial Intelligence (AI) and Machine Learning (ML) have automated solutions for tasks typically mastered mostly by humans because of their complexity, the inefficiency of standard algorithm solutions, and the lack of ideas on alternatives to solutions for architecture and programming[29]. Artificial Intelligence (AI) Solutions vary markedly from older standard programming solutions which do not require pre-known logic statements as well as a thorough understanding of their applications and rules.

Machine Learning (ML) trains learning AIs in creating patterns for different activities. Although traditional programs require hard code logics ML depend heavily on the data for learning. Deep Learning, which is able to train complicated model presentations to do comparatively challenging things, is one specific approach to ML. Overview and the associated constructs in **Figure 3** of the AI, ML and Deep learning.

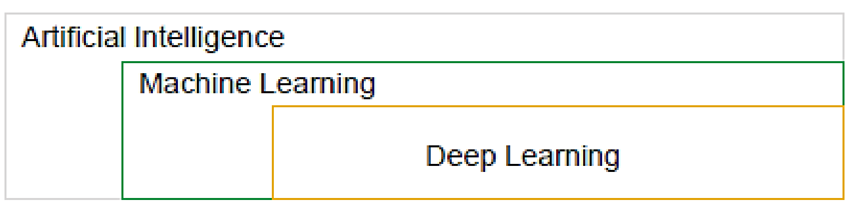

Figure 3 Classification of AI, ML and Deep Learning [4]

These fields will be discussed in the following paragraphs. It is important to highlight these topics for this master thesis before start talking about mBot robot and the practical part of the thesis.

3.2.1 Artificial Intelligence

Artificial intelligence (AI) is a large IT division devoted to constructing intelligent machines that carry out tasks usually requiring human intelligence. AI is a field of science with a range of approaches, but progression of deep learning and machine learning produces a paradigm change of almost every field of technology [4]. Moreover, AI is a computer science and engineering field that is used for the development of intelligent devices for various smart applications. AI operates and responds intelligently and individually like human beings by learning from experience and adapting itself to new participations [30], [31].

AI is a machine intelligence platform that provides enormous opportunities for the intelligent industrial revolution. It encourages the compilation of relevant information/data, recognition of alternatives, choice of alternatives, decision making, evaluating decision-making and smartly forecasting [32].

AI may be described as an abstract approach for seeking new or more useful solutions, human processes, or natural methods to solve problems [32]. Artificial intelligence is thus an abstract principle that can be used and applied by an intelligent agent. An agent is a system that operates with the managing activities of a host (in this case autonomous).

In the subject of AI, training agents who are capable of learning to enhance their performance have special reach. Their intelligence is also not founded on a predefined logic, but on a machine learning of massive data sets. As shown in **Figure 4**, an autonomous vehicle may be viewed as a hardware agent. They are fitted with sensors for environmental perception and drives to change their atmosphere or their own state. They observe their environments and check for other vehicle, pedestrians, or objects. The interior software agent is guided like an ideal goal by human feedback and schedules the next actions and determines a professional journey [13]. Then actors will speed up and monitor the vehicle to meet the driver's interest.

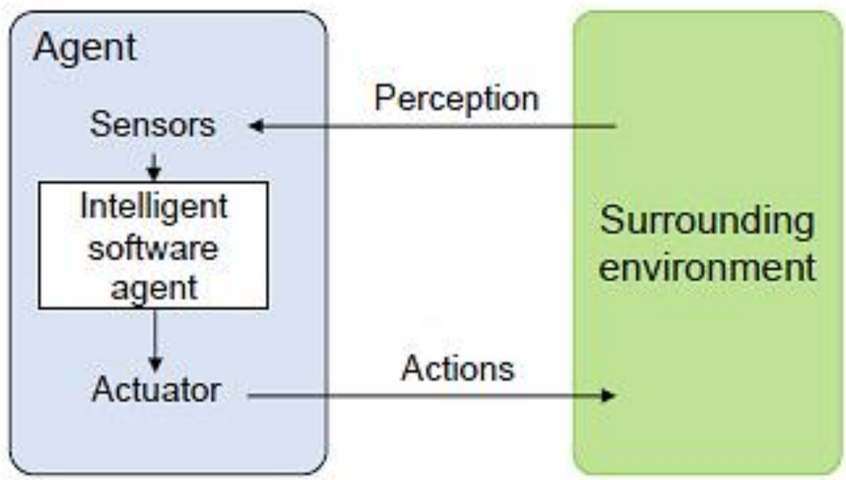

Figure 4 Hardware agent [33] [34], [35]

3.2.1.1 Benefits and Challenges of Artificial intelligence

The pattern of industrial revolution including technologies, automation and the sharing of data is shown in the **Figure 5**. New challenges in terms of competitiveness and demand are confronting traditional markets and drastic improvements to innovations in Industry 4.0.

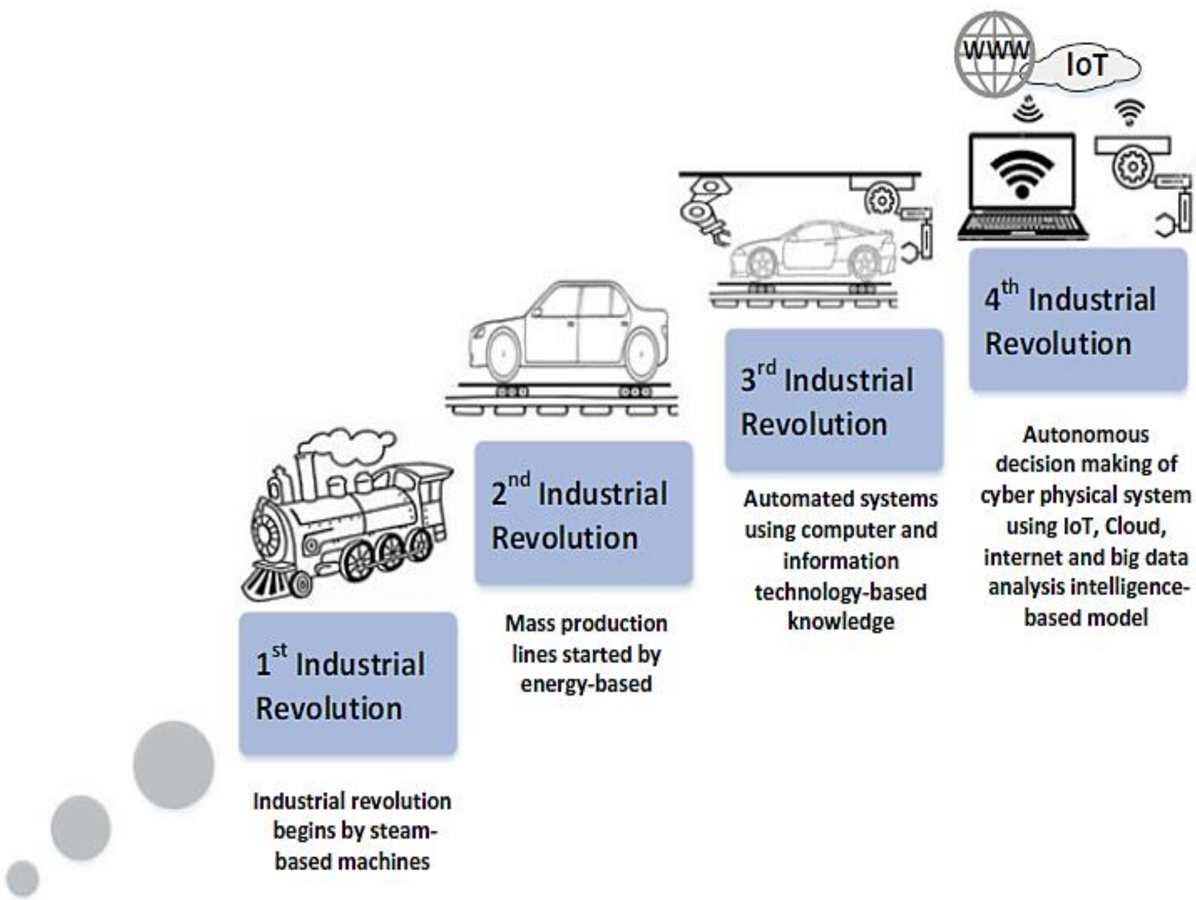

Figure 5 The fourth revolution of the industry [4]

Artificial Intelligence (AI) is one of the features that facilitates enhanced decision-making dynamics and better decision-making in industry 4.0 leading to improve the efficiency of business, reduced system failure, improved quality control, improved competitiveness, and reduced costs. AI provides many benefits like: (1) data use for automatic learning, (2) intelligence improvement in existing product, (3) smart learning algorithms are tailored to do some programming for the data, (4) data rationally analyzing and, (5) data accuracy improvement [36]. While AI is most likely to change the world today, it is limited by itself. The main challenges for AI are learning from experience, and information cannot be integrated into the learning experience. Moreover, any inexactness in data are particularly challenging and mirrored in the results.

3.2.1.2 Artificial Intelligence History and Approaches

AI is focused on the combination of huge amounts of data. Also, with intelligent algorithms, it can process data very rapidly with iterative analysis, which helps the machine to learn about characteristics or patterns within the data. AI has been more common as the vision error (less than 5 percent) decreased significantly recently relative to human vision error, as seen in the **Figure 6** [4], [36].

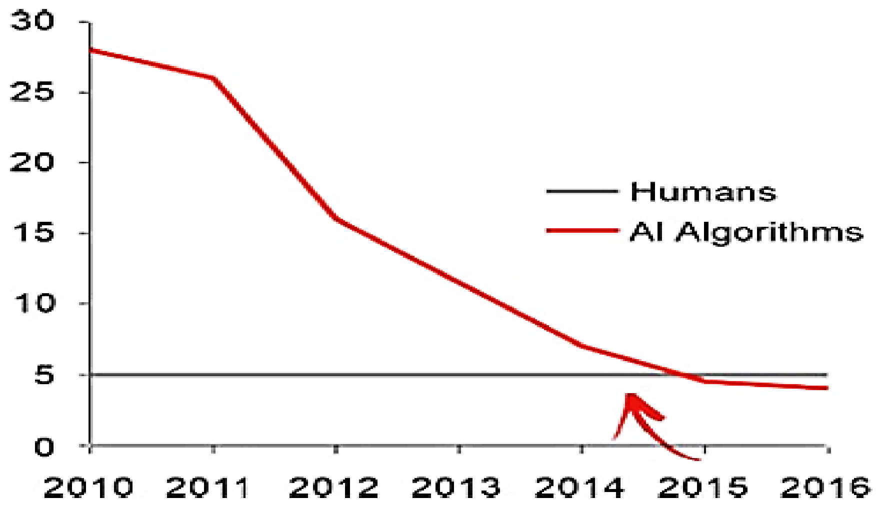

Figure 6 The rate (percent) of vision error from human and AI algorithms [4]

AI 's history started in ancient times, but John McCarthy initiated it in the 1950's. A brief development of AI is seen in **Figure 7** [4].

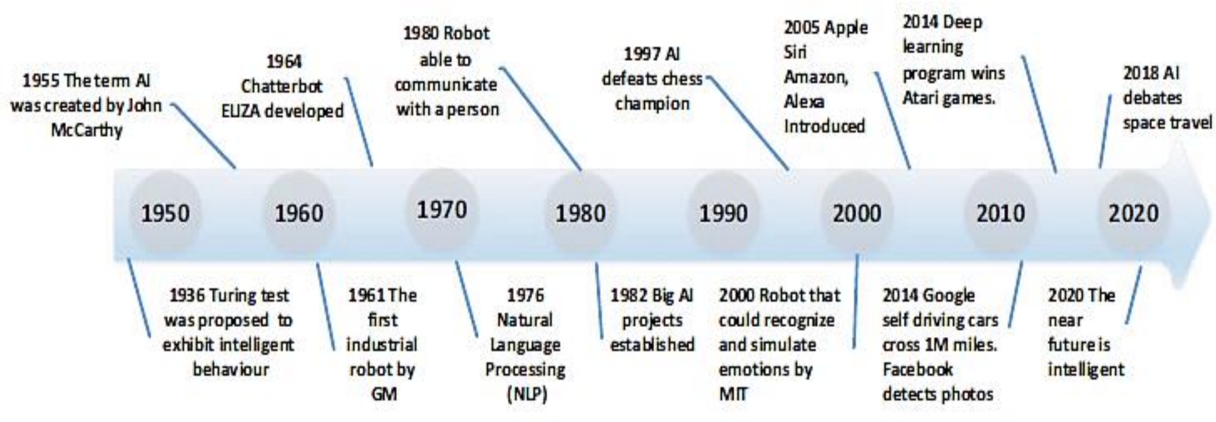

Figure 7 Artificial Intelligence (AI) scheme evolutionary diagram [4]

AI was inventing in three major areas: 1) Neural Networks (NNs), which were primarily focused on arouses excitement 'thinking machines' between the 1950s and the 1970s, 2) Machine Learning (ML) which became common AI approach, and 3) Deep Learning (DL) which is leading the breakthrough of this current decade. AI approaches diagram is shown in the **Figure 8** [4].

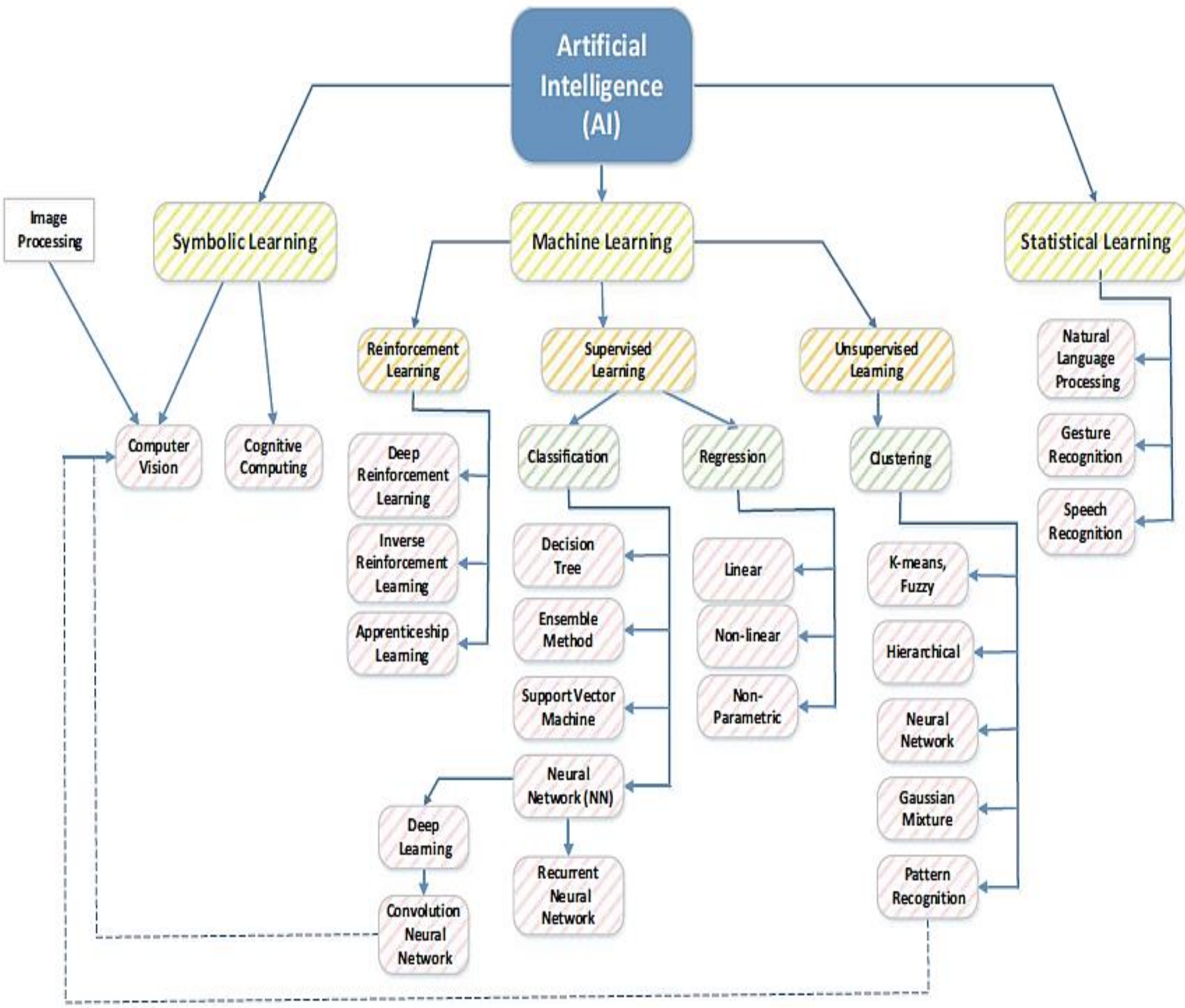

Figure 8 Approaches / apparatuses in artificial intelligence [4]

As can be shown above, AI is typically divided into three major areas: Machine learning (ML), Symbolic Learning and Statistical Learning. Here is a brief explanation about these three fields:

Symbolic Learning: Symbolic learning depends on human readable logic symbols that are generated by human intervention, challenges, search, and Symbolic learning rules. Mixtures of symbols were considered thinking through their interrelationships. In order to create a symbolic reasoning, humans begin to study the rules of the interrelationships of the phenomenon and then transfer their code into a program. It is possible to split symbolic learning into cognitive computation and computer vision [4], [37].

Statistical Learning: Statistical learning is an intense in mathematics and it works with the issue of discovering a data-based predictive function. Before constructing a model, we have to create a hypothesis. Statistical learning is dependent on rules based on some programming and formalized throughout the form of a relation between variables. Statistical learning itself is based upon a smaller dataset, with some properties, acting on assumptions like normality, no cross-linearity and homoscedasticity [38].

3.2.2 Machine learning

Machine learning is an artificial intelligence technology (AI), which offers systems the capability to learn and develop automatically from experience without expressly scripting them. Machine learning is targeted at developing computer software that can view and use data by itself. ML constructs and automates analytical / numerical models and algorithms which could be used in a specific role to boost the effectiveness of the algorithm [39].

The automotive industry's future will be characterized by machine learning (ML) and self-driving vehicles. It is no wonder that they are a dream pair. Algorithms for machine learning are most widely used for interpretation and decision-making in autonomous vehicles (AV). [40]

Many challenges, like driving, are fairly well performed by humanity, but computer software is difficult to master. Machine Learning is often used in place of traditional software where a task has the following characteristics [4].

To be manually coded, the task is too complex: Several laws are contained in the Road Traffic Law that have to be enforced when addressing a traffic condition on public roads. Although it is reasonably straightforward to understand each of these rules, for each one, the complexity of the whole system grows.

There is no excellent method to solve the problem: it is getting harder to find general solutions while considering different types of issues that can not be split into more workable sub-tasks. Identifying objects in images can be a specific example of this issue. Thus, it may be easy for humans to define the characteristics of the directed objects, but it is difficult to find simple sequential algorithmic methods to computationally implement the solution.

Intelligence must communicate in a vector and thus dynamic environment: This indicates that it is not possible to apply the method to all possible scenarios in advance. As a result, hardcoded and predefined rules and standards will not fulfill the high specification criteria of the entire product lifecycle. Machine Learning training rules in this case have two benefits. Well trained models of machine learning generalize well in unknown data, which means that no condition is required in advance. Similarity is nevertheless a requirement [31], [41]. In addition, by gathering feedback and constantly refining their internal models, reinforcement learners such as Q-learning algorithms are able to improve while they are used.

Therefore, machine learning offers decisive benefits when contemplating a self-driving program. The need for extra security precautions is a disadvantage that can not be leave unmentioned. ML models are useful to generalize and overcome related examples that are not used but appear to respond unpredictably to different details. In comparison, errors within a Machine Learning model are challenging to identify because they can be viewed as an abstract or black box. Machine learning may nevertheless be held in consideration, not only due to recent autonomous driving achievements.

It can be subdivided into three major forms of machine learning. Supervised learning (SL), Unsupervised learning (UL) and Reinforcement learning (RL). Which explained:

First, Unsupervised learning is a set of data that is focused entirely on input knowledge. An aggregation of data points into a sequence of data points is among the Unsupervised learning strategies [42], [43]. labelled training data containing an action input which is connected to the right result (Ground Truth), [4]. Then, in behind set of input and outcome, a supervised learner seeks operations and rules to predict more unknown tasks, identical to those it was trained on [44], [45].

Then we have, Unsupervised learning (UL) just involves unsolved data to identify trends (dimensional reduction) or clusters (cluster analysis) inside the dataset provided. Supervised learning (SL) builds an input and output-based paradigm for prediction. SL methods can be split into: (1) Regression that is an application for the interrelationship of variables. In machine learning, this is used to forecast the outcomes of an occurrence based on the relationship between data sets variables. (2) a classification that tries to determine precisely which one of the current observation categories is included in a package and that also tries to predict the target class for each data category [4].

Reinforcement learning (RL) is a third category that is often known as unsupervised learning but distinguished from it in many subjects. Here, by collecting input about its behavior in a complex environment, so the algorithm continues to improve. RL is a modern decision-making AI technology which will help AI move ahead in the field of machine learning. RL usually attempts to find a good mapping for identifying perceptions for the decision-maker, who deals with the environment, to take some steps to fix the problem. RL is a strong way to accelerate initial learning with excellent results [4]. **Table 3** provides a quick distinction of UL, SL and RL [46].

	Unsupervised learning	**Unsupervised learning**	**Reinforcement learning**
Model affection	Model does not affect the input data	Model does not affect the input data	Agent can affect its own observations
Learning structure	Learning underlying data structure	Learning to approximate reference answers	Learning optimal strategy by trial and error
Feedback	No feedback required	Needs correct answers	Needs feedback on agent's own actions

Table 3 Comparison with machine learning approaches [47]

The emphasis for more clarification is set on supervised learning while training a network with a pre-generated trajectory that functions as Ground Truth. For guided learning, there are two specific forms of potential assumptions. When the Algorithm looks for patterns to classify inputs as an output to a class using a classification method [4]. To create a model, there are several regression algorithms available, including linear regression and decision trees for basic predictions. For more complicated and nonlinear topics, Support vector regression and Artificial Neural Networks (ANN) are implemented. The complete table containing the ML algorithms explained is shown in **Figure 9**.

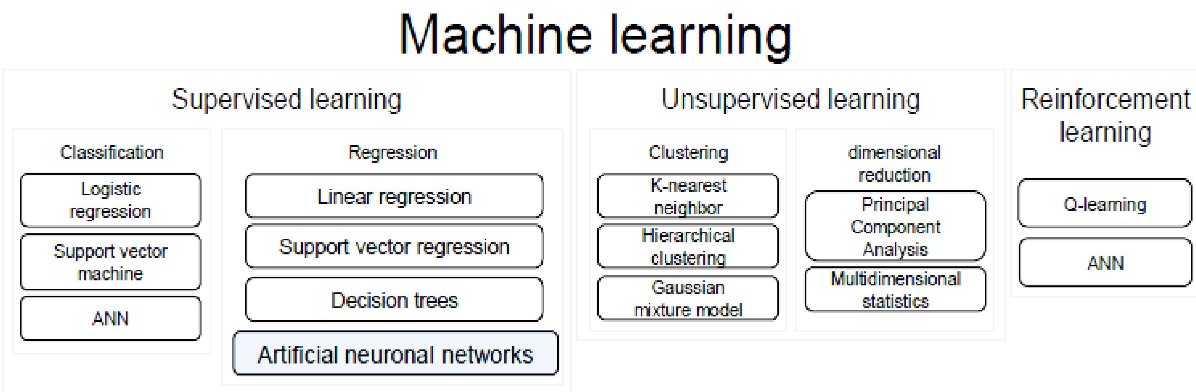

Figure 9 The algorithms of Machine learning [48]

3.3 Deep Learning

Deep Learning (DL) is one of the recent strong and scalable Machine learning that represents the world with a hierarchy of implications of abstract representations [4].

In most machine learning approaches, the implemented characteristics must be defined by domain experts, in order to minimize the complexity of the data and make patterns more accessible for learning algorithms. The major benefit of DL algorithms is to learn from data gradually in high-level features [49], [50].

DL focuses on conception, comprehension, and decision-making. It utilizes massive neural network layers with many processing units, which have many advantages to develop training techniques for the learning of complex patterns in several data. Commonly complex engineering technologies involves behavior recognition, video marking, image, voice recognition, object recognition and different kinds of applications. DL also transmits essential inputs into other fields of perception, like the transmission of audio, voice, and natural language [4].

While AI has many promising approaches, deep learning and reinforcement learning (RL) are the central technologies of AI, which creates motivating learning. Deep RL approach expands enhancement learning through the use of a deep neural network even without complex state space architecture [13]. Therefore, Deep RL refers to target-oriented algorithms for opening many new applications in such fields as engineering and a lot more.

3.4 Internet of Things (IoT)

In 1999, Kevin Ashton proposed the scientific word Internet of Things (IoT) [51]. The definition of 'things' has changed as technology advanced last decade, but without a human involvement the main purpose of a digital interface can make sense of information. In an unparalleled way, the Internet made interconnections possible between people. The next networking wave would arrive much quicker to connect objects and create an intelligent environment. Nine billion interconnected devices currently exist, more than the world' s population, and by 2020 it is predicted to hit 24 billion devices. A huge number of wired users have a key benefit in getting access to vast datasets that can be used for smart applications [52].

The Internet of Things (IoT) is the topic of a technological revolution of 4.0. This provides a global system for data aggregation and distribution of information / data from storage, actuation, sensing, state-of-the-art service and networking technologies. The combination of high-speed, durable, low-latency networking and AI and IoT technologies will make the transition to a fully intelligent, autonomous vehicle, an example of a combination of real-world and digital knowledge for industry 4.0[4].

Autonomous Vehicles (AVs) should be able to generate, record, translate, process and store vehicle data from different sources, including road conditions and incidents through a communication network, using various means of communication. The Internet of Things (IoT) is an emerging technology that incorporates a variety of network items such as: buildings and other products , vehicles, hardware-integrated systems , applications, sensors and network connectivity that allows them to collect and transmit data without human interference[51].

In **Figure 10**, the IoT definition evolves from machine-to-machine (M2M) communication. M2M links independent sensor systems to servers without (or little) human interventions, while IoT uses communication from machine to machine, integrating web apps, and connects it to cloud computing systems.

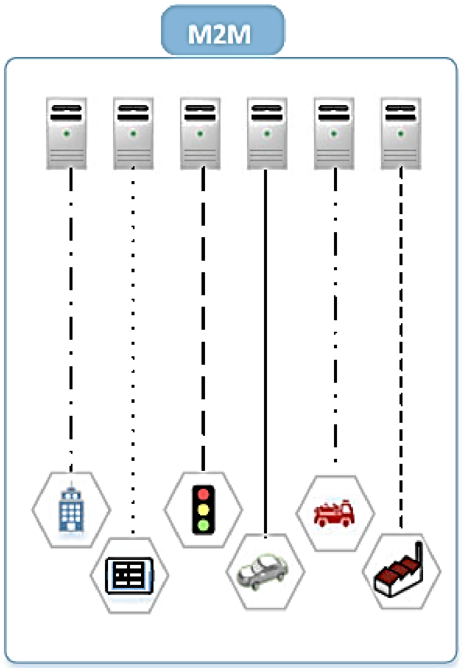

Figure 10 AV connectivity from Machine to Machine (M2M) and IoT Internet (IoT) [4]

3.4.1 Platform of the Internet of Things for Autonomous Driving

Typical IoT platform is an interconnected system that can enable the development of a massive number of data to be transferred and analyzed in a cloud computing with millions of simultaneous user connections. There are four major components on the standard IoT platform [4]:

1- The first part is the basic components sensors and hardware, which gather different data types from the real-world situation,

2- A second part is a communications network usually based on wireless infrastructure like Wi-Fi and telephone (3G, 4G, 5G).

3- Big data is the third component that reflects the generation of volume, speed, and varieties of data; this data should be transferred, processed, sorted, and proceeded.

4- The platform's fourth component is the cloud, which collects and processes data, with a range of cloud computing, analytics, and storage services.

IoT applications have generally been hosted in the cloud that will support the physical systems with reviews and decisions. For AVs, the cloud would be the unified management system that implements all the software components.

3.5 MBOT

For this particular project we have chosen mBot to represent autonomous vehicle principles, mBot is a simple, easy-to-use robot kit that offers realistic experience in graphics, electronics and robotics. This is a system all in one built for STEM training and for robotic learning [53].

mBot will be an all-in-one way to enjoy practical programming, electronics and robotics experience, all is enjoyable and innovative. It comes with a range of simple preassembled choices such as a vehicle for preventing barriers, a vehicle to navigate the line, a remote control car and is suitable for use in various games, such as balls, football match, sumo, etc. The module's mechanical system is compatible with the Makeblock framework and the majority of Lego modules while its electronic parts are created on the basis of the Arduino open-source ecosystem. This means that mBot has limitless flexibility and that we have to transform it using all of the mechanical parts and electronic components.

3.5.1 mBot features

- Electronics are based on the open source framework of Arduino.
- Suitable for individual users' beginners and supported by the IOS & Android Software.
- Two programming tools: mBlock is a drag-and-drop programming tool based on Scratch 3.0 and Arduino Integrated development environment (IDE).
- Fast and intuitive cables using RJ25 connector with color matching
- Aluminum chassis 2 mm wide, solid and Makeblock & Lego compatible.
- Achieve various projects like Wall avoidance, self-barking, line-following.
- There are free lessons about Arduino and mBlock and they are continuously growing.
- Provide us with innovation with state-of-the-art technology like AI. mBlock combines cognitive services from Microsoft and deep learning from Google into one tool.
- Create IoT apps in a real environment. mBlock is provided with an IoT teaching cloud service. We may use the feature to build some projects like the weather forecasts, the autonomous robotic watering system and intelligent lighting,

through working via robots or electronic modules. The easiest way for students to study IoT and to figure out how it operates in real life.

3.5.2 Vehicle specifications

- Software and programming: Arduino C and mBlock platform
- Arduino mBot board
- Raspberry Pi 2 Model B
- Inputs:
    - Light sensor
    - Ultrasonic sensor
    - Line-follower sensor
    - Button
    - Me PIR Motion sensor
    - IR Receiver

- Outputs:
    - RGB LED
    - IR Transmitter
    - Buzzer
    - 2 Motors
    - Ports
- Microcontroller: based on Arduino Uno
- Dimensions: 17 x 13 x 9 cm
- Weight: under 1070 g
- Power: 3.7V Lithium battery (charger on board) [54]

3.5.3 mBot structure

Figure 11, 12 show all the mBot components and the robot structure and mCore.

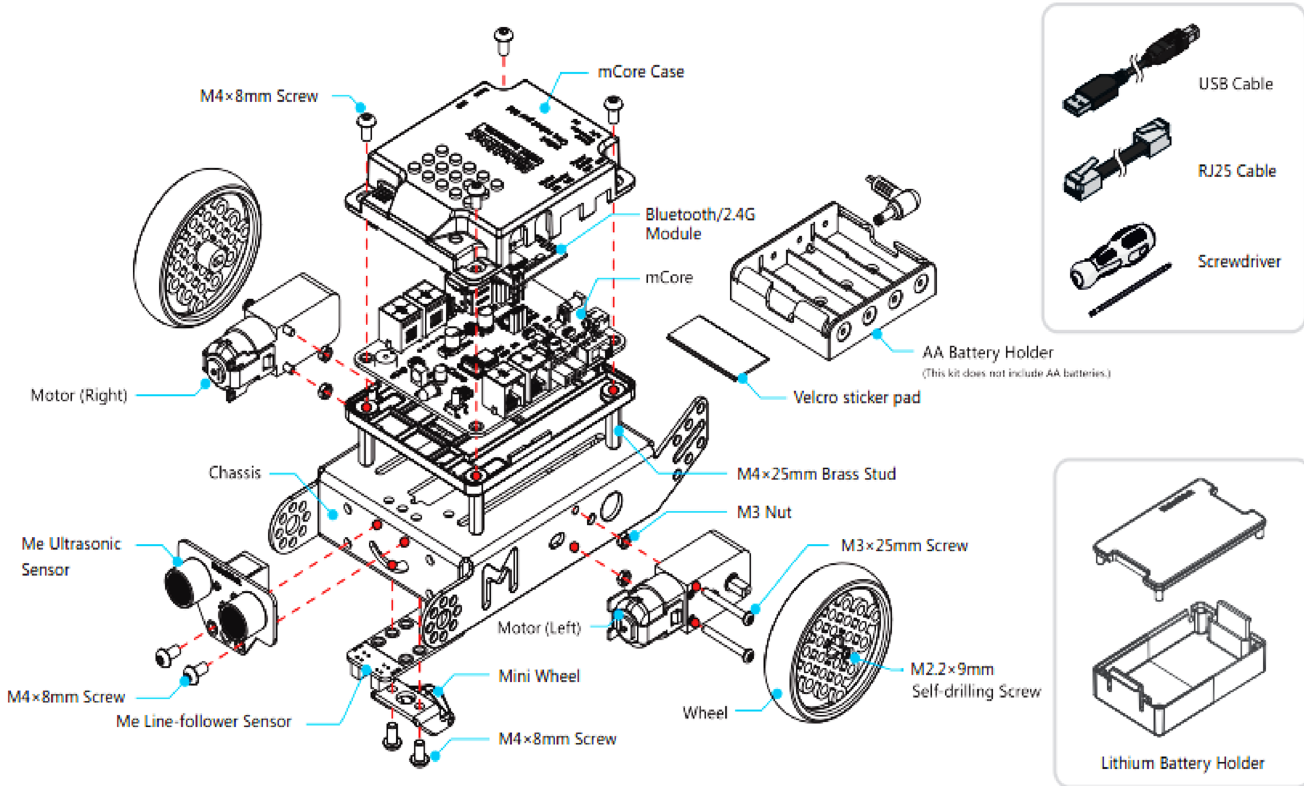

Figure 11 Parts list [55]

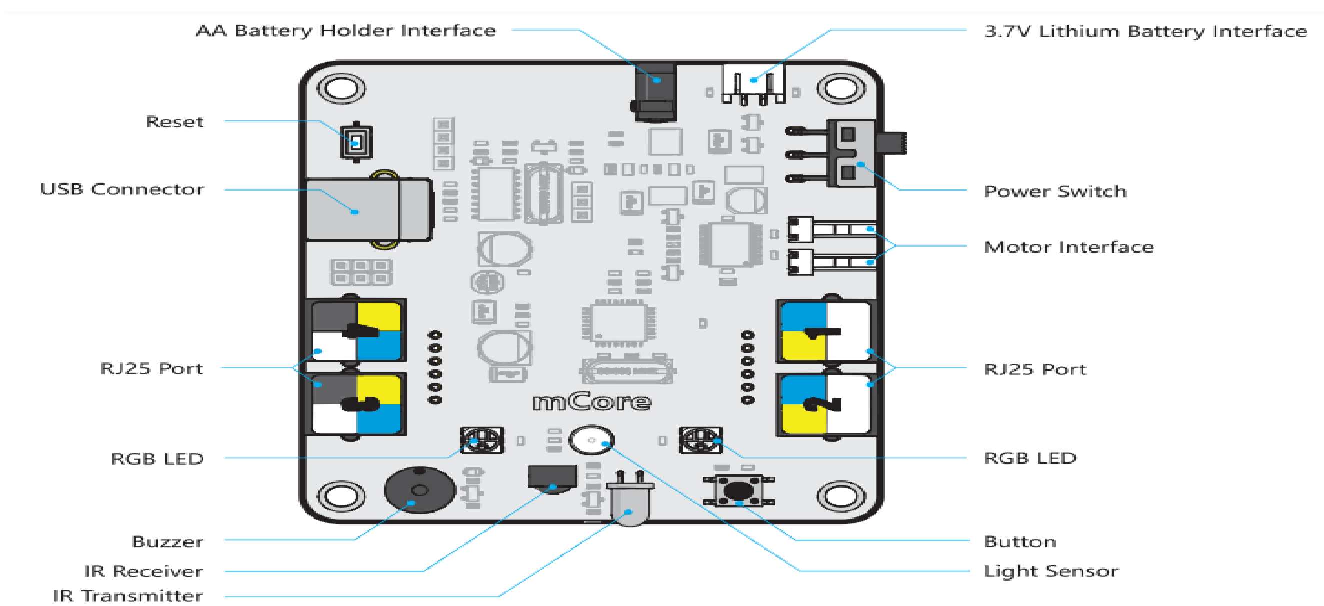

Figure 12 mCore [55]

4 Practical Part

In this section of this thesis we are going to conduct 5 different tests on mBot by doing 5 programs, where in this experiment each program has different implementation from other.

4.1 First program following the line

This program is the under Line- Following Mode mainly by using following line sensor, mBot moves and automatically follows the black line. Basic program for the infinite cycle.

The mBot consists of various of components (mechanical, electrical, electronic). Among these components, there are three sensors on the top and below and in front of the mBot. In this implementation the movement of the mBot clearly depends on the sensor below the mBot in which black dotted lines may be fumbling to be monitored on the black line and not crossed on the white page, as seen in the following **Figure 13**:

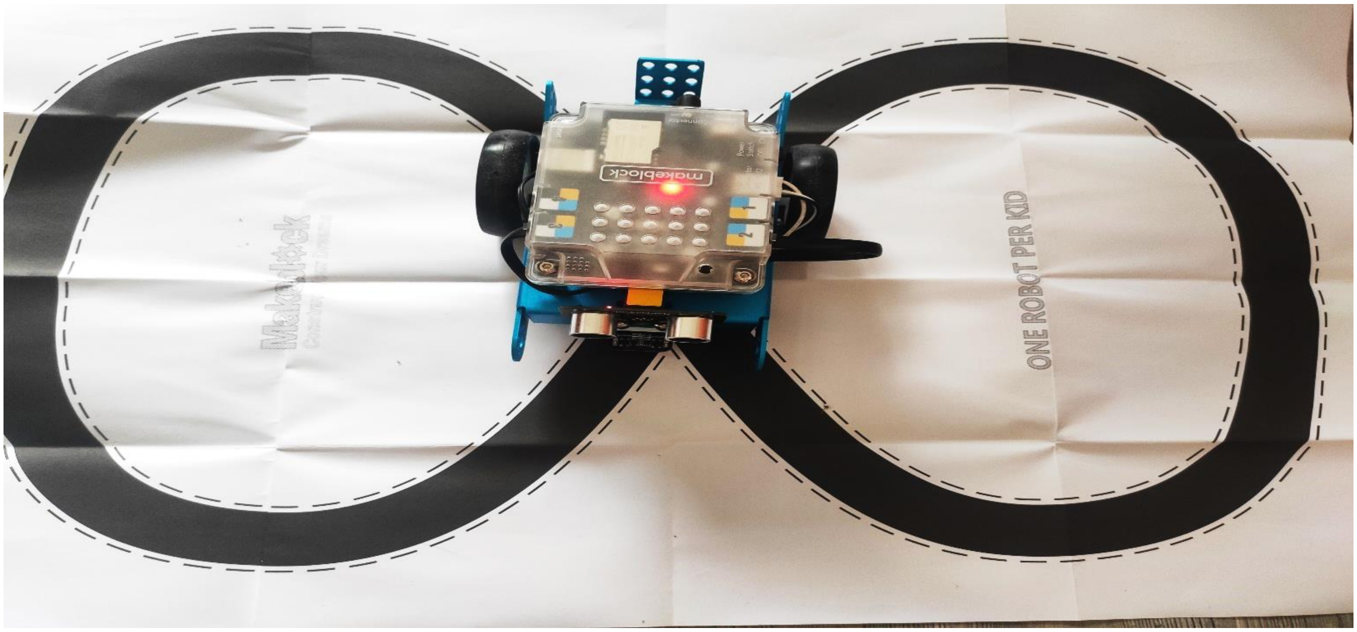

Figure 13 mBot will move and follow the black track automatically

According to data that is entered into the mBot through the computer, we can control the speed of the mBot and also set the actions of motion either forward or back to back. The first program details in mBlock framework are seen in **Figure 14**:

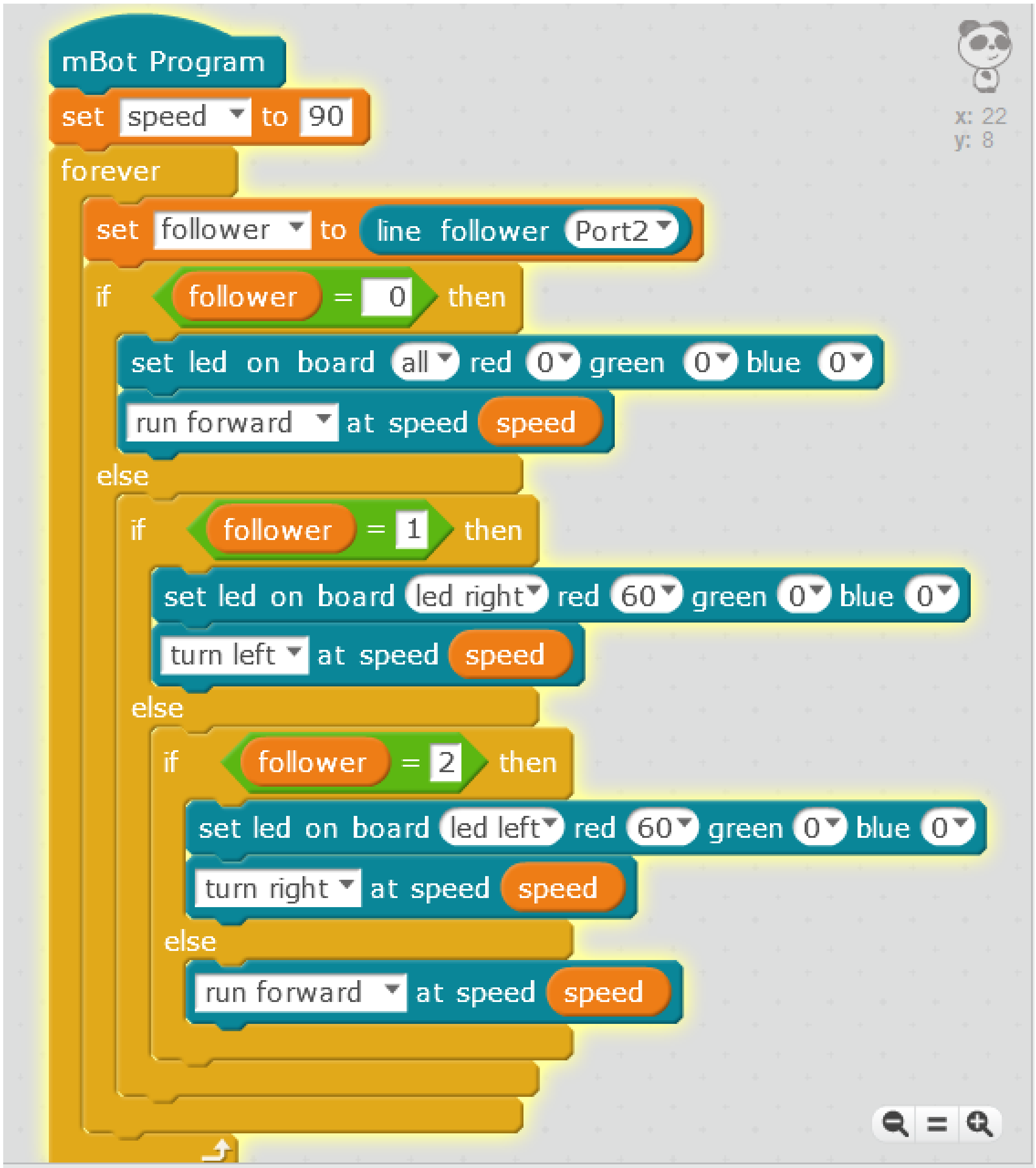

Figure 14 First progrm following the line in mBlock

The following is the first program (Follow the line) code presenting in Arduino C programming language:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include <MeMCore.h>

MeDCMotor motor_9(9);
MeDCMotor motor_10(10);
void move(int direction, int speed)
{
      int leftSpeed = 0;
      int rightSpeed = 0;
      if(direction == 1){
            leftSpeed = speed;
            rightSpeed = speed;
      }else if(direction == 2){
            leftSpeed = -speed;
            rightSpeed = -speed;
      }else if(direction == 3){
            leftSpeed = -speed;
            rightSpeed = speed;
      }else if(direction == 4){
            leftSpeed = speed;
            rightSpeed = -speed;
      }
      motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
      motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
}
double angle_rad = PI/180.0;
double angle_deg = 180.0/PI;
double speed;
double follower;
MeLineFollower linefollower_2(2);
MeRGBLed rgbled_7(7, 7==7?2:4);

void setup(){
    speed = 90;
}

void loop(){
    follower = linefollower_2.readSensors();
    if(((follower)==(  0))){
        rgbled_7.setColor(0,0,0,0);
        rgbled_7.show();
        move(1,speed);
    }else{
        if(((follower)==(1))){
            rgbled_7.setColor(1,60,0,0);
            rgbled_7.show();
            move(3,speed);
```

```
        }else{
            if(((follower)==(2))){
                rgbled_7.setColor(2,60,0,0);
                rgbled_7.show();
                move(4,speed);
            }else{
                move(1,speed);
            }
        }
    }
    _loop();
}

void _delay(float seconds){
    long endTime = millis() + seconds * 1000;
    while(millis() < endTime)_loop();
}

void _loop(){
}
```

4.2 Second program Following the line and avoid the objects

In this program the robot has the ability to follow the line as the first program, but also the robot can avoid any barrier in its way by using the ultrasonic sensor, when the robot’s sensor detect an object closer than 10 cm the robot will turn around automatedly to navigate the line again and then go back to the beginning, this program starts only when we press the button on the top of the robot.

Figure 15 is presenting the mBlock code by using functional blocks, we have two different functions in this program, the first function is read_sensor_data by using this function, it allows the robot to detect the data and understand the environment around the robot, the second function is presenting the following line code.

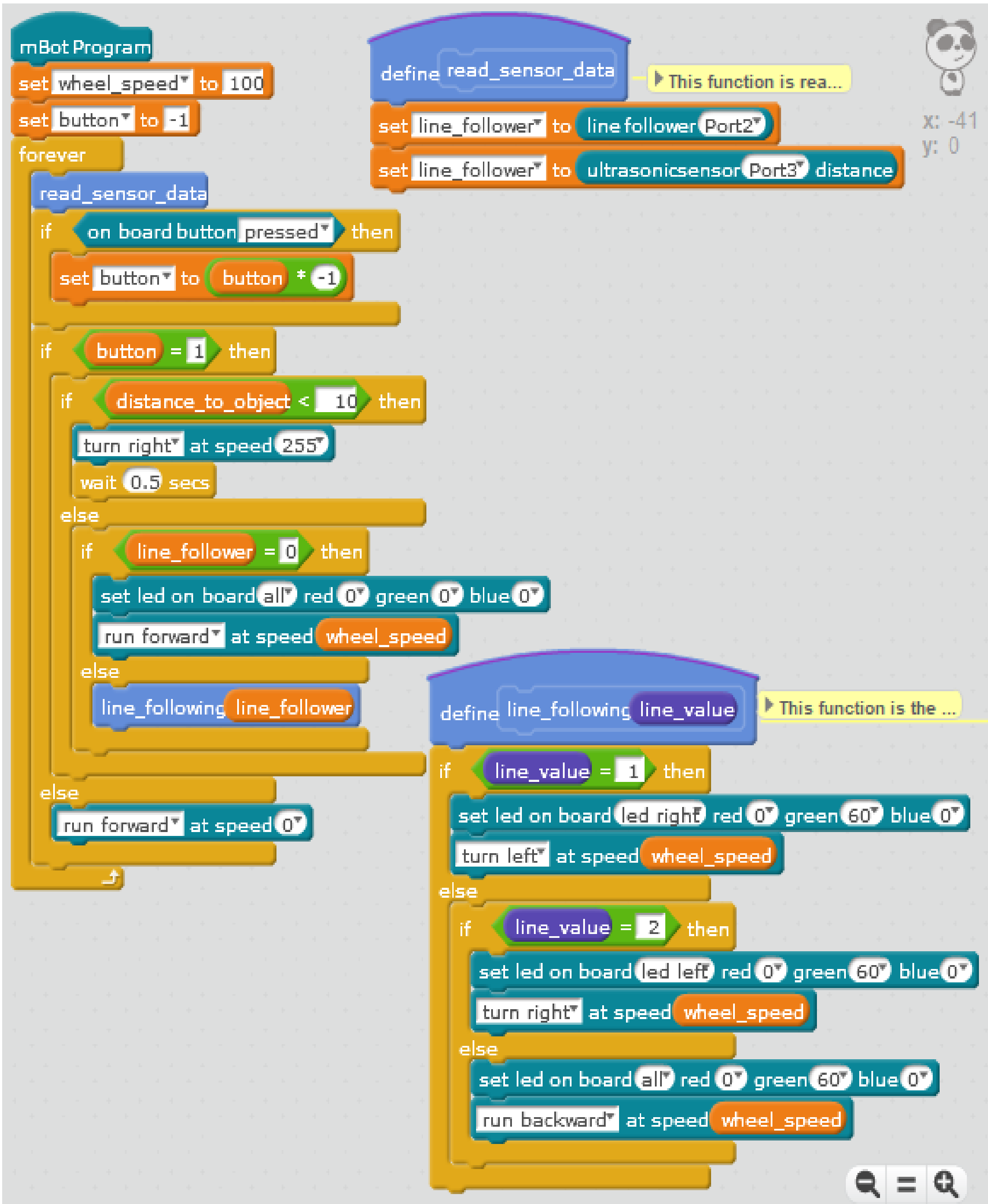

Figure 15 The second program mBlock code

The following is the second program (Following the line and avoid the objects) code presenting in Arduino C programming language:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include <MeMCore.h>

MeDCMotor motor_9(9);
MeDCMotor motor_10(10);
void move(int direction, int speed)
{
	int leftSpeed = 0;
	int rightSpeed = 0;
	if(direction == 1){
		leftSpeed = speed;
		rightSpeed = speed;
	}else if(direction == 2){
		leftSpeed = -speed;
		rightSpeed = -speed;
	}else if(direction == 3){
		leftSpeed = -speed;
		rightSpeed = speed;
	}else if(direction == 4){
		leftSpeed = speed;
		rightSpeed = -speed;
	}
	motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
	motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
}
double angle_rad = PI/180.0;
double angle_deg = 180.0/PI;
void read_sensor_data();
double line_follower;
MeLineFollower linefollower_2(2);
MeUltrasonicSensor ultrasonic_3(3);
void line_following(double line_value);
double wheel_speed;
MeRGBLed rgbled_7(7, 7==7?2:4);
double button;
double distance_to_object;

void read_sensor_data()
{
    line_follower = linefollower_2.readSensors();
    line_follower = ultrasonic_3.distanceCm();
}

void line_following(double line_value)
```

```
{
    if(((line_value)==( 1 ))){
        rgbled_7.setColor(1,0,60,0);
        rgbled_7.show();
        move(3,wheel_speed);
    }else{
        if(((line_value)==( 2 ))){
            rgbled_7.setColor(2,0,60,0);
            rgbled_7.show();
            move(4,wheel_speed);
        }else{
            rgbled_7.setColor(0,0,60,0);
            rgbled_7.show();
            move(2,wheel_speed);
        }
    }
}

void setup(){
    wheel_speed = 100;
    button = -1;
    pinMode(A7,INPUT);
}

void loop(){
    read_sensor_data();
    if((0^(analogRead(A7)>10?0:1))){
        button = (button) * (-1);
    }
    if(((button)==(1))){
        if((distance_to_object) < (  10)){
            move(4,255);
            _delay(0.5);
        }else{
            if(((line_follower)==(0))){
                rgbled_7.setColor(0,0,0,0);
                rgbled_7.show();
                move(1,wheel_speed);
            }else{
                line_following(line_follower);
            }
        }
    }else{
        move(1,0);
    }
    _loop();
}

void _delay(float seconds){
    long endTime = millis() + seconds * 1000;
    while(millis() < endTime)_loop();
}

void _loop(){
}
```

4.3 The Third Program Do not cross the line

This program uses the following line sensor to navigate the presence of a line, and guide the robot in a specific way that always keeps the robot inside the box (or any other enclosed form that we can take a place on the floor or the table.

Figure 16 is presenting the shape of the closed box.

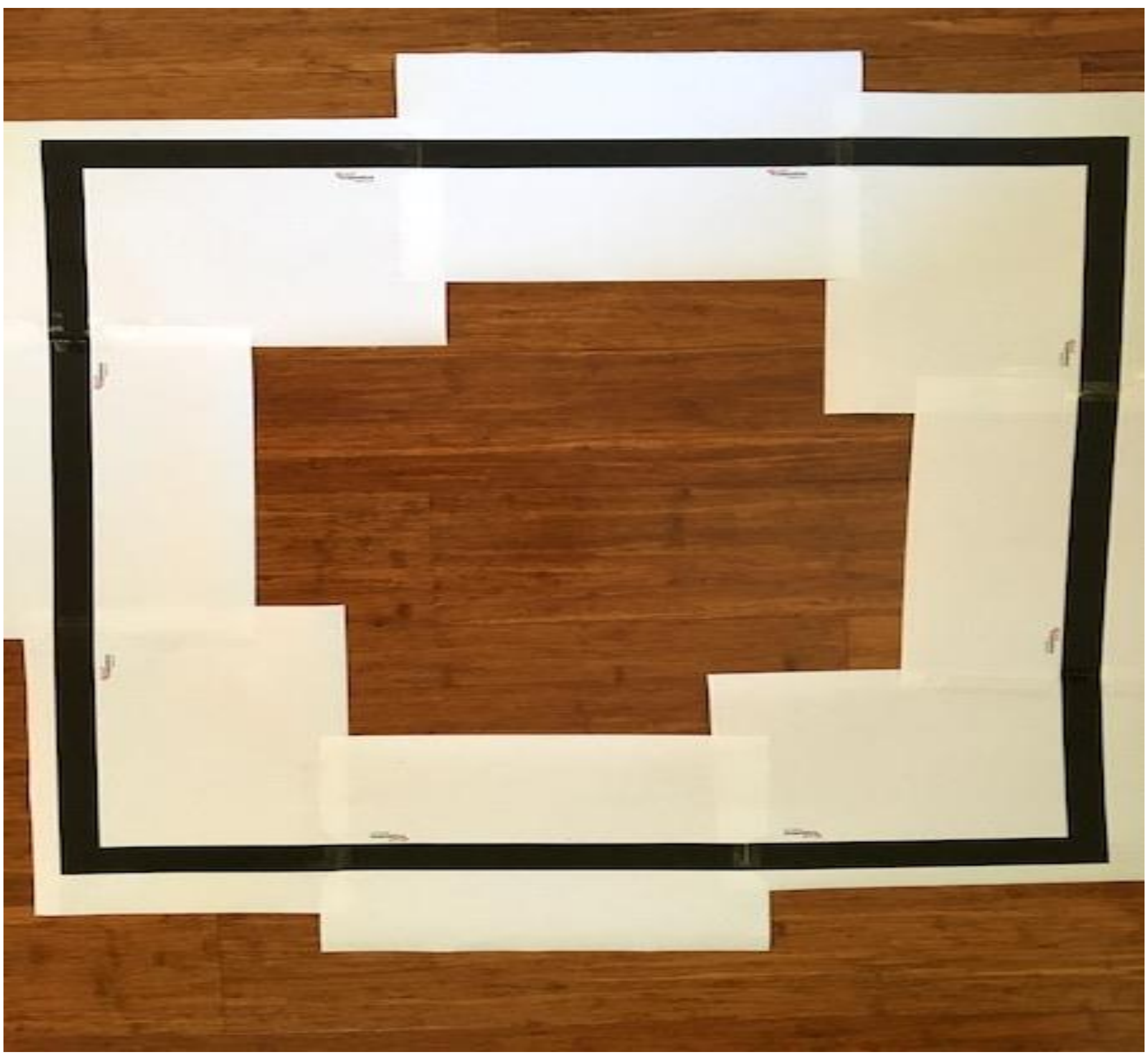

Figure 16 The closed shape box

Figure 17 is presenting the mBlock code by using functional blocks, we have two different functions in this program, the first function is follow_line This function implements the code for the line following functionality. We are passing a numerical value to the function, named "line_value". We use this value inside the function. The second function is get_sensor_data this subroutine reads sensor values and stores them in global variables.

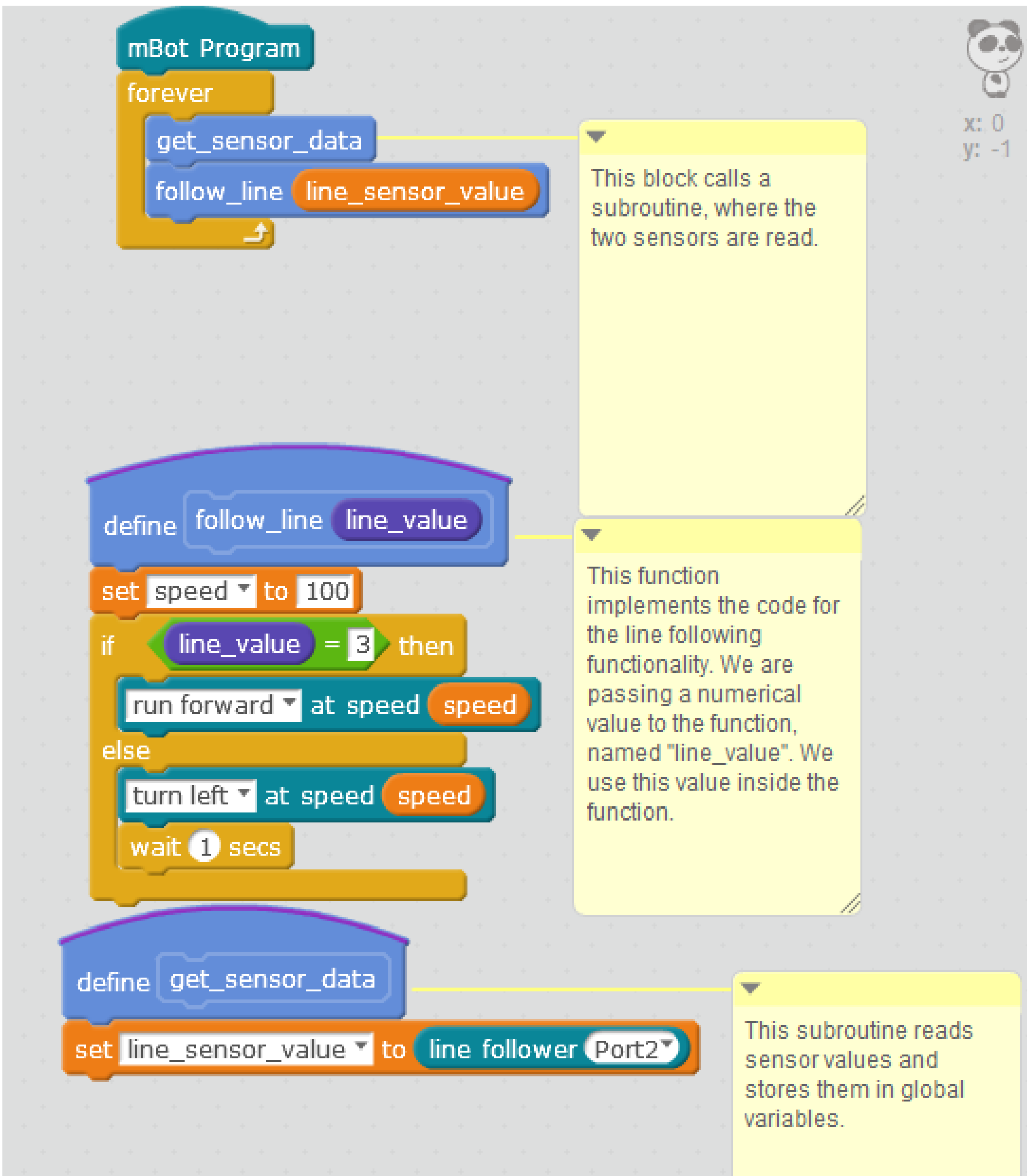

Figure 17 mBlock code for do not cross the line program

The following is the third program (do not cross the line) code presenting in Arduino C programming language:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include <MeMCore.h>

MeDCMotor motor_9(9);
MeDCMotor motor_10(10);
void move(int direction, int speed)
{
      int leftSpeed = 0;
      int rightSpeed = 0;
      if(direction == 1){
               leftSpeed = speed;
               rightSpeed = speed;
      }else if(direction == 2){
               leftSpeed = -speed;
               rightSpeed = -speed;
      }else if(direction == 3){
               leftSpeed = -speed;
               rightSpeed = speed;
      }else if(direction == 4){
               leftSpeed = speed;
               rightSpeed = -speed;
      }
      motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
      motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
}
double angle_rad = PI/180.0;
double angle_deg = 180.0/PI;
void follow_line(double line_value);
double speed;
void get_sensor_data();
double line_sensor_value;
MeLineFollower linefollower_2(2);

void follow_line(double line_value)
{
    speed = 100;
    if(((line_value)==(3))){
        move(1,speed);
    }else{
        move(3,speed);
        _delay(1);
    }
}

void get_sensor_data()
{
    line_sensor_value = linefollower_2.readSensors();
}
```

```

void setup(){
}

void loop(){
    get_sensor_data();
    follow_line(line_sensor_value);
    _loop();
}

void _delay(float seconds){
    long endTime = millis() + seconds * 1000;
    while(millis() < endTime)_loop();
}

void _loop(){
}
```

4.4 The Fourth Program Random Movement Without Making an Accident

Thin program does not require the follow line sensor all what we use for this particular program is the ultrasonic sensor to measure the distance, the program allows the robot to move randomly from any point to any another point without collision in any barrier or object close to robot by 15 cm.

Figure 18 is presenting the fourth program (random movement without making an accident) in mBlock code.

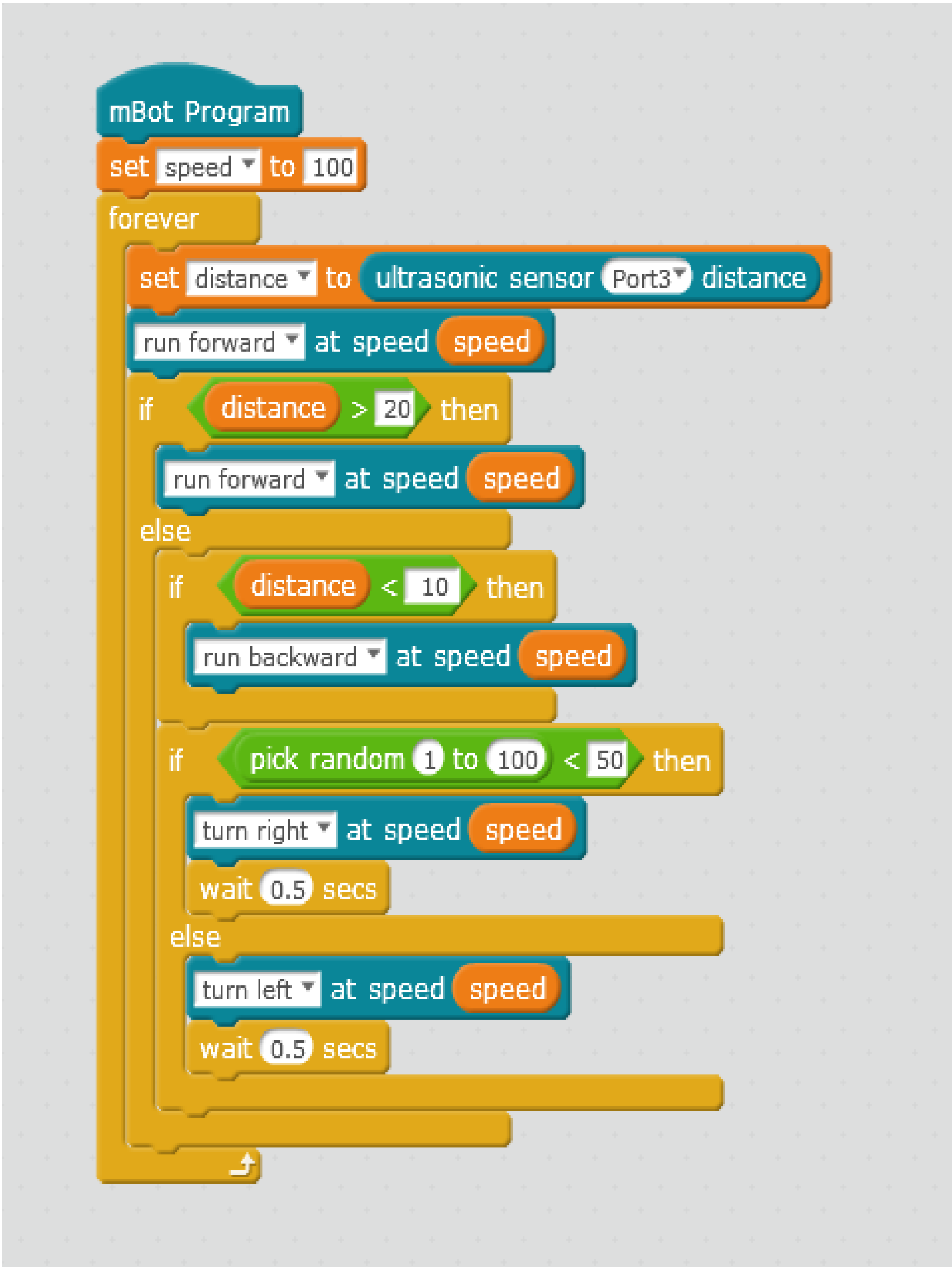

Figure 18 mBlock code for the fourth progra

The following is the fourth program (random movement without making an accident) code presenting in Arduino C programming language:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include <MeMCore.h>

MeDCMotor motor_9(9);
MeDCMotor motor_10(10);
void move(int direction, int speed)
{
        int leftSpeed = 0;
        int rightSpeed = 0;
        if(direction == 1){
                leftSpeed = speed;
                rightSpeed = speed;
        }else if(direction == 2){
                leftSpeed = -speed;
                rightSpeed = -speed;
        }else if(direction == 3){
                leftSpeed = -speed;
                rightSpeed = speed;
        }else if(direction == 4){
                leftSpeed = speed;
                rightSpeed = -speed;
        }
        motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
        motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
}
double angle_rad = PI/180.0;
double angle_deg = 180.0/PI;
double speed;
double distance;
MeUltrasonicSensor ultrasonic_3(3);

void setup(){
    speed = 100;
}

void loop(){
    distance = ultrasonic_3.distanceCm();
    if((distance) < (25)){
        if((distance) < (15)){
            move(2,speed);
            if((random(1,(100)+1)) < (55)){
                move(4,speed);
                _delay(0.5);
            }else{
                move(3,speed);
```

```
                _delay(0.5);
            }
            _delay(1);
        }
    }else{
        move(1,speed);
    }
    _loop();
}

void _delay(float seconds){
    long endTime = millis() + seconds * 1000;
    while(millis() < endTime)_loop();
}

void _loop(){
}
```

4.5 The Fifth Program mBot Dancing

The fifth program consists of certain data that automatically move the robot, and with these data we can also incorporate some music for the robot also making the robot as if dancing, this is the reason we named this program (Dancing). Furthermore, when the room light is off, the robot activity is detected. So, the robot will start the program only when the light sensor does not get any light.

Figure 19 the application of mBot robot presenting the fifth program when the light is off.

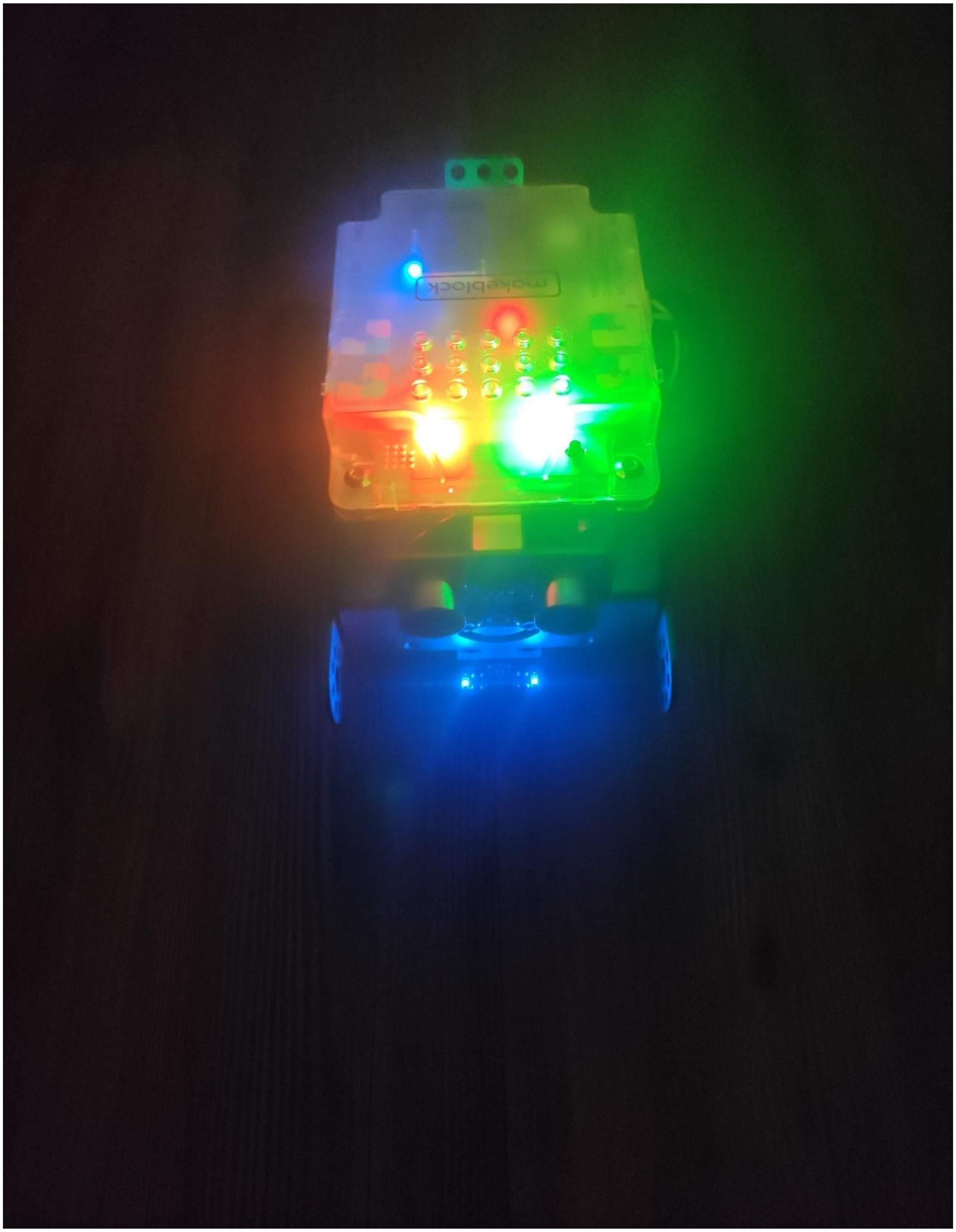

Figure 19 Fifth program presentation

Figure 20 is presenting the fifth program (mBot dancing) in mBlock code.

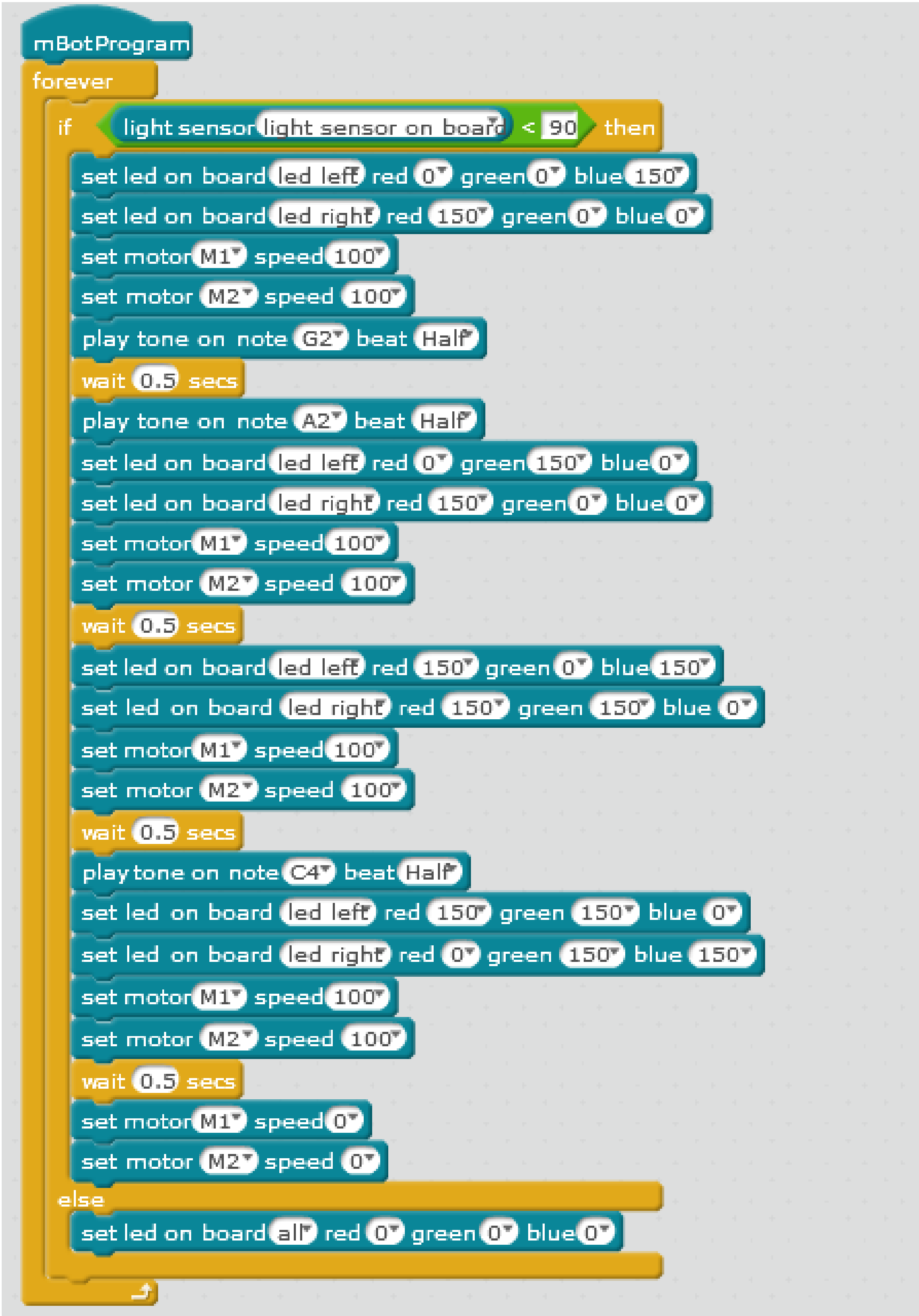

Figure 20 Fifth program mBlock code

The following is the fifth program (mBot dancing) code presenting in Arduino C programming language:

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include <MeMCore.h>

MeDCMotor motor_9(9);
MeDCMotor motor_10(10);
void move(int direction, int speed)
{
      int leftSpeed = 0;
      int rightSpeed = 0;
      if(direction == 1){
              leftSpeed = speed;
              rightSpeed = speed;
      }else if(direction == 2){
              leftSpeed = -speed;
              rightSpeed = -speed;
      }else if(direction == 3){
              leftSpeed = -speed;
              rightSpeed = speed;
      }else if(direction == 4){
              leftSpeed = speed;
              rightSpeed = -speed;
      }
      motor_9.run((9)==M1?-(leftSpeed):(leftSpeed));
      motor_10.run((10)==M1?-(rightSpeed):(rightSpeed));
}
double angle_rad = PI/180.0;
double angle_deg = 180.0/PI;
MeLightSensor lightsensor_6(6);
MeRGBLed rgbled_7(7, 7==7?2:4);
MeBuzzer buzzer;

void setup(){
}

void loop(){
    if((lightsensor_6.read()) < (90)){
        rgbled_7.setColor(2,0,0,150);
        rgbled_7.show();
        rgbled_7.setColor(1,150,0,0);
        rgbled_7.show();
        motor_9.run((9)==M1?-(100):(100));
        motor_10.run((10)==M1?-(100):(100));
        buzzer.tone(98, 500);
        delay(20);
        _delay(0.5);
        buzzer.tone(110, 500);
        delay(20);
        rgbled_7.setColor(2,0,150,0);
```

```
        rgbled_7.show();
        rgbled_7.setColor(1,150,0,0);
        rgbled_7.show();
        motor_9.run((9)==M1?-(100):(100));
        motor_10.run((10)==M1?-(100):(100));
        _delay(0.5);
        rgbled_7.setColor(2,150,0,150);
        rgbled_7.show();
        rgbled_7.setColor(1,150,150,0);
        rgbled_7.show();
        motor_9.run((9)==M1?-(100):(100));
        motor_10.run((10)==M1?-(100):(100));
        _delay(0.5);
        buzzer.tone(262, 500);
        delay(20);
        rgbled_7.setColor(2,150,150,0);
        rgbled_7.show();
        rgbled_7.setColor(1,0,150,150);
        rgbled_7.show();
        motor_9.run((9)==M1?-(100):(100));
        motor_10.run((10)==M1?-(100):(100));
        _delay(0.5);
        motor_9.run((9)==M1?-(0):(0));
        motor_10.run((10)==M1?-(0):(0));
    }else{
        rgbled_7.setColor(0,0,0,0);
        rgbled_7.show();
    }
    _loop();
}

void _delay(float seconds){
    long endTime = millis() + seconds * 1000;
    while(millis() < endTime)_loop();
}

void _loop(){
}
```

5 Results and Discussion

In practical section, we have been able to show the implementation in mBot robot in some different programs, Through the experiments that were made it was possible to prove some programs connected to the module we were using to implement an AV systems.

This thesis discusses how AVs could help to improve future mobility performance, safety, and cleanliness. However, it also emphasizes the importance of meeting many requirements to accomplish this goal. Otherwise, the introduction of AVs into the traffic streams cannot deliver the desired benefits.

Fully autonomous vehicles will probably not be offered soon. In the meantime, cooperative traffic management approaches should be implemented to ensure that technical problems come to end, which ensures progress as they start. In specific, legal and ethical issues should also be taken into consideration, which would decide whether humanity is prepared for future autonomous driving environments.

6 Conclusion

The rise of autonomous vehicles (AVs) has created a new movement in recent years to introduce different smart technologies and technologies to boost automated decision-making efficiency and quality. The combination of AI and IoT offers smart vehicles with high-performance, automated systems to allow complex and solid control systems in the environment. Although cloud computing services are historically the key components of AV applications, a new trend of computing has developed to solve technological problems including latency, network capacity and security.

This study proposes the concept architecture of a modern AI-based AV with IoT. it could be viewed as a simple architecture. Although these programs are just a proof of several future deployments, it shows that the integrated technologies will incorporate a remote-controlled vehicle framework.

6.1 Recommendation

For a long time now, we were interested in beginning an Arduino-based project, we decided to start a project based on Arduino and the Starter Robot Kit was the ideal gate for experiments. For those who want to bring somebody into open-source computer hardware's but want to feel stress free before plunging further into other Arduino DIY experiments, we highly suggest this package.

It was a pleasure to create the kit, since all the materials needed for the construction were included and all pieces were arranged carefully in the package. The design guide was quick but comprehensive to follow, and anyone with a larger Lego package could feel completely comfortable with the construction process. The parts of the kit were accurate and very near tolerance machined. The consistency of the parts and their fitting undoubtedly added to the satisfactory and durable design of the finish.

It was definitely important to try out the automatic navigation with the ultrasonic sensor. This robot really allows us to plug the board into the computer for programming purposes. In the real world, writing code and seeing the physical results is profoundly rewarding. We have been very interested in adding more sensors and actuators after plenty of time with the Arduino IDE to figure out what we can come up with.

7 Bibliography

[1] "Self-Driving Cars 101 | Union of Concerned Scientists." https://www.ucsusa.org/resources/self-driving-cars-101 (accessed Oct. 10, 2020).

[2] J. Anderson, N. Kalra, K. Stanley, P. Sorensen, C. Samaras, and O. Oluwatola, *Autonomous Vehicle Technology: A Guide for Policymakers*. 2016.

[3] K. F. Wu, L. Sasidharan, C. P. Thor, and S. Y. Chen, "Crash sequence based risk matrix for motorcycle crashes," *Accident Analysis and Prevention*, 2018, doi: 10.1016/j.aap.2018.03.022.

[4] H. Khayyam, B. Javadi, M. Jalili, and R. N. Jazar, "Artificial Intelligence and Internet of Things for Autonomous Vehicles," in *Nonlinear Approaches in Engineering Applications*, 2020.

[5] "Shared mobility on the road of the future," 2016. https://www.morganstanley.com/ideas/car-of-future-is-autonomous-electric-shared-mobility/.

[6] "What is an Autonomous Car? – How Self-Driving Cars Work | Synopsys." https://www.synopsys.com/automotive/what-is-autonomous-car.html (accessed Oct. 13, 2020).

[7] R. Lanctot, "Accelerating the Future: The Economic Impact of the Emerging Passenger Economy Autonomous," *Autonomous Vehicle Service*, 2017.

[8] A. B. T. Sherif, K. Rabieh, M. M. E. A. Mahmoud, and X. Liang, "Privacy-Preserving Ride Sharing Scheme for Autonomous Vehicles in Big Data Era," *IEEE Internet of Things Journal*, 2017, doi: 10.1109/JIOT.2016.2569090.

[9] W. Xu *et al.*, "Internet of vehicles in big data era," *IEEE/CAA Journal of Automatica Sinica*, 2018, doi: 10.1109/JAS.2017.7510736.

[10] M. Lavasani, X. Jin, and Y. Du, "Market penetration model for autonomous vehicles on the basis of earlier technology adoption experience," *Transportation Research Record*, 2016, doi: 10.3141/2597-09.

[11] "What is ADAS (Advanced Driver Assistance Systems)? – Overview of ADAS Applications | Synopsys." https://www.synopsys.com/automotive/what-is-adas.html (accessed Oct. 13, 2020).

[12] "The 6 Levels of Vehicle Autonomy Explained | Synopsys Automotive." https://www.synopsys.com/automotive/autonomous-driving-levels.html (accessed Oct. 12, 2020).

[13] “The Functional Components of Autonomous Vehicles - Expert Article | Robson Forensic.” https://www.robsonforensic.com/articles/autonomous-vehicles-sensors-expert/ (accessed Oct. 14, 2020).

[14] “Understanding How Ultrasonic Sensors Work | MaxBotix Inc.” https://www.maxbotix.com/articles/how-ultrasonic-sensors-work.htm (accessed Oct. 14, 2020).

[15] “IMU | Analog Devices.” https://www.analog.com/en/applications/markets/automotive-pavilion-home/autonomous-transportation-and-adas/imu.html (accessed Oct. 14, 2020).

[16] C. N. To *et al.*, “Autodriver Autonomous Vehicles Control Strategy,” in *Procedia Computer Science*, 2018, doi: 10.1016/j.procS.2018.08.021.

[17] H. Marzbani, H. Khayyam, C. N. To, D. V. Quoc, and R. N. Jazar, “Autonomous Vehicles: Autodriver Algorithm and Vehicle Dynamics,” *IEEE Transactions on Vehicular Technology*, 2019, doi: 10.1109/TVT.2019.2895297.

[18] A. H. M. Rubaiyat, Y. Fallah, X. Li, and G. Bansal, “Multi-sensor data fusion for vehicle detection in autonomous vehicle applications,” in *IS and T International Symposium on Electronic Imaging Science and Technology*, 2018, doi: 10.2352/ISSN.2470-1173.2018.17.AVM-257.

[19] J. Kocic, N. Jovicic, and V. Drndarevic, “Sensors and Sensor Fusion in Autonomous Vehicles,” in *2018 26th Telecommunications Forum, TELFOR 2018 - Proceedings*, 2018, doi: 10.1109/TELFOR.2018.8612054.

[20] D. Loebis, R. Sutton, and J. Chudley, “Review of multisensor data fusion techniques and their application to autonomous underwater vehicle navigation,” *Journal of Marine Engineering and Technology*, 2002, doi: 10.1080/20464177.2002.11020159.

[21] D. Viegas, P. Batista, P. Oliveira, and C. Silvestre, “Discrete-time distributed Kalman filter design for formations of autonomous vehicles,” *Control Engineering Practice*, 2018, doi: 10.1016/j.conengprac.2018.03.014.

[22] R. O. Duda, P. E. Hart, and D. G. Stork, “Pattern classification,” *New York: John Wiley, Section*, 2001.

[23] K. Yaguang and G. Siyu, “Urban traffic controller using fuzzy neural network and multiSensors data fusion,” in *Proceedings of IEEE ICIA 2006 - 2006 IEEE International Conference on Information Acquisition*, 2006, doi: 10.1109/ICIA.2006.305746.

[24] P. Stone *et al.*, “Artificial Intelligence and Life in 2030 One Hundred Year Study on Artificial Intelligence: Report of the 2015-2016 Study Panel,” 2016.

[25] “Artificial Intelligence and Autonomous Vehicles | by Suhasini Gadam | Data Driven Investor | Medium.” https://medium.com/datadriveninvestor/artificial-intelligence-and-autonomous-vehicles-ae877feb6cd2 (accessed Oct. 27, 2020).

[26] M. G. T. Forum., “Autonomous vehicles: Navigating the legal and regulatory issues of a driverless world.” https://www.mcca.com/wp-content/uploads/2018/04/Autonomous-Vehicles.pdf (accessed Oct. 27, 2020).

[27] “Five challenges in designing a fully autonomous system for driverless cars.” https://iiot-world.com/artificial-intelligence/five-challenges-in-designing-a-fully-autonomous-system-for-driverless-cars/ (accessed Oct. 27, 2020).

[28] B. Crawford, H. Khayyam, A. S. Milani, and R. N. Jazar, “Big Data Modeling Approaches for Engineering Applications,” in *Nonlinear Approaches in Engineering Applications*, 2020.

[29] “What is Artificial Intelligence? How Does AI Work? | Built In.” https://builtin.com/artificial-intelligence (accessed Oct. 20, 2020).

[30] L. Araujo and D. Santos, “Artificial intelligence and machine learning",” 2018.

[31] D. Silva, “10 ways in which autonomous cars will improve our lives",” 2018. .

[32] P. Louridas and C. Ebert, *Machine Learning*. 2016.

[33] P. H. Lin, A. Wooders, J. T. Y. Wang, and W. M. Yuan, “Artificial Intelligence, the Missing Piece of Online Education?,” *IEEE Engineering Management Review*, 2018, doi: 10.1109/EMR.2018.2868068.

[34] G. F. Elsayed *et al.*, “Adversarial examples that fool both computer vision and time-limited humans,” in *Advances in Neural Information Processing Systems*, 2018.

[35] “The Business of Artificial Intelligence.” https://hbr.org/cover-story/2017/07/the-business-of-artificial-intelligence (accessed Oct. 24, 2020).

[36] “Machines who think: a personal inquiry into the history and prospects of artificial intelligence,” *Choice Reviews Online*, 2004, doi: 10.5860/choice.42-0916.

[37] B. Sc. Thomas Nützel, “AI-based movement planning for autonomous and teleoperated vehicles including the development of a simulation environment and an intelligent agent,” 2018.

[38] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. 2013.

[39] Publ. T. Reitmaier, “Aktives Lernen für Klassifikationsprobleme unter der Nutzung von Strukturinformationen,” *Bonn: Gesellschaft für Informatik*, 2015.

[40] P. Langley, “Machine learning for adaptive user interfaces,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1997, vol. 1303, pp. 53–62, doi: 10.1007/3540634932_3.

[41] K. P. Murphy, *Machine learning: a probabilistic perspective (adaptive computation and machine learning series)*. 2012.

[42] C. Liu, S. Lee, S. Varnhagen, and H. E. Tseng, “Path planning for autonomous vehicles using model predictive control,” in *IEEE Intelligent Vehicles Symposium, Proceedings*, 2017, pp. 174–179, doi: 10.1109/IVS.2017.7995716.

[43] E. Alpaydin, “Introduction to Machine Learning Third Edition,” *Introduction to Machine Learning*, 2010.

[44] M. Kusy and R. Zajdel, “Application of Reinforcement Learning Algorithms for the Adaptive Computation of the Smoothing Parameter for Probabilistic Neural Network,” *IEEE Transactions on Neural Networks and Learning Systems*, 2015, doi: 10.1109/TNNLS.2014.2376703.

[45] G. Neu and C. Szepesvári, “Training parsers by inverse reinforcement learning,” *Machine Learning*, 2009, doi: 10.1007/s10994-009-5110-1.

[46] R. Müller and H.J. Lenz, “Business Intelligence,” *Berlin, Heidelberg: Springer Vieweg*, vol. 2013.

[47] P. Louridas and C. Ebert, *Machine Learning*. 2016.

[48] I. Goodfellow, Y. Bengio, and A. Courville, “RegGoodfellow, I., Bengio, Y., & Courville, A. (2016). Regularization for Deep Learning. Deep Learning, 216–261.ularization for Deep Learning,” *Deep Learning*, 2016.

[49] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, “An introduction to deep reinforcement learning,” *Foundations and Trends in Machine Learning*, 2018, doi: 10.1561/2200000071.

[50] L. Li, Y. Lv, and F. Y. Wang, “Traffic signal timing via deep reinforcement learning,” *IEEE/CAA Journal of Automatica Sinica*, 2016, doi: 10.1109/JAS.2016.7508798.

[51] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Future Generation Computer Systems*, 2013, doi: 10.1016/j.future.2013.01.010.

[52] B. Javadi, R. N. Calheiros, K. M. Matawie, A. Ginige, and A. Cook, “Smart nutrition monitoring system using heterogeneous internet of things platform,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, doi: 10.1007/978-3-319-97795-9_6.

[53] “mBot - STEAM Education Robotics Projects | makeblock.” https://www.makeblock.com/project/mbot (accessed Oct. 28, 2020).

[54] Marina Kreplak, “Sensor equipped Smart Vehicle and design of remote Control Companion App,” 2017.

[55] “mBot componants,” 2020. https://www.mblock.cc/wp-content/uploads/2017/06/mBot-V1-1-Blue-STD-shuo-ming-shu-EN-D1-2-2-7-40-4100-Edit-1-CC.pdf (accessed Oct. 28, 2020).