

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA  
Ministry of Higher Education and Scientific Research  
University of 8 Mai 1945 – Guelma  
Faculty of Science and Technology  
Department of Electro-technical and Automation Engineering

Réf:...../2023



## DISSERTATION

Presented for obtaining the **Academic MASTER diploma**

**Domain:** Science and Technology

**Branch:** Automation

**Specialty:** Automation and Industrial Computer Science

**By:** BENKIRAT Mohamed Achraf  
BOUGUERNE Omar

### Theme

**Integration of VLC technology into the smart factory  
and industry (4.0/5.0)**

Publicly defended, on 18/06/2023, in front of the jury composed of:

Mr. BABOURI Abdesselam	Professor	Univ.Guelma	President / supervisor
Mme.BOU CERREDJ Leila	MCA	Univ.Guelma	Examiner Principal
Mr. DEBECHE Mehdi	MAA	Univ.Guelma	Examiner
Mr. CHAABNA Ameer	MCB	Univ.Guelma	Co-Supervisor
Mr. CHOUABIA Halim	PhD student	Univ.Guelma	Co-Supervisor

**University Year : 2022/2023**

## ACKNOWLEDGMENTS

With heartfelt appreciation we would like to express our deepest gratitude to Allah whom without, this work could not be accomplished.

We are in debt to all the teachers we have met and learned from. A special gratitude goes to our kind, tolerant, and supportive supervisor, Pr. Babouri Abdessalam. Who devoted his time and energy to make this work possible. His enthusiasm and constant guidance were as a source of comfort. We are truly grateful for all his efforts.

We would like to take this opportunity to thank our teachers at the Department of electronic and automatic engineering, for their contributions in our intellectual growth and academic development.

We are immensely appreciative to the jury members Boucerredj Leila and Debeche Mehdi for the time invested in reading our thesis.

Foremost we are ineffably indebted to our family members. We would like to acknowledge the love and the patience of our parents. We are sincerely grateful for all their prayers and kind wishes. We would never forget their support and tolerance.

With sincere appreciation we would like to thank our siblings, for their faith and pleasant distractions.

Our worm thanks and appreciation goes to our friends, for their unwavering support. This work would have been more difficult without them.

Last but not the least, we would like to extend our deepest gratitude to all those who motivated us and helped to achieve this work both directly and indirectly. Thank you we are grateful beyond words

## **Abstract**

In this work, we studied and treated the artificial intelligence of robots, and make connection using Visible Light Communication (VLC) as a primary method for establishing connections with robots. More emphatically, deep learning and neural networks are the branches that will be used in this project as well. In addition, we are going to use python with its own library for image classification with pre-trained models for better performance. On the hardware side, we are going to use Arduino as a micro-controller, Raspberry Pi 4 as micro-processor, and VLC Car to build smart robot.

## **Résumé**

Dans ce travail, nous avons étudié et traité l'intelligence artificielle des robots, et établi des connexions en utilisant la communication par lumière visible (VLC) comme méthode principale pour établir des connexions avec les robots. Plus précisément, l'apprentissage profond et les réseaux neuronaux sont les branches qui seront utilisées dans ce projet. En outre, nous allons utiliser python avec sa propre bibliothèque pour la classification d'images avec des modèles pré-entraînés pour de meilleures performances. Sur le plan matériel, nous allons utiliser Arduino comme microcontrôleur, Raspberry Pi 4 comme micro-processeur, et VLC Car pour construire un robot intelligent.

## المخلص

في هذا العمل، درسنا وعالجنا الذكاء الاصطناعي للروبوتات، واتصلنا باستخدام الاتصال بالضوء المرئي كطريقة أساسية لإنشاء اتصالات مع الروبوتات. على وجه التحديد، فإن التعلم العميق والشبكات العصبية هي الفروع التي سيتم استخدامها في هذا المشروع أيضاً. بالإضافة إلى ذلك، سنستخدم البايثون مع مكتبته الخاصة لتصنيف الصور مع نماذج مدربة مسبقاً لأداء أفضل. على جانب الأجهزة، سنستخدم اردوينو كوحدة تحكم دقيقة، راسبيري 4 كمعالج صغير و سيارة الاتصال بالضوء المرئي، لبناء روبوت ذكي.

# TABLE OF CONTENTS

General introduction	1
<b>Chapter I: INTRODUCTION ABOUT ARTIFICIAL INTELLIGENCE, ROBOT, WIRELESS COMMUNICATION AND VISIBLE LIGHT COMMUNICATION</b>	
I.1. Introduction	3
I.2. Artificial Intelligence AI	3
I.2.1. History of AI	3
I.2.2. What Is Artificial Intelligence AI?	4
I.2.3. Types of Artificial Intelligence	4
I.2.4. Machine Learning ML	6
I.2.5. Neural Network NN	7
I.2.6. Deep Learning DL	8
I.2.7. Machine Learning vs. Deep Learning vs. Neural Networks	9
I.3. Robot	10
I.3.1. What is Robotics?	10
I.3.2. What is Robot	11
I.3.3. Types of Robots	13
I.3.4. Fields of Use	14
I.3.5. Control a robot	15
I.4. Wireless Communication: Introduction, Types and Applications	16
I.4.1. What is Wireless Communication?	17
I.4.2. What is Electromagnetic Wave?	18
I.4.3. Advantages of Wireless Communication	18
I.4.4. Disadvantages of Wireless Communication	20
I.4.5. Types of Wireless Communication Systems	21
I.5. Visible Light Communications	23
I.5.1. What Is the Electromagnetic Spectrum?	23
I.5.2. The VLC Block Diagram	25
I.5.3. LED	26
I.5.4. Canal LOS and NLOS	26
I.5.5. Photodiode	27
I.5.6. Received Signal Strength RSS	28
I.5.7. Light-Fidelity Li-Fi	28
I.5.8. Indoor	29
I.5.9. Outdoor	32
I.6. Conclusion	34
<b>CHAPTER II MODELLING AN INTELLIGENT CLASSIFICATION SYSTEM</b>	
II.1. Introduction	35
II.2. Python	35
II.2.1. What is Python?	35
II.2.3. Python Use Cases	36
II.2.4. Features and Benefits of Python	36
II.2.5. Python and AI	36
II.2.6. Tools Available for Python Development:	37
II.2.7. Comparing Python With Other Languages	37
II.3. Classification	38

II.3.1. Image classification	39
II.3.2. What is image classification?	40
II.3.3. Image Classification and object detection	40
II.3.4. How Image Classification Works	41
II.3.5. Algorithms and models: Supervised and unsupervised classification	42
II.3.5.1. Supervised	42
II.3.5.2. Unsupervised	43
II.3.6. Deep Neural Networks for Image Classification	43
II.3.7. Key takeaways	44
II.4. Neural networks	44
II.4.1. How do neural networks work?	44
II.4.2. Simple Neural Network architecture	45
II.4.3. Convolution Neural Network architecture	46
II.4.4. Deep Neural Network Architecture	47
II.4.5. What Is a Pre-trained Model?	48
II.4.5.1. Applications of Pre-training	49
II.4.6. What is MobileNetV2?	50
II.4.6.1. MobileNetV2 Architecture	51
II.5. My Experience	54
II.6. Our code of Simulation	55
II.7. Conclusion	60
<b>CHAPTER III COMMAND AND TRAINING OF USING VLC COMMUNICATION</b>	
III.1. Introduction	61
III.2. Raspberry pi	61
III.2.1. What Raspberry Pi models have been released?	63
III.2.2. How does Raspberry Pi work?	63
III.2.3. Top Uses of Raspberry Pi	64
III.2.4. Differences Between Arduino and Raspberry Pi	65
III.3. About this application	66
III.3.1. Our scenario	66
III.3.2. Command and training	67
III.3.3. Futur work	74
III.4. Conclusion	74
General conclusion	75
List of References	76
Annex 1	83
Annex 2	87

## LISTE DE FIGURES

- FIG I.1:** Artificial Intelligence
- FIG I.2:** Types of artificial intelligence
- FIG I.3:** Deep learning vs machine learning
- FIG I.4:** Subset of Artificial Intelligence
- FIG I.5:** Machine Learning system
- FIG I.6:** Neural Network
- FIG I.7:** Deep Learning
- FIG I.8:** Robot and machines
- FIG I.9:** Robot
- FIG I.10:** Robot type
- FIG I.11:** Wireless System
- FIG I.12:** Wireless type
- FIG I.13:** Propagation of electromagnetic wave
- FIG I.14:** Range of electromagnetic waves and the placement of the visible spectrum
- FIG I.15:** Visible Light Communication environment
- FIG I.16:** Block Diagram of the transceiver
- FIG I.17:** LED – Light Emitting Diode
- FIG I.18:** Geometry of LOS and NLOS channels
- FIG I.19:** Photodiode icon and examples
- FIG I.20:** Application scenarios for VLC/LiFi; (a) indoor applications and (b) outdoor applications
- FIG I.21:** Li-Fi internet networking.
- FIG I.22:** HOSPI robot
- FIG I.23:** V2V and I2V communication
- FIG I.24:** VLC Underwater Communication
- Fig II.1:** Image classification
- Fig II.2:** Image classification & Object detection
- Fig II.3:** Supervised & Unsupervised learning
- Fig II.4:** neural network architecture
- Fig II.5:** Convolutional neural network
- Fig II.6:** Concept of a neural network with the input values (green) and weights (blue).
- Fig II.7:** Timeline of how the state-of-the-art models have improved over time
- Fig II.8:** MobilenetV2
- Fig II.9:** Inverted Residual Blocks
- Fig II.10:** Depth wise convolution layer
- Fig II.11:** MobileNetV2 Architecture
- Fig III.1:** Raspberry pi
- Fig III.2:** Concept of our System
- Fig.III.3:** Our Raspberry pi + webcam
- Fig III.4:** Result of our code
- Fig III.5:** Emitter + Receiver
- Fig III.6:** Raspberry + Emitter connection



**Fig III.7:** VLC-Car

**Fig III.8:** Embodiment of our idea

## **List of Tables**

**Tab.I.1:** Machine Learning vs Deep Learning

**Tab.II.1 :** Python vs Java

**Tab.II.2 :** Python vs JavaScript

**Tab.II.3:** Python vs C++

## **Liste of abbreviation**

VLC: Visible Light Communication

LED: Light-Emitting Diode

AI: Artificial intelligence

ML: Machine Learning

ANN: Artificial neural networks

DL: Deep Learning

SNN: simulated neural network

CNC: Computerized Numerical Control

RUR: Rassum's Universal Robots

TV: Television

GPS: Global Positioning System

RF: Radio Frequency

GSM: Global System for Mobile communication

RFID: Radio Frequency Identification

EM: Electromagnetic

4G: Fourth Generation

WI-FI: Wireless Fidelity

WLAN: Wireless Local Area Network

IR: Infrared

OWC: Optical wireless communication

GBPS: Gigabits per second

LI-FI: Light Fidelity

THZ: Terahertz

5G: Fifth generation technology standard for broadband cellular networks

KHz: Kilohertz

GHz: Gigahertz

RSS: Received Signal Strength

DB: Decibels

UV: Ultraviolet radiation

VL: Viral load

P2P: point-to-point

FIG: Figure

V2I: Vehicle-To-Infrastructure

V2X: Vehicle-to-everything

V2V: Vehicle-To-Vehicle

LoT: Internet of Things

LOS: Line Of Sight

MRI: Magnetic Resonance Imaging

HF: High frequency

TOA: Time of Arrival

AOA: angle of arrival

ITS: Intelligent Transport Systems

SNR: Signal to Noise Ratio

IDE: Integrated Development Environment

HTML: Hypertext Markup Language

OOPs: Object-Oriented Programming

CNN: Convolution Neural Network

ChatGPT: Chat Generative Pre-Trained Transformer

PC: personal computer

GPU: Graphics processing unit

SD: Secure Digital

OS: operating system

HDMI: High-Definition Multimedia Interface

VPN: virtual private network

CUPS: Common Unix Printing System

HTTP: Hypertext Transfer Protocol

PHP: Hypertext Preprocessor

MHz: Megahertz

API: Application Programming Interface

PD: Photodiode

## General introduction

Although, artificial intelligence seems like a new technique, based on our research, WE concluded that it is deeply rooted in the past. According to Greek mythology, the concepts of artificial intelligence were used, and the model of artificial neurons was first presented in 1943 by Warren McCulloch and Walter Bates. Seven years later in 1950 Alan Turing published a research paper about artificial intelligence entitled "Computer Machines and Intelligence". In order to reduce the human burden, they wanted to rank it along with robots, which provide many benefits, especially the tasks that most people cannot undertake or perform. Therefore, it IS also useful in industrial environments.

VLC is considered as a new wireless connection that uses visible light not only for lighting, but also for data transmission. Its main advantage is the use of LED lighting systems, which makes them more popular and significantly reduces the installation costs.VLC technology is evolving against the backdrop of the growing demand for wireless communications in many fields. More emphatically, wireless communications have begun to involve in several areas, thus it is clear that there is a need for new wireless communication technology. VLC allows high-speed connections due to the large bandwidth it offers.

So as to apply them, we have to program these modern technologies, as an example of these programs is the so- called python, this latter, Works in a widely used and high-level programming language. It is also used for programming robots. Consequently, to improve their performance, they should be combined with classification or so-called image classification, to make its work or vision similar. And to ensure the success of human compensation in multiple fields.

This thesis consists of three chapters. The first deals with different types of artificial intelligence, robots, wireless communications and VLC. While the second chapter is devoted to the python programming language and classification. In the third chapter, we will study how to program and operate the images classification technology through Phyton using raspberry pi with cooperation of VLC technology (transmitter and receiver).

## **Chapter I**

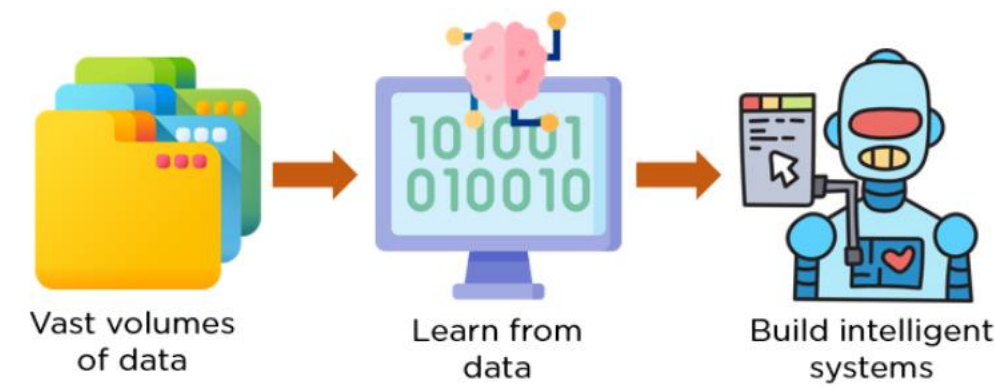
# **INTRODUCTION ABOUT ARTIFICIAL INTELLIGENCE, ROBOT, WIRELESS COMMUNICATION AND VISIBLE LIGHT COMMUNICATION**



### I.1. Introduction

In our time, technological advancement has become almost impossible due to the speed of development. However, focusing on specific areas allows us to keep up with this progress. In this manuscript, we will focus on several areas. To start with artificial intelligence, that is a breakthrough in technology. Then, Robots and the development of future devices. In addition to wireless communication for its flexibility of use, made it increasingly popular. And finally, Visible Light Communication (VLC), which is based on LED lights and can serve as a complementary technology to current wireless frequency standards.

### I.2. Artificial Intelligence AI



**Fig I.1.**Artificial Intelligence [1]

#### I.2.1. History of AI

The origins of AI can be traced back to the 1950s when researchers started toying with the idea of creating machines with human-like learning and thinking capabilities. The Dartmouth Conference, which took place in 1956 and was led by John McCarthy, Claude Shannon, Nathaniel Rochester, and Marvin Minsky, is often seen as a seminal moment in the history of AI. Since then, AI has undergone a multitude of developmental phases. [2]

## I.2.2. What Is Artificial Intelligence AI?

AI refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving.

The ideal characteristic of artificial intelligence is its ability to rationalize and take actions that have the best chance of achieving a specific goal. [2]

Artificial intelligence is based on the principle that human intelligence can be defined in a way that a machine can easily mimic it and execute tasks, Specifically, it focuses on investigating how the human brain functions, resonates, plans, makes decisions, represents knowledge, learns from the environment.

Artificial intelligence can be divided into two different categories: weak and strong. Weak AI tends to be simple and single-task oriented, for example, answering questions posed to virtual assistants. In contrast, strong AI carries out tasks that are more complex and human-like and can be found in applications such as self-driving cars or hospital operating rooms.

## I.2.3. Types of Artificial Intelligence

Artificial Intelligence can be divided based on capabilities and functionalities. [3]

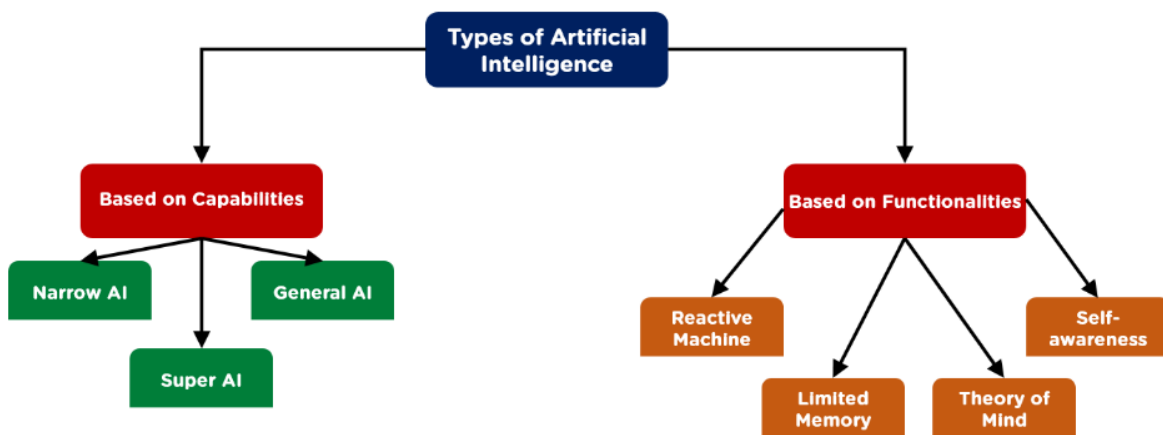


Fig I.2.Types of Artificial Intelligence [3]

There are three types of Artificial Intelligence-based on capabilities.[4]

- **Narrow AI:** Designed to complete very specific actions and it cannot independently learn skills beyond its design.
- **General AI:** Designed to learn, think, and perform at similar levels to humans, the goal of designing artificial general intelligence is to be able to create machines that are capable of performing multifunctional tasks and acting as lifelike.
- **Super AI:** AI able to surpass the knowledge and capabilities of humans it's theorized to become the backbone technology of completely self-aware AI and other individualistic robots.

Under functionalities, we have four types of Artificial Intelligence Functionality concerns how an AI applies its learning capabilities to process data, responds to stimuli and interacts with its environment. As such, AI can be sorted by four functionality types.

- **Reactive Machines:** Reactive machines are AI systems that have no memory and are task-specific, meaning that an input always delivers the same output.
- **Limited Theory:** Limited memory AI, unlike reactive machines, can look into the past and monitor specific objects or situations over time.
- **Theory of Mind:** Theory of mind AI could have the potential to understand the world and other entities' thoughts and emotions.
- **Self-awareness:** Could have the potential to be aware of themselves, and their ability to sense or predict others' feelings in the final stage of AI.

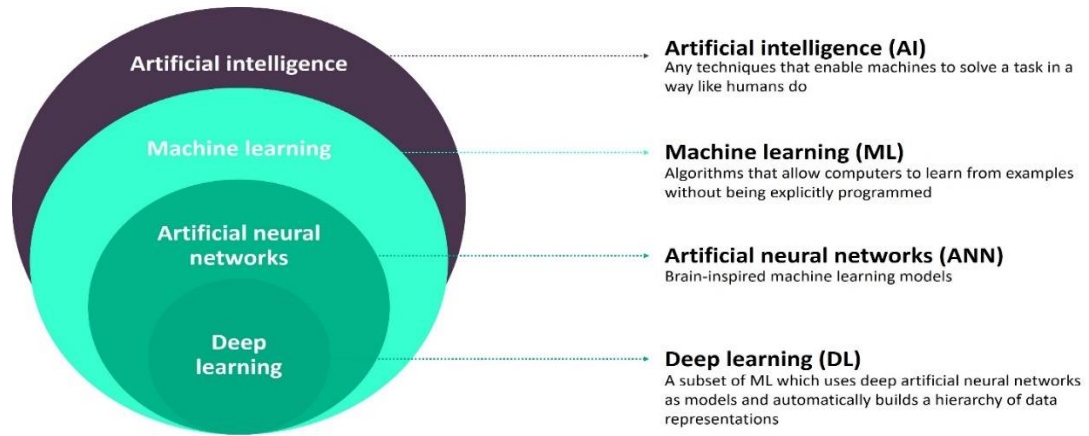


Fig I.3. Deep Learning Vs Machine Learning [4]

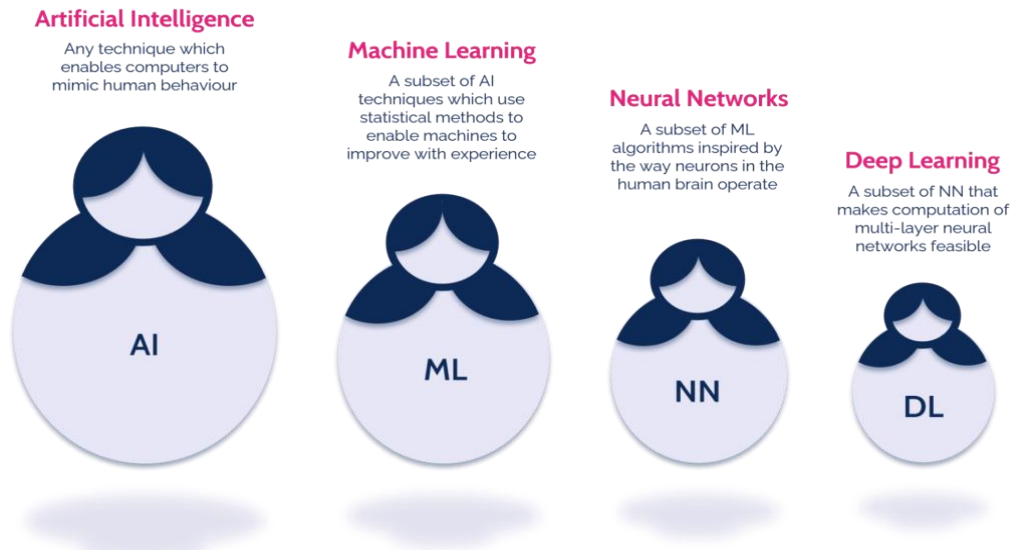


Fig I.4. Subset of Artificial Intelligence [4]

### I.2.4. Machine Learning ML

A machine learning algorithm is fed data by a computer and uses statistical techniques to help it “learn” which focuses on the use of data and algorithms to imitate humans’ way of learning, gradually improving its accuracy, without necessarily having been specifically programmed for that task. Instead, ML algorithms use historical data as input to predict new output values. Thus, ML consists of both supervised learning (where the expected output for the input is known

thanks to labeled data sets) and unsupervised learning (where the expected outputs are unknown due to the use of unlabeled data sets).[5]

### Machine Learning

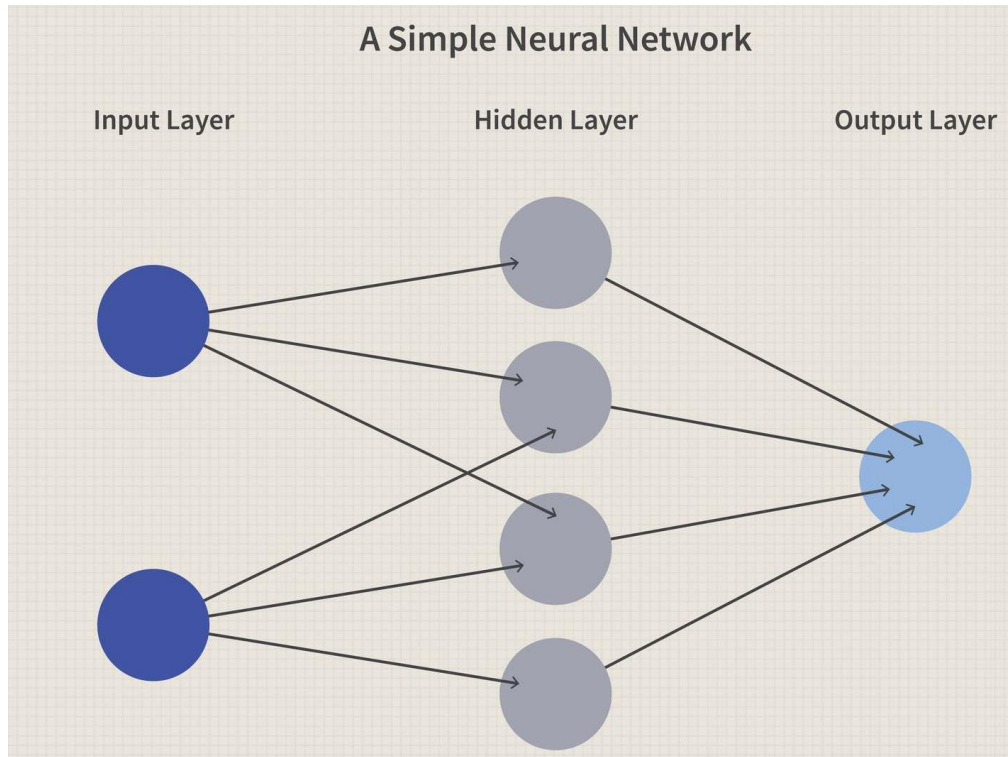


**Fig I.5.** Machine Learning System [6]

#### I.2.5. Neural Network NN

Neural Networks, also known as Artificial Neural Networks (ANNs) or Simulated Neural Networks (SNNs), are a subset of [machine learning](#) and are at the heart of [deep learning](#) algorithms. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to one another.[7]

Artificial neural networks (ANNs) are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold.



**Fig I.6.** Neural Network [8]

### I.2.6. Deep Learning DL

Deep learning is a subset of machine learning that trains a computer to perform human-like tasks, such as speech recognition, image identification, and prediction-making. It improves the ability to classify, recognize, detect, and describe using data. [9]

## Deep Learning

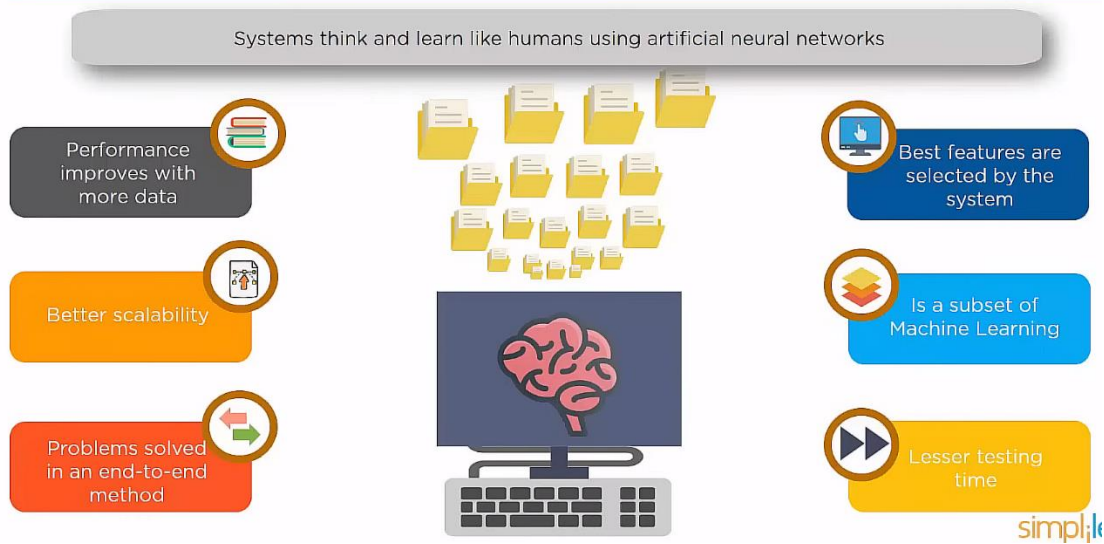


Fig I.7. Deep Learning [6]

### I.2.7. Machine Learning vs. Deep Learning vs. Neural Networks

Since deep learning and machine learning tend to be used interchangeably, it's worth noting the nuances between the two. Machine learning, deep learning, and neural networks are all sub-fields of artificial intelligence. However, neural networks are actually a sub-field of machine learning, unlike deep learning that is a sub-field of neural networks.[9]

The way in which deep learning and machine learning differ is in how each algorithm learns. "Deep" machine learning can use labeled datasets, also known as supervised learning, to inform its algorithm, but it doesn't necessarily require a labeled dataset. Deep learning can ingest unstructured data in its raw form (e.g., text or images), and it can automatically determine the set of features which distinguish different categories of data from one another. This eliminates some of the human intervention required and enables the use of larger data sets. You can think of deep learning as "scalable machine learning".

Classical, or "non-deep", machine learning is more dependent on human intervention to learn. Human experts determine the set of features to understand the differences between data inputs, usually requiring more structured data to learn.

Neural networks, or artificial neural networks (ANNs), are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network by that node.

The “deep” in deep learning is just referring to the number of layers in a neural network. A neural network that consists of more than three layers—which would be inclusive of the input and the output—can be considered a deep learning algorithm or a deep neural network. The neural network that has only three layers is just a basic neural network.

**Table I.1:** Machine Learning vs Deep Learning

	<b>Machine Learning</b>	<b>Deep Learning</b>
<b>Training dataset</b>	<b>Small</b>	<b>Large</b>
<b>Choose your own features</b>	<b>Yes</b>	<b>No</b>
<b># of classifiers available</b>	<b>Many</b>	<b>Few</b>
<b>Training time</b>	<b>Short</b>	<b>Long</b>

### I.3.Robot

#### I.3.1. What is Robotics?

Robotics is a branch of engineering that involves the conception, design, manufacture and operation of [robots](#). The objective of the robotics field is to create intelligent machines that can assist humans in a variety of ways. [10]

Robotics can take on a number of forms. A robot may resemble a human, or it may be in the form of a robotic application.



While the field of robotics and exploration of the potential uses and functionality of robots have grown substantially in the 20th century.

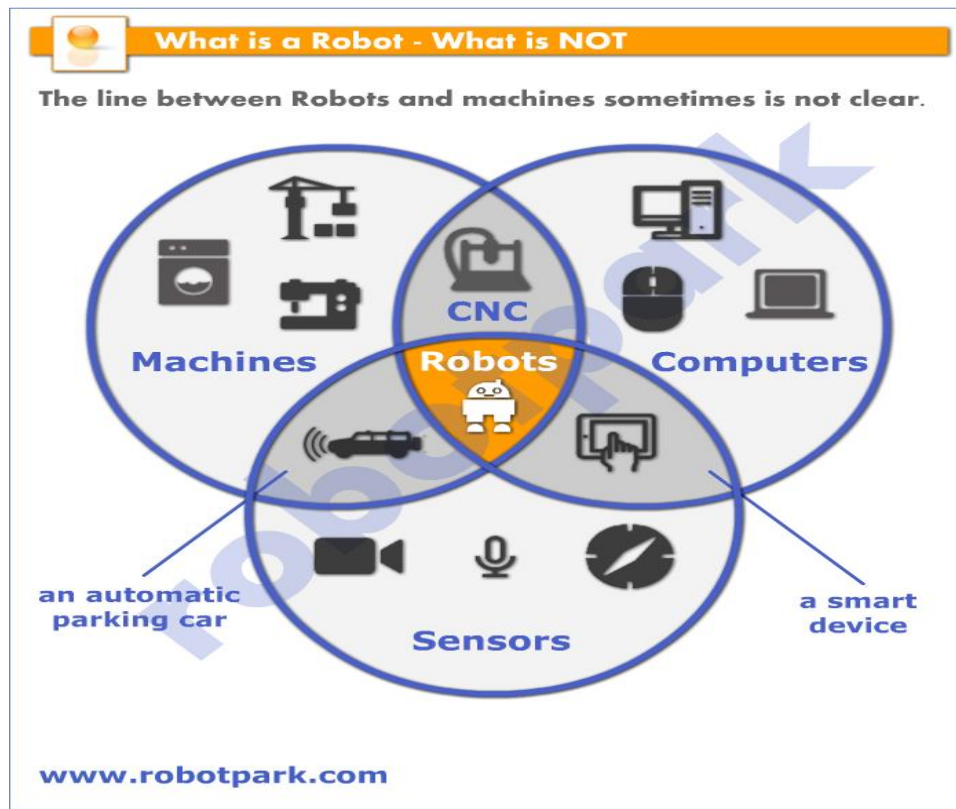


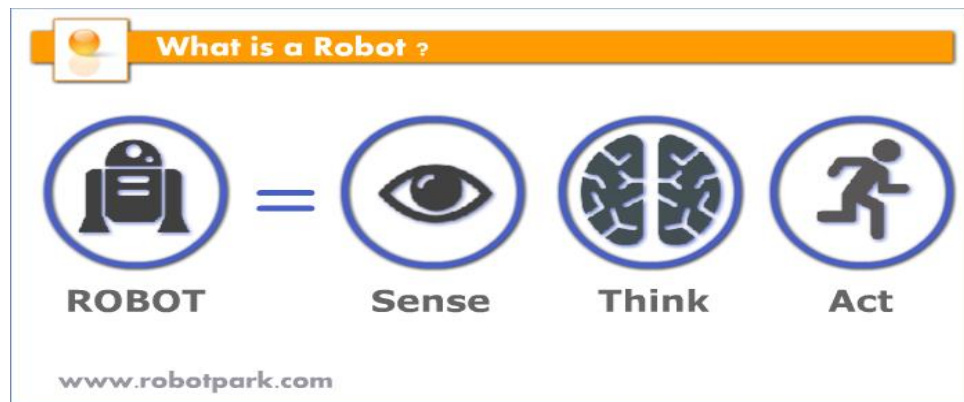
Fig I.8. Robot and Machines [11]

### I.3.2. What is Robot

Is a machine designed to resemble a human being and to perform specific tasks based on pre-programmed instructions. The term “robot” has its roots in the Slavic word “robota,” which translates to “forced laborer.” The first use of the word robot was by the Czech novelist Karel Capek in a 1920 play titled *Rassum’s Universal Robots (RUR)*. Initially developed in the 1960s, robots are typically constructed using metallic and composite materials. These machines are designed to respond directly to human commands. What was once a futuristic vision in movies, robots are now utilized across a wide range of industries, and their importance to the future of humanity cannot be overstated. Robotic technology has revolutionized several industries such as national defense, healthcare, manufacturing, homeland security, education, and even consumer goods. In fact, doctors are already utilizing robotics to perform intricate surgeries that would

have been deemed impossible before. The significance of robots lies in their ability to execute dangerous tasks that are beyond human capacity.[12]

There is no one simple definition of what is a robot. There are lots of different kinds of machines, some of them is called a robot, some of them not, only being anthropomorphic is not enough for something to be called a robot. A robot has to Sense, Think and Act. A human body shaped mechanical object would not be a robot. So we can say the physical appearance of a machine is not enough to name it as a robot.



**Fig I.9.**Robot [11]

- **Sense**

Robots use different kinds of sensors to interact with their environment. These could be light sensors, temperature sensors, pressure sensors, proximity sensors, and distance sensors...

- **Plan**

Robots are supposed to show some kind of intelligence, like solving complex problems, having social interactions or the ability to learn virtually anything. They accomplish all of this through Artificial Intelligence. Artificial intelligence allows a robot to assess its status and generate a plan according to this assessment to act. If your machine lacks significant AI, it probably isn't a robot.

- **Act**

In order for a robot to engage with its surroundings, it must do more than simply sense. The concept of "acting" takes on a broader significance in the context of robotics, with numerous forms of expression. For a robot, acting refers to the capacity to manipulate its environment in a variety of ways. These tasks can range from simple actions like moving objects to complex actions like performing surgical procedures or exploring outer space.

So, "acting robot" could potentially refer to a robot that is designed to mimic human emotions and actions. [12]

### **I.3.3. Types of Robots**

It's not easy to define what robots are, and it's not easy to categorize them either. Each robot has its own unique features, and as a whole robots vary hugely in size, shape, and capabilities. Still, many robots share a variety of features. [11]

- Stationary (Manipulator) Robots
- Wheeled robots
- Legged Robots
- Flying Robots
- Swarm Robots
- Swarm Robots
- Micro Robots
- Nano Robots
- Soft/Elastic Robots

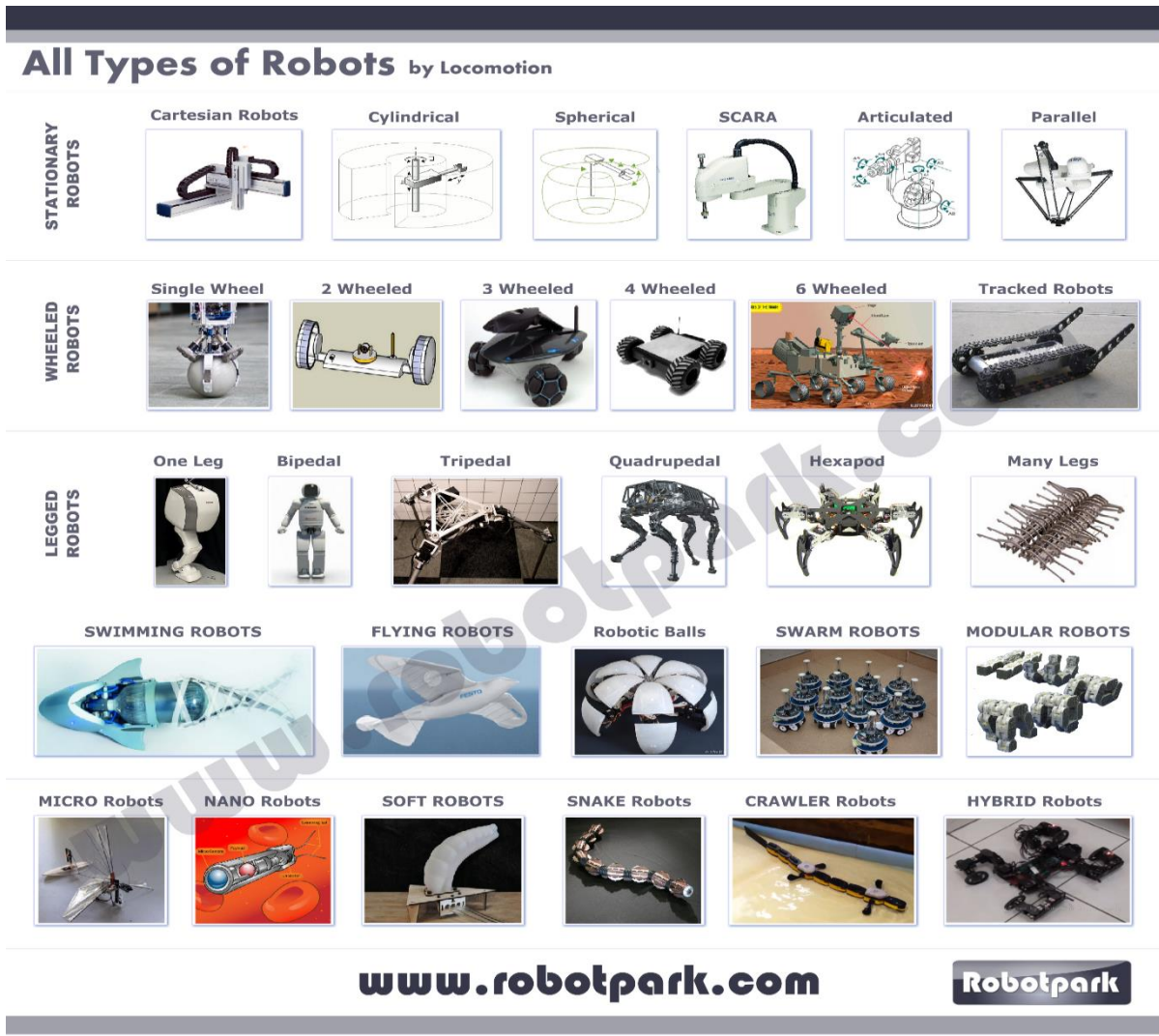


Fig I.10. Robot Type [11]

### I.3.4. Fields of Use

- **Industrial**

Industrial robots are used in the manufacturing industry, particularly for tasks that require great speed, precision, and endurance. They can be found on assembly lines across all types of industries, handling jobs that range from loading materials to welding parts. Their main advantage over humans is their precision; they do not tire or make mistakes, even when they perform repetitive tasks. [13]

- **Medical**

Medical robots are developed to perform a variety of tasks in healthcare settings. They can be found in hospitals, doctors' surgeries, and ambulances.

- **Military & Security**

Robots are advantageous as they can be freely used in dangerous places and avoid putting human life at risk. They are basically used in locations that are hard to reach or are prone to health hazards and accidents.

- **Education**

Education robots are designed to be used in schools and other educational institutions. They can help teachers to deliver lessons or allow students to investigate topics that interest them without the risk of injury. These robots are made particularly for children.

And many robots are used to day Consumer, Self-Driving Cars, Disaster Response, Research Humanoids, Tele presence Exoskeletons, Underwater and Transportation.

### **I.3.5. Control a robot**

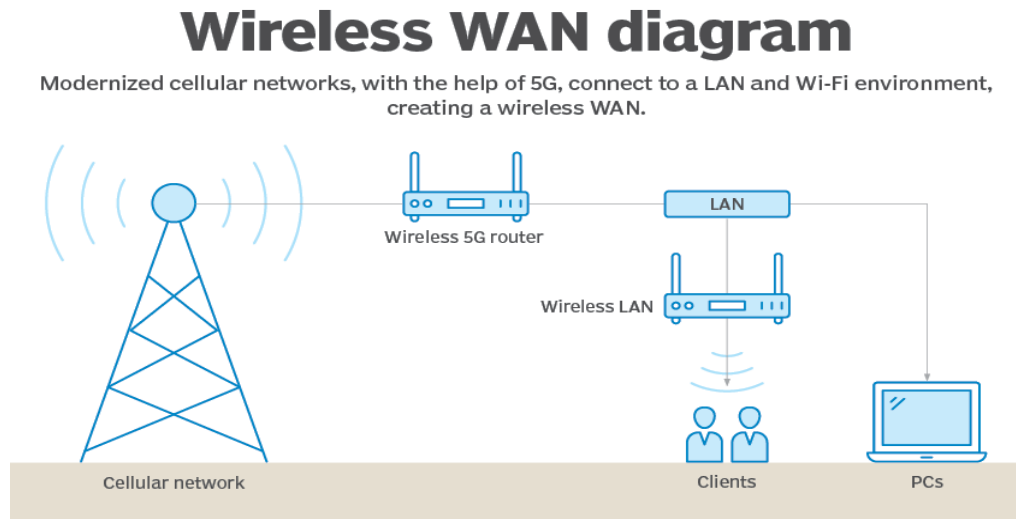
To control a robot, there are various techniques based on the application and the robot type. Below are some techniques that are often used: Manual control, a method of direct human intervention through input devices like joysticks, governs the motion and behavior of robots. This approach is frequently employed for straightforward, rudimentary robots carrying out undemanding tasks, such as remote-controlled toys.

To execute a designated function, programmed control is necessary. This involves crafting a set of instructions, or a program, for the robot to follow. It's possible to write this program in various programming languages and either store it on the robot's onboard computer or on a separate device.

By relying on sensors like microphones, cameras and pressure sensors, sensor-based control enables robots to detect and automatically respond to alterations in their surroundings. This not only allows them to adapt to new conditions but also execute intricate tasks with greater precision.

AI control utilizes advanced algorithms and machine learning techniques to train robots to learn and evolve. This innovative approach proves especially beneficial for tasks requiring robots to analyze complicated data inputs and make informed decisions.

In determining the most suitable control technique for a robot, the technology at hand and the task requirements play an instrumental role. It is typical to employ a fusion of techniques to deliver optimum outcomes.



**Fig I.11.**Wireless System [14]

### I.4. Wireless Communication: Introduction, Types and Applications

Wireless Communication is the fastest-growing and most vibrant technological area in the communication field. Wireless Communication is a method of transmitting information from one point to another, without using any connection like wires, cables, or any physical medium.

With the help of Wireless Communication, the transmitter, and receiver can be placed anywhere between a few meters (like a T.V. Remote Control) to a few thousand kilometers (Satellite Communication). Systems in our day-to-day life are: Mobile Phones, GPS Receivers, Remote Controls, Bluetooth Audio, Wi-Fi, etc. [13] [14]

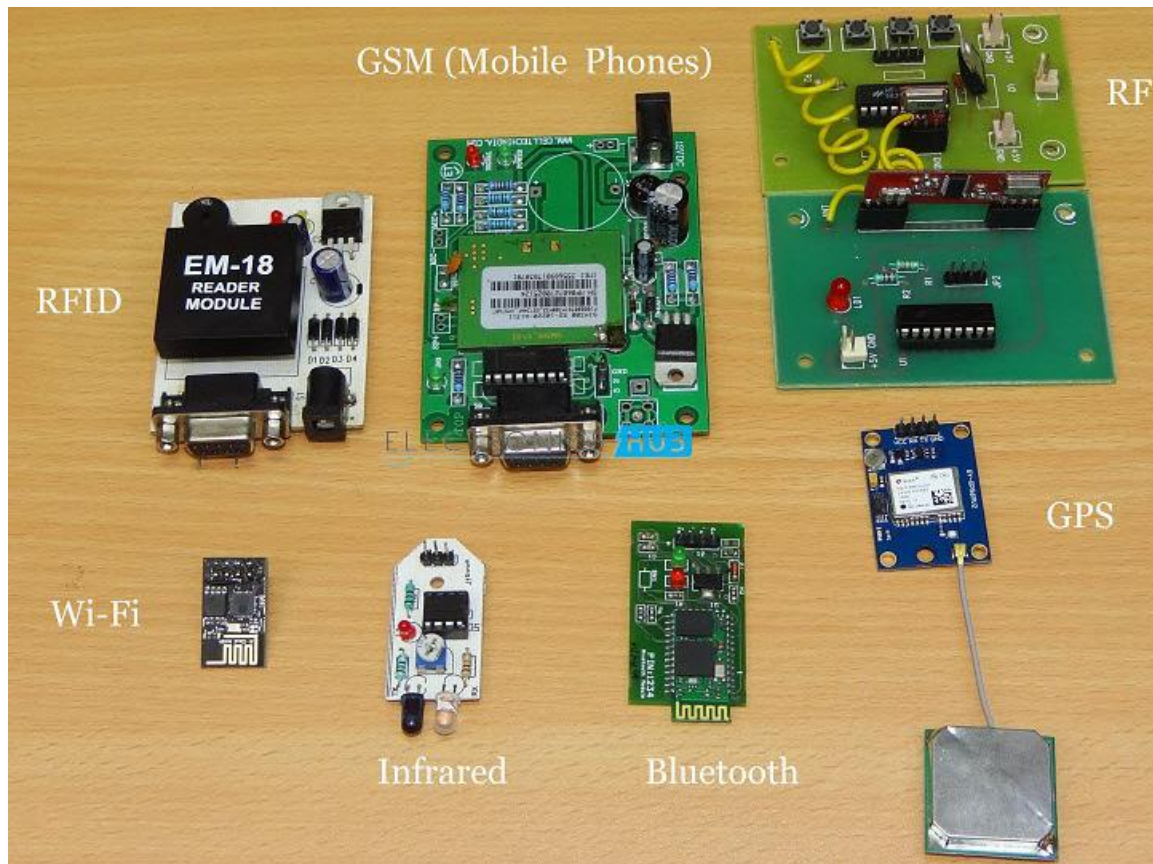


Fig I.12. Wireless Type [14]

#### I.4.1. What is Wireless Communication?

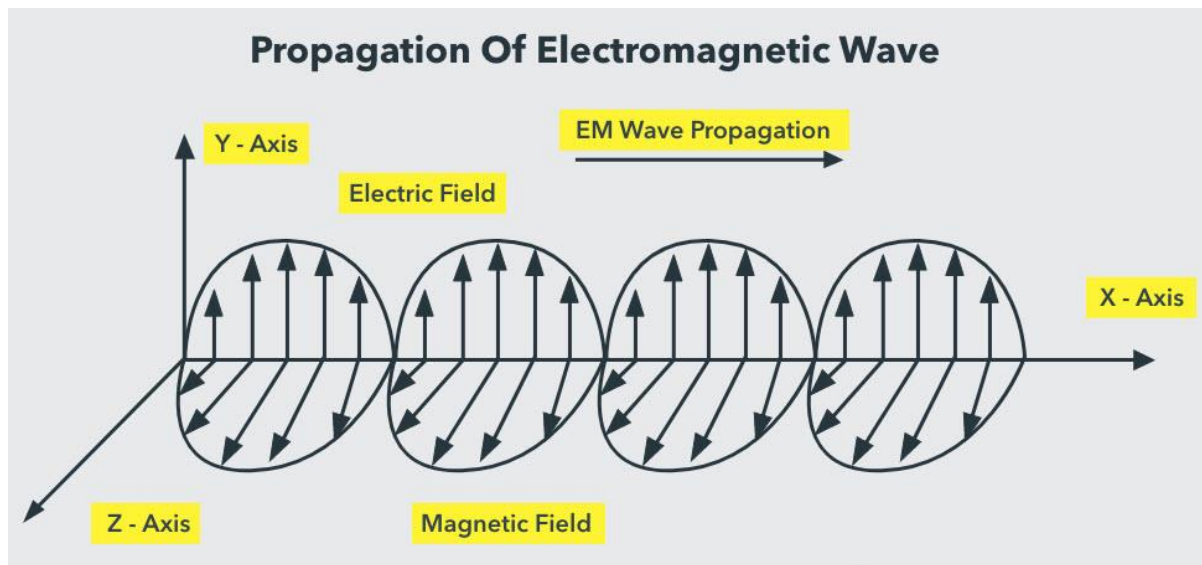
Wireless communication involves the transmission of information over a distance without the help of wires, cables or any other forms of electrical conductors.

Wireless communication is a broad term that incorporates all procedures and forms of connecting and communicating between two or more devices using a wireless signal through wireless communication technologies and devices.[14][15]

Data travels through electromagnetic signals broadcast from sending facilities to intermediate and end-user devices...

### I.4.2. What is Electromagnetic Wave?

Electromagnetic waves or EM waves are waves that are created as a result of vibrations between an electric field and a magnetic field. In other words, EM waves are composed of oscillating magnetic and electric fields.[16]



**Fig I.13.** Propagation of Electromagnetic Wave [17]

### I.4.3. Advantages of Wireless Communication

There are numerous advantages of Wireless Communication Technology, Wireless Networking and Wireless Systems over Wired Communication like Cost, Mobility, Ease of Installation, and Reliability etc. [18]

- **Cost- effectiveness**

Wired communication entails the use of connection wires. In wireless networks, communication does not require elaborate physical infrastructure or maintenance practices. Hence the cost is reduced.[19]



**Example**—Any company providing wireless communication services does not incur a lot of costs, and as a result, it is able to charge cheaply with regard to its customer fees.

- **Flexibility**

Wireless communication enables people to communicate regardless of their location. It is not necessary to be in an office or some telephone booth in order to pass and receive messages. Miners in the outback can rely on satellite phones to call their loved ones, and thus, help improve their general welfare by keeping them in touch with the people who mean the most to them.

- **Convenience**

Wireless communication devices like mobile phones are quite simple and therefore allow anyone to use them, wherever they may be. There is no need to physically connect anything in order to receive or pass messages.

**Example** – Wireless communications services can also be seen in Internet technologies such as Wi-Fi. With no network cables hampering movement, we can now connect with almost anyone, anywhere, anytime.

- **Accessibility**

The wireless technology helps easy accessibility as the remote areas where ground lines can't be properly laid, are being easily connected to the network.

**Example** –In rural regions, online education is now possible. Educators no longer need to travel to far-flung areas to teach their lessons. Thanks to live streaming of their educational modules.

- **Constant connectivity**

Constant connectivity also ensures that people can respond to emergencies relatively quickly.

**Example** –A wireless mobile can ensure you a constant connectivity though you move from place to place or while you travel, whereas a wired land line can't.

- **Ease of Installation**

The setup and installation of wireless communication network's equipment and infrastructure is very easy as we need not worry about the hassle of cables. Also, the time required to setup a wireless system like a Wi-Fi network for example, is very less when compared to setting up a full cabled network.

#### **I.4.4. Disadvantages of Wireless Communication**

- **Security**

Utilizing wireless networks raises serious security concerns. An improperly installed or managed wireless network could pose serious security risks. A wireless network does not require physical connections like wires. They merely require a wireless adaptor, which raises the risk of hacking because hackers may easily enter the network. A worsening of the situation may occur if the network has password protection.[20]

- **Speed**

The wireless network's speed is less rapid than that of wired networks. In a wireless network, file transfers and file sharing are much slower. The location of the user on the network affects performance as well. The connection degrades more rapidly the further the user is away from the network. For vast areas or buildings, this is a major issue.

- **Prone to Interference**

There is a high chance of interference and jamming in wireless networks because of environmental variables like dust storms or fog. When too many users are in one area, the air band that the signals are transported across can become overloaded. A wireless network could become unreliable due to fog, radiation, radio waves, or any other type of interference since wireless networks are particularly prone to interference.

- **Requires Basic Computer Knowledge**

A basic understanding of computers is necessary to set up a wireless network. Installing a wireless network might be difficult for people with little computer skills. Security is risky, and those networks are easy targets for hackers.

- **Health Concerns**

Continuous exposure to any type of radiation can be hazardous. Even though the levels of RF energy that can cause the damage are not accurately established, it is advised to avoid RF radiation to the maximum.

### **I.4.5. Types of Wireless Communication Systems**

Different types of Wireless Communication Systems are developed. Some of the important Wireless Communication Systems available today are Cordless telephones, Zigbee, GPS, Wi-Fi, satellite television, and wireless computer parts. Current wireless phones include 4 and 5G networks, Bluetooth, Infrared Communication, and Wi-Fi AND Li-Fi technologies.

- **Satellite Communication**

Satellite Communication System is an important type of Wireless Communication. Satellite Communication Networks provide worldwide coverage independent of population density. Satellite Communication Systems offer telecommunication (Satellite Phones), positioning and navigation (GPS), broadcasting, internet, etc. Other wireless services like mobile, television broadcasting, and other radio systems are dependent on Satellite Communication Systems.[19]

- **Global Positioning System GPS**

GPS is solely a subcategory of satellite communication. GPS provides different wireless services like navigation, positioning, location, speed, etc. with the help of dedicated GPS receivers and satellites.[15]

- **Wireless Local Area Network WLAN**

Wi-Fi (Wireless Fidelity): Wi-Fi is a popular wireless networking technology that uses radio waves to transmit data between devices. Wi-Fi networks can be set up in homes, businesses, and public places such as coffee shops, airports, and libraries. Wi-Fi networks offer high-speed internet access and support a wide range of devices, including smartphones, tablets, laptops, and smart home devices.[19]

- **Infrared Communication**

Infrared Communication is another commonly used wireless communication in our daily lives. It uses the infrared waves of the Electromagnetic (EM) spectrum. Infrared (IR) Communication is used in remote controls of Televisions, cars, audio equipment, etc. [21]

- **Optical wireless communication**

Optical Wireless Communication (OWC) is a type of wireless communication that uses light to transmit data between devices. This technology has gained popularity in recent years due to its ability to provide high-speed data transfer rates, high security, and low interference.[22]

OWC uses a transmitter to send information in the form of modulated light waves, which are received by a Photodetector (PD) at the other end. Light waves can be transmitted using a variety of methods, including infrared (IR) and visible light (VLC/Li-Fi).

One of the main advantages of OWC is its high data transfer rates, which can reach up to several gigabits per second (Gbps). This is much faster than traditional wireless communication methods, such as Wi-Fi and Bluetooth.

OWC is also highly secure, as light waves do not penetrate walls or other obstacles, making it difficult for third parties to intercept the data transmission. Additionally, the use of directional antennas can further increase security by limiting the area in which the data can be received.

Another advantage of OWC is its low interference, as it does not use radio frequencies that can be affected by other wireless devices or environmental factors.

OWC has a wide range of applications, including indoor and outdoor wireless communication, Li-Fi (which uses visible light for indoor wireless communication), and underwater communication.

### I.5. Visible Light Communications

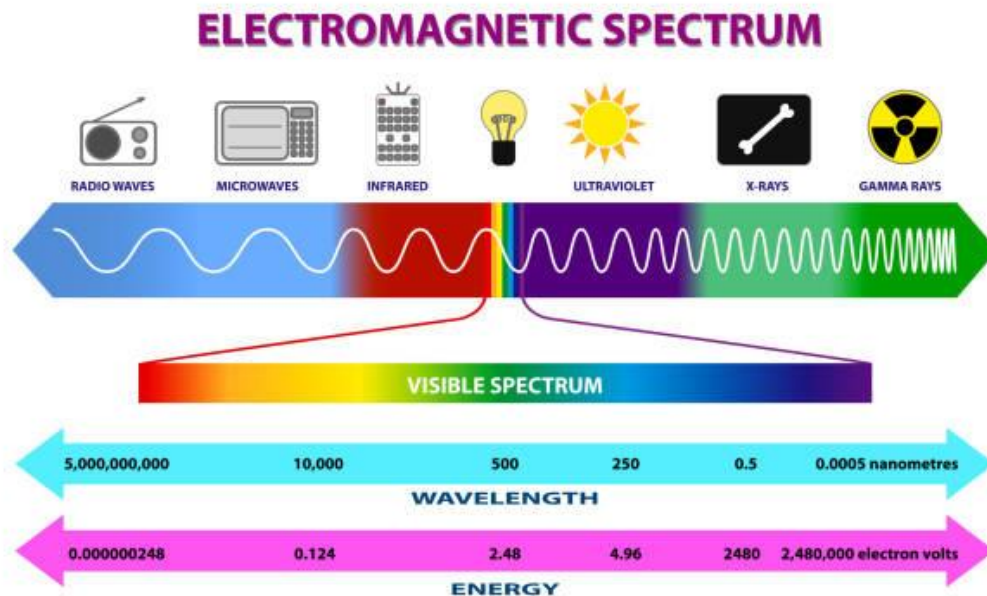
#### I.5.1. What Is the Electromagnetic Spectrum?

The electromagnetic spectrum can be defined by all the kinds of light in the universe, including those that the human eye cannot see and in fact the majority of the lights are invisible for humans.

A relatively small percentage of the electromagnetic spectrum is represented by the light we can see, which is made up of each hue in the rainbow.

In addition to visible light, there are other forms of radiation that cannot be seen by human eyes, such as radio waves, microwaves, infrared radiation, ultraviolet rays, X-rays, and gamma rays.

All light, or electromagnetic radiation, travels through space at 186,000 miles (300,000 kilometers) per second — the speed of light. That’s about as far as a car will go over its lifetime, traveled by light in a single second. [23]



**Fig I.14.** Range of Electromagnetic Waves and the Placement of the Visible Spectrum [23]

Within the subset of OWC, there's a significant interest in VLC. VLC, as the name indicates, is an OWC technology which utilizes the VL spectrum. It is anticipated that VLC can be complementary technology to the prevailing RF systems and can fulfill the requirements of IoT [24]

- The Visible Light Communication (VLC) refers to a technology that uses visible light as a Signal Transmitter, The Air as Transmission Medium and a suitable photodiode as a Signal Receiver. It uses Light between 400 THz (780 nm) and 800 THz (375 nm) as an optical carrier for data communication and illumination [25].
- The heart of this technology is the high brightness LED's [26]. Light Emitting Diode can be switched **on** and **off** faster since operating speed of LED'S is less than  $1\mu\text{s}$  , causing the light source to appear continuously which make a kind of data transmission using binary codes as switching "**on** " is a logical "**1** " and switching "**off** " is a logical "**0**" , in other word .
- Visible Communication is typically implemented using white LED light bulbs. These devices are normally used for illumination by applying constant current through the LED, by fast variations of current, the optical output can be made to vary at extreme high speeds. Unseen by the human eye [26],[27], this variation is used to carry high speed data. When signals reach the receiver through the indoor wireless channel, the photodiode will convert the optical signals to electrical ones and the original information will be recovered. The visible light communication based on LED is a novel developing technique in the optical wireless communication field [28].

VLC provides cable free communication and has also great benefits, such as, no Interference with Radio Frequency (RF) devices, Safety, and Free of Band Licensing so it is considered to be a safe alternative to radio frequencies that it can be used in hospitals, space stations and other electromagnetic interference sensible locations as shown in fig I.15. [29], [30], [31].

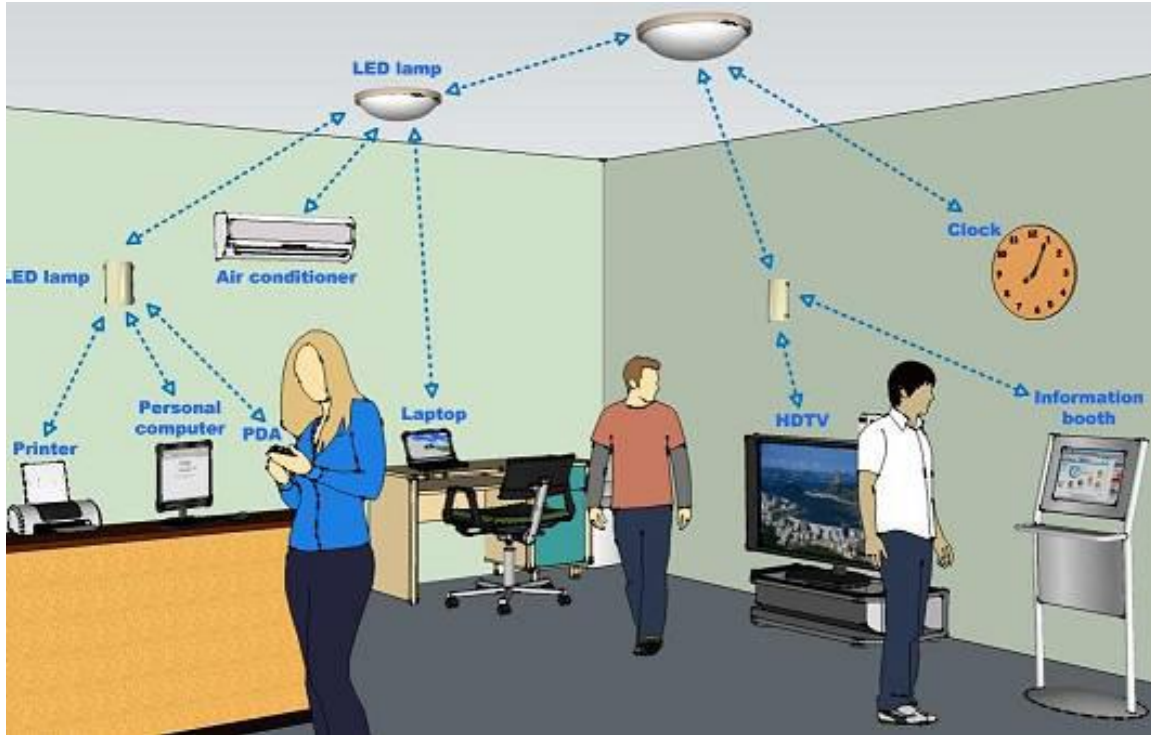


Fig I.15. Visible Light Communication Environnement [23]

I.5.2. The VLC Block Diagram

Fig I.16. shows block diagram of the desired transceiver system which consists of two parts, the first part is the transmitter section which contains a computer (1) works as data source, buffer, amplifier and white LED which transmits sends data by flashing light at speeds undetectable to the human eye [32],[33]. This signal reaches the receiver, the second part, and the Photodiode will convert the optical signal to electrical ones and the original signal will be recovered and appeared on the computer (2) and the reverse process is also performed if we want to send data from computer (2) to computer (1), where it is a Transceiver VLC system.

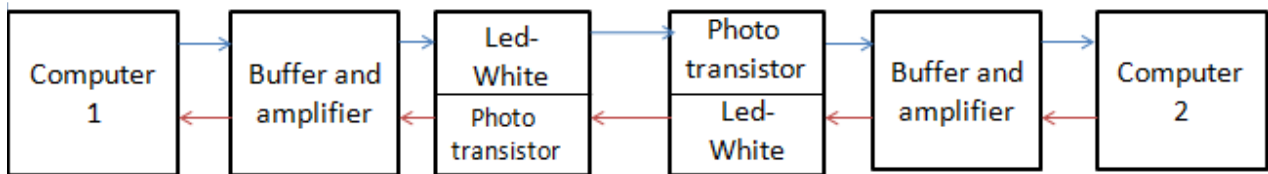
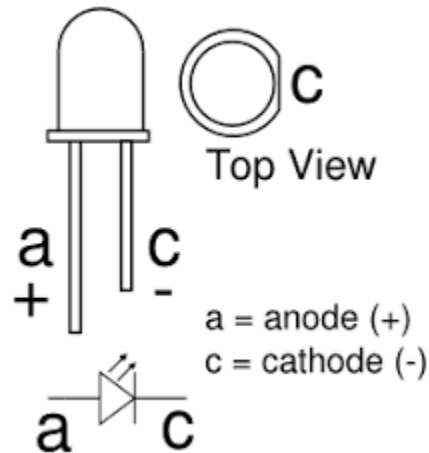


Fig I.16. Block Diagram of the Transceiver [23]

### I.5.3. LED

Due to its characteristics such as price, LED light bulbs have become the main medium used for Visible Light Communication. In addition, LED light bulbs became increasingly popular, integrating various environments where it would be advantageous to use light as a form of communication [34].



**Fig I.17.** LED – Light Emitting Diode

### I.5.4. Canal LOS and NLOS

There are many criteria for classifying the different types of optical links. As the presence of a direct link between the transmitter and the receiver (LOS, NLOS), the opening angle of the transmitter and the field of view (FOV) of the receiver.



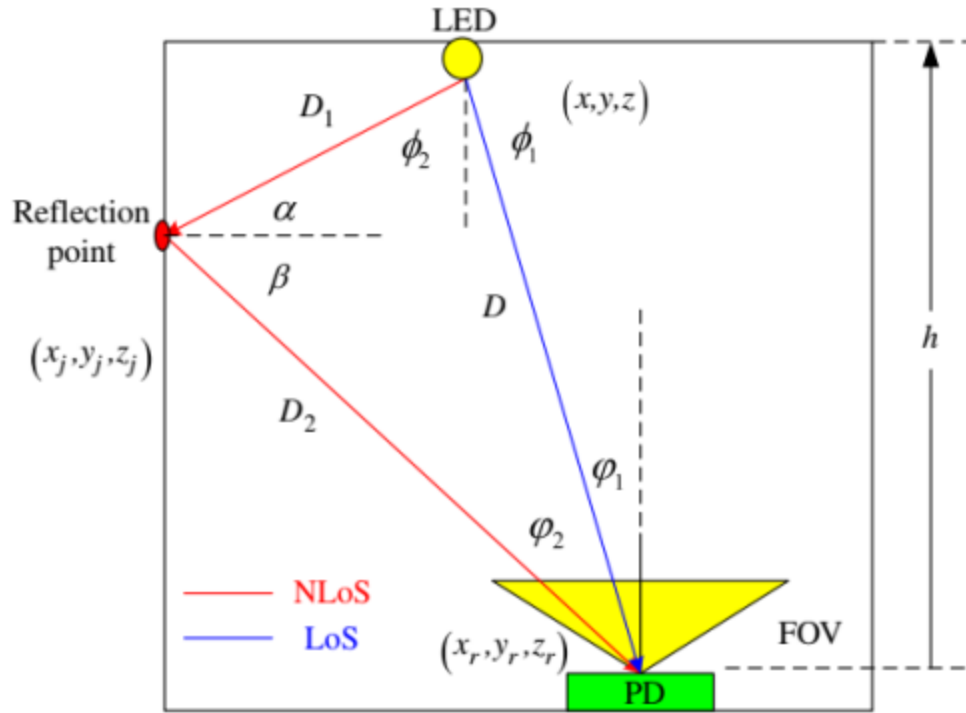


Fig I.18. Geometry of LOS and NLOS channels

### I.5.5. Photodiode

A Photodiode converts incident light into current. It works on a principle called photoconduction, whereas LED works on the principle of electro-luminance. The photodiode is a type of photo-detector that converts the light to either current or voltage [35].

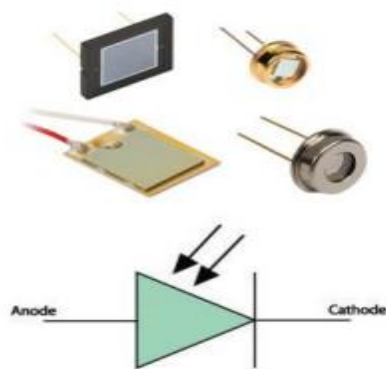


Fig I.19. Photodiode Icon and Examples [35]

### I.5.6. Received Signal Strength RSS

The strength of wireless signals as measured by the receiver that are typically measured in units of decibels (dB). If the power output of the transmitter is known, the RSS value can be used to estimate the distance between the transmitter and receiver. [36]

### I.5.7. Light-Fidelity Li-Fi

Li-Fi is closely related to VLC; However, Li-Fi offers a full networking solution which allows multiuser communication and seamless maneuverability of the User Equipment UE by configuring attocells within heterogeneous wireless systems. Other fundamental differences include [35]:

- VLC can only use VL spectrum, while, Li-Fi uses VL spectrum on the down-link and have the possibility to manipulate IR, VL or UV spectrum in the uplink. [37, 38, 39, 40]
- VLC systems are unidirectional [41] or bidirectional [42], however, Li-Fi should invariably be bidirectional. [39]
- Seamless user mobility is essential for Li-Fi systems, whilst, it is not a prerequisite for VLC8.
- Li-Fi systems should support multi-user communication via multiple access techniques, however, VLC systems can use SISO System, SIMO system, MISO system and MIMO system (multi input, single input, multi output). Indeed, a VLC system can be regarded as a Li-Fi system if they support multi-user communications and mobility of the UE. Conversely, a Li-Fi system is equivalent to a VLC system if only VL spectrum is used in both up-link and down-link [35].



**Fig I.20.** Application Scenarios for VLC/LiFi; (a) Indoor Applications and (b) Outdoor Applications [35]

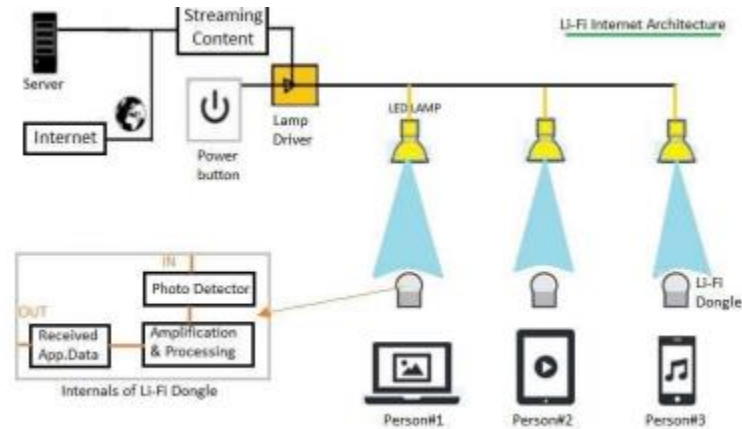
As both the technologies are somewhat alike, their applications are practically similar. Fig I.20. shows various application scenarios for both VLC and Li-Fi, it displays the indoor applications for both VLC and Li-Fi. In the indoor environments, both lighting and communication is administered together, however, Li-Fi also supports mobility. Some outdoor applications of both systems are suggested in Fig I.20.(b). The outdoor applications may consist of either communication scenarios, e.g., vehicle taillights and traffic lights; or both communication and lighting prospects, such as street lights and vehicle headlights. Smartphone to street lamp communication may be an example of both VLC and Li-Fi, whilst, V2X and V2V communication scenarios are examples of VLC.

### I.5.8. Indoor

- **Mobile connectivity (Li-Fi)**

Harald Haas first coined the term light in 2011 Fidelity (Li-Fi). Li-Fi is a high-speed, two-way, fully connected, visible light wireless a communication system, similar to Wi-Fi, which uses radio frequencies Communicate. The Wi-Fi signal has the problem of interfering with other RF signals.

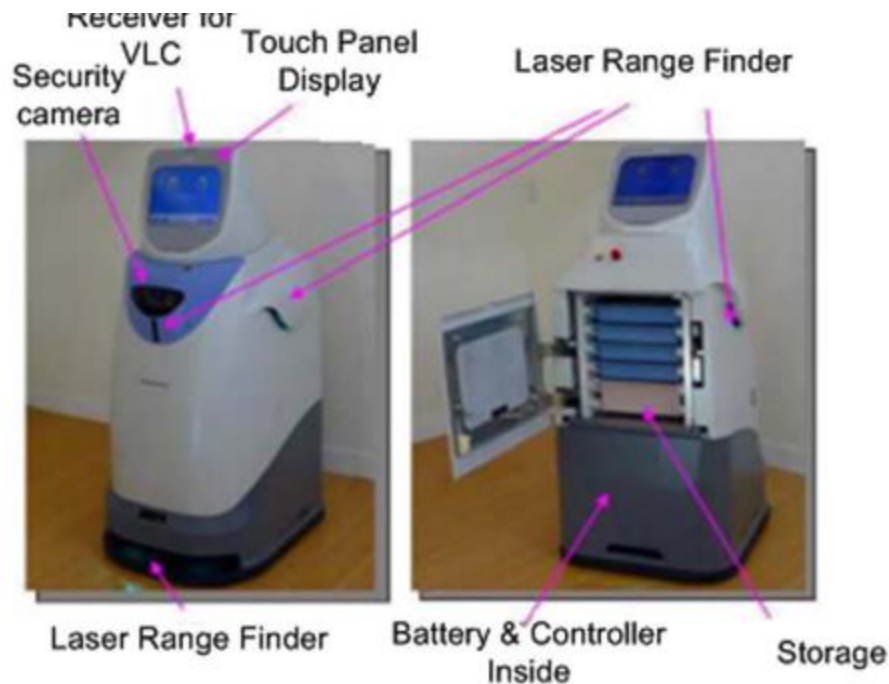
For example, it can interfere with signals from a pilot's navigation device on an airplane. As a result, Li-Fi may be a better option in areas that are susceptible to electromagnetic radiation, such as airplanes. Li-Fi also supports the Internet of Things (IoT). Speed up to 10 Gbit/s Implemented with Li-Fi, it is 250 times faster than ultrafast broadband [43].



**Fig I.21.** Li-Fi Internet Networking [42]

- **Hospitals and Healthcare**

In hospitals, electromagnetically sensitive areas (e.g., MRI Scanner) might switch to VLC since it does not interfere with radio wave of other machines. A robot named HOSPI is proposed for transportation Hospital. The control system improvement in HOSPI is achieved by installing the Robot construction and navigation sensors.



**Fig I.22.** HOSPI Robot [11]

- **Wi-Fi Spectrum Freed**

Wi-Fi cannot keep up with growing demand for wireless data. VLC can offer higher data rates than Wi-Fi. This can be done cheaply thanks to radio frequency (HF) components and antenna systems no longer required.

- **Airplanes**

Airplane cabins do not welcome radio waves. LED is already used for lighting it can also provide multimedia in place of cables Serve. This decreases the expense and weight of building the airplane.

- **Positioning**

In order to use VLC for high-precision positioning, the receiver must receive take the signal from the LEDs in the room and use the different LEDs to calculate the distance to the Algorithm to determine the exact location of the receiver. One of these techniques is based on RSS (received signal strength), commonly used in radio frequency systems. However, increasing the distance between the transmitter and receiver degrades the signal strength. So obstacles can

interfere with RSS because they block or reflect waves limits the accuracy of this method. Another widely used procedure is through time of Arrival (TOA) calculation. However, this technique requires transmission of strictly synchronized signals between transmitter and receiver, which may require more expensive features. Finally, there is another method that takes angle of arrival (AOA) into account. This technique is uncommon in RF systems because it requires a direct line of sight (hurry up). However, VLC systems require LOS for communication so AOA methods are also feasible. [44].

### I.5.9. Outdoor

- **Vehicles and Transportation**

Widespread use of LEDs in transportation applications and the growing interest in Intelligent Transportation Systems (ITS) presents a number of opportunities for application. While previous studies have evaluated the effectiveness of vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) communication using wireless technologies in terms of improving vehicle safety. Curbside installation since VLC link is much easier visible equipment. Installed extras such as LED traffic illuminated or LED signs may be used. Also, since the transmitter is designed for lighting purposes (thus generally high radiant power); signal-to-noise ratio(SNR) is high for VLC. The visible spectrum is not regulated globally and its bandwidths range from 400 to 790 THz. Large (390 THz) available bandwidth attractive opportunities for ITS applications. There are also V2I and V2Vcommunication via radio technology can be used simultaneously with VLC, with anyone have a different spectrum. VLC provides additional functionality if the sink includes one image sensor or camera. In particular, by using image or video processing to detect and recognize moving vehicles, security applications can be integrated. The VLC system, for instance, incorporates adaptive cruise control, collision warning, pedestrian detection, and range estimation for nearby vehicles [45].

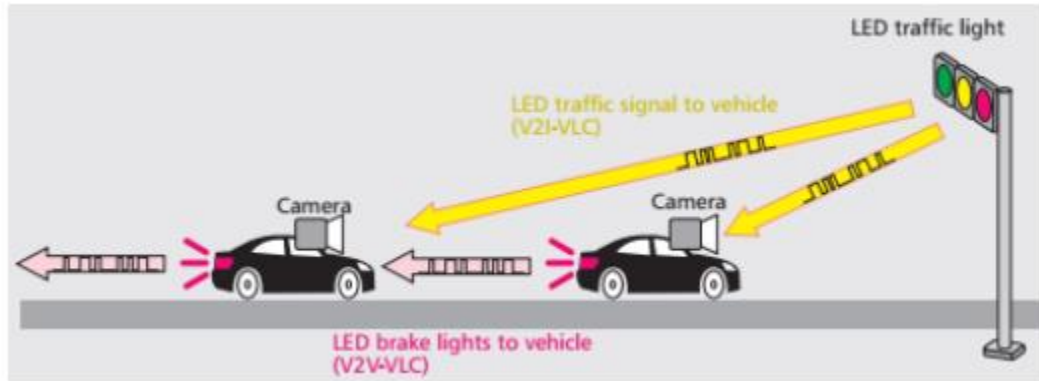


Fig I.23. V2V and I2V communication [45]

- **Underwater communication**

HF waves do not propagate well in seawater due to its favorable properties conductivity. Therefore, underwater communication should use VLC communication network. Another use for VLC in underwater communications is the Untethered Remotely Operated Vehicle (UTROV). UTROV can be used for a variety of tasks, including maintaining marine observatories.[43]

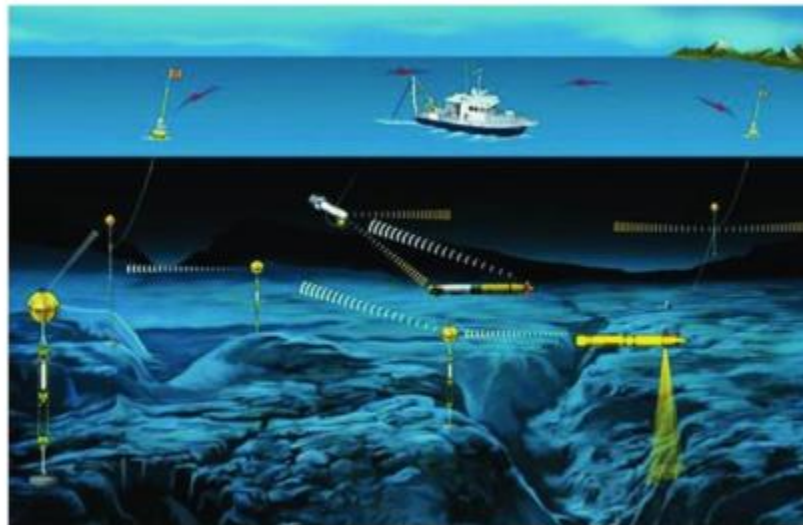


Fig I.24. VLC Underwater Communication [43]

### **I.6. Conclusion**

this chapter, we have defined artificial intelligence, provided a historical overview of it, described its several types—major groups and subsidiary groups—and discussed its applications in many fields. We also discussed Robot, with a brief historical review of the word's etymology and touching briefly on its sorts and range of applications. Following that, we discussed wireless communication and looked at its application (definition), electromagnetic waves, as well as various forms of wireless transmission and their benefits and drawbacks. Finally, we learned about the VLC and the several categories or sections that it is employed.



**CHAPTER II**

**MODELLING AN INTELLIGENT**

**CLASSIFICATION SYSTEM**

### II.1. Introduction:

Human simulation technology and robot education are developing very quickly to the point that we can nearly argue that a robot is a human being. In this study, we will discuss Python programming as a method for teaching the robot to identify items from images. More emphatically, we will talk about Python in general, including what is it and all of its applications.

How does the picture classification process work? How is it programmed? Yet, we discussed our experience, which we have obtained in these fields, and the techniques that would provide better outcomes.

### II.2. Python

#### II.2.1. What is Python?

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for rapid application development and for use as a scripting or glue language to connect existing components. Python is simple, easy to learn syntax, emphasizes readability, and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed. [46]

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

### II.2.3. Python Use Cases

The popular cases Python can be used for: [47]

1. AI and machine learning
2. Data analytics
3. Data visualization
4. Programming applications
5. Web development
6. Game development
7. Language development
8. Finance
9. Search engine optimization
10. Design

Professionally, Python is great for backend web development, data analysis, artificial intelligence, and scientific computing. Developers also use Python to build productivity tools, games, and desktop apps

### II.2.4. Features and Benefits of Python

- Compatible with a variety of platforms including Windows, Mac, Linux, Raspberry Pi, and others
- Uses a simple syntax comparable to the English language that lets developers use fewer lines than other programming languages
- Operates on an interpreter system that allows code to be executed immediately, fast-tracking prototyping
- Can be handled in a procedural, object-orientated, or functional way

### II.2.5. Python and AI

AI researchers are fans of Python. Google TensorFlow, as well as other libraries (Scikit-Learn, Keras), establish a foundation for AI development because of the usability and flexibility it offers Python users. These libraries, and their availability are essential because they enable developers to focus on growth and building. [48]

**II.2.6. Tools Available for Python Development:** [49]

- **PyCharm** is a proprietary and open-source IDE for Python development
- **Python Tools for Visual Studio** is a free and open-source plug-in for Visual Studio.
- **Jupyter Notebook** has everything you'll ever need for interactive development, documentation, and code execution. The Notebook comes equipped with a console-based approach for document authoring.
- **Colaboratory ("Colab" for short)** is a data analysis and machine learning tool that allows you to combine executable Python code and rich text along with charts, images, HTML, LaTeX and more into a single document stored in Google Drive. It connects to powerful Google Cloud Platform runtimes and enables you to easily share your work and collaborate with others.

[50]

**II.2.7. Comparing Python With Other Languages**

Python, compared with the most liked programming languages such as Java, JavaScript, and C++ also, thought to be the most useful. [51]

**Tab.II.1 : Python vs Java**

Point of Difference	Python	Java
Program run time	Python program runs slower than Java	Java program runs faster than python
Program development time	Takes less time than Java	Takes more time for developing the same program
Length of the code	Python code is 3-5 times shorter than Java code	Java code is longer than Python
Type	Python is dynamically typed	Java is not dynamically typed

**Tab.II.2 : Python vs JavaScript**

Point of Difference	Python	JavaScript
Object-based	A python is an object-based programming language	JavaScript is also an object-based scripting language
Type	Python can be used as a scripting language as well as a programming language	JavaScript is only used for the scripting language

**Tab.II.3 : Python vs C++**

Point of Difference	Python	C++
Length of code	Python code is 5-10 time shorter than C++ code	C++ code is 5-10 times longer than Python code
OOPS	Python is pure OOPS programming language	C++ is not a pure OOPS programming language

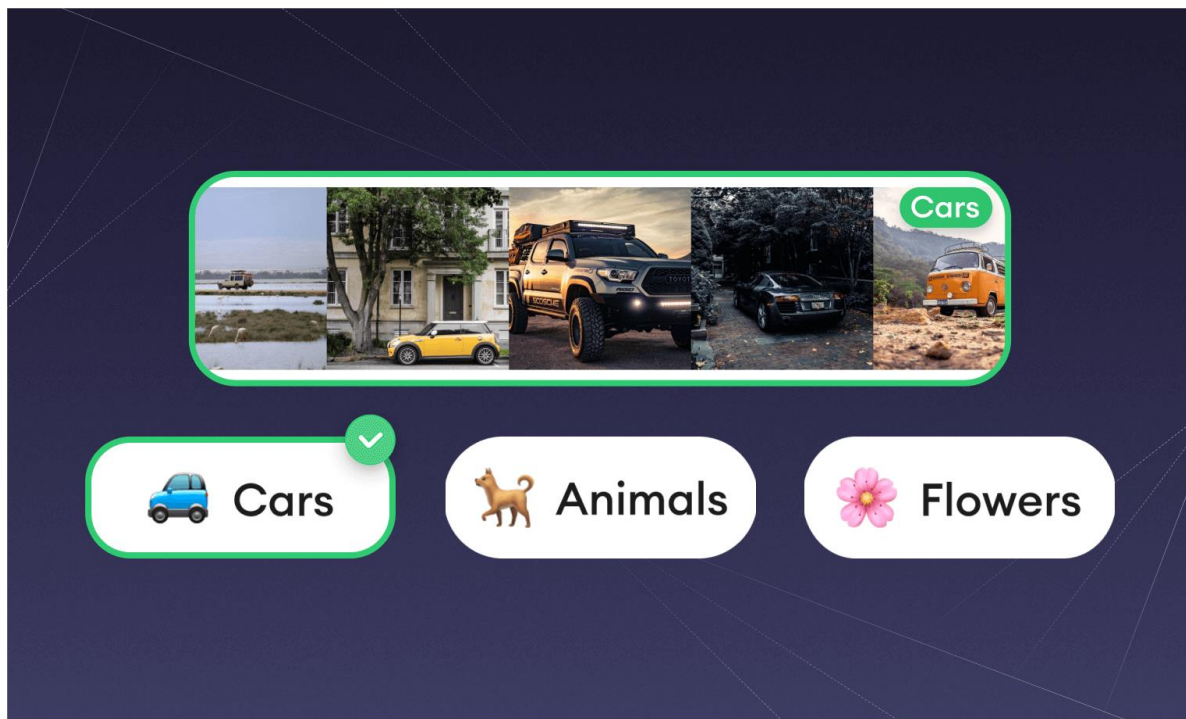
### II.3. Classification

Classification is the process of identifying and grouping objects or ideas into predetermined categories. In data management, classification enables the separation and sorting of data according to set requirements for various business or personal objectives.

In machine learning (ML), classification is used in predictive modeling to assign input data with a class label. [52]

### II.3.1. Image classification

We often underestimate the everyday paths we cross with technology when we're unlocking our smartphones with facial recognition or reverse image searches without giving much thought to it. At the root of most of these processes is the machine's capability to analyze an image and assign it a label, similar to distinguishing between different plant species for plant phenotypic recognition. Image classification brings that human capability to the world of tech. Essentially, both technology and AI have evolved to possess eyes of their own and perceive the world through computer vision. Image classification acts as a foundation for many other vital computer vision processes that only grow more advanced as we go. Let's focus on what image classification is exactly in machine learning and expand further from there. We've compiled the only guide to image classification that you'll need to learn the basics — and even something more. [52]



**Fig II.1** : Image classification [52]

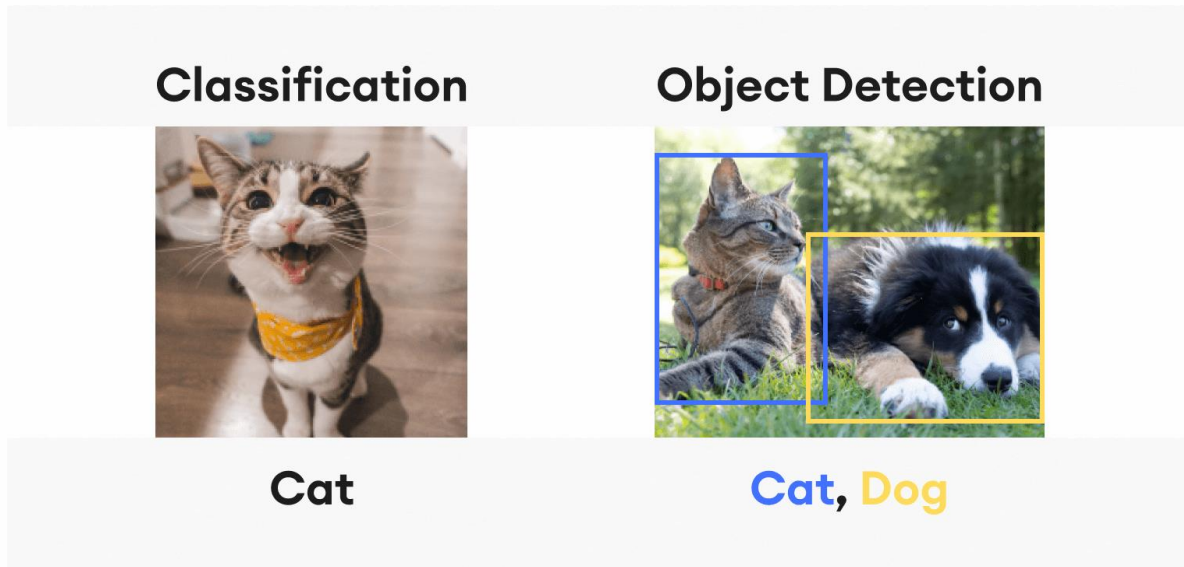
### II.3.2. What is image classification?

Image classification is a vital computer vision task that has an integral role in modern technology. It involves assigning the overall image a label or tag, which are taken from a pre-existing database acquired through a training model. The process is simple from a surface level, but, in fact, it involves analyzing individual pixels of the image before determining an appropriate label for the overall image. As an end result, we are able to acquire a selection of data for each corresponding image. We can go on to categorize and analyze the information from the labeling. However, it is important that the data labeling is completed accurately and correctly in the training phase to avoid discrepancies in the data. That can be ensured through model training, which is just as possible with the help of publicly available datasets. Why is image classification important, and what significance does it have for the average person? In practical use, image classification is apparent across many industries from environmental and agriculture through remote sensing, land and urban planning, surveillance, geographic mapping, disaster control, item identification, and much more. [53]

### II.3.3. Image Classification and object detection

Image classification, object detection, object localization— all of that may be a tangled mess in your mind, and that’s completely alright if you are newly introduced to these concepts. All of these mentioned terms are integral components in computer vision and image annotation. There are subtle, yet particular, differences among the three that we’ll break down now.

We’ve already defined that image classification assigns a specific label to the image. Whereas, with object localization, we refer to the process of locating the main object, or one that is of interest, in a given image or video. [53]



**Fig II.2:** Image classification & Object detection [53]

Object detection, on the other hand, is the method of assigning labels to individual items in an image, as opposed to image classification, which assigns a label to the entire picture. Object detection, as the name implies, recognizes the target items inside an image, labels them, and specifies their position. One of the most prominent components of object detection is the “bounding box”, which indicates the location of a particular object or an image, and what the label of that object is. Essentially, object detection combines image classification and object localization. [53]

#### II.3.4. How Image Classification Works

It’s a known fact that the image we see as a whole is made up of hundreds to thousands of tiny pixels. Before computer vision can determine and label the image as a whole, it needs to analyze the individual components of the image to make an educated assumption. That is why image classification is executed via a computer system that analyzes a given image in the form of pixels. It accomplishes this by treating the picture as an array of matrices, the size of which is determined by the image resolution. The pixels of the digital image are taken and grouped into what we know as “classes”.



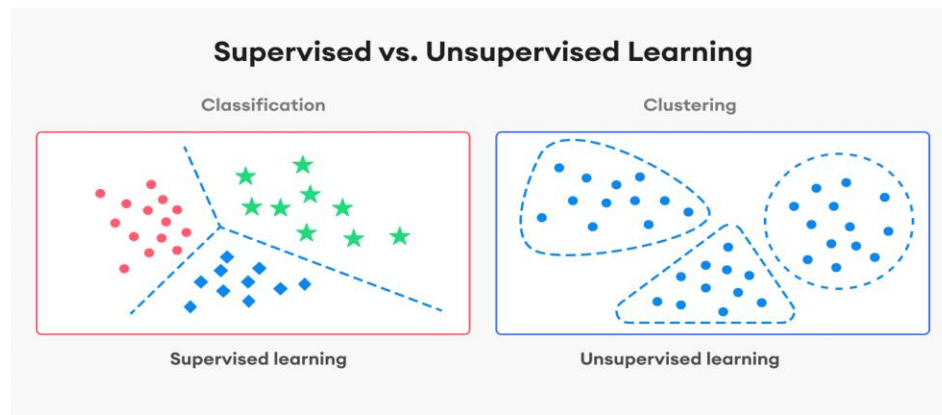
From here, the process will differ based on the algorithm, but before observing the various algorithms, Let's look at it more broadly to see how it functions. The chosen algorithm will divide the image into a series of key attributes in order to prevent relying exclusively on the final classifier. Those attributes help the classifier determine what the image is about and which class it belongs to. Because the rest of the stages are dependent on it, the feature extraction procedure is arguably the most critical step in classifying a picture. [53]

Overall, the image classification flow looks something like this:

**Image pre-processing -> feature extraction -> object classification**

### II.3.5. Algorithms and models: Supervised and unsupervised classification

There isn't one straightforward approach for achieving image classification, and the two most notable kinds we will look further at today are supervised and unsupervised classification.



**Fig II. 3:** Supervised & Unsupervised learning [53]

#### II..3.5.1. Supervised

In the case of supervised classification, the system first must be trained with some initial reference information before it can apply the information acquired to future visual material. During the classification process in this model, the algorithm refers to the trained data and draws similarities between that data and the new input. Since it has been trained with data prior to receiving the new data, it can implement the knowledge gained from patterns of that data and classify the new images based on that.[53]

The process doesn't end there, however, supervised image classification algorithms, in their turn, can be divided into single-label classification and multi-label classification. As the name suggests, single-label classification refers to a singular label that is assigned to an image as a result of the classification process. It is by far the most common type of image classification we witness on a daily basis.

If single-label classification generalized the image and assigned it a single class, then the number of classes an image can be assigned with multi-label classification is uncountable. In what instances is multi-label classification exceptionally helpful? In the field of medicine, for example, medical imaging may show several diseases or anomalies present in a single image for the patient.

### **II.3.5.2. Unsupervised**

With unsupervised algorithms, no pre-existing tags are given to the system, only raw data. The system interprets the data on its own terms, recognizes patterns, and draws unique conclusions from the data without human interference. How does it know what to look for and then properly classify it? Unsupervised classification makes avid use of a concept called clusterization to achieve this. Clusterization is the unsupervised, natural locating and grouping (or “clustering”) of data into groups. However, this method will provide you with the different clusters that you would need to decide a class for in another way. There is a plethora of different clusterization algorithms in their turn, with some of the most notable ones being K-Means, Agglomerative Clustering, BIRCH, and Mini-Batch K-Means.

There isn't a single best choice out of these clusterization algorithms. Instead, it is optimal to test various ones until you settle on the one that works best with the specific task at hand. [53]

### **II.3.6. Deep Neural Networks for Image Classification**

Deep learning has proven to take computer vision tasks to an even higher level of accuracy and efficiency, all thanks to convolutional neural networks (CNNs). It aimed to emulate the neural networks of the human mind in order to complete specific computer processes with minimal human interference. The variety of layers, starting with the input layer to the hidden

inner layers, and output layer, are what make the network considered “deep”. In brief, this is how image classification is done via CNNs:

- The input image is fed into the network.
- Various filters are applied to the image in order to generate a feature map.
- A pooling layer is applied to each of those maps.
- The pooled layers are flattened into a vector, then that vector is connected to the neural network.
- The final fully-connected output layer with the classified features is received.

Fully grasping the use of CNNs for image classification requires a much deeper dive into the technical aspect of the model. That deserves a separate crash course if you want to go beyond the basics of image classification. [53]

### II.3.7. Key takeaways

Image classification is a branch of computer vision that deals with categorizing and identifying groupings of pixels or vectors inside an image using a set of predetermined tags or categories on which an algorithm has been trained. To expand on these last two concepts, we need to distinguish between supervised and unsupervised categorization.

In either case, you must have a large number of different datasets with precisely labeled data in order to create successful image classifiers, or else there will be discrepancies in the data labeling. State-of-the-art CNN classification is another reliable method of image classification that generates highly accurate results, remains favorable by specialists.[53]

## II.4. Neural networks

### II.4.1. How do neural networks work?

The human brain is the inspiration behind neural network architecture. Human brain cells, called neurons, form a complex, highly interconnected network and send electrical signals to each other to help humans process information. Similarly, an artificial neural network is made of artificial neurons that work together to solve a problem. Artificial neurons are software modules,

called nodes, and artificial neural networks are software programs or algorithms that, at their core, use computing systems to solve mathematical calculations. [54]

### II.4.2. Simple Neural Network architecture:

A basic neural network has interconnected artificial neurons in three layers: [55]

- **Input Layer:**

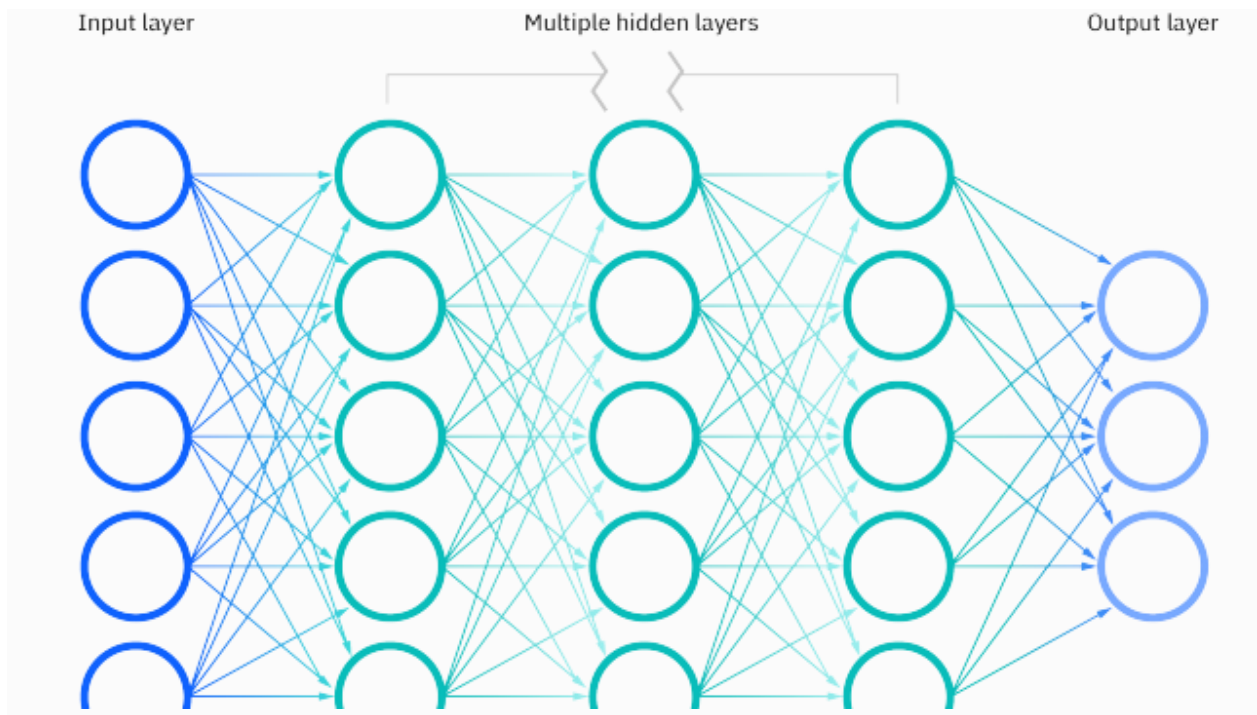
Information from the outside world enters the artificial neural network from the input layer. Input nodes process the data, analyze or categorize it, and pass it to the next layer.

- **Hidden Layer:**

Hidden layers take their input from the input layer or other hidden layers. Artificial neural networks can have a large number of hidden layers. Each hidden layer analyzes the output from the previous layer, processes it further, and passes it to the next layer.

- **Output Layer:**

The output layer gives the final result of all the data processing by the artificial neural network. It can have single or multiple nodes. For instance, if we have a binary (yes/no) classification problem, the output layer will have one output node, which will give the result as 1 or 0. However, if we have a multi-class classification problem, the output layer might consist of more than one output node.



**Fig II.4:** neural network architecture [56]

### II.4.3. Convolution Neural Network architecture:

CNN layers can be of four main types: Convolution Layer, ReLu Layer, Pooling Layer, and Fully-Connected Layer.

- **Convolution Layer:**

A convolution is the simple application of a filter to an input that results in an activation. The convolution layer has a set of trainable filters that have a small receptive range but can be used to the full-dept of data provided. Convolution layers are the major building blocks used in convolutional neural networks.

- **ReLu Layer:**

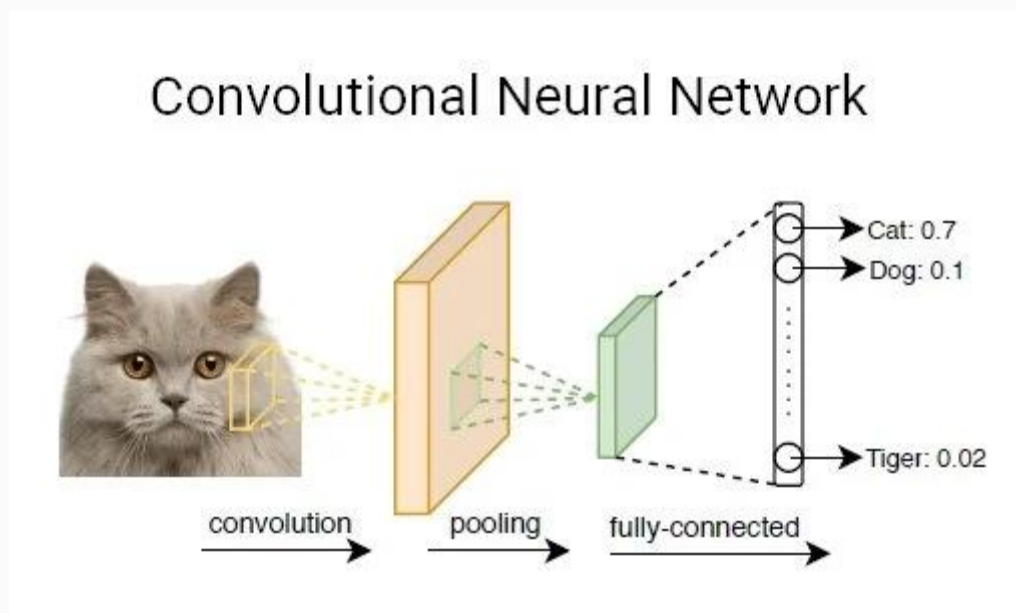
ReLu layers, also known as Rectified linear unit layers, are activation functions applied to lower overfitting and build the accuracy and effectiveness of the CNN. Models that have these layers are easier to train and produce more accurate results.

- **Pooling Layer:**

This layer collects the result of all neurons in the layer preceding it and processes this data. The primary task of a pooling layer is to lower the number of factors being considered and give streamlined output.

- **Fully-Connected Layer:**

This layer is the final output layer for CNN models that flattens the input received from layers before it and gives the result.

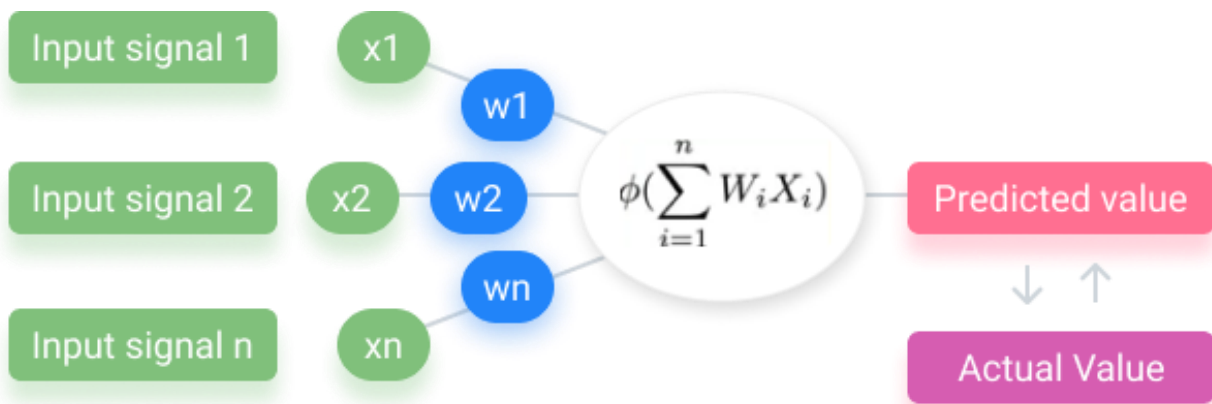


**Fig II.5:** Convolutional neural network [54]

### II.4.4. Deep Neural Network Architecture

Deep neural networks, or deep learning networks, have several hidden layers with millions of artificial neurons linked together. A number, called weight, represents the connections between one node and another. The weight is a positive number if one node excites another, or negative if one node suppresses the other. Nodes with higher weight values have more influence on the other nodes.

Theoretically, deep neural networks can map any input type to any output type. However, they also need much more training as compared to other machine learning methods. They need millions of examples of training data rather than perhaps the hundreds or thousands that a simpler network might need. [56]



**Fig II.6:** Concept of a neural network with the input values (green) and weights (blue). [53]

### II.4.5. What Is a Pre-trained Model?

Simply put, a pre-trained model is a model created by developers to solve a similar problem. Instead of building a model from scratch to solve a similar problem, you use the model trained on other problem as a starting point.

For example, if you want to build a self-learning car, you can spend years to build a decent image recognition algorithm from scratch, or you can take a pre-trained model from Google which was built on ImageNet data to identify images in those pictures.

A pre-trained model saves huge efforts required to re-invent the wheel. [57]

II.4.5.1. Applications of Pre-training

The applications of pre-training can be grouped into three categories: Transfer Learning, classification, and feature extraction. [58]

Pre-trained models can also be applied to classification tasks e.g., image classification. Image classification refers to the task of identifying what a given image represents. There are a number of pre-trained models out there today trained specifically for image classification. These models have been trained on large image datasets and hence can be used for an image classification task.

Given below is a rough timeline of how the state-of-the-art models have improved over time

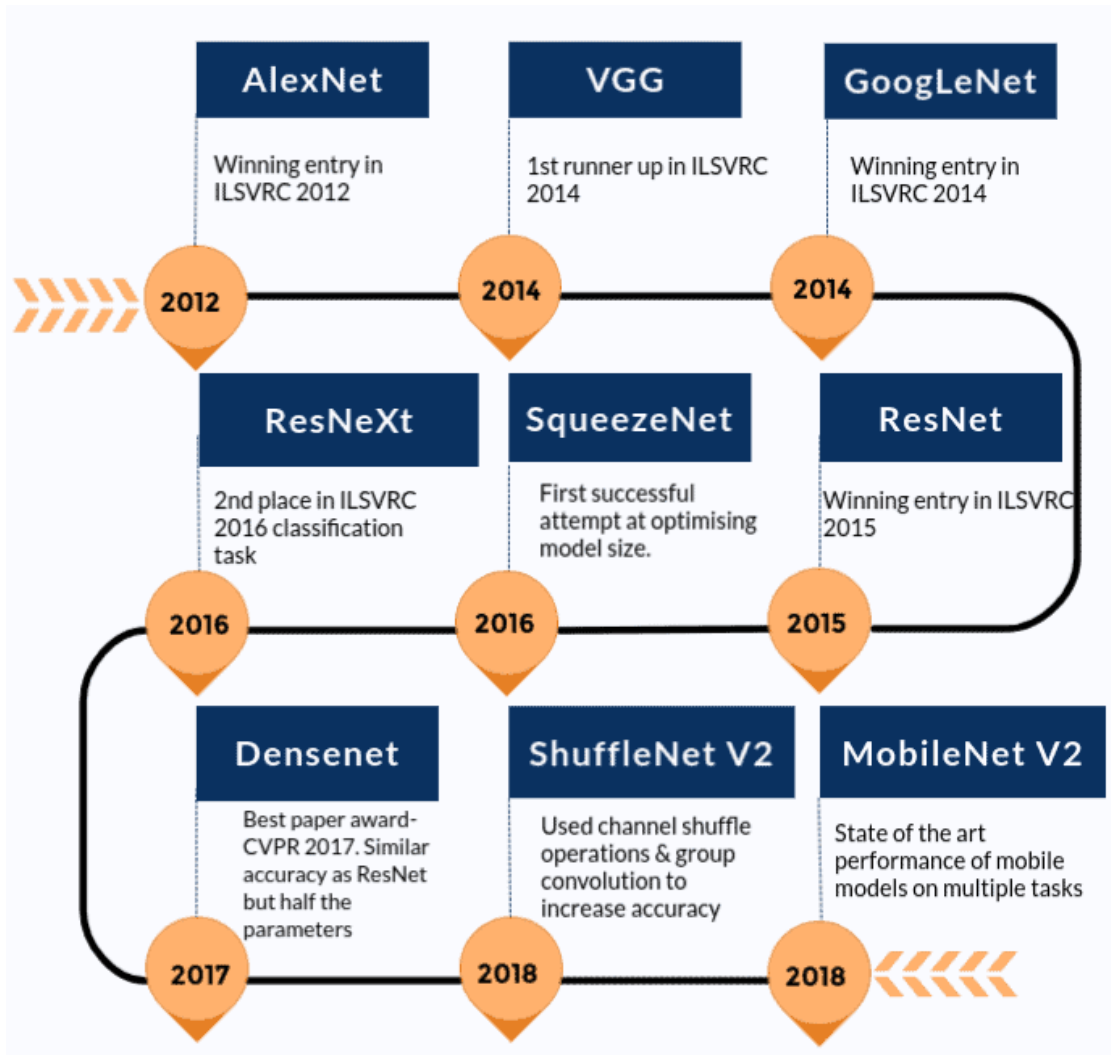


Fig II.7: Timeline of how the state-of-the-art models have improved over time [59]



II.4.6. What is MobileNetV2?

MobileNetV2 is a classification model (distinct from MobileNetSSDv2) developed by Google. It provides real-time classification capabilities under computing constraints in devices like smartphones. This implementation leverages transfer learning from ImageNet to your dataset. [60]

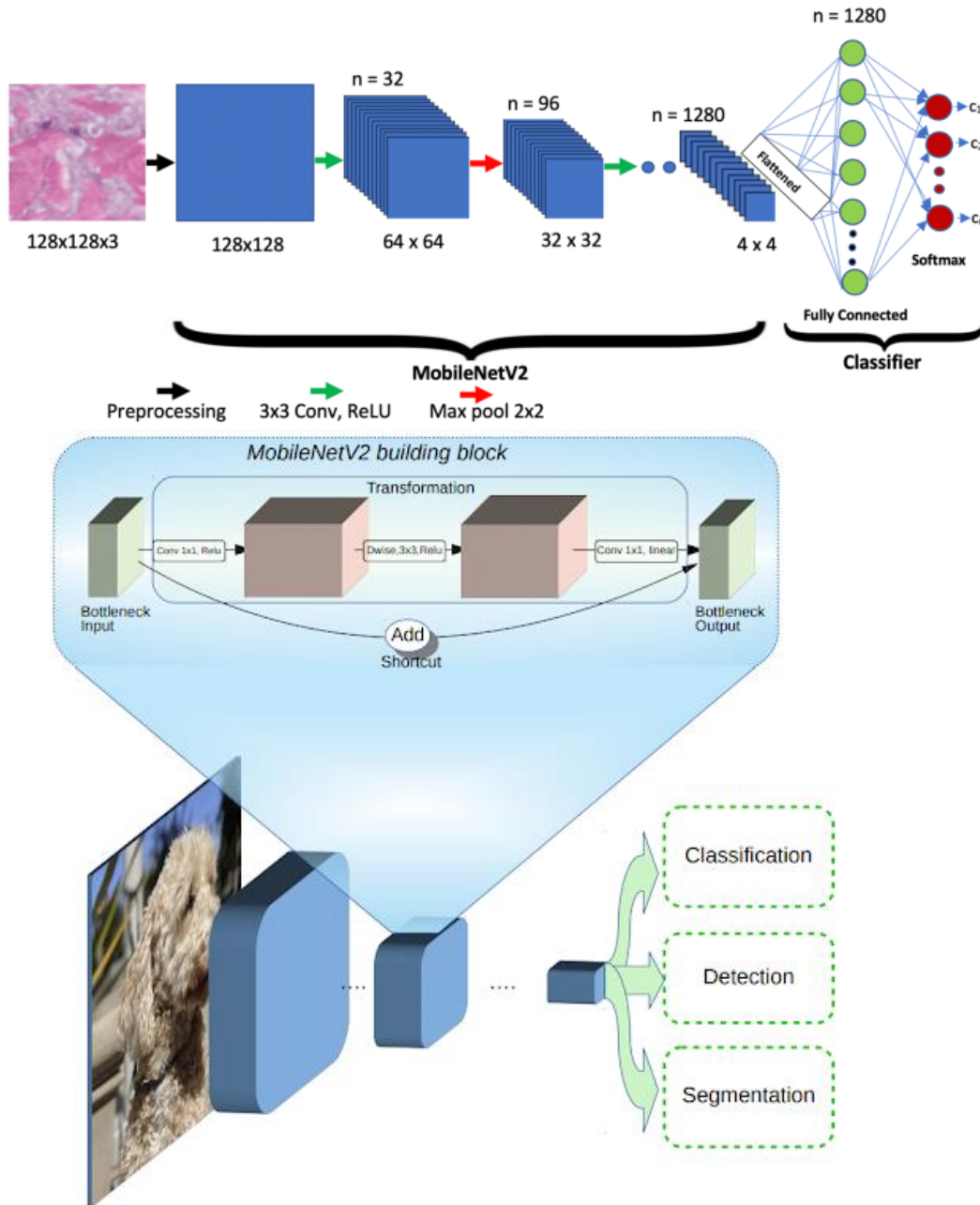


Fig II.8: MobilenetV2 [60]

### II.4.6.1. MobileNetV2 Architecture

MobileNetV2 is built around a key concept called "inverted residual blocks." These blocks consist of a lightweight depth wise separable convolution followed by a linear bottleneck layer to enhance the representation power of the network. The inverted residual blocks are designed to strike a balance between computational efficiency and model accuracy. [61]

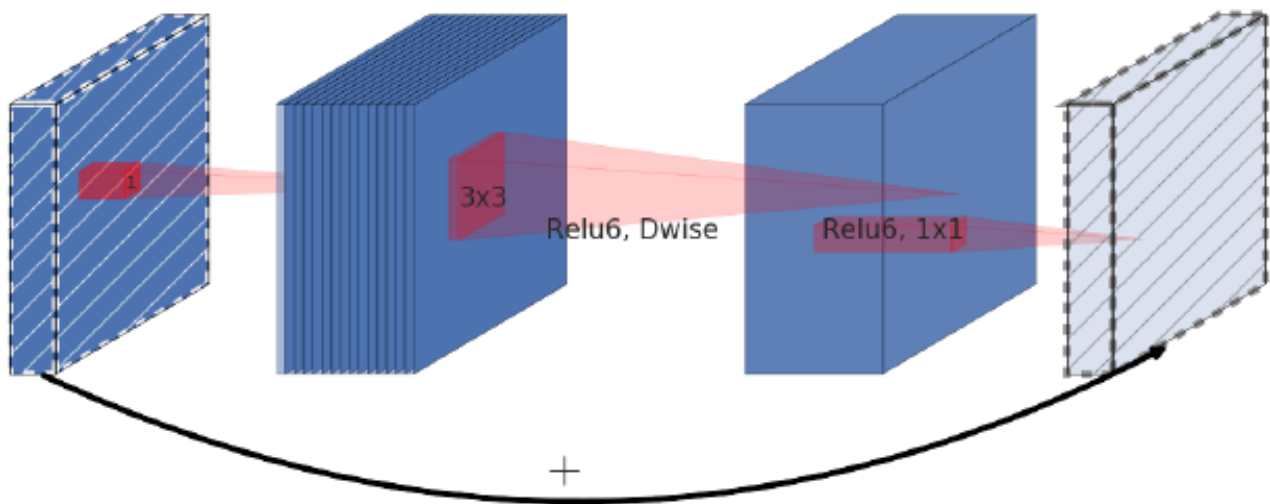
Here is an overview of the MobileNetV2 architecture:

- **Initial Convolutional Layer:**

The network starts with a standard convolutional layer that performs a 3x3 convolution with a stride of 2. This layer helps to reduce the spatial dimensions of the input. [63]

- **Inverted Residual Blocks:**

MobileNetV2 consists of multiple stacked inverted residual blocks. Each block consists of the following layers:



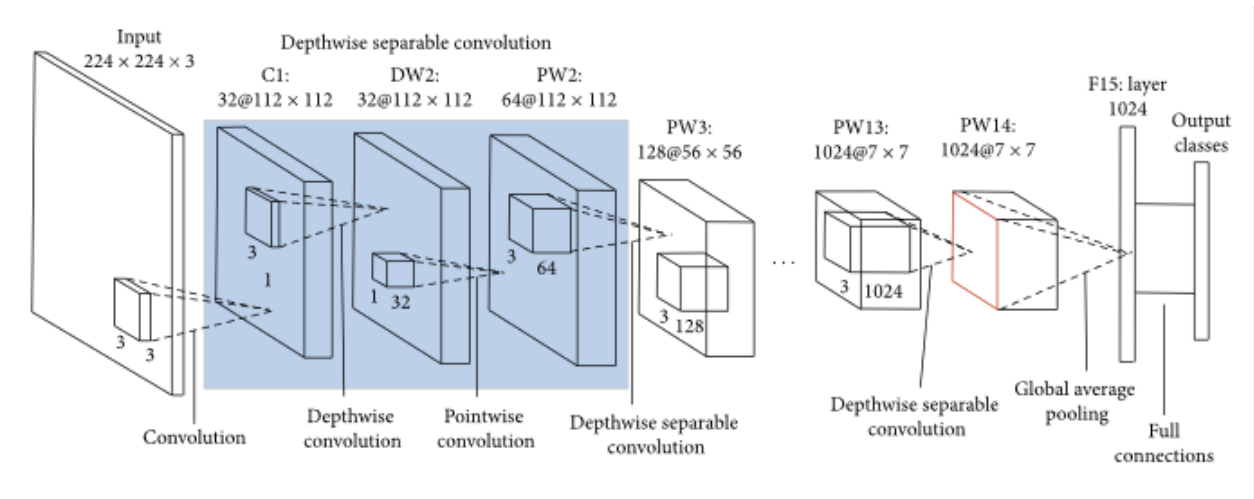
**Fig II.9:** Inverted Residual Blocks [60]

- Expansion Layer:**

This layer expands the number of channels of the input tensor using a 1x1 convolution. It helps to create a higher-dimensional representation that captures more information. [62]

**b. Depth wise Convolution:**

The expanded tensor is then subjected to depth wise convolution, where each input channel is convolved with its own set of filters. This operation reduces the computational costsignificantly.[64]



**Fig II.10:** Depth wise convolution layer [60]

**c. Linear Bottleneck Layer:**

The depth wise convolution output is passed through a linear bottleneck layer, which consists of a 1x1 convolution followed by a non-linear activation function. This layer allows the network to learn more complex features and increases its representation power. [65]

**d. Skip Connection:**

Some of the inverted residual blocks have a skip connection (also known as a residual connection) that directly passes the input to the output. This helps in propagating the gradients and improves the flow of information through the network.

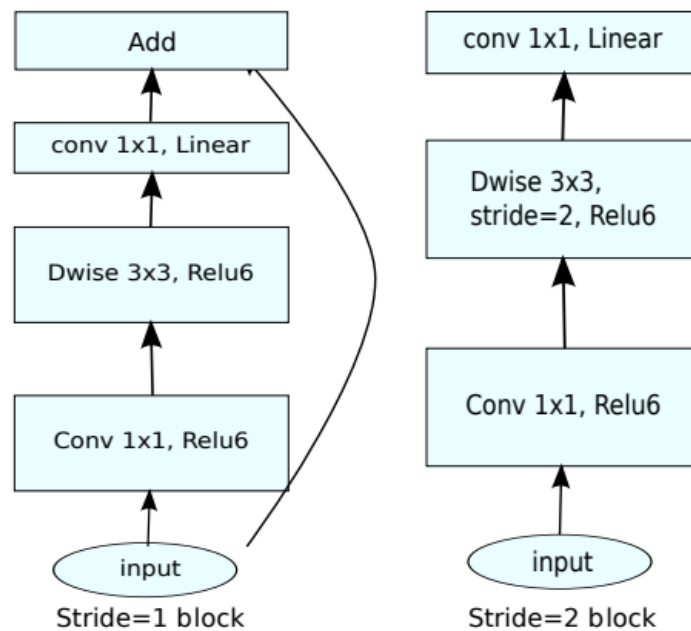
- **Middle Flow:**

After several initial inverted residual blocks, MobileNetV2 includes a middle flow that repeats the same pattern of inverted residual blocks multiple times. This helps to increase the network's capacity for feature extraction. [66]

- **Final Layers:**

The middle flow is followed by a few additional inverted residual blocks, similar to the initial ones. Finally, there is a global average pooling layer that reduces the spatial dimensions to 1x1, followed by a fully connected layer with SoftMax activation for classification.

MobileNetV2 achieves a good balance between model size, accuracy, and computational efficiency by utilizing depthwise separable convolutions, linear bottleneck layers, and skip connections. It is widely used for on-device image classification tasks, where resource constraints are a concern, such as mobile applications and edge computing scenarios



**Fig II.11:** MobileNetV2 Architecture [60]

### II.5. My Experience

In this section, I would like to discuss my experience with the simulation as a practical component of the second chapter.

This is my first time working with Python. I started by installing it on my PC and watching YouTube tutorials on how to do it. The first thing I learned about was the different code editors I could use. I started with Visual Studio but found it hard to work with for my project. So I switched to PyCharm, which was much better. Since I'm still learning, PyCharm's working process was helpful and efficient, in a way that I could write my code directly in the program and it would quickly point out any error or issue.

Later, a friend of me suggested to try Jupyter Notebook, and I found it to be a great tool as well. It has cells where I can write sections of code and test them individually to identify and fix bugs one step at a time.

While working with Python, I often use code snippets from various sources such as ChatGPT, GitHub, and online databases. Initially, I started developing with a dataset of cats and dogs, which I got from Google. I downloaded around 25,000 photos of cats and dogs to use in my project. Once I got that working, I replaced the dataset with food product images like drinks, coffee, cakes, and water.

More emphatically, the cat and dog images were directly downloaded to my PC, and I would only modify the code and dataset when needed. Sometimes the code works, other times it doesn't. Thus, whenever, I encounter problems, I would seek assistance from ChatGPT to understand the issue and get suggestions to fix it.

However, one drawback I faced with Jupyter Notebook was its dependence on my PC's performance, similar to PyCharm. Once, I ran a particular code, it took six hours to finish, only to provide incorrect or unsuccessful results. This prompted me to switch to Google Colab, an online code editor. This latter, doesn't require downloading databases, as it can extract data directly from Google or from an external hard drive. It also uses the GPU from the cloud, which makes the code execution faster. I gradually tested my code on Google Colab, and most of the time, I was able to generate the correct code that allowed smooth training and produced amazing results. For example, when I presented a picture of a cat, the program recognized it correctly, and the same happened with dog pictures.

Ultimately, after changing my dataset, which now consisted of around 4,000 food product images (equally distributed for each product), I uploaded these images to Google Drive, worked with them. I used the code with MobileNetV2, which is a pre-trained model that doesn't require a large dataset to achieve good results. Last but not least, the findings of training it with 4,000 images proved to be excellent.

### II.6. Our code of Simulation

I didn't share all of the code but i only shared the important parts:

```
# Mount your Google Drive by adding the following code at the beginning
from google.colab import drive
drive.mount('/content/drive/')

#Import the necessary modules
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
from sklearn.model_selection import train_test_split
from google.colab.patches import cv2_imshow

#Load the image filenames and name my data set in range of 5 latter
file_names = os.listdir('/content/drive/MyDrive/image resized2/')
for i in range(5):
    name = file_names[i]
    print(name[0:5])

#Count the number of images for each class
file_names = os.listdir('/content/drive/MyDrive/image resized2/')
sodaa_count = 0
water_count = 0
anything_else_count = 0

for img_file in file_names:
    name = img_file[0:5]
    if name == 'sodaa':
        sodaa_count += 1
    elif name == 'water':
        water_count += 1
    else:
        something_count += 1
print('Number of sodaa images =', sodaa_count)
print('Number of water images =', water_count)
print('Number of anything else =', anything_else_count)

# creating a for loop to assign labels
filenames = os.listdir('/content/drive/MyDrive/image resized2/')

labels = []

for i in range(4000):
    file_name = filenames[i]
    label = file_name[0:5]
    if label == 'sodaa':
        labels.append(1)
    elif label == 'water':
```

```

    labels.append(0)
else:
    labels.append(2)

# Split the data into training and testing sets
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)
# Load the MobileNetV2 model and Building the Neural Network
modelimport tensorflow as tf
import tensorflow_hub as hub

mobilenet_model = 'https://tfhub.dev/google/tf2-preview/mobilenet_v2/feature_vector/4'

pretrained_model = hub.KerasLayer(mobilenet_model,

input_shape=(224,224,3), trainable=False)
num_of_classes = 3

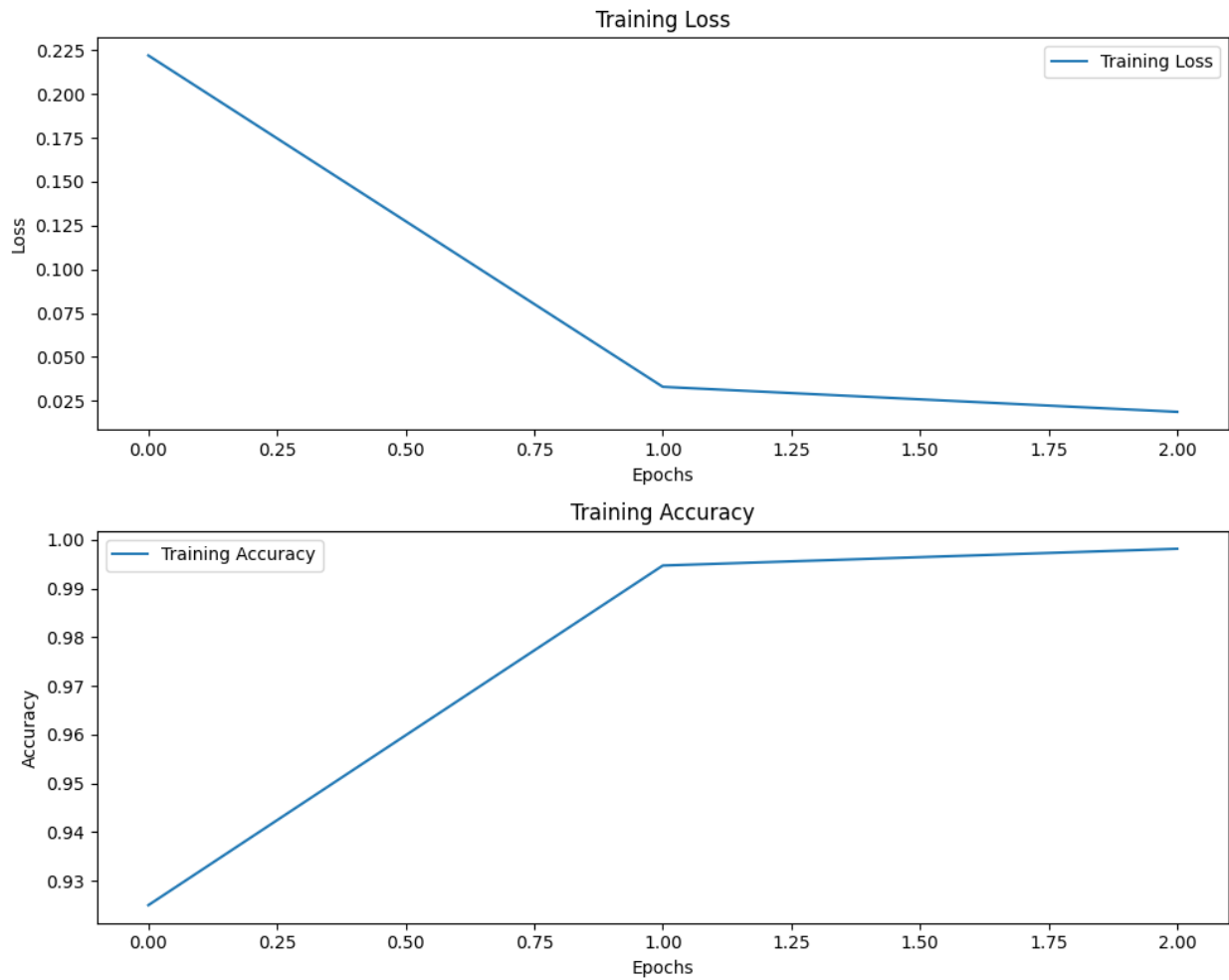
model = tf.keras.Sequential([
    pretrained_model,
    tf.keras.layers.Dense(num_of_classes)

])
model.summary()

# configuring the training process of your model
model.compile(
    optimizer = 'adam',
    loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
    metrics = ['acc']
)

# Train your model
import matplotlib.pyplot as plt
history = model.fit(X_train_scaled, Y_train, epochs=3)
# Extract the training loss and accuracy from the history object
training_loss = history.history['loss']
training_accuracy = history.history['acc']
# Create a figure and axes
fig, ax = plt.subplots(2, 1, figsize=(10, 8))
# Plot the training loss
ax[0].plot(training_loss, label='Training Loss')
ax[0].set_xlabel('Epochs')
ax[0].set_ylabel('Loss')
ax[0].set_title('Training Loss')
ax[0].legend()
# Plot the training accuracy
ax[1].plot(training_accuracy, label='Training Accuracy')
ax[1].set_xlabel('Epochs')
ax[1].set_ylabel('Accuracy')
ax[1].set_title('Training Accuracy')
ax[1].legend()
# Display the plot
plt.tight_layout()
plt.show()

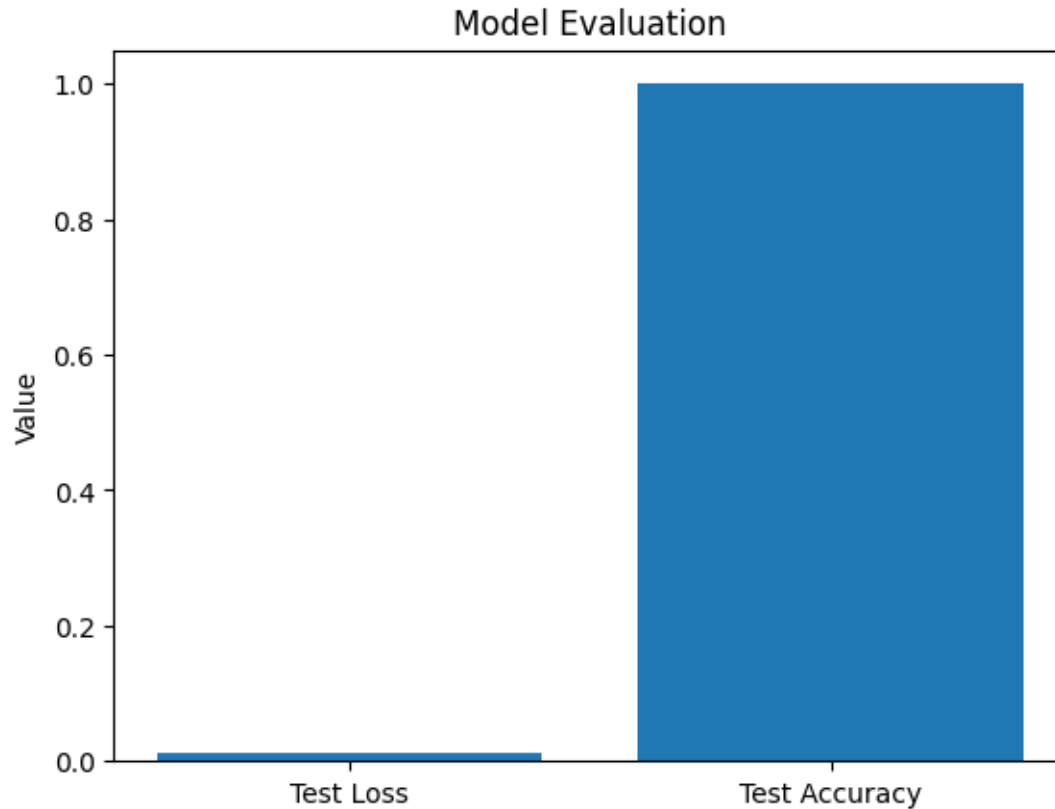
```



```
# Evaluate your model on the test set
score, acc = model.evaluate(X_test_scaled, Y_test)
print('Test Loss =', score)
print('Test Accuracy =', acc)

# Create a bar plot for the evaluation results
fig, ax = plt.subplots()
labels = ['Test Loss', 'Test Accuracy']
values = [score, acc]
ax.bar(labels, values)
ax.set_ylabel('Value')
ax.set_title('Model Evaluation')
plt.show()
```





```

# Save your model
model.save('/content/my_model.h5')

# Predictive System
from google.colab.patches import cv2_imshow
import cv2
import numpy as np
from tensorflow.keras.models import load_model

# Prompt the user for the image path
input_image_path = input('Path of the image to be predicted: ')

# Read the input image
input_image = cv2.imread(input_image_path)

# Display the input image
cv2_imshow(input_image)

# Preprocess the input image
input_image_resize = cv2.resize(input_image, (224, 224))
input_image_scaled = input_image_resize / 255.0
input_image_reshaped = np.reshape(input_image_scaled, (1, 224, 224, 3))

# Make a prediction on the input image
input_prediction = model.predict(input_image_reshaped)
input_pred_label = np.argmax(input_prediction)

# Print the prediction
if input_pred_label == 0:
    print('The image represents water')
elif input_pred_label == 1:

```

```
print('The image represents soda')  
else:  
    print('Unable to make a prediction')
```

Path of the image to be predicted: /content/download (1).jpg



1/1 [=====] - 0s 24ms/step

The image represents a water

Path of the image to be predicted: /content/download.jpg



1/1 [=====] - 1s 968ms/step

The image represents a sodaa

Path of the image to be predicted: /content/cat.jpg



1/1 [=====] - 0s 41ms/step

Unable to make a prediction

### II.7. Conclusion

In this chapter, we have defined what python is, as well as the foremost critical cases of its usage, advantages, and interdependence with artificial intelligence, and the finest ways to utilize it. On one hand, we compared it with other programs to clarify things. We also talked about image classification, its definition, how it works, and its most important models.

On the other hand, we discussed Neural Network's definition and its sorts, furthermore, we shed light on pre-trained show, its meaning and how to utilize it. In addition to the mobilenetv2 along with its types.

Eventually, we talked about our involvement and experience that we picked up with the use of Python in many of the tools available to develop it, and how to reach the desired result of our work.

## **CHAPTER III**

### **COMMAND AND TRAINING OF USING VLC**

### **COMMUNICATION**

### III.1. Introduction

In this chapter, we will talk about Raspberry Pi, its important uses and the tasks it performs. Then, we will study how to program and run image classification technology using python on Raspberry PI, with the assistance of VLC (transmitter and receiver) technology. We will also present the results of the studied system and its performance.

### III.2. Raspberry pi

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

What's more, the Raspberry Pi has the ability to interact with the outside world, and has been used in a wide array of digital maker projects, from music machines and parent detectors to weather stations and tweeting birdhouses with infra-red cameras. We want to see the Raspberry Pi being used by kids all over the world to learn to program and understand how computers work.[67]

# CHAPTER III: COMMAND AND TRAINING OF ROBOT USING VLC COMMUNICATION

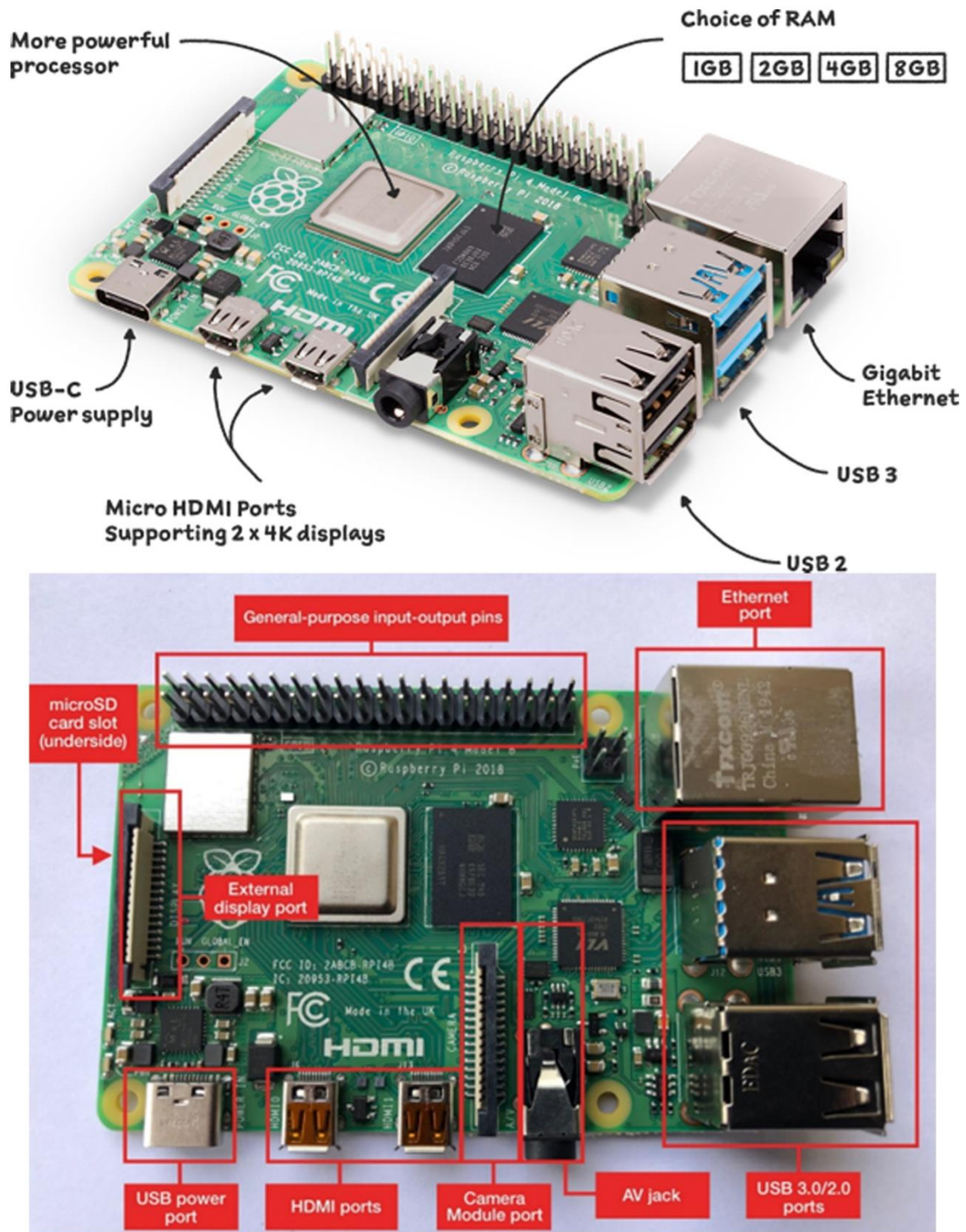


Fig III.1: Raspberry pi [68] [69]

### III.2.1. What Raspberry Pi models have been released?

There have been many generations of the Raspberry Pi line: from Pi 1 to 4, and even a Pi 400. There has generally been a Model A and a Model B of most generations. Model A has been a less expensive variant, and tends to have reduced RAM and fewer ports (such as USB and Ethernet). The Pi Zero is a spinoff of the original (Pi 1) generation, made even smaller and cheaper. Here's the lineup so far: [70]

- Pi 1 Model B (2012)
- Pi 1 Model A (2013)
- Pi 1 Model B+ (2014)
- Pi 1 Model A+ (2014)
- Pi 2 Model B (2015)
- Pi Zero (2015)
- Pi 3 Model B (2016)
- Pi Zero W (2017)
- Pi 3 Model B+ (2018)
- Pi 3 Model A+ (2019)
- Pi 4 Model A (2019)
- Pi 4 Model B (2020)
- Pi 400 (2021)

### III.2.2. How does Raspberry Pi work?

Raspberry Pi is a programmable device. It comes with all the critical features of the motherboard in an average computer but without peripherals or internal storage. To set up the Raspberry computer, you will need an SD card inserted into the provided space. The SD card should have the operating system installed and is required for the computer to boot. Raspberry computers are compatible with Linux OS. This reduces the amount of memory needed and creates an environment for diversity.

After setting up the OS, one can connect Raspberry Pi to output devices like computer monitors or a High-Definition Multimedia Interface (HDMI) television. Input units like mice or keyboards

should also be connected. This minicomputer's exact use and applications depend on the buyer and can cover many functions. [71]

### III.2.3. Top Uses of Raspberry Pi

Raspberry Pi has reached an unprecedented level of popularity. Originally created for educational purposes, it has become a go-to solution for tech enthusiasts looking for something to tinker with. This has led to the emergence of several relevant use cases. They are: [71]

- **Constructing a desktop PC:**

One can use Raspberry Pi to construct a typical desktop personal computer. The hardware includes Raspberry Pi, a micro-SD card with an operating system installed, a constant power source, and an output display device like an old monitor or television. It is also essential to have a USB mouse and keyboard. With all these, the user can work with fully functional devices for a very cheap cost.

- **Controlling IoT robots:**

Robotics is a vital part of today's technology that promises to strongly influence the future, particularly the Internet of Things (IoT). Raspberry Pi, therefore, is playing a crucial role in the technology of the future. Currently, there are several robot controller Raspberry Pi projects. Anyone can order fully packaged robot parts with DIY instructions and programs from many Pi communities. You can also choose to build your robot from scratch.

Raspberry Pi provides the best core a robot can have. Its miniature, lightweight nature, combined with the unit's low price, makes it simply perfect. The Pi Zero is very popular for robotics because it boasts an even slimmer and more compact size than other Pi models.

- **Replacing web servers:**

One other practical application of the Raspberry Pi computer is its use as a web server. This simply means configuring the computer to be able to host HTTP websites. It can function as a web server on the internet directly or in a local network such as a home or office. To do this, one must install specific software – the complete LAMP stack comprising Linux, Apache, MySQL, and PHP. After this, one can use the www directory to save HTML files, and the Raspberry Pi can function as a fully functional web server.

- **Attaching to surveillance cameras:**

Businesses, offices, and even homes need surveillance cameras to prevent and apprehend security threats. For some, this can be very expensive, especially for small businesses. However,



Raspberry Pi comes in as an excellent alternative. Combining it with a camera module allows anyone to set up their personal surveillance system.

- **Supporting digital signage:**

Most businesses now use digital signage to achieve a great deal of marketing. Information like the latest product, restaurant menus, adverts, appropriate behavior, maps, etc., can be displayed on large screens or specialized platforms. Raspberry Pi is not left out in this application – the minicomputer is perfect for displaying text and images. It can also display animations but will require more power and data consumption. Overall, using Raspberry Pi for digital signage is a cost-effective process.

- **Conducting network penetration tests:**

Computer and network security is a big deal for everyone in today's hyperconnected society. Personal and financial information stored on various sites may get into the wrong hands. This has created a massive market for cybersecurity software. But the only way to ensure that you have adequate security is to test it. With Raspberry Pi, you can create an offensive security hacking tool to 'attack' your network. Any breach found is immediately repaired and closed down. .

### III.2.4. Differences Between Arduino and Raspberry Pi [72]

- You can build a robot using either, but depending on the board, the robot will have different capabilities. Arduino is an electronic board with a simple microcontroller, whereas Raspberry Pi is a full-fledged computer.
- Unlike Arduino, Raspberry Pi has its own operating system and thanks to that, it can carry out complex operations like robot control, monitoring weather and many others. Arduino works based on simple instructions that its IDE (Integrated Development Environment) provides.
- Raspberry Pi has a superb processing power – up to 1.6 GHz (depending on the board), whereas that of Arduino is up to 16 MHz (depending on the board).
- Arduino will come in handy for controlling motors, LEDs, or interfacing sensors, whereas Raspberry Pi is good for developing software applications.
- Arduino and Raspberry Pi have different power requirements. Although both can be powered by USB, Raspberry Pi needs more current than Arduino does. This means that

you'll need a power adapter for Raspberry Pi, but a USB port of your computer will do the job in the case of Arduino.

- Since Arduino is a controller board, you can plug and unplug the power as you see fit. As a full-fledged computer, Raspberry Pi needs to be properly turned off before powering it down or after using it.
- Arduino's open-source software and hardware enables you to create your own customized Arduino board, and you can find many alternatives for it such as Teensy, ESP32, Adafruit, something that's much harder with Raspberry Pi as it's not open-source.
- In Raspberry Pi, the key programming languages used to develop applications are C, C++, Python, Ruby and Scratch. As to Arduino, it can be programmed with the use of C or C++.
- With Ethernet or WiFi, you can easily connect to the Internet with Raspberry Pi. Most Arduino boards don't support wireless connectivity and the Internet but it's possible with extra modules or shields

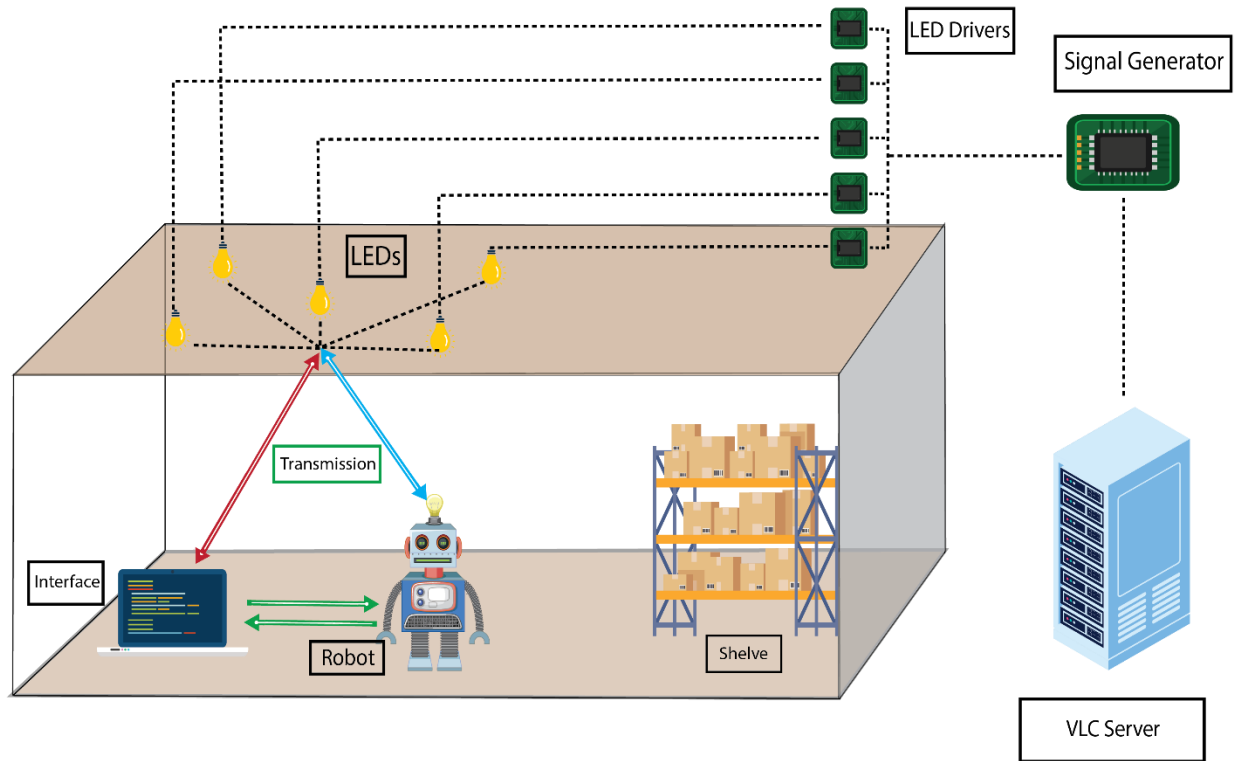
### III.3. About this application

#### III.3.1. Our scenario

In this particular section, our focus is the project we have been working on. We have set out to control a robot via VLC technology that will enter a supermarket and carry out the task of identifying food products that it comes across. There are various possibilities that could occur during this process, but we are particularly interested in the scenario where we instruct the robot to examine a specific shelf and accurately identify all the products it contains through the use of image classification.

Our initial testing phase concentrated only on bottles of water and soda, but assuming the outcomes are successful, we intend to expand our dataset to other food products. At present, we have gained control over the VLC car and are giving our full attention to training the robot on image classification, making sure that the results are accurately transmitted and received via the VLC interface. The interface we have established features a reciprocal function that allows for the same process to be carried out on the robot, which means the robot transforms into an emitter

and receiver, obtaining the command and transmitting the outcome to the interface, which also functions as an emitter and receiver. The interface conveys orders to the robot and obtains its outcomes.



**Fig III.2:** Concept of our System

### III.3.2. Command and training

- After finishing the simulation process, we saved the "soda\_water\_model.h5" of the model for direct use in real-time image classification on the Raspberry Pi 4 and webcam. where we added code that allows image classification on Real Time using the model that we installed and the picture below Shows it: (photo Raspberry + Cam).



**Fig.III.3:** Our Raspberry pi + webcam

- this is the code we are working with:

```
import tensorflow as tf
import tensorflow_hub as hub
import cv2
import numpy as np
import time
import serial
# Load the model
model = tf.keras.models.load_model('/home/raspberry/Downloads/soda_water_model.h5',
custom_objects={'KerasLayer': hub.KerasLayer})

# Set up the video capture
video_capture = cv2.VideoCapture(0) # 0 for the default camera

# Set up the serial connection
serial_connection = serial.Serial('/dev/ttyACM0', 115200) # Replace with the correct port and baud
rate

# Set the initial time
start_time = time.time()
prediction_interval = 5 # Interval in seconds for prediction

while True:
    # Capture frame-by-frame from the camera
    ret, frame = video_capture.read()

    # Display the captured frame
    cv2.imshow('Video', frame)

    # Preprocess the frame
    frame_resize = cv2.resize(frame, (224, 224))
    frame_scaled = frame_resize / 255.0
    frame_reshaped = np.reshape(frame_scaled, (1, 224, 224, 3))

    # Check if the prediction interval has elapsed
    elapsed_time = time.time() - start_time
    if elapsed_time >= prediction_interval:
        # Make prediction
        prediction = model.predict(frame_reshaped)
        pred_label = np.argmax(prediction)

        # Convert the prediction label to a result code
        if pred_label == 0:
            result_code = 'W'
```

```
elif pred_label == 1:
    result_code = 'S'
else:
    result_code = 'N'
print("Result Code:", result_code)

# Convert the result code to a string and send it over the serial connection
result_string = str(result_code)
serial_connection.write(result_string.encode())
# Reset the timer
start_time = time.time()
# Break the loop when 'q' is pressed
if cv2.waitKey(1) & 0xFF == ord('q'):
    break
# Release the video capture, close the window, and close the serial connection
video_capture.release()
cv2.destroyAllWindows()
serial_connection.close()
```

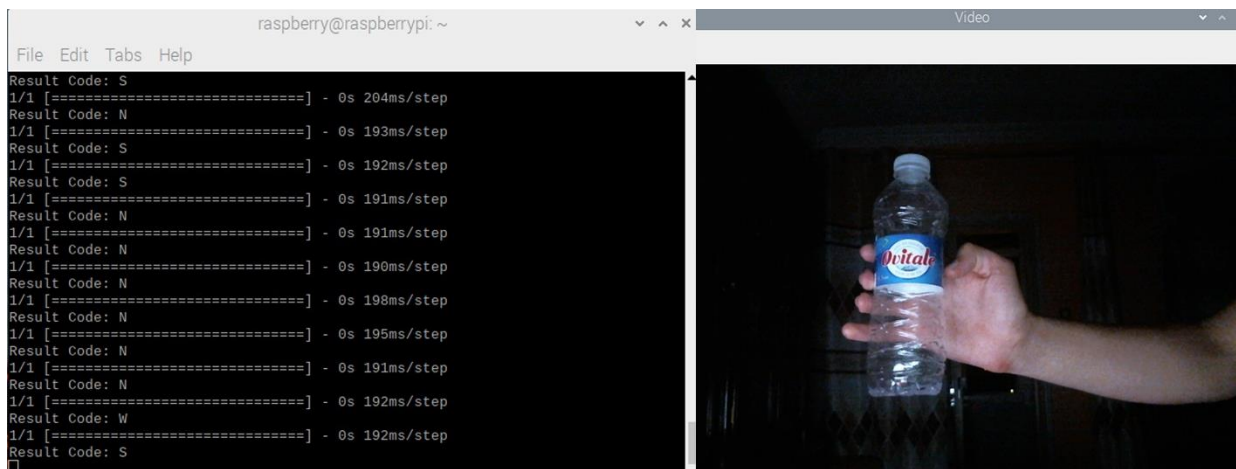
The code you provided seems to be a script that uses a trained TensorFlow model to perform real-time predictions on video frames captured from a camera. The predictions are then sent over a serial connection. Here's a breakdown of what the code does:

- It imports the necessary libraries: tensorflow, tensorflow\_hub, cv2 (OpenCV), numpy, time, and serial.
- It loads a pre-trained model using `tf.keras.models.load_model` function, specifying the path to the model file (`soda_water_model.h5`). The model is loaded with a custom object `{'KerasLayer': hub.KerasLayer}` which indicates the usage of a KerasLayer from TensorFlow Hub.
- It sets up the video capture using `cv2.VideoCapture(0)` to capture frames from the default camera (index 0).
- It establishes a serial connection using `serial.Serial('/dev/ttyACM0', 115200)` by specifying the correct port (`/dev/ttyACM0`) and baud rate (115200). You may need to replace the port with the appropriate one for your device.
- It defines a function `read_serial()` to read data from the serial connection using `serial_connection.readline()`. The received data is decoded and stripped of leading/trailing whitespaces, and then printed.

- It initializes the `start_time` variable with the current time and sets the `prediction_interval` to 5 seconds, indicating the interval at which predictions should be made.
- It starts an infinite loop that captures frames from the camera, displays them using `cv2.imshow`, and performs the prediction logic.
- Inside the loop, the frame is preprocessed by resizing it to (224, 224) using `cv2.resize`, scaling the pixel values to the range [0, 1], and reshaping it to match the expected input shape of the model.
- It checks if the prediction interval has elapsed by calculating the elapsed time as the difference between the current time and `start_time`.
- If the prediction interval has elapsed, it makes a prediction using `model.predict` on the preprocessed frame. The prediction is an array of probabilities for different classes. The class with the highest probability is determined using `np.argmax`.
- The predicted label is converted to a result code ('W', 'S', or 'N').
- The result code is sent over the serial connection using `serial_connection.write`.
- The `start_time` is reset to the current time to start the next prediction interval.
- It calls the `read_serial()` function to check if any data is available on the serial connection.
- It breaks the loop and exits the program when the 'q' key is pressed.

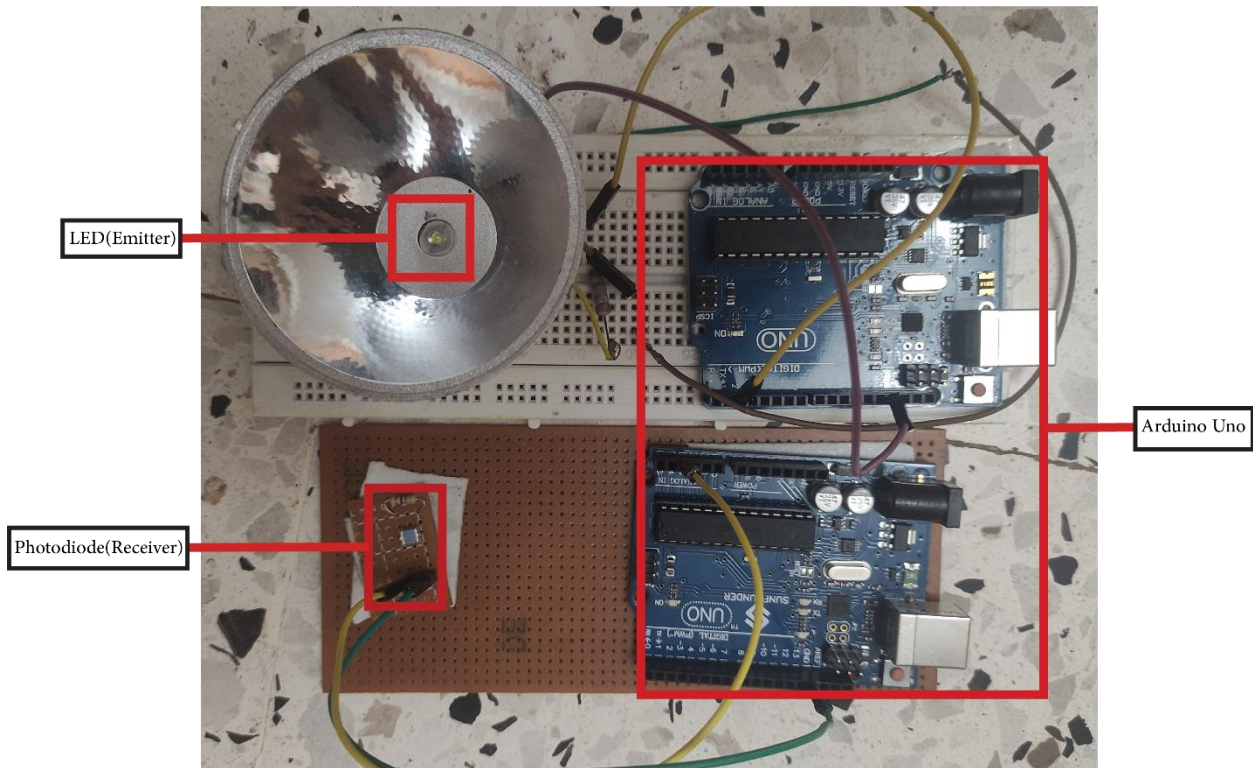
## CHAPTER III: COMMAND AND TRAINING OF ROBOT USING VLC COMMUNICATION

- Finally, it releases the video capture, closes the display window, and closes the serial connection.
- When we run the code on the Raspberry pi, the following results appear based on what the camera sees: bottle of water, Soda, or otherwise:(result picture).



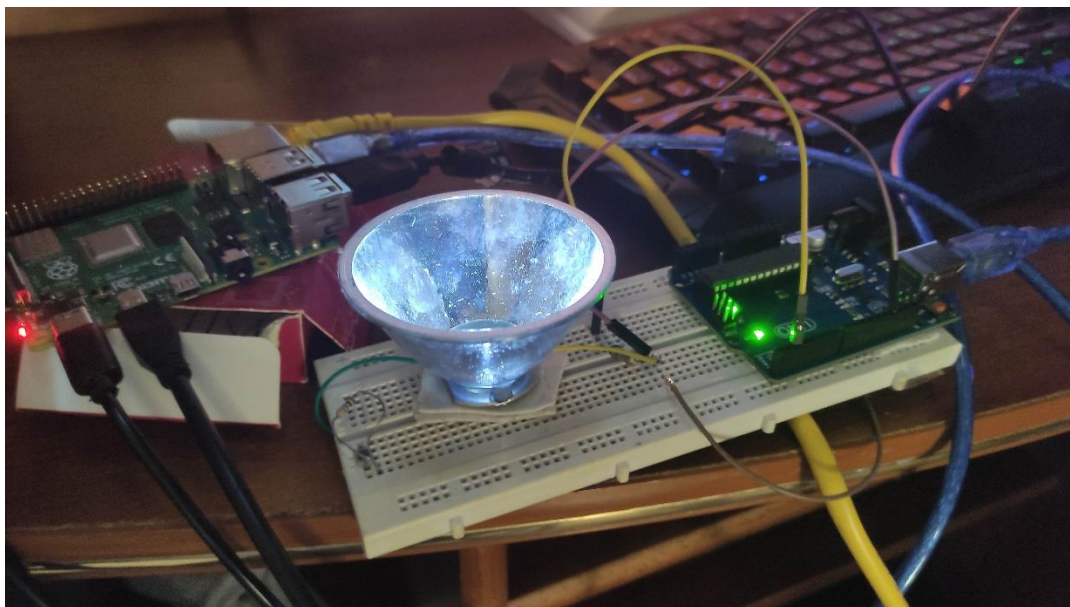
**Fig III.4:** Result of our code

- Upon the completion of this process, we brought the VLC emitter receiver represented by Arduino Uno + Breadboard programmed with a special code that transmits information via LED (emitter), with regard to the (receiver) that is represented by Arduino Uno, it is programmed with a special code to receive information using the photodiode that receives its information through led. The following pictures represent this:



**Fig III.5: Emitter + Receiver**

- We connect the emitter to the raspberry pi to transfer the acquired data through VLC technology, and the following images show this:

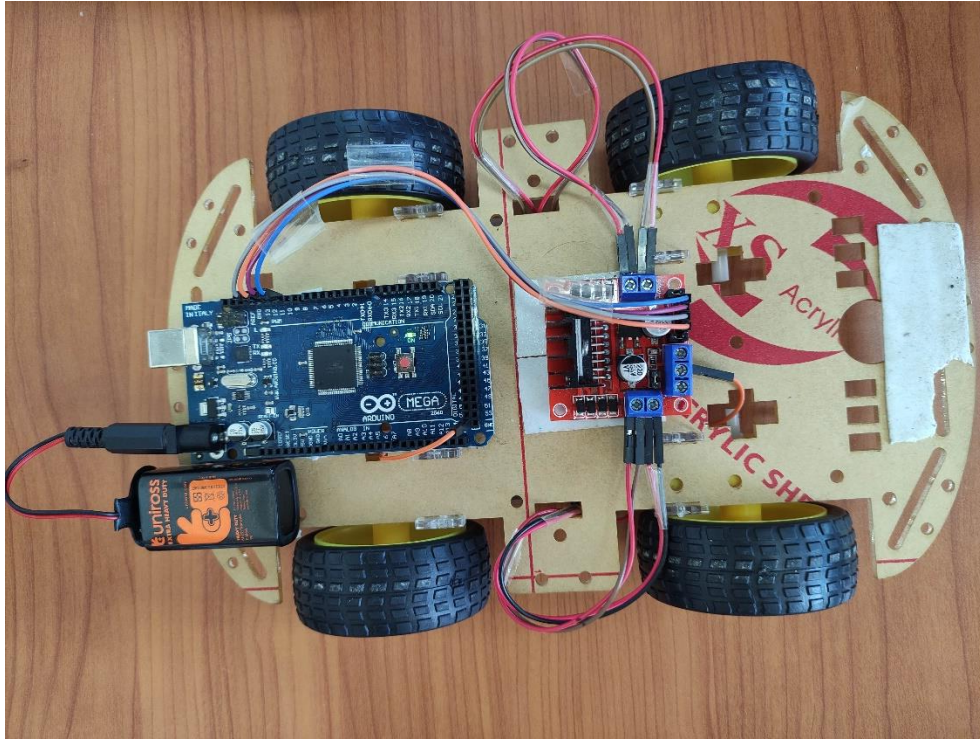


**Fig III.6: Raspberry + Emitter connection**



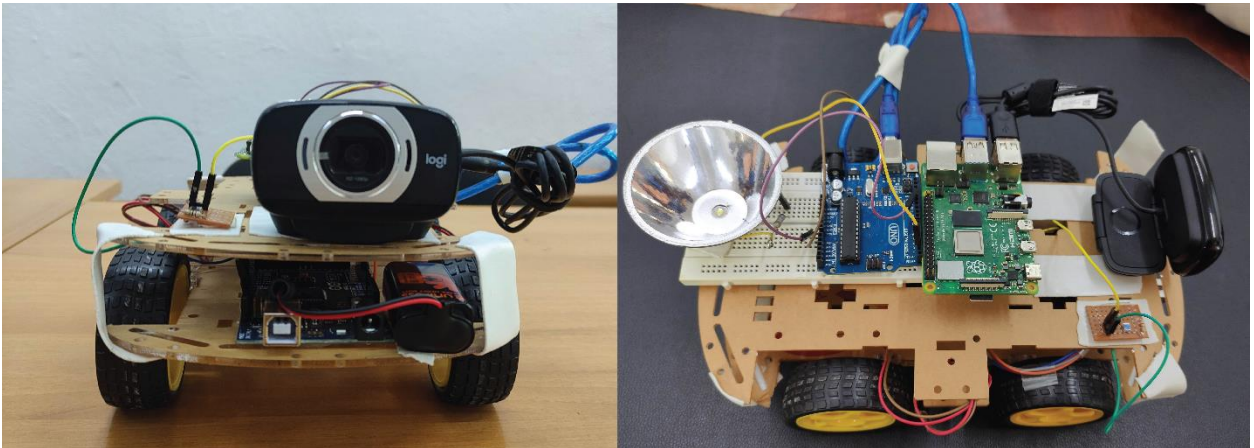
## CHAPTER III: COMMAND AND TRAINING OF ROBOT USING VLC COMMUNICATION

- Thence, we can combine the Work WE have done with the previous work with VLC car (picture of VLC Car).



**Fig III.7:** VLC-Car

- This latter becomes an advanced and intelligent robot that automatically classifies things and is controlled as a whole by VLC technology (This is the embodiment of our idea).



**Fig III.8:** Embodiment of our idea

### III.3.3. Future work

- Dig deeper into image classification by adding different data to help your robot recognize different things.
- Adding object recognition technology that allows the robot to recognize everything it sees at once.
- Expanding the use of VLC technology.
- Developing the robot's structure and seek to use it in various fields.

### III.4. Conclusion

In this chapter, we introduced the raspberry Pi and its various models, up to the latest version. We discussed its functionality, important uses, and compared several aspects between Raspberry Pi and Arduino.

Additionally, we learned how to train a robot for image classification in real-time using a Raspberry Pi 4 and webcam. We also covered sending the obtained results through VLC technology.

## **General conclusion:**

Through the continuous development of technology, including wireless communication technology, most technologies use radio waves or high frequency RF, which suffers from interference and prolonged use. Consequently, there are alternatives that solve most of these problems.

Such as, VLC which provides high bandwidth and prevents radio interference in sensitive areas that are electromagnetic and non-hazardous to health. All of this makes VLC the preferred technology for future wireless communications.

As an example, there are, Li-Fi, hospital robots, underwater. etc. Due to the development of artificial intelligence, robots, and programming languages it will make many tedious things easier for humans in the future.

In this study, we identified and explored how to train a robot image classification in real time using the available means, which are raspberry pi 4 and web camera. In addition, we wanted to send the results obtained from the latter i.e., (the results of the operation carried out by the raspberry pi) via the VLC technology that is previously used in the Arduino (transmitter, receiver), and therefore the raspberry pi outputs must be linked with the Arduino to be transferred

## List of References

- [1] Biswal, A. (2023). 7 Types of Artificial Intelligence That You Should Know in 2023. *Simplilearn.com*. <https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/types-of-artificial-intelligence>
- [2] Frankenfield, J. (2023). Artificial Intelligence: What It Is and How It Is Used. Investopedia. <https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp>
- [3] Betz, S. (2022). 7 Types of Artificial Intelligence. Built In. <https://builtin.com/artificial-intelligence/types-of-artificial-intelligence>
- [4] Coursera. (2023). 4 Types of AI: Getting to Know Artificial Intelligence. Coursera. <https://www.coursera.org/articles/types-of-ai>
- [5] What is Machine Learning? | IBM. (n.d.). <https://www.ibm.com/topics/machine-learning>
- [6] Simplilearn Americas, Inc. (n.d.). Simplilearn | Online Courses - Bootcamp & Certification Platform. Simplilearn.com. <https://www.simplilearn.com/>
- [7] What are Neural Networks? | IBM. (n.d.). <https://www.ibm.com/topics/neural-networks#:~:text=Neural%20networks%2C%20also%20known%20as,neurons%20signal%20to%20one%20another>
- [8] Investopedia. (2019, June 6). Investopedia. <https://www.investopedia.com/>
- [9] What is Deep Learning and Why It Matters? (n.d.). SAS. [https://www.sas.com/en\\_us/insights/analytics/deep-learning.html](https://www.sas.com/en_us/insights/analytics/deep-learning.html)
- [10] Hanna, K. T. (2021). robotics. WhatIs.com. <https://www.techtarget.com/whatis/definition/robotics#:~:text=Robotics%20is%20a%20branc>
- [11] Robotpark. (n.d.). <https://www.robotpark.com/>
- [12] WHAT is a ROBOT. (n.d.). <https://www.robotpark.com/What-is-a-Robot>

- [13]Tran, T. (2022, January 7). 6 Examples of Robots in Everyday Life. Orient Software. <https://www.orientsoftware.com/blog/robots-in-everyday-life/>
- [14]WirelessCommunication - Overview. (n.d.-  
b). [https://www.tutorialspoint.com/wireless\\_communication/wireless\\_communication\\_overview.htm](https://www.tutorialspoint.com/wireless_communication/wireless_communication_overview.htm)
- [15]Moozakis, C., & Kerner, S. M. (2023). What is wireless communications? Everything you need to know. Mobile Computing. <https://www.techtarget.com/searchmobilecomputing/definition/wireless>
- [16]What is Electromagnetic Waves? Definition of Electromagnetic Waves, Electromagnetic Waves Meaning - The Economic Times. (n.d.). The Economic Times. <https://economictimes.indiatimes.com/definition/electromagnetic-waves>
- [17]ElectronicsHub. (n.d.). ElectronicsHub. <https://www.electronicshub.org/>
- [18]Wireless Communication - Overview.(n.d.). [https://www.tutorialspoint.com/wireless\\_communication/wireless\\_communication\\_overview.htm](https://www.tutorialspoint.com/wireless_communication/wireless_communication_overview.htm)
- [19]Teja, R. (2021d). Wireless Communication: Introduction, Types and Applications. ElectronicsHub. <https://www.electronicshub.org/wireless-communication-introduction-types-applications/>
- [20]Advantages and Disadvantages of Wireless Communication - Javatpoint. (n.d.). www.javatpoint.com. <https://www.javatpoint.com/advantages-and-disadvantages-of-wireless-communication>
- [21]Teja, R. (2021). Wireless Communication: Introduction, Types and Applications. ElectronicsHub. <https://www.electronicshub.org/wireless-communication-introduction-types-applications/>
- [22]Jenila, C., &Jeyachitra, R. K. (2021). Green indoor optical wireless communication systems: Pathway towards pervasive deployment. Digital Communications and Networks, 7(3), 410–444. <https://doi.org/10.1016/j.dcan.2020.09.004>

- [23] The Electromagnetic Spectrum. (n.d.).  
HubbleSite.org. <https://hubblesite.org/contents/articles/the-electromagnetic-spectrum>
- [24] R. Boubezari et al. “Smartphone camera based visible light communication”. In: Journal of Lightwave Technology 34.17 (2016), pp. 4121–4127.)
- [25] U.SUGANYA1, C.SUBHALAKSHMIPRIYA, “Li – FI ( Light Fidelity ) Technology ”, INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS ” , Vol.3 Issue No.1, PP. 26-32, January 2015 )
- [26] K. Sindhubala and B. Vijayalakshmi, “Design and Implementation of Visible Light Communication System In Indoor Environment”, ARPN Journal of Engineering and Applied Sciences ” , VOL. 10, NO. 7, APRIL 2015 )
- [27] PokpoomChanthosot, VittayaTipsuwanporn, ViriyaKrongratana and ThanapornLilawatthanun, “The Indoor Use Development for Visible Light Communication”, Proceedings of the World Congress on Engineering and Computer Science 2015 Vol I WCECS 2015, October 21-23, 2015, San Francisco, USA
- [28] Sathiya.T, Prof.E.Divya, Prof.S.Raja, “Visible Light Communication for Wireless Data Transmission”, International Journal of Innovative Research In Electrical, Electronics, Instrumentation and Control Engineering”, Vol. 2, Issue No. 2, February 2014.
- [29] Rahul R. Sharma, Akshay Sanganal, Sandhya Pati, “Implementation of A Simple Li-Fi Based System”, International Journal of Computing and Technology, Volume 1, Issue No. 9, October 2014.
- [30] Abhishek Kurup, Vipin Tiwari, Selvanathiya, “Implementation and Demonstration of LI-FI Technology”, International Journal of Research in Engineering and Technology, Volume: 03, Issue No. 03, Mar-2014.

- [31] Megha Goyal, Dimple Sapru, Asha Bhagashra, “New Epoch of Wireless Communication: Light Fidelity”, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 1, Issue No. 2, April 2013.
- [32] Rajan Sagotra, Reena Aggarwal, “Visible Light Communication”, International Journal of Computer Trends and Technology (IJCTT) , volume 4 , Issue No. 4 , April 2013.
- [33] M. Saadi<sup>1</sup>, L. Wattisuttikulkiy , Y. Zhao , P. Sangwongngam, “Visible Light Communication: Opportunities, Challenges and Channel Models”, International Journal of Electronics & Informatics , Vol. 2, No. 1, 2013.
- [34] L. E. M. Matheus, A. B. Vieira, L. F. M. Vieira, M. A. M. Vieira and O. Gnawali, "Visible Light Communication: Concepts, Applications and Challenges," in IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3204-3237, Fourthquarter 2019, doi: 10.1109/COMST.2019.2913348.
- [35] Durgesh Gujjari —VISIBLE LIGHT COMMUNICATION| Submitted in partial fulfilment of the requirements for the degree of Master of Applied Science Dalhousie University Halifax, Nova Scotia August 2012.)
- [36] .igi-global.com/dictionary/received-signal-strength-rss/43649)
- [37] S. Dimitrov and H. Haas. Principles of LED light communications: towards networked Li-Fi. Cambridge University Press, 2015
- [38] H.-H. Lu et al. “A 56Gb/s PAM4 VCSEL-based LiFi transmission with two-stage injectionlocked technique”. In: IEEE Photonics Journal 9.1 (2017), pp. 1–8.
- [39]H. Haas. “LiFi: Conceptions, misconceptions and opportunities”. In: IEEE Photonics Conference (IPC). IEEE. 2016, pp. 680–681.
- [40] M. Ayyash et al. “Coexistence of WiFi and LiFi toward 5G: concepts, opportunities, and challenges”. In: IEEE Communications Magazine 54.2 (2016), pp. 64–71.
- [41] S. Li, A. Pandharipande, and F. M. J. Willems. “Unidirectional visible light communication and illumination with LEDs”. In: IEEE Sensors Journal 16.23 (2016), pp. 8617–8626.

- [42] W. Yuanquan and C.Nan. "A high-speed bi-directional visible light communication system based on RGB-LED". In: China Communications 11.3 (2014), pp. 40–44.
- [43]Latif Ullah Khan, —Visible light communication: Applications, architecture, standardization and research challenges|| Digital Communications and Networks, Volume 3, Issue 2, 2017, Pages 78-88, ISSN 2352-8648
- [44] L. E. M. Matheus, A. B. Vieira, L. F. M. Vieira, M. A. M. Vieira and O. Gnawali, "Visible Light Communication: Concepts, Applications and Challenges," in IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3204-3237,Fourthquarter 2019,doi: 10.1109/COMST.2019.2913348
- [45] T. Yamazato et al., "Image-sensor-based visible light communication for automotive applications," in IEEE Communications Magazine, vol. 52, no. 7, pp. 88-97, July 2014,doi: 10.1109/MCOM.2014.6852088
- [46] *What is Python? Executive Summary.* (n.d.). Python.org.  
<https://www.python.org/doc/essays/blurb/>
- [47] Fran. (2021, December 30). *What is Python used for? | 10 practical Python uses - FutureLearn.* FutureLearn. <https://www.futurelearn.com/info/blog/what-is-python-used-for>
- [48] *What is Python? | Teradata.* (n.d.). <https://www.teradata.com/Glossary/What-is-Python#:~:text=Python%20is%20an%20interpreted%2C%20object,British%20comedy%20group%20Monty%20Python>
- [49] Buckley, I. (2021). The 13 Best Browser IDEs Every Programmer Should Know About.  
*MUO.* <https://www.makeuseof.com/tag/programmer-browser-ides/>
- [50] *Colaboratory.* (n.d.). Google Workspace Marketplace.  
<https://workspace.google.com/marketplace/app/colaboratory/1014160490159>
- [51] Malik, A. (n.d.). *How Python Is Different From Other Languages.* <https://www.c-sharpcorner.com/article/how-python-is-different-from-other-languages/>



- [52] Techopedia. (2021, June 21). *What is Classification? - Definition from Techopedia*.  
<https://www.techopedia.com/definition/13779/classification>
- [53] SuperAnnotate AI Inc. (n.d.). *What is image classification? Basics you need to know / SuperAnnotate*. <https://www.superannotate.com/blog/image-classification-basics>
- [54] Boesch, G. (2023). *A Complete Guide to Image Classification in 2023*. *viso.ai*.  
<https://viso.ai/computer-vision/image-classification/>
- [55] *What is a Neural Network? - Artificial Neural Network Explained - AWS*. (n.d.). Amazon Web Services, Inc. <https://aws.amazon.com/what-is/neural-network/>
- [56] *What are Neural Networks? | IBM*. (n.d.). <https://www.ibm.com/topics/neural-networks>
- [57] Gupta, D. (2023). *Transfer Learning and the Art of Using Pre-trained Models in Deep Learning*. *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2017/06/transfer-learning-the-art-of-fine-tuning-a-pre-trained-model/>
- [58] Budu, E., & Budu, E. (2023). *What Does Pre-training a Neural Network Mean? | Baeldung on Computer Science*. *Baeldung on Computer Science*.  
<https://www.baeldung.com/cs/neural-network-pre-training>
- [59] Shrimali, V., & Shrimali, V. (2023). *Pre Trained Models for Image Classification - PyTorch*. *LearnOpenCV – Learn OpenCV, PyTorch, Keras, Tensorflow With Examples and Tutorials*. <https://learnopencv.com/pytorch-for-beginners-image-classification-using-pre-trained-models/>
- [60] *MobileNet V2 Classification Classification Model*. (n.d.).  
<https://roboflow.com/model/mobilenet-v2-classification>
- [61] Hollemans, M. (n.d.). *New mobile neural network architectures*.  
<https://machinethink.net/blog/mobile-architectures/>

- [62] Wang, C. (2018, August 14). A Basic Introduction to Separable Convolutions - Towards Data Science. *Medium*. <https://towardsdatascience.com/a-basic-introduction-to-separable-convolutions-b99ec3102728>
- [63] *CS231n Convolutional Neural Networks for Visual Recognition*. (n.d.). <https://cs231n.github.io/convolutional-networks/>
- [64] Potrimba, P. (2023). What is a Convolutional Neural Network? *Roboflow Blog*. <https://blog.roboflow.com/what-is-a-convolutional-neural-network/>
- [65] Wong, W. (2022, January 4). What is Residual Connection? - Towards Data Science. *Medium*. <https://towardsdatascience.com/what-is-residual-connection-efb07cab0d55>
- [66] TenTen-Teng. (n.d.). *GitHub - TenTen-Teng/MobileNet-V2: Build MobileNet-V2 in Keras 2.1.4, TensorFlow 1.4.0*. GitHub. <https://github.com/TenTen-Teng/MobileNet-V2>
- [67] *What is a Raspberry Pi?* (2015, August 20). Raspberry Pi Foundation. <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>
- [68] Ltd, R. P. (n.d.). *Buy a Raspberry Pi 4 Model B – Raspberry Pi*. Raspberry Pi. <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
- [69] Python, R. (2023). Build Physical Projects With Python on the Raspberry Pi. *realpython.com*. <https://realpython.com/python-raspberry-pi/>
- [70] Nuttall, B. (n.d.). *What is a Raspberry Pi?* Opensource.com. <https://opensource.com/resources/raspberry-pi>
- [71] *Raspberry Pi Models and Features*. (2022b, August 26). Spiceworks. <https://www.spiceworks.com/tech/networking/articles/what-is-raspberry-pi/>
- [72] *Leo Rover Blog - Raspberry Pi or Arduino – when to choose which?* (n.d.). <https://www.leorover.tech/post/raspberry-pi-or-arduino-when-to-choose-which>

# Annex 1

## VLC Emitter Code:

Sending outcomes of serial connection via VLC

```
#include <TimerOne.h>
#include <util/atomic.h>

#define SYMBOL_PERIOD 500
#define WORD_LENGTH 10
#define SYNC_SYMBOL 0xD5
#define ETX 0x03
#define STX 0x02

#define OUT_LED() DDRD |= (1 << 2)
#define SET_LED() PORTD |= (1 << 2)
#define CLR_LED() PORTD &= ~(1 << 2)

unsigned char frame_buffer[38];
char frame_index = -1;
char frame_size = -1;
unsigned char bit_counter = 0;
unsigned long int manchester_data;

char com_buffer[32]; // Declaration of com_buffer as a global variable

void to_manchester(unsigned char data, unsigned long int* data_manchester) {
    unsigned int i;
    (*data_manchester) = 0x02;
    (*data_manchester) = (*data_manchester) << 2;
    for (i = 0; i < 8; i++) {
        if (data & 0x80)
            (*data_manchester) |= 0x02;
        else
            (*data_manchester) |= 0x01;
        (*data_manchester) = (*data_manchester) << 2;
        data = data << 1;
    }
    (*data_manchester) |= 0x01;
}

void emit_half_bit() {
    if (manchester_data & 0x01) {
        SET_LED();
    }
}
```

```

} else {
    CLR_LED();
}
bit_counter--;
manchester_data = (manchester_data >> 1);
if (bit_counter == 0) {
    manchester_data = 0xAAAAAAAA;
    if (frame_index >= 0) {
        if (frame_index < frame_size) {
            to_manchester(frame_buffer[frame_index], &manchester_data);
            frame_index++;
        } else {
            frame_index = -1;
            frame_size = -1;
        }
    }
    bit_counter = WORD_LENGTH * 2;
}
}

```

```

void init_frame(unsigned char* frame) {
    memset(frame, 0xAA, 3);
    frame[3] = SYNC_SYMBOL;
    frame[4] = STX;
    frame_index = -1;
    frame_size = -1;
}

```

```

int create_frame(char* data, int data_size, unsigned char* frame) {
    memcpy(&(frame[5]), data, data_size);
    frame[5 + data_size] = ETX;
    return 1;
}

```

```

int write(char* data, int data_size) {
    if (frame_index >= 0)
        return -1;
    if (data_size > 32)
        return -1;
    create_frame(data, data_size, frame_buffer);
    ATOMIC_BLOCK(ATOMIC_RESTORESTATE) {
        frame_index = 0;
        frame_size = data_size + 6;
    }
    return 0;
}

```

```

}

int transmitter_available() {
    if (frame_index >= 0)
        return 0;
    return 1;
}

void init_emitter() {
    manchester_data = 0xFFFFFFFF;
    bit_counter = WORD_LENGTH * 2;
}

const int LED_PIN = 2; // Pin connected to the LED for VLC

void setup() {
    Serial.begin(115200); // Set the baud rate for serial communication with the
computer
    OUT_LED();
    init_frame(frame_buffer);
    init_emitter();
    Timer1.initialize(SYMBOL_PERIOD);
    Timer1.attachInterrupt(emit_half_bit);

    pinMode(LED_PIN, OUTPUT);
}

void loop() {
    if (Serial.available() > 0) {
        char receivedData = Serial.read();
        Serial.println("we got to the loop");

        // Send the received data over VLC

        // Perform actions based on the received data
        if (receivedData == 'W') {
            // Water detected
            char* msg = "This is water ";
            char com_buffer_nb_bytes = 0;
            sendOverVLC(msg);
            // Perform the desired actions for water classification
        } else if (receivedData == 'S') {
            // Soda detected
            char* msg = "This is soda ";
            char com_buffer_nb_bytes = 0;

```

```

    sendOverVLC(msg);
    // Perform the desired actions for soda classification
} else if (receivedData == 'N') {
    // Neither water nor soda detected
    char* msg = "There is nothing";
    char com_buffer_nb_bytes = 0;
    sendOverVLC(msg);
    // Perform the desired actions for neither classification
}
}
}

void sendOverVLC(char* data) {
    // Convert the data to a format suitable for VLC (e.g., encoding, modulation)
    Serial.println("sending....");
    static int i = 0;
    memcpy(com_buffer, data, 11);
    com_buffer[11] = i + '0';
    if (write(com_buffer, 12) < 0) {
        delay(10);
    } else {
        i++;
        if (i > 9)
            i = 0;
    }
    // Rest of the code...
}

```

## Annex 2

### Receiver Code:

```
#include <TimerOne.h>
#include "receiver_types.h"
#define INT_REF
enum receiver_state frame_state = IDLE ;
#define SENSOR_PIN 3
#define SYMBOL_PERIOD 500
#define SAMPLE_PER_SYMBOL 4
#define WORD_LENGTH 10
#define SYNC_SYMBOL 0xD5
#define ETX 0x03
#define STX 0x02
char frame_buffer[38] ;
int frame_index = -1 ;
int frame_size = -1 ;
unsigned int signal_mean = 0 ;
unsigned long acc_sum = 0 ;
unsigned int acc_counter = 0 ;
long shift_reg = 0;
void ADC_setup() {
    ADCSRA = bit (ADEN);
    ADCSRA |= bit (ADPS0) | bit (ADPS1) | bit (ADPS2);
#ifdef INT_REF
    ADMUX = bit (REFS0) | bit (REFS1);
#else
    ADMUX = bit (REFS0) ;
#endif
}
void ADC_start_conversion(int adc_pin) {
    ADMUX &= ~(0x07) ;
    ADMUX |= (adc_pin & 0x07) ;
    bitSet (ADCSRA, ADSC) ;
}
int ADC_read_conversion() {
    while (bit_is_set(ADCSRA, ADSC));
    return ADC ;
}
#define START_SYMBOL 0x02
#define STOP_SYMBOL 0x01
```

```

#define START_STOP_MASK ((STOP_SYMBOL << 20) | (START_SYMBOL << 18) |
STOP_SYMBOL)
#define SYNC_SYMBOL_MANCHESTER (0x6665)
inline int is_a_word(long * manchester_word, int time_from_last_sync, unsigned
int * detected_word) {
    if (time_from_last_sync >= 20 || frame_state == IDLE) {
        if (((*manchester_word) & START_STOP_MASK) == (START_STOP_MASK)) {
            (*detected_word) = ((*manchester_word) >> 2) & 0xFFFF;
            if (frame_state == IDLE) {
                if ((*detected_word) == SYNC_SYMBOL_MANCHESTER) return 2 ;
            }
            return 1 ;
        } else if (frame_state != IDLE && time_from_last_sync == 20) {
            (*detected_word) = ((*manchester_word) >> 2) & 0xFFFF;
            return 1 ;
        }
    }
    return 0 ;
}
inline int insert_edge( long * manchester_word, char edge, int edge_period, int
* time_from_last_sync, unsigned int * detected_word) {
    int new_word = 0 ;
    int is_a_word_value = 0 ;
    int sync_word_detect = 0 ;
    if ( ((*manchester_word) & 0x01) != edge ) {
        if (edge_period > (SAMPLE_PER_SYMBOL + 1)) {
            unsigned char last_bit = (*manchester_word) & 0x01 ;
            (*manchester_word) = ((*manchester_word) << 1) | last_bit ;
            (*time_from_last_sync) += 1 ;
            is_a_word_value = is_a_word(manchester_word, (*time_from_last_sync),
detected_word);
            if (is_a_word_value > 0) {
                new_word = 1 ;
                (*time_from_last_sync) = 0 ;
                if (is_a_word_value > 1) sync_word_detect = 1 ;
            }
        }
        if (edge < 0) {
            (*manchester_word) = ( (*manchester_word) << 1) | 0x00 ;
        } else {
            (*manchester_word) = ( (*manchester_word) << 1) | 0x01 ;
        }
        (*time_from_last_sync) += 1 ;
        is_a_word_value = is_a_word(manchester_word, (*time_from_last_sync),
detected_word);
    }
}

```



```

    if (sync_word_detect == 0 && is_a_word_value > 0) {
        new_word = 1 ;
        (*time_from_last_sync) = 0 ;
    }
} else {
    new_word = -1 ;
}
return new_word ;
}

#define EDGE_THRESHOLD 4
int oldValue = 0 ;
int steady_count = 0 ;
int dist_last_sync = 0 ;
unsigned int detected_word = 0;
int new_word = 0;
char old_edge_val = 0 ;
void sample_signal_edge() {
    char edge_val ;
    int sensorValue = ADC_read_conversion();
    ADC_start_conversion(SENSOR_PIN);
    if ((sensorValue - oldValue) > EDGE_THRESHOLD) edge_val = 1 ;
    else if ((oldValue - sensorValue) > EDGE_THRESHOLD) edge_val = -1;
    else edge_val = 0 ;
    oldValue = sensorValue ;
    if (edge_val == 0 || edge_val == old_edge_val || (edge_val != old_edge_val &&
steady_count < 2)) {
        if ( steady_count < (4 * SAMPLE_PER_SYMBOL)) {
            steady_count ++ ;
        }
    } else {
        new_word = insert_edge(&shift_reg, edge_val, steady_count, &(dist_last_sync),
&detected_word);
        if (dist_last_sync > (8 * SAMPLE_PER_SYMBOL)) {
            dist_last_sync = 32 ;
        }
        steady_count = 0 ;
    }
    old_edge_val = edge_val ;
}

int add_byte_to_frame(char * frame_buffer, int * frame_index, int * frame_size,
enum receiver_state * frame_state , unsigned char data) {
    if (data == SYNC_SYMBOL) {
        (*frame_index) = 0 ;
        (*frame_size) = 0 ;
    }
}

```

```

    (*frame_state) = SYNC ;
    return 0 ;
}
if ((*frame_state) != IDLE) {
    frame_buffer[*frame_index] = data ;
    (*frame_index) ++ ;
    if (data == STX) {
        (*frame_state) = START ;
        return 0 ;
    } else if (data == ETX) {
        (*frame_size) = (*frame_index) ;
        (*frame_index) = -1 ;
        (*frame_state) = IDLE ;
        return 1 ;
    } else if ((*frame_index) >= 38) {
        (*frame_index) = -1 ;
        (*frame_size) = -1 ;
        (*frame_state) = IDLE ;
        return -1 ;
    } else {
        (*frame_state) = DATA ;
    }
    return 0 ;
}
return -1 ;
}
void setup() {
    int i;
    Serial.begin(115200);
    Serial.println("Start of receiver program");
    ADC_setup();
    ADC_start_conversion(SENSOR_PIN);
    Timer1.initialize(SYMBOL_PERIOD / SAMPLE_PER_SYMBOL);
    Timer1.attachInterrupt(sample_signal_edge);
}
void loop() {
    int i;
    unsigned char received_data;
    char received_data_print ;
    int nb_shift ;
    int byte_added = 0 ;
    if (new_word == 1) {
        received_data = 0 ;
        for (i = 0 ; i < 16 ; i = i + 2) {

```

```
received_data = received_data << 1 ;
if (((detected_word >> i) & 0x03) == 0x01) {
    received_data |= 0x01 ;
} else {
    received_data &= ~0x01 ;
}
}
received_data = received_data & 0xFF ;
new_word = 0 ;
if ((byte_added = add_byte_to_frame(frame_buffer, &frame_index, &frame_size,
&frame_state, received_data)) > 0) {
    frame_buffer[frame_size - 1] = '\0';
    Serial.println(&(frame_buffer[1]));
}
}
}
```