

PREVENTION OF WILD-LIFE COLLISION IN RAILWAY TRACKS

Submitted in partial fulfilment of the requirements for the award of

Bachelor of Engineering Degree in
Electronics and Communications Engineering

By

Kalaivani A (38130094)

Hemalatha J (38130083)



**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING**

**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC**

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI -600119

MAY 2022



DEPARTMENT OF ELECTRONICS AND COMMUNICATION

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Kalaivani A (38130094)** and **Hemalatha J (38130083)** who carried out the project entitled “**PREVENTION OF WILD-LIFE COLLISION IN RAILWAYTRACKS**” under our supervision from November 2021 to May 2022.

Internal Guide

Dr. Rajasekar, B.E, MTech, Ph.D.

HEAD OF THE DEPARTMENT

Dr. T. Ravi, M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We **Kalaivani A (38130094)** and **Hemalatha J (38130083)** hereby declare that the Project Report titled **“PREVENTION OF WILD-LIFE COLLISION IN RAILWAYTRACKS”** done by us under the guidance of **Dr. B. Rajasekar BE, M.Tech.,Ph.D** at Sathyabama Institute of Science and Technology, **Chennai-600119** is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE:

1.

2.

PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGMENT

We are pleased to acknowledge our sincere thanks to the Board of Management of SATHYABAMA for their kind encouragement in doing this project and completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA M.E.,Ph.D., Dean**, School of Electronics and Communication Engineering and **Dr.T.RAVI M.E.,Ph.D** Head of the department, Dept. of Electronics and communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would also like to express our sincere and deep sense of gratitude to our internal guide **Dr. B. Rajasekar BE, M.Tech.,Ph.D** for his valuable guidance and suggestions during the progressive reviews for our project.

We wish to express our thanks to all teaching and Non-teaching staff members of the Department of Electronics and Communication who were helpful in many ways for the completion of the project.

ABSTRACT

As high-speed railway infrastructure is being developed at break-neck pace throughout India. Train accidents are becoming more frequent. While number of people killed or injured in such accidents is well documented. Little research has been done to analyses the number of animal casualties in Indian railways. Rail transport is generally considered more environmentally friendly than road transport due to lower energy consumption, lower carbon dioxide emissions or higher transport capacity per unit. Moreover, rail embankments can serve as a refuge for some species and the rail infrastructure can have a positive effect on birds, as bridges, power lines or signboards can provide perches for hunting and nesting sites. Despite these positives, the rail infrastructure causes habitat fragmentation that affects the daily movement of wildlife. This leads to wildlife-train collisions (WTCs) that are usually not extensively studied, when compared to the same crashes on roads, due to a lack of relevant data. A collision of a heavy locomotive with an animal often causes minimal damage to the vehicle and presents a low risk of injury to passengers. WTCs can have a significant impact, however, on certain animal species, as trains are able to kill a number of animals in a single accident... The impacts of WTCs on society include, according to, ethical (animal suffering), social (working environment of train drivers), economic (crash-related damage and following compensation) and ecological (mortality) losses. Proficient reliable and autonomous monitoring of human settlements bordering forest areas can help reduce such animal-human conflicts. A broad range of techniques in computer vision and deep learning have shown enormous potential to solve such problems.

In this paper, a novel, efficient and reliable system is presented which automatically detects wild animals using computer vision. The proposed method uses the YOLO object detection model to ascertain presence of wild animals in images. The model is fine-tuned for identifying different entities – humans, and different types of animals (elephant, zebra, giraffe, lion and cheetah etc.). Once detected, based on the perceived information, notifications are sent to alert the concerned authorities. The design of a prototype for the proposed solution is also described, which uses Arduino devices equipped with cameras. The proposed method achieves an accuracy of 98.8% and 99.8% to detect animals and humans respectively.

TABLE CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF ABBREVIATIONS	X
	LIST OF FIGURES	XII
	LIST OF TABLES	XIII
1	INTRODUCTION	1
2	LITERATURE SURVEY	6
	2.1 ANIMAL DETECTION USING COMPUTER DETECTION	6
3	SCOPE OF THE PRESENT INVESTIGATION	8
	3.1 OBJECTIVE	
	3.2 FACTORS AFFECTING RAIL ACCIDENTS	8
	NEED TO PREVENT SUCH ACCIDENTS	8
	ADVANTAGES OF THE PROPOSED SYSTEM	9
	BLOCK DIAGRAM	11
	WORKING	11
4	EXPERIMENT, MATERIALS AND METHODS	13
	4.1 SOFTWARE USED	13
	4.2 HARDWARE USED	13

4.3 OVERVIEW OF THE YOLO OBJECT DETECTION ALGORITHM	13
4.4 SOFTWARE DESCRIPTION	14
4.4.1 PYTHON SPYDER IDE	14
4.5 PROGRAMMING LANGUAGE	14
4.6 HARDWARE DESCRIPTION	14
4.6.1 ARDIUNO UNO	14
4.6.2 OVERVIEW	15
4.6.3 SPECIFICATIONS	16
4.6.4 POWER	16
4.6.5 MEMORY	17
4.6.6 INPUT AND OUTPUT	17
4.6.7 PROGRAMMING	19
4.7 APR9600 RE-RECORDING VOICE IC	19
4.7.1 SCHEMATIC DIAGRAM	19
4.7.1.1 SCHEMATIC EXPLANATION	19
4.7.2 FUNCTIONAL DESCRIPTION	20
4.7.2.1 BLOCK DIAGRAM OF APR9600	22
4.7.2.2 SAMPLING RATIO AND VOICE QUALITY	22
4.8 LIQUID CRYSTAL DISPLAY	22

4.8.1 PIN DIAGRAM	23
4.9 NODEMCU	24
4.9.1 NODEMCU DEVOLPMENT BOARD/ KIT	25
4.9.2 NODEMCU WITH ARDIUNO IDE	25
4.9.3 NODEMCU DEV KIT v1.0 PIN DESCRIPTION	25
4.9.3.1 GPIO PINS	26
4.9.3.2 ADC CHANNEL	27
4.9.3.3 SPI PINS	27
4.9.3.4 12C INTER-INTEGRATED PINS	27
4.9.3.5 UART PINS	27
4.10 GPS MODULE WITH PATCH ANTENNA	27
4.10.1 SPECIFICATIONS	28
4.11 ENCODER WITH RF TRANSMITTER	29
4.11.1 SCHEMATIC DIAGRAM	29
4.11.2 SCHEMATIC EXPLANATION	29
4.11.3 HT12E ENCODER	30
4.12 DECODER WITH RF RECEIVER	30
4.12.1 SCHEMATIC DIAGRAM	31
4.12.2 FEATURES	31
4.12.3 HT12D DECODER	32
4.13 CONTROLLING ARDIUNO BOARD WITH SMART PHONE AND BLYNK VIA INTERNET	32

5	RESUL	33
6	CONCLUSION	37
	REFERENCES	38

LIST OF ABBREVIATIONS

GPS	GLOBAL POSITIONING SYSTEM
IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
LCD	LIQUID CRYSTAL DISPLAY
RF RX	RADIO FREQUENCY RECEIVER
RF TX	RADIO FREQUENCY TRANSMITTER
YOLO	YOU ONLY LOOK ONCE
PC	PERSONAL COMPUTER
OPENCV	OPEN SOURCE COMPUTER VISION LIBRARY
SRAM	STATIC RANDOM ACCESS MEMORY
GND	GROUND
APR	APACHE PORTABLE RUNTIME
PCB	PRINTED CIRCUIT BOARD
GPIO	GENERAL PURPOSE INPUT OUTPUT
ADC	ANALOG TO DIGITAL CONVERTER
SPI	SERIAL PERIPHERAL INTERFACE
IC	INTEGRATED CIRCUIT
UART	UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER
CMOS	COMPLEMENTARY METAL-OXIDE-SEMICONDUCTOR
ASK	AMPLITUDE SHIFT KEY

COM PORT	COMMUNICATION PORT
IoT	INTERNET OF THINGS
NODEMCU	NODE MICROCONTROLLER UNIT
WIFI	WIRELESS FIDELITY
VIA	BY WAY OF
NMOS	N-TYPE-METALOXIDE SEMICONDUCTOR
LSI	LARGE SCALE INTEGRATION
POT	PATCH ANTENNA ON TOP
LED	LIGHT EMITTING DIODE
ESP	ELECTRONIC STABILITY PROGRAM

LIST OF FIGURES

4.7.1	SCHEMATIC DIAGRAM OF APR
4.7.2.1	BLOCK DIAGRAM OF APR9600
4.8.1	PIN DIAGRAM OF LCD
4.9.4	PINOUT OF NODEMCU
4.10	FIGURE OF GPS MODULE
5.1	BLOCK DIAGRAM OF THE PROJECT

LIST OF TABLES

4.6.3	SPECIFICATIONS OF ARDUINO UNO
4.8.1	PIN DESCRIPTION OF LCD
4.10.2	ELECTRICAL SPECIFICATIONS

CHAPTER 1

1. INTRODUCTION

There have been increasing reports of wild animals entering villages or towns, especially in settlements surrounding forest areas, endangering human lives. Intrusions by animals causes huge losses are it in terms of crop loss or cattle being attacked. Increasing human population leading to decreasing forest cover is one of the leading causes for rise in human animal conflicts. Current methods to reduce such conflicts include installation of electric fences or have sentries watch for animals through the night. Electric fences cause severe injury to animals. Moreover, they require enormous initial investment and additionally have high maintenance costs. Recent developments in the field of computer science enable use of technology to create low-cost solutions to such problems. Computer vision is one such technology which could potentially solve most of the associated problems.

Most of these incidents don't generate international headlines; nor do the deaths of thousands of additional animals killed by trains worldwide each year. In fact, most wildlife-train collisions go unnoticed; their fatalities left uncounted — which has made it difficult for experts to study the problem and mitigate its impacts.

That puts us woefully behind similar research to reduce vehicle-wildlife collisions on roads, an active field of research for the past two decades. That's because car-animal collisions present a greater danger to human safety and property, according to a 2016 study surveying the emerging field of railway ecology. "Despite the field of road ecology rapidly expanding and the large footprint created by railways, there is a prominent lack of research related to railways and their effects on wildlife," the study found.

Here's what we do know: Like roads, railways fragment habitat and can affect all kinds of wildlife in varying ways. Collisions are the most common cause of mortality, but some animals die from electrocution or being stuck between the rails, leaving them susceptible to predation, starvation or dehydration.

Krizhevsky, who submitted the winning entry for the Image Net classification challenge, introduced a Deep Neural Network based solution for image classification.

It is now considered a landmark achievement in computer vision and has contributed to increased research in the field.

The main intent of this paper is to describe the design for a computer vision system, capable of detecting wild animals and tracking their movement. DCNNs could be leveraged to detect the presence of animals in the captured images.

In addition to detecting the presence of an animal, in order to effectively track them and monitor their actions, it is also necessary to localize the animals within the image. This is the task of object detection. Object detection systems predict regions of interest within images, and in addition classify entities within these regions. Thus, object detection is the ideal choice for the system proposed in this project.

This project introduces a novel method of reducing human animal conflicts, through constant and automatic monitoring of vulnerable areas using a system of cameras. The proposed solution is accurate and cost effective and to an extent, can be customized specifically for a particular region. The timing of this research, experts tell us, is important. With rail transit of products and materials on the rise and high-speed rail networks expected to grow as we work globally to lower our carbon footprint, the number of fatalities could soon increase unless we devote more resources to additional research and mitigation.

RAILWAY BARRIERS

Rail tracks can make for tough times if you're a toad — even a big one.

In Brazil a 2018 study found an estimated 10,000 Cururu toads (*Rhinella marina*) and related species, often called giant toads, were dying every year along a 500-mile stretch of railway. Researcher Rubem Dornas says they still don't know exactly why so many toads die, but it appears the tracks formed a barrier the toads can't cross while migrating. Despite the large size of the toads, which average about 4 to 6 inches in length, the researchers found they may not be able to jump or climb over rails more than 6 inches high.

“We think the main problem is the barrier effect caused by the rails,” says Dornas. Not all the fatalities are the result of being run over by passing trains. Some of the toads appeared to have died from desiccation due to extreme heat from the tracks. Most horrifyingly, others showed signs of barotrauma, where a sudden change in air pressure from the fast-moving train causes the inner organs to be blown out — the toads literally exploded from the inside. While additional research would help to better understand the problem and its population-level impacts for the toads, Dornas says that providing passage underneath the rails could be a useful solution. A 2018 study of endangered gopher tortoises (*Gopherus polyphemus*), which have been known to cross railways near John F. Kennedy Space Centre in Florida, came to a similar conclusion.

“We predict that nearly all tortoises in the vicinity of railways are susceptible to becoming entrapped or experiencing reduced movement and dispersal,” the researchers wrote. They recommended trenches that can create a safe passage underneath the tracks and an escape route for those that get caught between rails. While smaller in size, these trenches are similar in concept to corridor bridges and tunnels that are commonly used to help animals safely cross roadways. And while crossing structures may be used occasionally for railways — like a “landscape” bridge over railway tracks that was opened in Stockholm, Sweden in 2017 — it’s far less common. The biggest reason is simply financial — the structures take resources to build, and so far more investments have been devoted to reducing wildlife collisions on roads than rails. That could change with more interest in railway ecology and cheaper building options. ARC Solutions, a project of the Center for Large Landscape Conservation, held a recent design competition to rethink the materials and engineering used in wildlife overpasses to make them more sustainable and affordable. If those concepts come to fruition, it could mean an easier lift to develop safer crossing systems for all kinds of wildlife over both roads and rails.

WARNING SYSTEMS

Getting animals over or under tracks safely is helpful when a railway is an obstacle. But, as researchers found in Alberta, Canada, railways can also be a destination. And that requires a new set of solutions. Concern over grizzly bear

(*Ursus arctos*) deaths on the Canadian Pacific Railway between Banff and Yoho national parks has prompted years of study. “The causes for attraction of bears to the rail are really surprisingly complex and variable among individuals,” says Colleen Cassady St. Clair, a professor of biological sciences at the University of Alberta, who has been leading a team to determine why grizzlies end up on railways and how to prevent their deaths. “There just isn’t a single simple solution. “The biggest reason is that railways are a good place for a bear to find food. For one thing, the trains can spill grain from their cargo cars, leaving behind a steady supply of free food. The very existence of the railways also opens up avenues for grizzly dining. The carcasses of deer and other ungulates struck by trains are an attractant for bears. Railways are slightly warmer than adjacent forests, which attracts ants, another grizzly food, researchers have found. And palatable vegetation also grows along the tracks, providing grizzlies with a wide range of edible choices. This “edge habitat,” according to a 2017 study co-authored by St. Clair, has “higher species richness, diversity and cover for seven of the eight most commonly-occurring species that are consumed by grizzly bears.” Buffaloberry, a local fruit that’s an important source of nutrients for bears pre-hibernation, was even found to have more fruit, ripen earlier and have higher sugar content within 15 meters the railway lines compared to the nearby interior forest. More food along railways means the possibility of more train strikes on bears. So what to do about it? One tactic would be to limit the growth of vegetation that attracts bears and other wildlife, or, as Canadian Pacific Railways has done, remove vegetation from along the tracks that could obscure sight and sounds lines to make approaching trains easier to see and hear at certain problem locations. But St. Clair favours a different approach, developed by one of her graduate researchers, Jonathan Backs. He invented a warning system using a vibration sensor on the track that, farther down, triggers a ringing bell sound and flashing light 30 seconds before a train passes by a hotspot that has been designated for mitigation. Acoustic warning systems have been developed elsewhere. Poland is testing a system that deters wildlife from approaching trains by loudly broadcasting recordings of barking dogs and alarm calls of other animals. Japan is testing a similar system. St. Clair says that these acoustic warning systems are promising, but it’s too early to determine how effective they’ll ultimately be. One drawback could be that if predator sounds are used as a deterrent and no predator ultimately appears, animals will stop associating those warning sounds with a risk. To avoid that potential pitfall, the warning system

that Backs is developing in Alberta has a key difference: It's not meant to scare the animals, but to teach them.

"Our idea with this approach was to help the animals learn that these warning signals, which are not scary in themselves, are reliably associated with the train coming, which most animals seem to find scary," says Backs. "And then the animals would learn to get out of the way when they receive these warning signals rather than waiting for the train to arrive."

Backs says he's still analysing the data he has collected from trials of the warning system, but preliminary results are encouraging — and not just for bears. Other large animals appear to leave the tracks around six seconds earlier when the system is used.

And that's another reason St. Clair is excited about the potential of this system. Public interest in grizzlies helped spur the research, but the mitigation can be useful for all kinds of wildlife and all over the world. "The principles potentially apply to all animals," she says.

Backs says there is still more work to do to prove the concept and then find partner organizations to implement it. "The most important thing for me is to get it out there and make it real and put it in the hands of people who are working hard to keep animals safe," he says. "It might end up being only one part of a broader toolbox — different solutions are appropriate in different situations — but it's exciting that this could actually be used to save lives. "And for the field of railway ecology, more research is still needed, too, says St. Clair.

"We need a broader understanding of where mortality is a real problem, for which species and what the circumstances are that generate locations of higher vulnerability," she says. "Some ongoing work in Banff is trying to put together an entire database of animals that have been killed on the rail and determine what environmental and train operational factors seem to contribute to that vulnerability. With that information it will be possible to be more surgical, if you will, in applying the right kind of mitigate.

CHAPTER 2

2. LITERATURE SURVEY

The section is organized into two categories – some systems for animal detection have been reviewed, followed by methods of animal intrusion detection and prevention.

2.1 Animal Detection using Computer Vision

Zhang et al. describe a system for segmentation of animals from images captured through camera traps. The procedure employed uses a multi-level iterative graph cut to generate object region proposals and accurately recognize regions of interest. This is especially useful when the animal blends together with the background and is difficult to identify. These proposals segmented into background and foreground in the second stage. Feature vectors are extracted from each image using Alex Net architecture and combined together with the histogram of oriented gradients (HOG) to generate Fisher vectors. The system obtained an accuracy of 82.1% for animal and species detection.

Yousif combine deep learning classification with dynamic background modelling to evolve a swift and precise method for human and animal detection from highly cluttered camera trap pictures. Background modelling helps generate region proposals for foreground objects, which are then cashiered using the DCNN, resulting in improved efficiency and increased accuracy. The proposed system achieves 82% accuracy in segmenting images into human, animal and background patches.

Kellenberger use Unmanned Aerial Vehicles (UAV) to monitor animals and prevent poaching. A two-branch custom CNN is built using Alex Net as the backbone. The authors report a 60% accuracy over the dataset gathered from UAV images Kuzikus Wildlife Reserve Park, Namibia.

Norouzzadeh use the Snapshot Serengeti dataset and apply deep-neural networks to detect and identify animals in camera trap images. The system consists of multiple parts a) detection stage (whether there is an animal in the image), b) species identification stage, c) information stage, where the network reports additional data such as the count and attributes of the animals (standing, resting etc.) An ensemble

of nine models is used and obtains a top-1 accuracy of 99.4% for the species identification task, and the overall pipeline accuracy was around 93.8%.

Parham propose a multi-stage pipeline for animal detection and recognition. The fundamental steps include animal classification, animal localization and predicting animal characteristics, such as orientation. Animal localization is based on the YOLO object detection model. The proposed system achieves an overall detection accuracy of 76.58% over 6 species.

Matuska propose a novel system for monitoring animals, consisting of a computing unit, for extracting features of animals, and a separate module to track movements. SIFT and SURF are used for feature extraction, and are classified using an SVM classifier. Use of SIFT descriptors achieved inaccuracy of 94% for animal species classification. Sharma describe a system for animal detection that uses cross correlation filters for template matching. The training data is used as a baseline for classification, and new images are matched with images in the database to detect presence of animals. This system obtains an overall accuracy of 86.25%.

CHAPTER 3:

3.SCOPE OF THE PRESENT INVESTIGATION

OBJECTIVE:

In 2015, the Indian Transport Administration, in cooperation with Indian Railways, Enviroplanning AB, and the Indian University of Agricultural Sciences, initiated a project to develop novel approaches and test available systems for alerting and deterring wildlife from railways shortly before trains arrive. The goal is to prevent WTCs while still allowing animals to cross the railway when no trains are approaching.

Earlier studies from Poland (Werka and Wasilewski 2009; Babińska-Werka et al. 2015) have been promising, but further technical and methodological development is needed to develop a cost-effective and robust alternative to fences and crossing structures.

FACTORS CONTRIBUTING TO RAIL ACCIDENTS:

- The National Wildlife Action Plan (2002-2016) prescribes regulations and mitigation measures, such as slowing traffic speed, providing wildlife crossing structures and increasing driver awareness. But such measures have been rarely implemented on the ground.
- Trains operating during night-time don't take extra precautions. The risk of damage increases after sundown since that is when the wild animals are more active and the visibility is poor.
- Conversion of meter gauge to broad gauge along forested areas has allowed trains to run faster than before. Sometimes the prescribed speed limit is also violated.
- Inefficient and inadequate current methods being employed.

THE NEED TO PREVENT SUCH ACCIDENTS:

- These accidents are detrimental to both the animal and the human community and lead to: Injury to, or death of, vehicle occupants. While animals may be the primary victims, vehicle occupants may also suffer unforeseen injuries.

- Harm to endangered species and wildlife population. The number of deaths due to accidents rivals those due to poaching as a key threat to wildlife animals in India.
- Vehicle damage
- Economic losses incurred (repairing vehicles, cleaning up, providing compensation to victims etc.)
- Accidents result in delay in time/schedule of train. The delay caused has a domino effect since schedule of other trains is also affected, resulting in gross inconvenience. For e.g., after the Chapramari Forest train accident in 2013, the track was closed for over 12 hours, delaying multiple passenger and freight trains

We proposed to use sound waves to repel animals from the tracks of an approaching train and hence prevent such fatalities. Wild animals have an audibility range of 1212000 HZ and can hear infra sound. Through experimental means a particular frequency can be determined at which the animal experiences minor irritations. A device emitting the sound of this frequency can be placed in trains and switched ON when the train is passing through areas which abound with wild animals.

This can be done wither manually or automatically using a GPS which will along with the device. The device itself will produce the sound using an embedded C program that we have written. Although the idea the repelling animals using sound waves has been floating around for quite some time. What makes our proposal unique is that we have taken into account various factors which will affect the emitted sound frequencies.

This is one of the best ways and easiest ways to get the real time parameters. The best Ways to eradicate the Train-Vehicle collision system. There is no need of human monitoring for 24/7. By using this proposed system, the collision of wildlife animals is greatly reduced. This needs to be installed only in specified areas. When these ideas are implemented surely the rate of rail-accidents due to collision can be drastically reduced. The input is given as an image by Arduino IDE (python) code and this code is dumped into Arduino board by a connector RS232 where it makes the LCD to glow and display the animal's name and also makes the buzzer to produce noise. The sound is produced on the buzzer by APR9600. This device offers true single-chip voice recording, non-volatile storage, and playback capability for 40 to 60 seconds.

The device supports both random and sequential access of multiple messages. This information is stored and shared in the cloud by an IoT device (NodeMCU ESP8266). NodeMCU is open source platform; their hardware design is open for edit/modify/build. NodeMCU Dev. Kit/board consists of ESP8266 Wi-Fi enabled chip. GPS is used here to track the location. This POT (Patch Antenna On Top) GPS receiver providing a solution that high position and speed accuracy performances as well as high sensitivity and tracking capabilities in urban conditions & provides standard NMEA0183 strings in “raw” mode for any microcontroller.

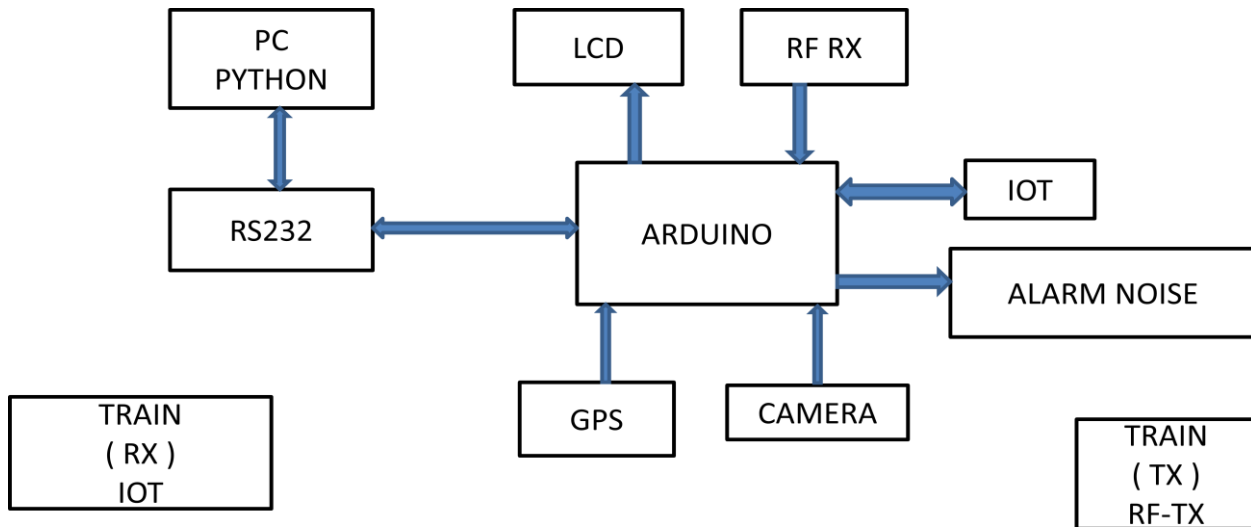
Notification is sent to the driver through blynk app so that he could slow down the train.

This project can also be done manually by dumping an embedded C code on to the transmitter (**encoder HT12E**) where this process happens again in the receiver end (**decoder HT12D**). Encoder IC (HT12E) receives parallel data in the form of address bits and control bits. The control signals from remote switches along with 8 address bits constitute a set of 12 parallel signals. The encoder HT12E encodes these parallel signals into serial bits. Transmission is enabled by providing ground to pin14 which is active low. The control signals are given at pins 10-13 of HT12E. The serial data is fed to the RF transmitter through pin17 of HT12E. Transmitter, upon receiving serial data from encoder IC (HT12E), transmits it wirelessly to the RF receiver. We have also connected a camera for live recording.

ADVANTAGES OF THE PROPOSED SYSTEM:

- Pre information of Train arrival.
- Only placed in the areas where needed.
- Accident avoidance.
- Safety and quality of service.
- Cost effective.
- Draw less power.
- Works both manually and automatically.

BLOCK DIAGRAM:



Working:

The input is given as an image by Arduino IDE (python) code and this code is dumped into Arduino board by a connector RS232 where it makes the LCD to glow and display the animal's name and also makes the buzzer to produce noise. The sound is produced on the buzzer by APR9600. This device offers true single-chip voice recording, non-volatile storage, and playback capability for 40 to 60 seconds.

The device supports both random and sequential access of multiple messages. This information is stored and shared in the cloud by an IoT device (NodeMCU ESP8266). NodeMCU is open source platform; their hardware design is open for edit/modify/build. NodeMCU Dev. Kit/board consists of ESP8266 Wi-Fi enabled chip. GPS is used here to track the location. This POT (Patch Antenna On Top) GPS receiver providing a solution that high position and speed accuracy performances as well as high sensitivity and tracking capabilities in urban conditions & provides standard NMEA0183 strings in "raw" mode for any microcontroller.

Notification is sent to the driver through blynk app so that he could slow down the train.

This project can also be done manually by dumping an embedded C code on to the transmitter (**encoder HT12E**) where this process happens again in the receiver end (**decoder HT12D**). Encoder IC (HT12E) receives parallel data in the form of address bits and control bits. The control signals from remote switches along with 8 address bits constitute a set of 12 parallel signals. The encoder HT12E encodes these parallel signals into serial bits. Transmission is enabled by providing ground to pin14 which is active low. The control signals are given at pins 10-13 of HT12E. The serial data is fed to the RF transmitter through pin17 of HT12E. Transmitter, upon receiving serial data from encoder IC (HT12E), transmits it wirelessly to the RF receiver. We have also connected a camera for live recording.

CHAPTER 4

4. EXPERIMENTAL DEVICES AND METHODS/ALGORITHMS USED

Software Used:

Operating System	:	Windows 7 / 8/ 10
Language	:	Python
IDE	:	Spyder

Hardware Used:

- ARDUINO
- LCD
- ALARM
- GPS
- RF RX
- PC

Overview of the YOLO Object Detection Algorithm:

YOLO (You Only Look Once) real-time object detection algorithm, which is one of the most effective object detection algorithms that also encompasses many of the most innovative ideas coming out of the computer vision research community. Object detection is a critical capability of autonomous vehicle technology.

It's an area of computer vision that's exploding and working so much better than just a few years ago. At the end of this article, we'll see a couple of recent updates to YOLO by the original researchers of this important technique.

SOFTWARE DESCRIPTION:

Python Spyder IDE

It is always necessary to have interactive environments to create software applications and this fact becomes very important when you work in the fields of Data Science, engineering, and scientific research. The Python Spyder IDE has been created for the same purpose.

Spyder is an open-source cross-platform IDE. The Python Spyder IDE is written completely in Python. It is designed by scientists and is exclusively for scientists, data analysts, and engineers. It is also known as the Scientific Python development IDE and has a huge set of remarkable features which are discussed below.

Programming language

OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. There are bindings in Python, Java and MATLAB/OCTAVE. The API for these interfaces can be found in the online documentation Wrappers in other languages such as C#, Perl, Ch., Haskell, and Ruby have been developed to encourage adoption by a wider audience.

Since version 3.4, **OpenCV.js** is a JavaScript binding for selected subset of OpenCV functions for the web platform.

All of the new developments and algorithms in OpenCV are now developed in the C++ interface.

HARDWARE DESCRIPTION

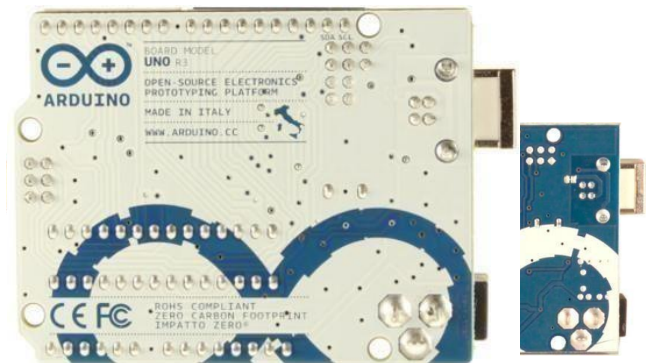
Arduino Uno

Arduino Uno R3 Front



Uno R3 Back

Arduino



Overview of the Arduino

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

Revision 3 of the board has the following new features:

- pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.

- At mega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

Specifications

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage(recommended)	7-12 V
Input Voltage(limits)	6-20 V
Digital I/O pins	14(of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50mA
Flash Memory	32 KB(ATmega328) of which 0.5 KB used by boot loader
SRAM	2 KB(ATmega328)
EEPROM	1 KB(AT mega328)
Clock Speed	16 MHz

Power:

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall wart) or battery. The adapter can be connected by plugging a 2.1mm Centre positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode(), digital Write(), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 KOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the `analogWrite()` function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the SPI library.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e., 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical.

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes preprogrammed with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C headerfiles).

You can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available.

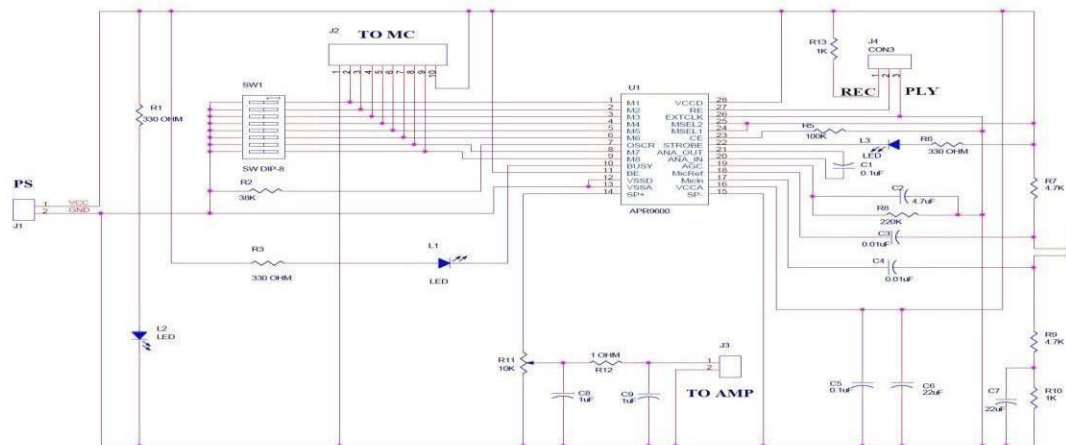
The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user contributed tutorial for more information.

APR9600 Re-Recording Voice IC

Schematic diagram



Schematic Explanation:

The APR9600 device offers true single-chip voice recording, non-volatile storage, and playback capability for 40 to 60 seconds. The device supports both random and sequential access of multiple messages. Sample rates are user selectable, allowing designers to customize their design for unique quality and storage time needs. Integrated output amplifier, microphone amplifier, and AGC circuits greatly simplify system design. the device is ideal for use in portable voice recorders, toys, and many other consumer and industrial applications. APLUS integrated achieves these high levels of storage capability by using its proprietary analog/multilevel storage technology implemented in an advanced Flash non-volatile memory process, where each memory cell can store 256 voltage levels. This technology enables the APR9600 device to reproduce voice signals in their natural form. It eliminates the need for encoding and compression, which often introduce distortion.



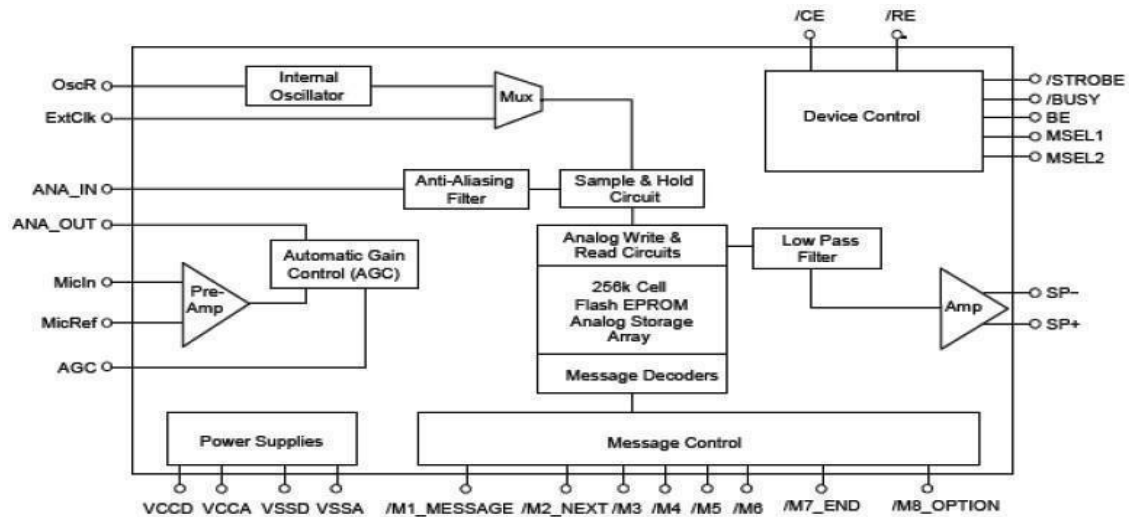
Functional Descriptions:

APR9600 block diagram is included in order to describe the device's internal architecture. At the left-hand side of the diagram are the analog inputs. A differential microphone amplifier, including integrated AGC, is included on-chip for applications requiring use. The amplified microphone signals fed into the device by connecting the ANA_OUT pin to the ANA_IN pin through an external DC blocking capacitor. Recording can be fed directly into the ANA_IN pin through a DC blocking capacitor, however, the connection between ANA_IN and ANA_OUT is still required for playback. The next block encountered by the input signal is the internal anti-aliasing filter. The filter automatically adjusts its response according to the sampling frequency selected so

Shannon's Sampling Theorem is satisfied. After anti-aliasing filtering is accomplished, the signal is ready to be clocked into the memory array. This storage is accomplished through a combination of the

Sample and hold circuit and the analog write/read circuit. These circuits are clocked by either the Internal Oscillator or an external clock source. When playback is desired the previously stored recording is retrieved from memory, low pass filtered, and amplified as shown on the right-hand side of the diagram. The signal can be heard by connecting a speaker to the SP+ and SP- pins. Chip-wide management is accomplished through the device control block shown in the upper right-hand corner.

Message management is provided through the message control block represented in the lower centre of the block diagram. More detail on actual device application can be found in the Sample Application section. More detail on sampling control can be found in the Sample Rate and Voice Quality section. More detail on Message management and device control can be found in the Message Management section.



Block Diagram of APR9600

Sampling Rate & Voice Quality:

According to Shannon's sampling theorem, the highest possible frequency component introduced to the input of a sampling system must be equal to or less than half the sampling frequency if aliasing errors are to be eliminated. The APR9600 automatically filters its input, based on the selected sampling frequency, to meet this requirement. Higher sampling rates increase the bandwidth and hence the voice quality, but they also use more memory cells for the same length of recording time. Lower sampling rates use fewer memory cells and effectively increase the duration capabilities of the device, but they also reduce incoming signal bandwidth. The APR9600 accommodates sampling rates as high as 8 kHz and as low as 4 kHz. You can control the quality/duration trade off by controlling the sampling frequency.

An internal oscillator provides the APR9600 sampling clock. Oscillator frequency can be Changed by changing the resistance from the OSCR pin to GND. Table 2 summarizes resistance values and the corresponding sampling frequencies, as well as the resulting input bandwidth and duration.

Liquid Crystal Display:

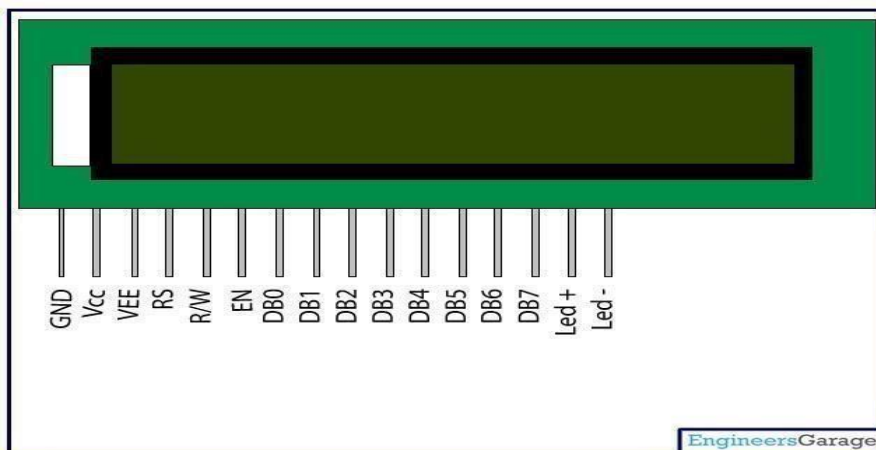
LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over segments and other multi segment LEDs. The reasons being: LCDs are economical;

easily programmable; have no limitation of displaying special & even custom characters(unlike in seven segments), animations and so on.

A **16x2 LCD** means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

Pin Diagram:



Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	V _{cc}
3	Contrast adjustment; through a variable resistor	V _{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable

7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V _{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

NODEMCU:



NodeMCU is an open source [LUA](#) based firmware developed for ESP8266 Wi-Fi chip. By exploring functionality with ESP8266 chip, NodeMCU firmware comes with ESP8266 Development board/kit i.e., NodeMCU Development board.

NodeMCU Development Board/kit v0.9 (Version1)

Since NodeMCU is open-source platform, their hardware design is open for edit/modify/build. NodeMCU Dev Kit/board consist of ESP8266 Wi-Fi enabled chip. The **ESP8266** is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information, about ESP8266, you can refer Module. There is Version2 (V2) available for NodeMCU Dev Kit

How to start with NodeMCU?

NodeMCU Development board is featured with Wi-Fi capability, analog pin, digital pins and serial communication protocols. To get start with using NodeMCU for IoT applications first we need to know about how to write/download NodeMCU firmware in NodeMCU Development Boards. And before that where this NodeMCU firmware will get as per our requirement. There is online NodeMCU custom builds available using which we can easily get our custom NodeMCU firmware as per our requirement.

How to write codes for NodeMCU?

After setting up ESP8266 with Node-MCU firmware, let's see the IDE (Integrated Development Environment) required for development of NodeMCU.

NodeMCU with Arduino IDE

Here is another way of developing NodeMCU with a well-known IDE i.e., Arduino IDE. We can also develop applications on NodeMCU using Arduino development environment. This makes easy for Arduino developers than learning new language and IDE for NodeMCU.

NODEMCU DEVELOPMENT KIT/BOARD:

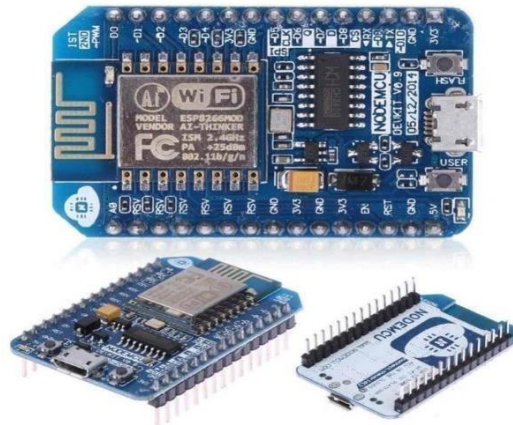
NodeMCU Development Kit/Board consist of ESP8266 Wi-Fi chip.

ESP8266 chip has GPIO pins, serial communication protocol, etc. features on it.

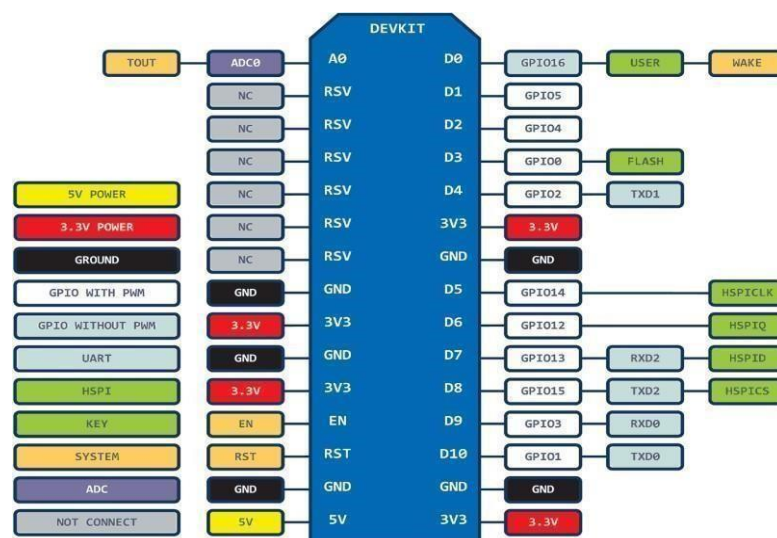
ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. For more information about ESP8266, you can refer ESP8266 Wi-Fi Module.

The features of ESP8266 are extracted on NodeMCU Development board. NodeMCU (LUA based firmware) with Development board/kit that consist of ESP8266 (Wi-Fi enabled chip) chip combines NodeMCU Development board which make it stand-alone device in IoT applications.

Let's see 1st version of NodeMCU Dev. Kit and its pinout as shown in below image.



NodeMCU Development Board v0.9 (Version1)



a. NodeMCU Dev Kit v0.9(V1) Pinouts

2nd version of NodeMCU Dev Kit and its Pinout as shown in below images.

GPIO (General Purpose Input Output) Pins:

NodeMCU has general purpose input output pins on its board as shown in above pinout diagram. We can make it digital high/low and control things like LED or switch on it. Also, we can generate PWM signal on these GPIO pins.

ADC (Analog to Digital Converter) channel (A0):

NodeMCU has one ADC channel/pin on its board.

SPI (Serial Peripheral Interface) Pins:

NodeMCU based ESP8266 has Hardware SPI (HSPI) with four pins available for SPI communication. It also has SPI pins for Quad-SPI communication. With this SPI interface, we can connect any SPI enabled device with NodeMCU and make communication possible with it.

I2C (Inter-Integrated Circuit) Pins:

NodeMCU has I2C functionality support on ESP8266 GPIO pins. Due to internal functionality on ESP-12E we cannot use all its GPIOs for I2C functionality. So, do tests before using any GPIO for I2C applications.

UART (Universal Asynchronous Receiver Transmitter) Pins:

NodeMCU based ESP8266 has two UART interfaces, UART0 and UART1. Since UART0 (RXD0 & TXD0) is used to upload firmware/codes to board, we can't use them in applications while uploading firmware/codes.

GPS MODULE WITH PATCH ANTENNA:



This is a third generation POT (Patch Antenna on Top) GPS module. This POT GPS receiver providing a solution that high position and speed accuracy performances as well as high sensitivity and tracking capabilities in urban conditions and provides standard NMEA0183 strings in “raw” mode for any microcontroller. The module provides current time, date, latitude, longitude, speed, altitude and travel direction / heading among other data, and can be used in a host of applications, including navigation, tracking systems, fleet management, mapping and robotics.

This is a standalone GPS Module and requires no external components except power supply decoupling capacitors. It is built with internal RTC Back up battery. It can be directly connected to Microcontroller's USART. The module is having option for connecting external active antenna if necessary.

The GPS chipsets inside the module are designed by MediaTek Inc., which is the world's leading digital media solution provider and largest fab-less IC Company in Taiwan. The module can support up to 51 channels. The GPS solution enables small form factor devices. They deliver major advancements in GPS performances, accuracy, integration, computing power and flexibility. They are designed to simplify the embedded system integration process.

SPECIFICATIONS:

- Supply: 3.3V, 45mA
- Chipset: MTK MT3318
- Antenna: High gain GPS patch antenna from Cirocomm
- Data output: CMOS UART interface at 3.3V
- Protocol: NMEA-0183@9600bps (Default) at update rate of 1 second
- Protocol message support: GGA, GSA, RMC, VTG
- No. of Satellite simultaneously tracked: 51
- Tracking Sensitivity: On-module antenna: -157 dBm
- Position Accuracy: <3 m
- Max. Update Rate?5Hz (Default: 1 Hz)
- Time to First Fix (Open sky and stationary position)
- Obscuration recovery: 0.1 second average
- Hot start: <1 second's average

- Warm start: <34 second's average
- Cold start: <36 second's average
- Size: 26mm x 26mmx11.7mm

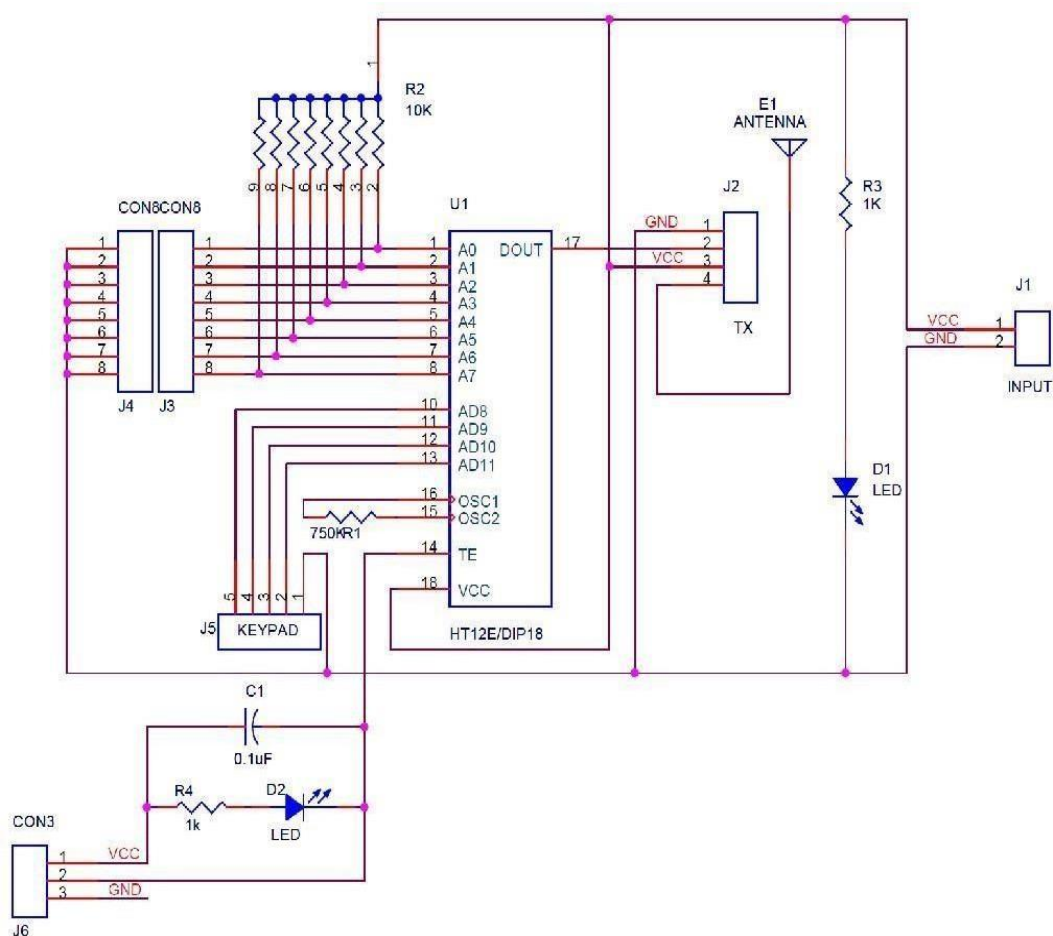
ELECTRICAL SPECIFICATIONS .

ELECTRICAL SPECIFICATIONS						
ITEM	SYMBOL	CONDITION	STANDARDVALUE			UNIT
			MIN.	TYP.	MAX.	
Input Voltage	VDD	VDD = + 5V	4.7	5.0	5.3	V
		VDD = + 3V	2.7	3.0	5.3	V
Supply Current	IDD	VDD = 5V	—	1.2	3.0	mA
Recommended LC Driving Voltage for Normal Temp. Version Module	VDD - V0	-20°C	—	—	—	V
		0°C	4.2	4.8	5.1	
		25°C	3.8	4.2	4.6	
		50°C	3.6	4.0	4.4	
		70°C	—	—	—	

LED Forward Voltage	VF	25° C		—	4 . 2	4.6	V
LED Forward Current	IF	25° C	Array	—	130	26 0	m A
			Edge	—	2 0	40	
EL Power Supply Current	IEL	Val = 110VAC:400Hz		—	—	5.0	m A

ENCODER WITH RF TRANSMITTER

Schematic diagram: -



Schematic Explanation:

This radio frequency (RF) transmission system employs Amplitude Shift Keying (ASK) with transmitter/receiver (Tx/Rx) pair operating at 434 MHz the transmitter module takes serial input and transmits these signals through RF. The transmitted signals are received by the receiver module placed away from the source of transmission. The system allows one-way communication between two nodes, namely, transmission and reception. The RF module has been used in conjunction with a set of four channel encoder/decoder ICs. Here HT12E & HT12D have been used as encoder and decoder respectively.

The encoder converts the parallel inputs (from the remote switches) into serial set of signals. These signals are serially transferred through RF to the reception point. The decoder is used after the RF receiver to decode the serial format and retrieve the original signals as outputs. These outputs can be observed on corresponding LEDs.

Encoder IC (HT12E) receives parallel data in the form of address bits and control bits. The control signals from remote switches along with 8 address bits constitute a set of 12 parallel signals. The encoder HT12E encodes these parallel signals into serial bits.

Transmission is enabled by providing ground to pin14 which is active low. The control signals are given at pins 10-13 of HT12E. The serial data is fed to the RF transmitter through pin17 of HT12E. Transmitter, upon receiving serial data from encoder IC (HT12E), transmits it wirelessly to the RF receiver.

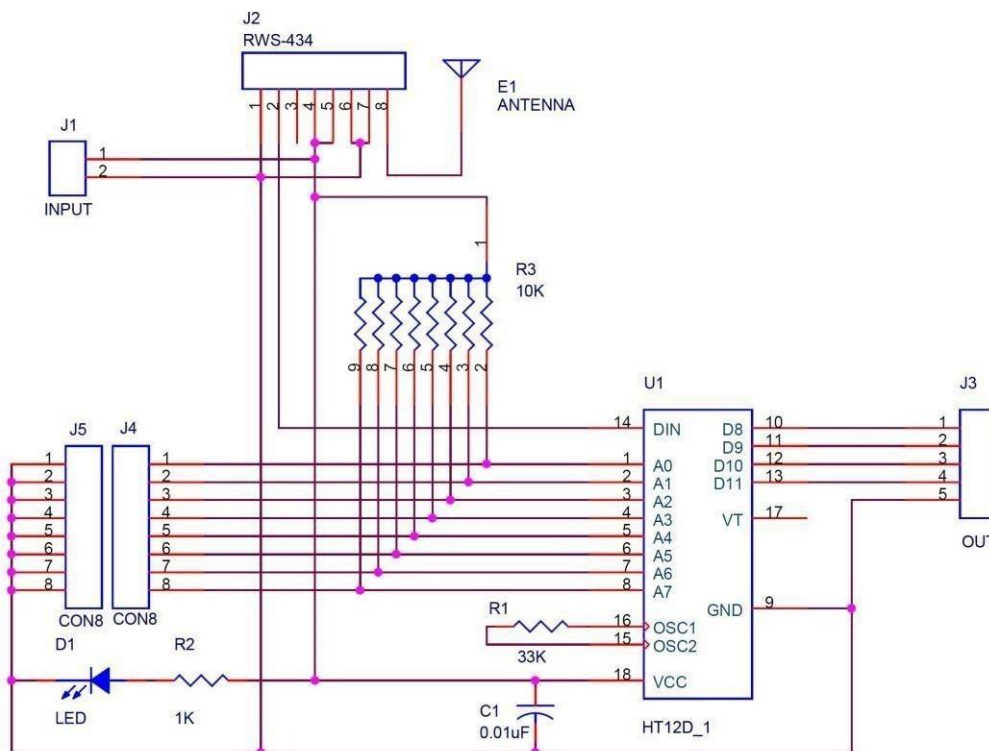
HT12E ENCODER

HT12E is an encoder integrated circuit of 212 series of encoders. They are paired with 212 series of decoders for use in remote control system applications. It is mainly used in interfacing RF and infrared circuits. The chosen pair of encoder/decoder should have same number of addresses and data format. Simply put, HT12E converts the parallel inputs into serial output. It encodes the 12-bit parallel data into serial for transmission through an RF transmitter. These 12 bits are divided into 8 address bits and 4 data bits.

HT12E has a transmission enable pin which is active low. When a trigger signal is received on TE pin, the programmed addresses/data are transmitted together with the header bits via an RF or an infrared transmission medium. HT12E begins a 4-word transmission cycle upon receipt of a transmission enable. This cycle is repeated as long as TE is kept low. As soon as TE returns to high, the encoder output completes its final cycle and then stops.

RF RECEIVER WITH DECODER:

Schematic Diagram:



Features:

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Capable of decoding 12 bits of information
- series of encoders

- Binary address setting
- Received codes are checked 3 times
- Address/Data number combination
- HT12D: 8 address bits and 4 data bits
- HT12F: 12 address bits only
- Built-in oscillator needs only 5% resistor
- Valid transmission indicator
- Easy interface with an RF or an infrared transmission medium
- Minimal external components

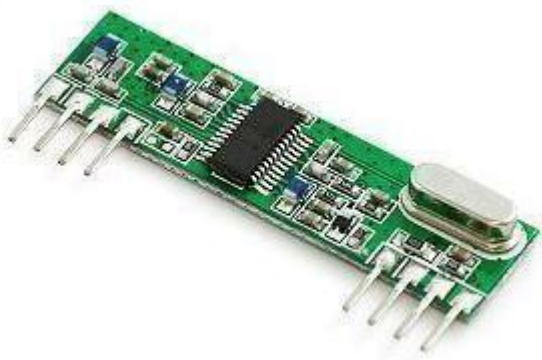
Decoder:

This radio frequency (RF) transmission system employs Amplitude Shift Keying (ASK) with transmitter/receiver (Tx/Rx) pair operating at 434 MHz

The transmitter module takes serial input and transmits these signals through RF. The transmitted signals are received by the receiver module placed away from the source of transmission. The system allows one-way communication between two nodes, namely, transmission and reception. The RF module has been used in conjunction with a set of four channel encoder/decoder ICs. Here HT12E & HT12D have been used as encoder and decoder respectively. The encoder converts the parallel inputs (from the remote switches) into serial set of signals. These signals are serially transferred through RF to the reception point. The decoder is used after the RF receiver to decode the serial format and retrieve the original signals as outputs. These outputs can be observed on corresponding LEDs.

The 212 decoders are a series of CMOS LSIs for remote control system applications. They are paired with Holtek's 212 series of encoders (refer to the encoder/decoder cross reference table). For proper operation, a pair of encoder/decoder with the same number of addresses and data format should be chosen. The decoders receive serial addresses and data from a programmed 212 series of encoders that are transmitted by a carrier using an RF or an IR transmission medium. They compare the serial input data three times continuously with their local addresses. If no error or unmatched codes are found, the input data codes are decoded and then transferred to the output pins. The VT pin also goes high to indicate a valid transmission.

The 212 series of decoders are capable of decoding information's that consist of N bits of address and 12_N bits of data. Of this series, the HT12D is arranged to provide 8 address bits and 4 data bits, and HT12F is used to decode 12 bits of address information.



Controlling Arduino board with smartphone and Blynk via internet:

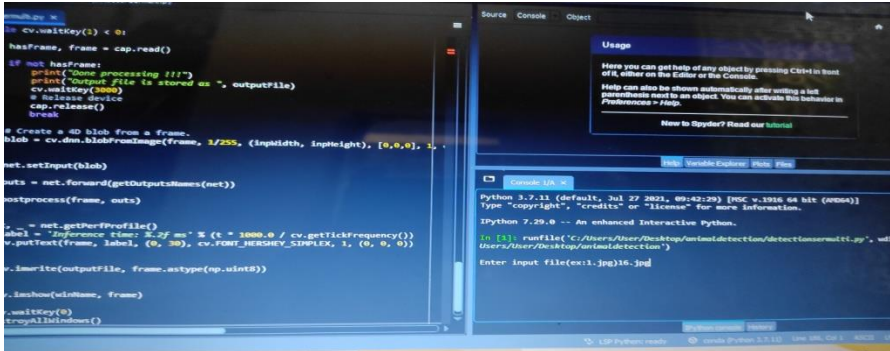
This is my first work concerning the use of Blynk application to control Arduino board. Blynk allow us to create applications and then use it to control Arduino board connected to a PC with internet access, from anywhere in the world, (for instance, controlled, servos, receive data, etc.), with a smartphone. It is one of the most interesting actions! The connection can also be established by Bluetooth between smartphone and Arduino board, but this will be not presented in this work.

Blynk can be downloaded from Google play store ([for Android](#)). & App store ([for apple](#)), providing us the dashboard as well as the connectivity to Arduino, (it is a virtual connectivity). Programming Blynk is very simple, push and drag widgets form the tools bar and allocating them pins on Arduino board. For such project can be used an ordinary Arduino board, without internet shield, connected to a PC with internet access and a smartphone. The role of PC is to ensure the connectivity of the Arduino board to the internet and to upload the Arduino code. For this purpose, it is necessarily to install Blynk library on the PC and made some settings. The Arduino codes used for Blynk have the same structure as ordinary codes but contain specific parts to communicate with Android devices. I will present a simple's example, taken from internet, and partially modified by me. You can see that creating new codes is almost similar with the coding of the ordinary Arduino.

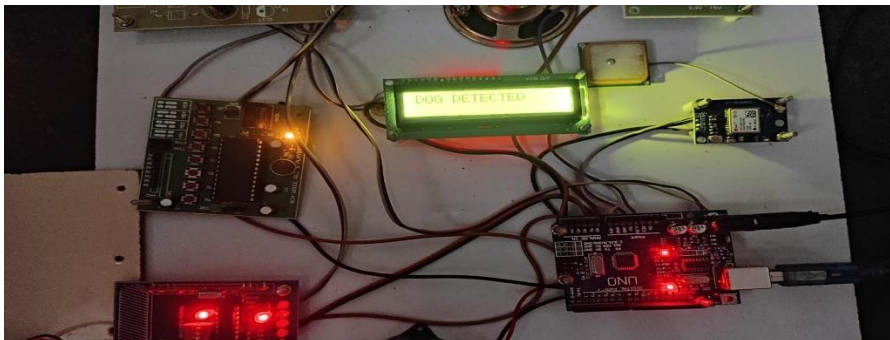
CHAPTER 5

5. RESULT

RESULT FOR DOG DEDUCTION :



Input is given to the python code, file name (16.jpg).

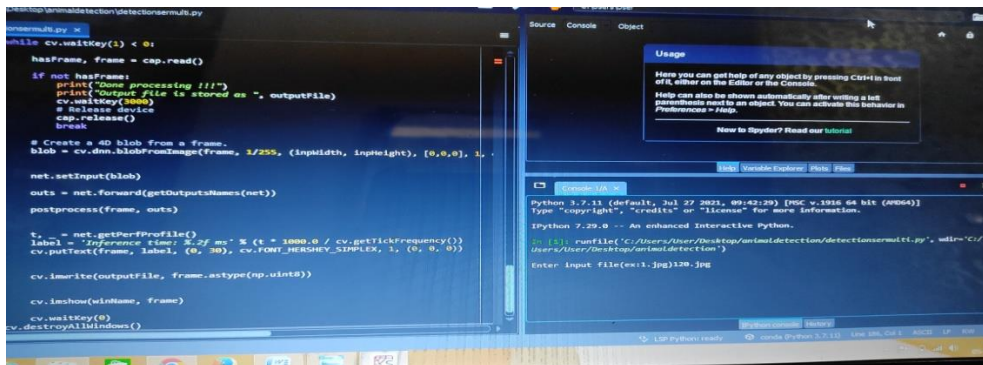


Animal name is displayed on LCD and device emitting the noise.



Output was showing in the monitor.

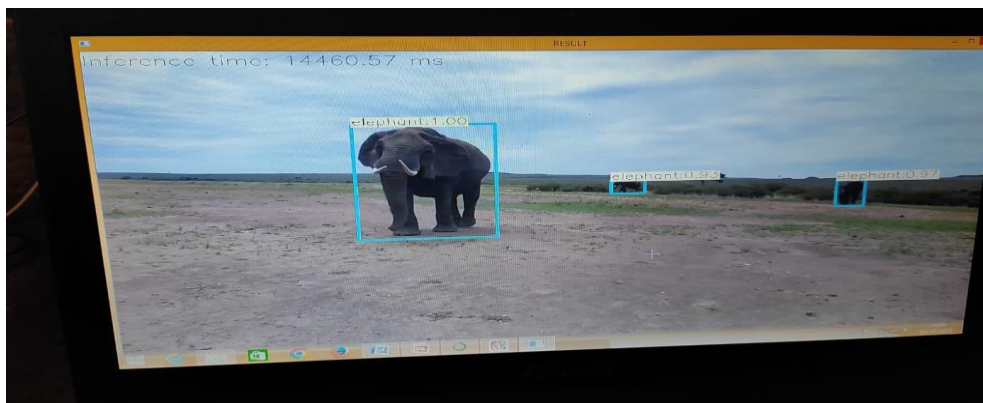
RESULT FOR ELEPHANT DEDUCTION :



Input is given to the python code. File name (120.jpg).

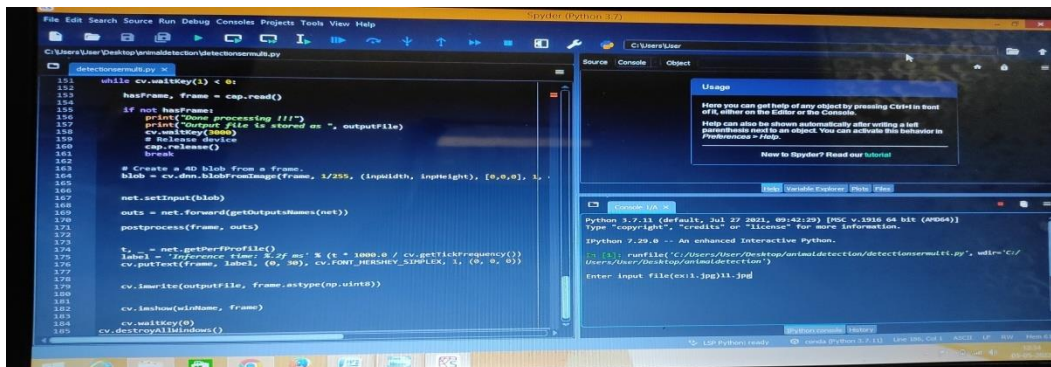


Animal name is displayed on LCD and device emitting the noise.

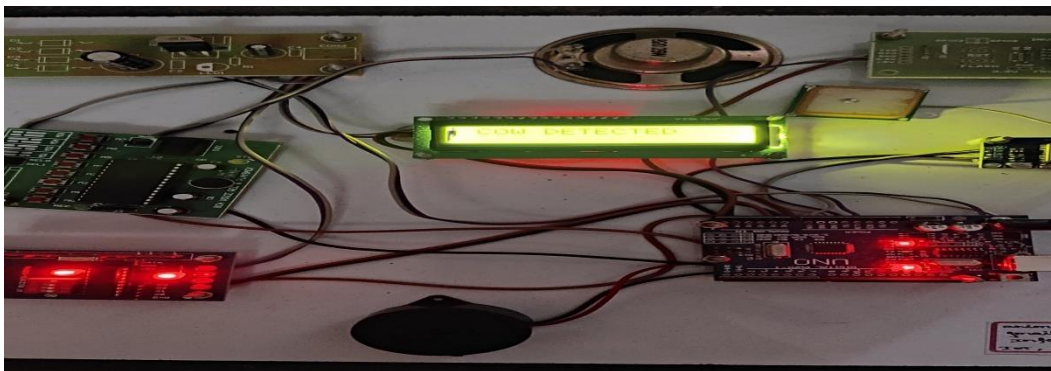


Output was showing in the monitor.

RESULT FOR COW DEDUCTION :



Input is given to the python code. File name (11.jpg)

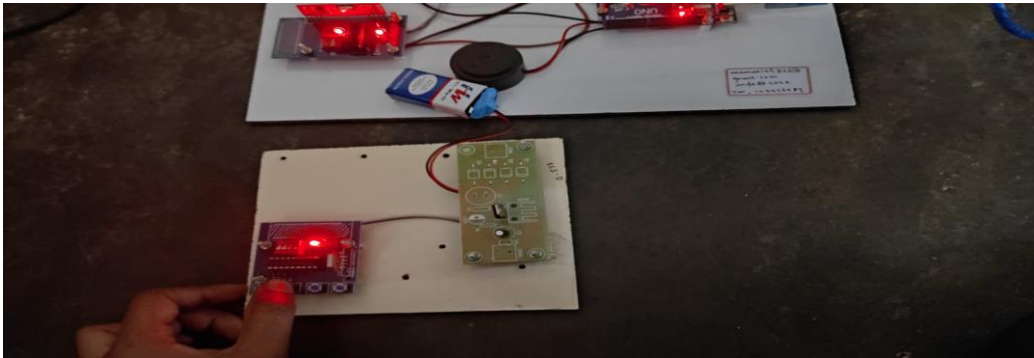


Animal name is displayed on LCD and device emitting the noise.

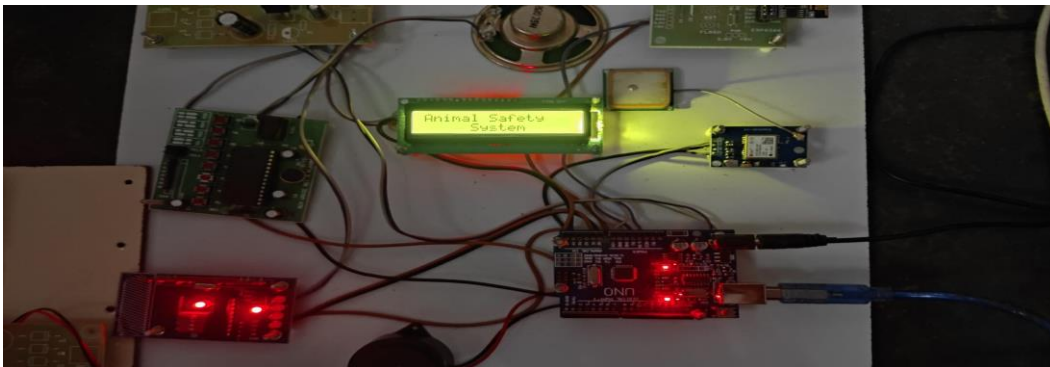


Output was showing in the monitor.

RESULT FOR MANUAL METHOD :



Input signal is given through the transmitter by manually.



Receiver receives the signal and LCD displays the animals name. APR emits the noise automatically.

BLINK APPLICATION :



When animal is deducted in the monitor the result will be send to the train driver through the blink app.

CHAPTER 6

6. Conclusion

The proposed system attempts to reduce human-animal conflicts by continuous and automatic monitoring of vulnerable areas using computer vision to detect animal intrusions. The intrusion detection pipeline consists of three stages – animal detection, animal tracking and user alerts and notifications. The proposed system is cost-effective and highly efficient, with an average accuracy of 98.8% in detecting and identifying animals in images. Although the prototype described in this paper is trained to recognize five different species of animals, it is easily extendable to detect and track other types of animals with sufficient training data. The choice of species can also be region specific, thereby providing a unique edge over other existing solutions. Such a system if implemented on a large scale, has potential to largely reduce casualties due to animal intrusions.

References:

1. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: A. Leonardis, H. Bischof, A. Pinz (eds.) *Computer Vision – ECCV 2006*, pp. 404–417. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
2. Cortes, C., Vapnik, V.: Support-vector networks. *Machine Learning* 20(3), 273–297(1995)
3. G, P., Bagal, M.U.: A smart farmland using raspberry pi crop vandalization prevention & intrusion detection system. *International Journal of Advance Research and Innovative ideas in Education* 1(S), 62–68 (2016)
4. Kellenberger, B., Volpi, M., Tuia, D.: Fast animal detection in uav images using convolutional neural networks. In: *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 866–869 (2017)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (eds.) *Advances in Neural Information Processing Systems* 25, pp. 1097–1105. Curran Associates, Inc. (2012)
6. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. *Lecture Notes in Computer Science* pp. 1–11. Springer, Cham (2016)
7. Lowe, D.G.: Distinctive image features from scale-invariant key points. *International Journal of Computer Vision* 60(2), 91–110 (2004)
8. Lukešič, A., Vojtíšek, T., Čechovin, L., Matas, J., Kristan, M.: Discriminative correlation filter tracker with channel and spatial reliability. *International Journal of Computer Vision* 126(7), 671–688 (2018)
9. Matuska, S., Hudec, R., Benco, M., Kamencay, P., Zachariasova, M.: A novel system for automatic detection and classification of animal. In: *2014 ELEKTRO*, pp. 76–80 (2014)
10. Norouzzadeh, M.S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M.S., Packer, C., Clune, J.: Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences* 115(25), E5716–E5725 (2018)

10. Parham, J., Stewart, C., Crall, J., Rubenstein, D., Holmberg, J., Berger-Wolf, T.: An animal detection pipeline for identification. An animal detection pipeline for identification. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1075–1083 (2018)
11. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016)
12. Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517– 6525 (2017)
13. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference of Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
14. Sharma, S., Shah, D., Bhavsar, R., Jaiswal, B., Bamniya, K.: Automated detection of animals in context to Indian scenario. In: 2014 5th International Conference on Intelligent Systems, Modeling and Simulation, pp. 334–338 (2014)
15. Suganthi, N., Rajathi, N., M, F.I.: Elephant intrusion detection and repulsive system. International Journal of Recent Technology and Engineering 7(4S), 307– 310(2018)
16. Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A., Packer, C.: Snapshot Serengeti, high frequency annotated camera trap images of 40 mammalian species in an African savanna. Scientific data 2, 150,026 (2015)
17. Yousif, H., Yuan, J., Kays, R., He, Z.: Fast human-animal detection from highly cluttered camera-trap images using joint background modeling and deep learning classification. In: 2017 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–4 (2017)
18. Zhang, Z., He, Z., Cao, G., Cao, W.: Animal detection from highly cluttered natural scenes using spatiotemporal object region proposals and patch verification. IEEE Transactions on Multimedia 18(10), 2079–2092 (2016)