

SMART IRRIGATION USING BLYNK APP AND WITESP NODE MCU & ANIMAL DETECTION

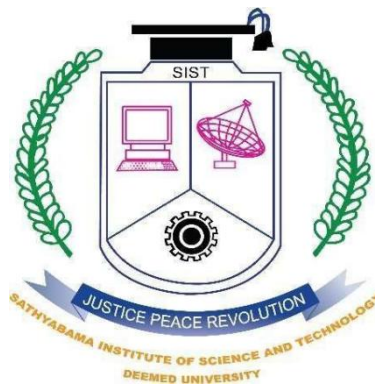
A Project Report Phase - II

Submitted in partial fulfillment of the requirements for the award of a
Bachelor of Engineering degree in Electronics and Communication Engineering

by

GUBBALA UDAY SUBRAMANYA KUMAR (39130149)

GOGULAMUDI PREMCHAND (39130141)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE A UNIVERSITY)**

**Accredited with a grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119**

APRIL 2023



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE A UNIVERSITY)
Accredited with an "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of GUBBALA UDAY SUBRAMANYA KUAMR (39130149) And GOGULAMUDI PREMCHAND (39130141) who carried out the project entitled "SMART IRRIGATION USING BLYNK APP AND WITH ESP NODE MCU & ANIMAL DETECTION" under our supervision from November 2022 to April 2023.

Internal guide

Dr. R. M. JOANY M.E., Ph.D.,

Head of the Department

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

Submitted for viva-voce Examination held on 20/04/2023

Internal Examiner

External Examiner

DECLARATION

We, Gubbala Uday Subramanya Kumar (39130149) and Gogulamudi Premchand (39130141) hereby declare that the Project Report entitled "SMART IRRIGATION USING BLYNK APP AND WITH ESP NODE MCU & ANIMAL DETECTION" done by us under the guidance of Dr. R.M. JOANY, M.E., Ph.D., Associate Professor Department of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

Date: 20/04/2023

Place: CHENNAI

1. *Uday Subramanya Kumar*
2. *Premchand*

Signature of the candidates

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this Project and for completing it successfully. We are grateful to them.

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

We would like to express our sincere and deep sense of gratitude to our guide **Dr.R.M.JOANY, M.E., Ph.D.**, Associate Professor for her valuable guidance, suggestions, and constant encouragement paved way for the successful completion of our Project.

We wish to express our thanks to all the Teaching and Non- teaching staff members of the Department of Electronics and Communication Engineering who are helpful in many ways in the completion of the Project.

ABSTRACT

India is mainly an agriculture-based country. Agricultural farm security is important to protect agricultural produce. Valuable investments can be ruined intentionally or unintentionally by a person who intends to rob or destroy the property or by aals. Introduction of modern technology into agriculture has enabled a view to think about building security systems for the farm lands. IoT technology helps in building various applications for Smart Agriculture. When Vision is embedded to IoT, the possibilities are endless. The implementation of IoT agriculture starts with intelligent irrigation for the majority of fields. Optimizing the water schedule and quantity of water helps us to save water, money, and have the best crop on the field. Sensor-based IoT technology gathers soil moisture, temperature data transmits this information to farm irrigation systems from sensors. A platform responds to these signals and the drip irrigation switches on as soon as there is insufficient water in the soil. This project is designed to overcome the problem of irrigation by reducing the usage of water while watering the plants. Animal attack in the farm area is considered as the major threat, which will reduce the amount of crop. The main reason for this is the expansion of cultivated land. Machine learning model is designed to detect the animal entering the farm and plays the appropriate sound to shoo an animal away from the farm such that the crops are prevented from damage. Different types of packages and concepts of the convolutional neural network is used to design the model to achieve the desired aim in the project.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	v
	LIST OF FIGURES	vii
1	INTRODUCTION	1
	1.1 Smart Irrigation System	1
	1.2 Irrigation	3
	1.3 Animal Intrusion Detection Agriculture	6
	1.4 Objective	9
	1.5 Problem Statement	10
2	LITERATURE SURVEY	11
3	SYSTEM DESIGN	15
	3.1 Existing System	15
	3.2 Proposed System	15
	3.3 Methodology	18
	3.4 System Implementations	21
4	SYSTEM REQUIREMENTS	28
	4.1 NODE MCU	28
	4.2 Hardware Details	30
	4.3 DHT11 Sensor and its Working	32
	4.4 Applications	34
	4.5 Relay	34
	4.6 Software Tools	38
5	RESULTS AND DISCUSSION	40
	5.1 Results	40
	5.2 Discussion	42

CHAPTER No.	TITLE	PAGE No.
	5.3 Difference Between the Existing System and the Proposed System (When Micro-controller is ESP8266 MCU vs ESP8266 MCU)	43
	5.4 Difference Between the Existing System and the Proposed System (When Micro-controller is ARDUINO MCU vs ESP8266 MCU)	43
6	SUMMARY AND CONCLUSION	45
	6.1 Future Work	46
	6.2 Research Work	47
	6.3 Implementations Issues	48
	REFERENCES	50
	APPENDIX	52
	A. SOURCE CODE	52
	B. SCREENSHOT	60
	C. RESEARCH PAPER	63

LIST OF FIGURES

FIGURE No.	FIGURE NAME	PAGE No.
Fig 1.1	Overview of Automated Irrigation System	5
Fig 3.1	Block Diagram of Proposed System	16
Fig 3.2	Blynk App Screenshot	17
Fig 3.3	Image Processing Flow	19
Fig 3.4	System Architecture	20
Fig 3.5	Convolution Layer	24
Fig 3.6	Pooling	24
Fig 3.7	Flattering	25
Fig 3.8	Fully Connected	25
Fig 4.1	NODE MCU Development Board	29
Fig 4.2	NODE MCU Chip	29
Fig 4.3	Humidity Sensor	30
Fig 4.4	Specification Sensor	32
Fig 4.5	DHT11Sensor	33
Fig 4.6	Relay	34
Fig 4.7	Relay Basic Section	35
Fig 4.8	Relay Basic Section	36
Fig 4.9	Flyback Diode Block Diagram 1	37
Fig 4.10	Flyback Diode Block Diagram 2	37
Fig 5.1	Bird Detection	40
Fig 5.2	Notification in Blynk App	41
Fig 5.3	Detection of 3D Image	41
Fig 5.4	Smart Irrigation Output	42

CHAPTER– 1

INTRODUCTION

1.1 SMART AGRICULTURE SYSTEM

The contemporary world is in a transition stage where problems concerning global issues, such as global warming and alternative energy sources, are combined with new challenges demanding immediate solutions. Society's focus has shifted from economic growth to sustainable development, where environmental, social, and economic aspects are considered together, rather than separately. Policies that promote sustainability in all sectors of the economy (manufacturing, agriculture, and services) are now considered as a part of good governance. Problems such as climate change, population growth, and poverty (especially hunger), occur in a context of a gradual depletion of natural resources and the fear of diminishing coal energy reserves. These are some of the global issues that are thought to require multidisciplinary approaches in order to be addressed successfully. In this Master's project, we focus on agricultural production and cultivation. This overall process has a significant role in fulfilling the basic human need for food. The production, preparation, packaging, distribution, etc. of food also generates a lot of income. This Master's thesis project aims to exploit modern technologies and tools to improve the monitoring and management of crops, to improve the efficiency and sustainability of farming and food production. To this end, we have designed a system for precision agriculture, which relies on a wireless sensor network combined with a service to provide individual farmers with access to data that they find useful. The system utilizes wireless sensor nodes that collect and transmit data about the quality of the water supply, the soil, and other parameters in an agricultural field. While such sensor-based systems have been investigated earlier, one of the key innovations to be explored in this

Master's thesis project is the combination of these sensor systems with a service-driven business model to increase their ease of use and to amplify the gains that can be realized via an integrated system. The goal is to give a farmer a more complete picture of the current and historic crop status in order to foster better informed decision making. It is expected that such decisions will benefit both farming and irrigation by saving time and resources. Factors such as the diversity of conditions which vary depending on location (for example weather, presence of insects, and disease) combined with the inability to predict the future characteristics of the environment during the different seasons over time complicate the decision-making process and require specialized knowledge.

This project is an attempt to bring some of these micro-environmental sources of information into the decision-making process of farmers. Agriculture is considered as the basis of life for the human species as it is the main source of food grains and other raw materials. It plays vital role in the growth of country's economy. It also provides large ample employment opportunities to the people. Growth in agricultural sector is necessary for the development of economic condition of the country. Unfortunately, many farmers still use the traditional methods of farming which results in low yielding of crops and fruits. But wherever automation had been implemented and human beings had been replaced by automatic machinery, the yield has been improved. Hence there is need to implement modern science and technology in the agriculture sector for increasing the yield. Most of the papers signifies the use of wireless sensor network which collects the data from different types of sensors and then send it to main server using wireless protocol. The collected data provides the information about different environmental factors which in turns helps to monitor the system. Monitoring environmental factors is not enough and complete solution to improve the yield of the crops. There are

number of other factors that affect the productivity to great extent. These factors include attack of insects and pests which can be controlled by spraying the crop with proper insecticide and pesticides. Secondly, attack of wild animals and birds when the crop grows up. There is also possibility of thefts when crop is at the stage of harvesting. Even after harvesting, farmers also face problems in storage of harvested crop. So, in order to provide solutions to all such problems, it is necessary to develop integrated system which will take care of all factors affecting the productivity in every stages like; cultivation, harvesting and post harvesting storage. This paper therefore proposes a system which is useful in monitoring the field data as well as controlling the field operations which provides the flexibility. The aim of our project is to minimize this manual intervention by farmers and save their time and resources using automated systems to irrigate land and fertilize crops. Our project consists of a Moisture sensor, pH sensor and water level sensor to monitor the humidity level, pH value and the water level of the land respectively. The measured value is displayed by the LCD display. If the Moisture level is low, it will turn ON the motor for watering. And if the water level increase the particular level, the motor will work in opposite direction. If the pH value is low, both the motor will turn ON and mix the fertilizer with the water and send it to the field.

1.2 IRRIGATION

Irrigation system uses valves to turn irrigation ON and OFF. These valves may be easily automated by using controllers and solenoids. Automating farm or nursery irrigation allows farmers to apply the right amount of water at the right time, regardless of the availability of labor to turn valves on and off. In addition, farmers using automation equipment are able to reduce runoff from over watering saturated soils, avoid irrigating at the wrong time of day, which will improve crop performance by ensuring adequate water and nutrients when needed. Automatic Drip Irrigation is a

valuable tool for accurate soil moisture control in highly specialized greenhouse vegetable production and it is a simple, precise method for irrigation. It also helps in time saving, removal of human error in adjusting available soil moisture levels and to maximize their net profits. Irrigation is the artificial application of water to the soil usually for assisting in growing crops. In crop production it is mainly used in dry areas and in periods of rainfall shortfalls, but also to protect plants against frost. Types of Irrigation

- Surface irrigation
- Localized irrigation
- Drip Irrigation
- Sprinkler irrigation

The conventional irrigation methods like overhead sprinklers, flood type feeding systems usually wet the lower leaves and stem of the plants. The entire soil surface is saturated and often stays wet long after irrigation is completed. Such condition promotes infections by leaf mold fungi. On the contrary the drip or trickle irrigation is a type of modern irrigation technique that slowly applies small amounts of water to part of plant root zone. Water is supplied frequently, often daily to maintain

favorable soil moisture condition and prevent moisture stress in the plant with proper use of water resources. Drip irrigation saves water because only the plant's rootzone receives moisture. Little water is lost to deep percolation if the proper amount is applied. Drip irrigation is popular because it can increase yields and decrease both water requirements and labor. Drip irrigation requires about half of the water needed by sprinkler or surface irrigation. Lower operating pressures and flow rates result in reduced energy costs. A higher degree of water control is attainable. Plants can be supplied with more precise amounts of water. Disease and insect damage is reduced because plant foliage stays dry. Operating cost is usually reduced. Irrigations may continue during the

irrigation process because rows between plants remain dry.

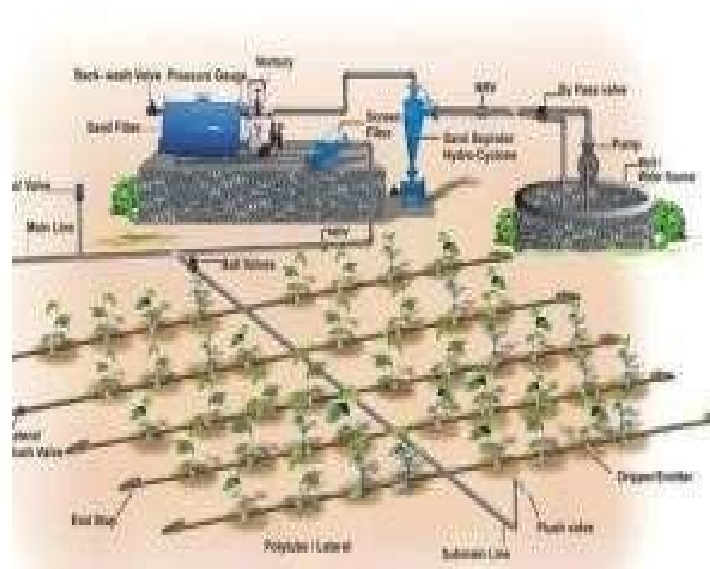


Fig. 1.1 Overview of Automated Irrigation System

The above fig 1 explains about important parameters to be measured for automation of irrigation system are soil moisture. The entire field is first divided into small sections such that each section should contain one moisture sensor. These sensors are buried in the ground at required depth. Once the soil has reached desired moisture level the sensors send a signal to the micro controller to turn on the relays, which control the motor. In proposed system, automated irrigation mechanism which turns the pumping motor ON and OFF on detecting the dampness content of the earth. In the domain of farming, utilization of appropriate means of irrigation is significant. The benefit of employing these techniques is to decrease human interference. This automated irrigation project, the soil sensor senses the moisture content by giving input signal to an Arduino board which operates on ATmega328 micro-controller, is programmed to collect the input signal of changeable dampness circumstances of the earth via dampness detecting system.

1.2.1 Precision Agriculture

The term “precision agriculture” refers to a contemporary approach of applying new technologies utilizing sensors in order to optimize the agricultural cultivation processes. Precision agriculture-based system design principles are increasingly used in research projects and commercial products to provide solutions for crop status monitoring, water supply regulation for irrigation, fertilizer management, pest control, and automated harvesting. Such systems benefit the complete process by reducing cost through automation and saving time. Another significant feature that precision agriculture provides to farmers is the ability to prevent hazardous incidents and to proactively monitor their crops and the local environmental conditions. The effectiveness of precision agriculture is based on the analysis of accurate sets of measurements in soft real-time. Parameters such as the soil condition and humidity are aggregated and analyzed, in order to extract useful information that a farmer can use as a recommendation or guidance; or even to apply fully automated procedures to the crop cultivation process chain.

1.3 ANIMAL INTRUSION DETECTION AGRICULTURE

Nowadays, huge data on wildlife activity and behavior can be obtained over larger space and time domain. Camera-trap methods and many other digital technologies can be used in wildlife monitoring and analysis due to relatively low cost and easy to-use. With the growth in data on wildlife, the study related to wildlife has become more convenient such as studying the effects of climate change on wildlife, alterations in habitat, impact of human intervention on animals and biodiversity over different seasons, areas and species. For monitoring wildlife, sensor cameras are placed on trees in a region creating a stationary camera-trap network. The camera traps are activated; each time motion is sensed, and a short video of

animals activities are recorded with details about the surroundings (illumination levels, humidity, temperature, and location). Camera-trap networks are vital for acquisition of wildlife data without any disturbance. Moreover, camera-trap networks are economically feasible, easy to deploy at larger space and have low maintenance cost; as a result, they are widely used for wildlife monitoring. We can easily obtain data about the visual aspects of animal from the camera-trap images that help to know the behavior and biometric features of species along with the relevant features related to wildlife habitat and surroundings. Recently, a huge set of camera-trap images have been acquired which challenges the capability of manual annotation and image processing. There is a dire need to design multiple tools for automated processing of these huge camera-trap images such as animal identification, segmentation, extraction, and tracking. In this work, we propose a method to detect wildlife animal using CNN based on camera-trap images.

In India 24% of the area is covered by forest. Forestry is the source of major income for over 400 million people in India, most commonly in rural. Population in India is increasing at high rate so providing shelter to all of them is a big challenge. Hence, most of the people live around forest to fulfil their requirements. Since people live closer to forest they often come across wild animals and for safety of human being, often wild animals are killed or human beings are attacked by animals. This paper includes two ways to solve these issues: one where humans should get warning whenever wild animal comes near residential area and other is to keep wild animal away from human habitat. The applications of computer vision include, detecting the moving object which is a major step. Human can easily notice and distinguish objects, but it is difficult for computer to identify and recognize object like humans, that is why developing an object detection and recognition system which is adaptable and not complex is important. The activities of the interested object such as

(humans, animals, vehicles, etc) are monitored using a video camera in a scene for Optical surveillance. Extracting the foreground object is important when grouping, tracking and examining the activities of the concerned object, which is why background subtraction is applied. This paper includes Foreground detection algorithm for detection of animal from a video captured from a still camera. Background subtraction method is used in this paper as it is one of the efficient method for real-time segmentation of moving objects in image alternation of the captured video.

Back propagation algorithm is used to compare the training data set which is taken from Caltec-101 database and the extracted foreground test image. Back Propagation is a popular method of training Artificial Neural Networks (ANN) and used in combination with an optimization procedure such as gradient descent. The procedure of the algorithm works in two stage, propagation and weight update. When an input vector is offered to the network, it is propagated forward through the network, layer by layer, passing through all the layers until it reaches the output layer. The output of the network is then compared with the expected output, using a loss function

Object segmentation and detection from the background based on the motion of object are a necessary step for automated analysis from image sequences. Several studies are based on background and foreground modeling for object detection, however, challenges are involved with complex dynamic scene modeling. The image sequences captured by camera traps consist of natural and dynamic scenes that are challenging to analyze using existing techniques. The natural scenes are usually highly cluttered with swinging trees, waving water, shifting shadows, changing weather, rains, etc. Also, the natural camouflage of animals poses another difficulty for analyzing natural scenes. The prime challenge for wildlife detection is to design models that can handle

complex backgrounds and efficiently detect animals from dynamic scenes. Conventional approaches based on motion are inefficient with dynamic scenes.

Lately, techniques based on deep neural networks are employed for object detection such as Region-based Convolution Neural Networks (RCNN), Fast-RCNN, and Faster-RCNN. Generally, the object detection can be divided into two steps: first; the detection of image regions using region proposal methods that may contain desired object and second; classification step that detects whether the regions contain the desired object or not. Object detection in terms of animal detection deals with problem of accuracy and speed due to the highly dynamic and highly cluttered image sequences obtained from camera traps. The available region proposal approaches create a huge amount of candidate regions. We observe that DCNN is computationally comprehensive and also requires performing region classification multiple times for all candidate regions. Hence, it is important to study the distinct characteristics of camera-trap image sequences in extemporization domain to model an efficient region proposal approach that creates a small number of candidate regions. Hence, we used the camera-trap image sequences that are analyzed using Iterative Embedded Graph Cut (IEGC) technique to create a small group of candidate animal regions. Also for better candidate animal region classification, DCNN features are used with different classifiers such as Support Vector Machine (SVM) and its variants and Ensemble classifiers.

1.4 OBJECTIVE

- The objective of our project is to provide protection from the attacks of animals and thus minimizing the probable loss of the farmer. To detect intrusion around the field.
- The system is capable of turning on/Off automatically and warding off the animals thus protecting the fields from any damage.

- In IoT-based smart farming, a system is built for monitoring the crop field with the help of sensors like soil moisture, humidity sensor.

1.5 PROBLEM STATEMENT

In the case of traditional irrigation system water saving is not considered. Since, the water is irrigated directly in the land, plants under go high stress from variation in soil moisture, therefore plant appearance is reduced. The absence of automatic controlling of the system result in improper water control system. The major reason for these limitations is the growth of population which is increasing at a faster rate. At present there is emerging global water crisis where managing scarcity of water has become a serious job. This growth can be seen in countries which have shortage of water resources and are economically poor. So this is the serious problem in agriculture area. So we want to design an Smart Irrigation System which is based on PROTEUS software programming using Arduino micro controller that operate automatically by sensing the moisture content of the soil and turn ON/OFF the pump using relay without the intervention of farmer and hence save water.

Developing a system to detect an intruder without human intervention which can be installed in farmlands which lacks physical security or manual security.

CHAPTER – 2

LITERATURE SURVEY

2.1 IOT BASED MONITORING SYSTEM IN SMART AGRICULTURE

Author: S. R. Prathibha ; AnupamaHongal ; M. P. Jyothi

Internet of Things (IoT) plays a crucial role in smart agriculture. Smart farming is an emerging concept, because IoT sensors capable of providing information about their agriculture fields. The paper aims making use of evolving technology i.e. IoT and smart agriculture using automation. Monitoring environmental factors is the major factor to improve the yield of the efficient crops. The feature of this paper includes monitoring temperature and humidity in agricultural field through sensors using CC3200 single chip. Camera is interfaced with CC3200 to capture images and send that pictures through MMS to farmers mobile using Wi-Fi.

2.2 IOT BASED SMART SENSOR AGRICULTURE STICK FOR LIVE TEMPERATURE AND MOISTURE MONITORING USING ARDUINO, CLOUD COMPUTING AND SOLAR TECHNOLOGY

Author: AnandNayyarEr. VikramPuri

Internet of Things (IoT) technology has brought revolution to each and every field of common man's life by making everything smart and intelligent. IoT refers to a network of things which make a self-configuring network. The development of Intelligent Smart Farming IoT based devices is day by day turning the face of agriculture production by not only enhancing it but also making it cost-effective and reducing wastage. The aim / objective of this paper is to propose a Novel Smart IoT based

Agriculture Stick assisting farmers in getting Live Data (Temperature, Soil Moisture) for efficient environment monitoring which will enable them to do smart farming and increase their overall yield and quality of products. The agriculture stick being proposed via this paper is integrated with Arduino Technology, Breadboard mixed with various sensors and live data feed can be obtained online from Thingspeak.com. The product being proposed is tested on Live Agriculture Fields giving high accuracy over 98% in data feeds.

2.3 IOT BASED SMART AGRICULTURE MONITORING SYSTEM

Author: Dr N Suma, Sandra Rhea Samson, S. Saranya

Agriculture is the primary occupation in our country for ages. But now due to migration of people from rural to urban there is hindrance in agriculture. So to overcome this problem we go for smart agriculture techniques using IoT. This project includes various features like GPS based remote controlled monitoring, moisture & temperature sensing, intruders scaring, security, leaf wetness and proper irrigation facilities. It makes use of wireless sensor networks for noting the soil properties and environmental factors continuously. Various sensor nodes are deployed at different locations in the farm. Controlling these parameters are through any remote device or internet services and the operations are performed by interfacing sensors, Wi-Fi, camera with microcontroller. This concept is created as a product and given to the farmer's welfare.

2.4 ANIMAL DETECTION USING DEEP LEARNING ALGORITHM

Author: N. Banupriya¹, S. Saranya², rashmi swaminathan³, sanchitha harikumar⁴, sukitha palanisamy

Checking of wild animal in their common environment is crucial. This

proposed work develops an algorithm to detect the animals in wild life. Since there are many different animals manually identifying them can be a difficult task. This algorithm classifies animals based on their images so we can monitor them more efficiently. Animal detection and classification can help to prevent animal-vehicle accidents, trace animals and prevent theft. This can be achieved by applying effective deep learning algorithms.

2.5 SCRUTINY OF METHODS FOR IMAGE DETECTION AND RECOGNITION OF DIFFERENT SPECIES OF ANIMALS

Author: Elham Mohammed Thabit A. ALSADI, Nidhal K. El Abbadi

Animal detection-based study is useful in many real-life applications. Techniques involved in animal detection are useful in observing the locomotive behavior of the engaged animal and in result it prevent harmful interruption of animals in residential areas. There are some branches of research in animal detection. Some of these branches will therefore be discussed in this journal. Humans have developed many algorithms and techniques to gain a better understanding of animal behaviour. Therefore, for early preventive measures, these technologies can also serve as a warning system for humans from encroachment of dangerous wild animals. Such tasks can be reduced to three main branches, namely animal detection, tracking and recognition. Through these papers, new approaches for study and a variety of technologies/algorithms implemented in the past are identified and appropriate ways for solving the research gaps are suggested to fill the gap.

2.6 ARTIFICIAL NEURAL NETWORK BASED IMAGE PROCESSING FOR WILD ANIMAL DETECTION AND MONITORING

Author: Archana H, Poornima K, Raksha V, Shwetha H R, Supreeth H S G

Animals leaving the forest area and entering the human habitat is increasing day by day. Animals entering the agricultural areas placed near the forest destroy crops or even attack on people therefore there is a need of system which detects the animal presence and gives warning about that in the view of security purpose. In this paper wild animal which enters the human habitation are recognized. The movement of animals in the input video is detected using Foreground detector algorithm. Extraction of foreground animal is done using Background subtraction based Gaussian mixture model. Morphological filters are used to remove the noise from binary image obtained from background subtraction. Training and recognition is done using back propagation algorithm. If trained images are matched with the test image, then the animal is recognized. The project aims to safe guard the wild life and animal life and it is a great help to forest department.

CHAPTER - 3

SYSTEM DESIGN

3.1 EXISTING SYSTEM

The current system mostly functions as a surveillance system in farms. The most important thing that should be taken into consideration that this system does give the assurance to create a barrier or to act as a physical barrier between the farm and the wild animals. The existing or the displayed system also needs to take into consideration that all the animals are different in size. According to the threat level provided by animals crossing or trespassing on farmland, appropriate precautions must be adopted to prevent animals from entering the restricted area. Many crop protection systems exist, but intrusion detection is rarely one of them. Farmers also use conventional measures like as scarecrows, which are ineffective in keeping wild animals and birds away, despite being simple to set up and effective in preventing bird ingress to some extent. They also use electric fencing, which has no channel for fast alerting of breaches occurring, and diagnosing problems is difficult and can be lethal at times. They are also quite expensive, with substantial maintenance costs. Some other measures used by farmers to avoid intrusion of wild animals are erecting physical barriers, manual inspection which is quite exhaustive.

3.1.1 Disadvantages

- Accuracy is very low.
- Manual Monitoring and Control
- Short distance communication

3.2 PROPOSED SYSTEM

- In the Internet of Things (IoT) design, everything which are wherever will be on the system in some structure. The motivation

behind this examination paper depends on the Internet of Things (IoT) innovation which is being applied to the farming part.

- The proposed system is used to automate the process of irrigation. Node MCU ESP8266 micro-controller is used to read data from sensors like soil moisture, Humidity. This system continuously gets data from sensors using a micro-controller and connects with the Mobile Applications and the cloud to send data.
- Protection of our home land is more of a necessity than a luxury. Detection system is a term that we all have been familiar with for the past few years. Over the years, the hardware and the technology used to develop this system has been constantly upgraded to assist in detecting intrusions with the help of machine learning.

3.2.1 Block Diagram

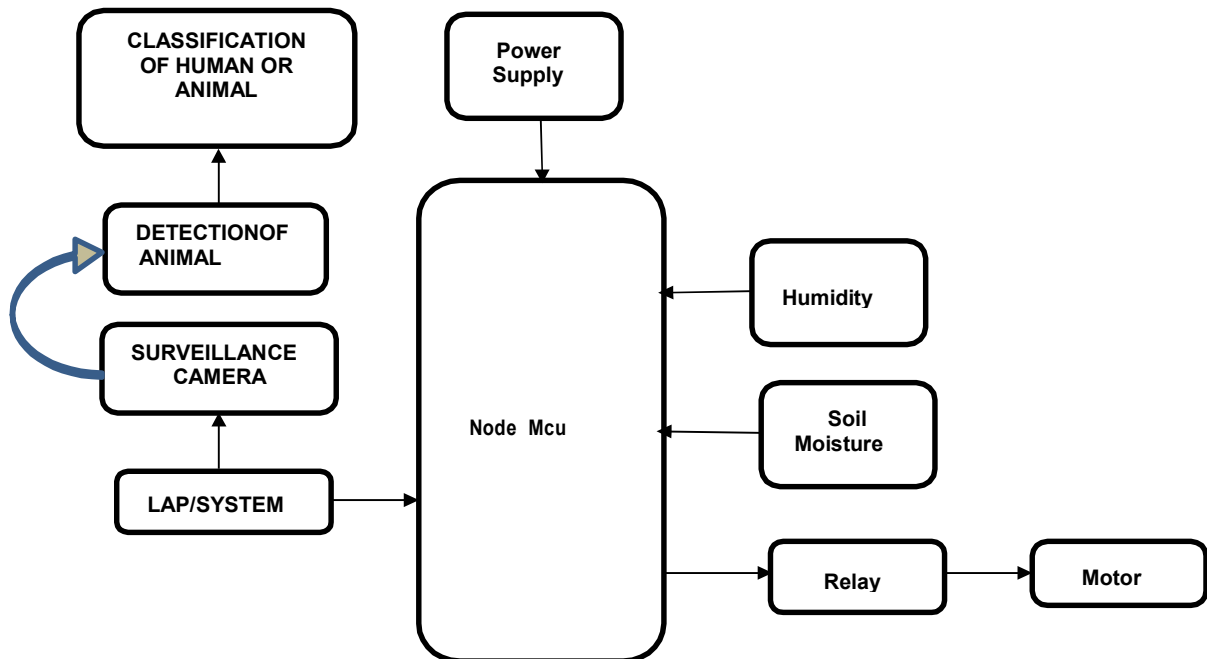


Fig. 3.1 Block Diagram of Proposed System

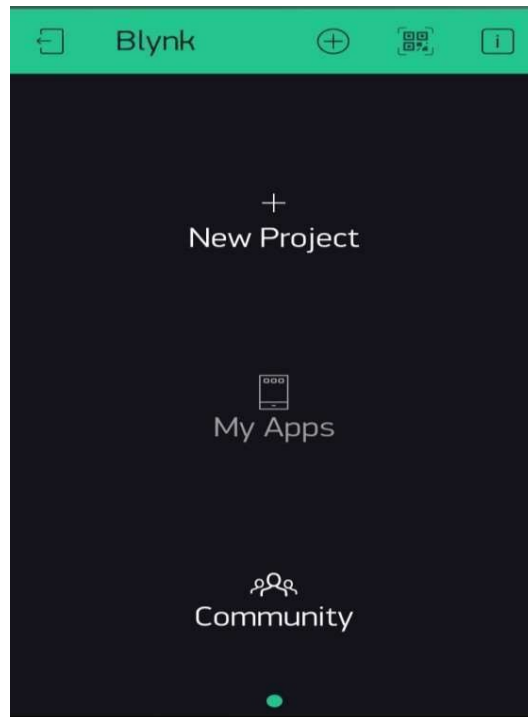


Fig. 3.2 Blynk App Screenshot

- We use machine learning to detect animals, entering into our farm by using deep neural network concept, a division in computer vision. In this project, we will monitor the entire farm at regular intervals through a camera which will be recording the surrounding throughout the day. With the help of a machine learning model, we detect the entry of animals and we play appropriate sounds to drive the animal away
- Moisture sensor used to detect the water level or moisture of the agriculture land

3.2.2 Advantages

- More Accuracy.
- This will improve the efficiency of the crops.
- We can monitor the cultivation field from any place.
- Protect the crop from animals.

3.3 METHODOLOGY

The system comprises of two stages, training and testing. In the training stage, a set of images are provided as visual examples. In the testing stage, a newly captured image i.e. the test image is given as input to the classifier. With the help of the knowledge gained from training, the test image is accordingly classified into the most favorable class.

3.3.1 Receiving the input image

In the proposed system, the test image is fed as input which is then converted into a binary pattern. A set of previously labelled images are present in the dataset whose features are matched with those of the test image, in order to determine the species of the animal present.

3.3.2 Feature Extraction

The test image which is received as input can be transformed into a reduced set of features. The selected features may contain the significant information from the input data, due to which the desired task can be performed by using this reduced amount of data instead of the initial unaltered data. Deep neural networks recognize features from images, and determine multiple levels of representation, with higher-level features depicting more abstract characteristics of the data.

3.3.3 Classifying the Species in an image

For the task of classification of species, the corresponding output layer generates the probabilities of the animal detected in the image belonging

to one of the possible classes. Even though providing such a result would save a huge amount of human effort that will be needed in recognizing the correct species, the testing of this hypothesis will require human knowledge.

3.3.4 Image Processing Flow

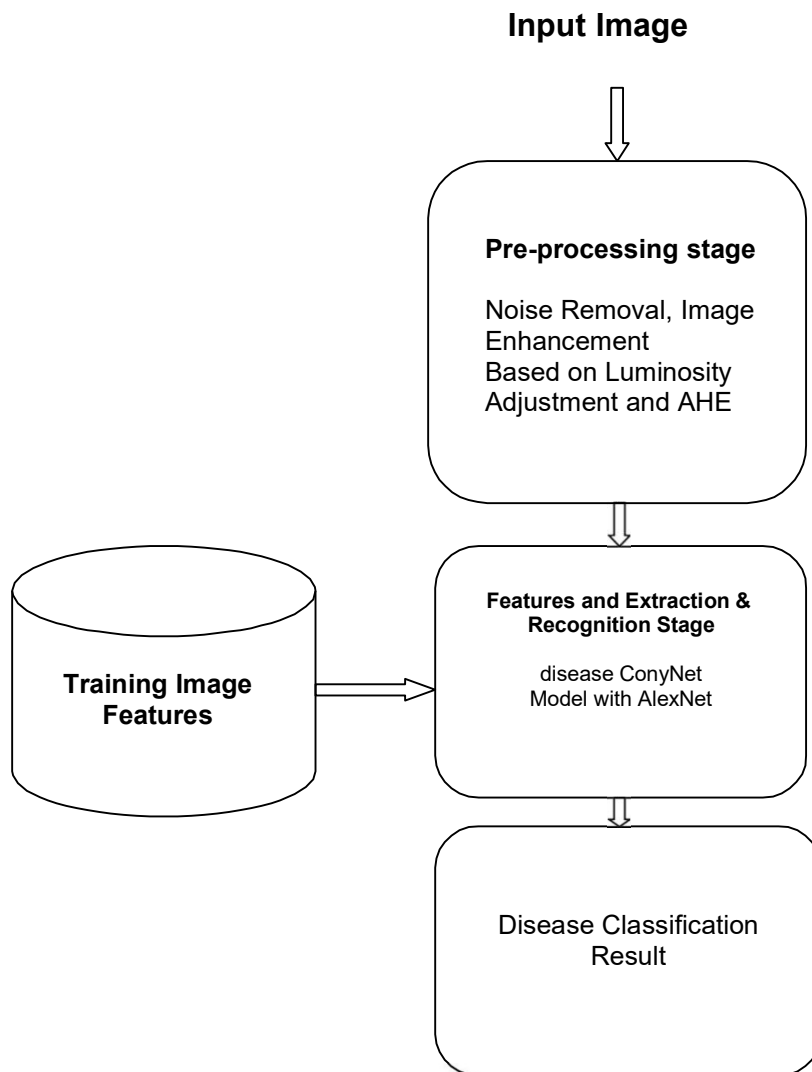


Fig. 3.3 Image Processing Flow

3.3.5 System Architecture

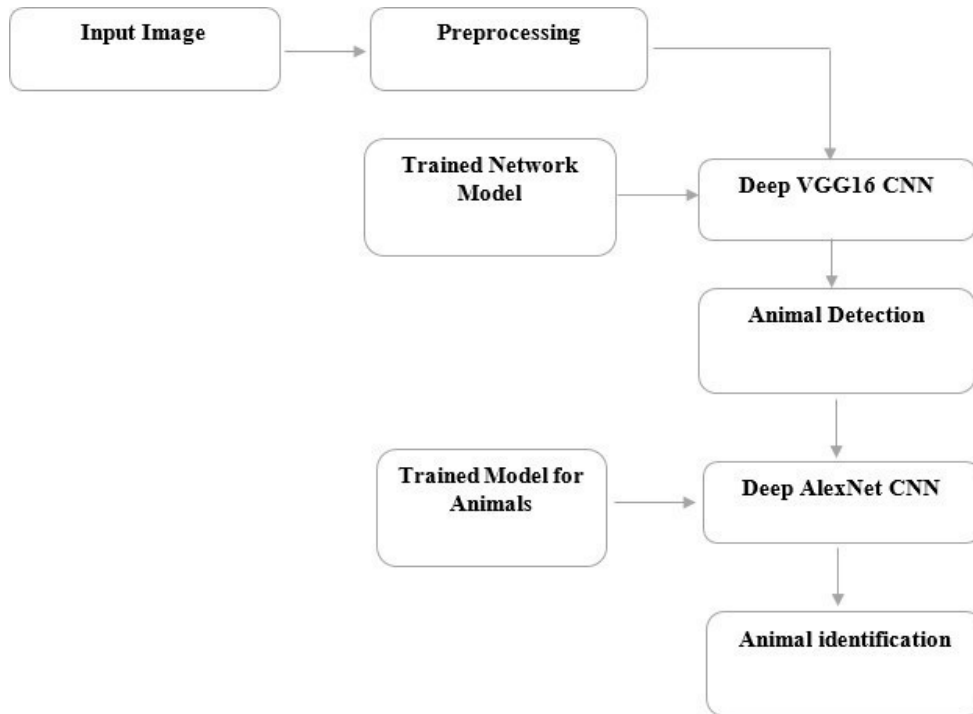


Fig. 3.4 System Architecture

3.3.6 Open CV 4.0

Open CV is an open source computer vision library which is very popular for performing basic image processing tasks such as blurring, image blending, enhancing image as well as video quality, thresholding etc. In addition to image processing, it provides various pre-trained deep learning models which can be directly used to solve simple tasks at hand. The programmers have to download and load the model using Open CV instructions in order to do the task of inference on their own dataset.

3.3.7 Keras 2.2.4

Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages.

3.3.8 Tensor Flow

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications. TensorFlow is a free and open source software for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. TensorFlow is Google Brain's second generation system. While the reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs (with optional CUDA and SYCL extensions for general purpose computing on graphics processing units). TensorFlow is available on 64-bit Linux, macOS, Windows, and mobile computing platforms including Android and iOS.

3.4 SYSTEM IMPLEMENTATION

3.4.1 Design Considerations

Dataset is large enough as possible. Previous results of CNN framework for animal classifications is taken into consideration.

3.4.2 Description of the Proposed Algorithm

The aim of the proposed algorithm is to compare the overall recognition accuracy of animals by using two deep learning neural network

frameworks – CNN and Faster RCNN and highlight the importance of deep learning technology in classification problems. The proposed algorithm is consisting of five main steps.

3.4.3 Input Image

The image uploaded by the user is considered as a input image (test image) and is sent to the network framework. The image is rescaled to a fixed size.

3.4.4 CNN

A convolutional neural network (CNN) is a deep neural network framework, in which every neuron will be connected to a region of neurons below it that allows handling of less amount of weights and it is mostly applied in visual imagery for classification and detection problems. It is a version of multilayer perceptrons. The name itself indicates that the network is employed with a mathematical operation called Convolution in order to extract and learn features from the images by the network.

3.4.5 RCNN

Faster RCNN is the recently proposed deep learning neural network framework mainly used for object detection. It comprises of two modules in a single unified network. The first module is a deep convolution neural network that works as a Region Proposal Network (RPN) and proposes Regions of Interest (ROI). The second module is a Fast RCNN detector that categorizes each of the proposed ROIs. This unification of RPN with the detector lowers testing computation time without any loss in the detection performance

3.4.6 ALGORITHM

3.4.6.1 Convolutional Neural Network

A convolutional neural network (CNN) is a subset of artificial neural

networks which uses perceptron, a machine learning algorithm for supervised learning to analyze large amount of data. CNNs can be used for image processing and natural language processing (NLP) applications and any kind of cognitive tasks. A convolutional neural network (CNN) has an input layer, an output layer and many numbers of hidden layers. Few of these layers are convolved, using mathematical models to carry on results to succeeding layers.

Input image will have the raw pixel values of color channels Red, Green (G), Blue (B).

The next layer is called, CONV layer. This layer will calculate the output of neurons that are connected to local regions to the input layer. Each calculates a point product between their weights and a small area connected to the input layer.

Third layer is called RELU layer, which will apply a unit wise activation function. This makes the size of the volume unaffected. • The next layer is called the POOL layer, which performs down sampling operation along the spatial dimensions (width and height) resulting in size such as [16x16x12].

• Last layer is FC (i.e. fully-connected) layer; this layer will calculate the class scores, resulting in volume of size. Like traditional Neural Networks, each neuron in this layer is connected to all the weights in the previous set.

3.4.7 Convolutional Layer

Fig. 1 shows the concept of convolutional layer; this layer extracts features from an input image. Convolution conserves the association between pixels by learning image features by means of small squares of input image data. In this mathematical operation, two inputs such as image matrix and a filter is used. Convolution of an image data with various filters can do operations like edge detection, blur and sharpening of images.

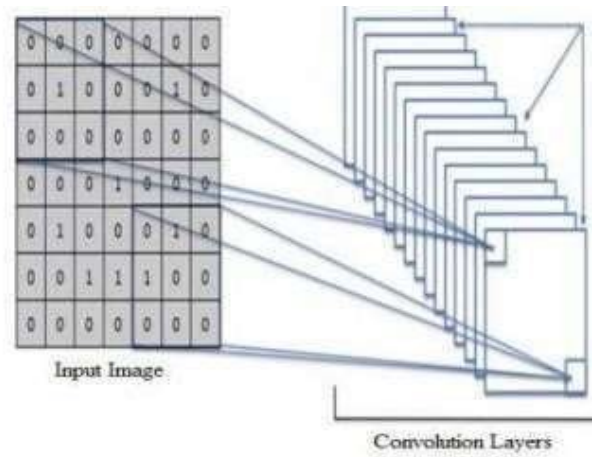


Fig. 3.5 Convolution Layers

3.4.8 Pooling

Pooling layer would lessen the number of parameters when a large image is given as input. Max pooling is done by taking the largest element from the revised feature map. The objective of max pooling is to down sample an input image, reducing its dimensions etc. This is shown in Fig. 2.

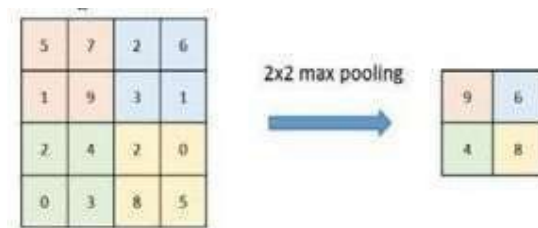


Fig. 3.6 Pooling

3.4.9 Flattening

Flattening is the procedure for converting the two dimensional array set into a single, long continuous linear vector. It receives the output from the convolutional layers, flattens its structure to create a single, long feature vector to be use the next layer for the final classification. This is shown in Fig. 3

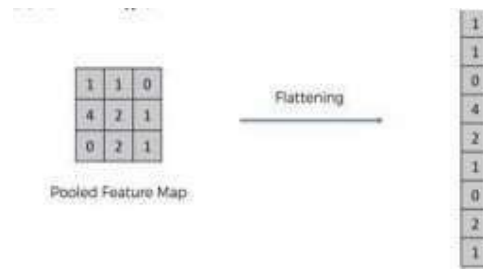


Fig. 3.7 Flattering

3.4.10 Fully connected

Fig.4 shows the hidden layers inside a Convolutional Neural Network that are called as Fully Connected Layers. These are a specific type of hidden layer which must be used within the CNN. This is used to combine the features into more attributes that predict the outputs more accurately

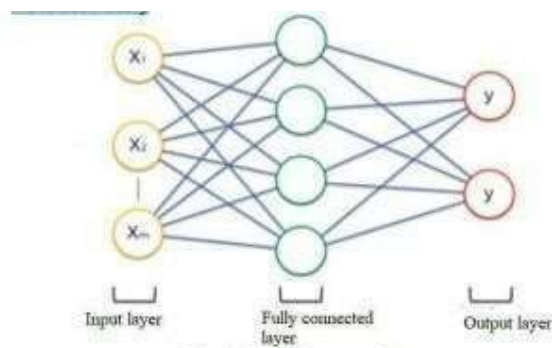


Fig. 3.8 Fully Connected

3.5 OPEN CV

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez (which was later acquired by Intel). The library is cross-platform and free for use under the open-source BSD license. OpenCV supports the deep learning frameworks TensorFlow, Torch/PyTorch and Caffe

3.5.1 History

Officially launched in 1999, the OpenCV project was initially an Intel Research initiative to advance CPU-intensive applications, part of a series of projects including real-time ray tracing and 3D display walls. The main contributors to the project included a number of optimization experts in Intel Russia, as well as Intel's Performance Library Team. In the early days of OpenCV, the goals of the project were described as:

- Advance vision research by providing not only open but also optimized code for basic vision infrastructure. No more reinventing the wheel.
- Disseminate vision knowledge by providing a common infrastructure that developers could build on, so that code would be more readily readable and
- Advance vision-based commercial applications by making portable, performance-optimized code available for free – with a license that did not require code to be open or free itself.

The first alpha version of OpenCV was released to the public at the IEEE Conference on Computer Vision and Pattern Recognition in 2000, and five betas were released between 2001 and 2005. The first 1.0 version was released in 2006. A version 1.1 “pre-release” was released in October 2008.

The second major release of the OpenCV was in October 2009. OpenCV 2 includes major changes to the C++ interface, aiming at easier, more type-safe patterns, new functions, and better implementations for existing ones in terms of performance (especially on multi-core systems). Official releases now occur every six months and development is now done by an independent Russian team supported by commercial corporations. In August 2012, support for OpenCV was taken over by a non-profit foundation OpenCV.org, which maintains a developer and user site.

On May 2016, Intel signed an agreement to acquire Itseez, the leading developer of OpenC

3.5.2 *OpenCV's application areas:*

- 2D and 3D feature toolkits
- Ego motion estimation
- Facial recognition system
- Human–computer interaction (HCI)
- Mobile robotics
- Motion understanding
- Object identification
- Motion tracking
- Augmented reality

To support some of the above areas, OpenCV includes a statistical machinelearning library that contains:

- Boosting
- Decision tree learning
- Gradient boosting trees
- Expectation-maximization algorithm
- k-nearest neighbor algorithm

CHAPTER – 4

SYSTEM REQUIREMENTS

4.1 NODE MCU

4.1.1 Node MCU ESP8266 (IoT)

Node MCU is an open source IoT stage. It incorporates firmware which keeps running on the ESP8266 Wi-Fi SoC from Expressive Systems, and equipment which depends on the ESP-12 module. The term NodeMCU typically refers to the firmware, whereas the board is termed Devkit. NodeMCU Devkit 1.0 consists of associate ESP-12E on a board that facilitates its use. It additionally contains a transformer, a USB interface. The expression “NodeMCU” of course alludes to the firmware as opposed to the improvement units. The firmware utilizes the Lua scripting dialect

The NodeMCU (Node Micro Controller Unit) is an open source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266 is designed and manufactured by Express, contains all crucial elements of the modern computer: CPU, RAM, networking (wi-fi), and even a modern operating system and SDK. When purchased at bulk, the ESP8266 chip costs only \$2 USD a piece. That makes it an excellent choice for this system design. The NodeMCU aims to simplify ESP8266 development. It has two key components.

- i. An open source ESP8266 firmware that is built on top of the chip manufacturer’s proprietary SDK. The firmware provides a simple programming environment based on eLua (embedded Lua), which is a very simple and fast scripting language with an established developer community. For new comers, the Lua scripting language is easy to learn. And to add on NodeMCU can be programmed with the Android IDE too.
- ii. A development kit board that incorporates the ESP8266 chip on a standard circuit board. The board has a built-in USB port that is already wired up with the chip, a hardware reset button, Wi-Fi antenna, LED lights, and standard-sized GPIO (General Purpose Input Output) pins that can

plug into a bread board. Figure 2 below shows the NodeMCU development board.



**Fig 4.1 :- NodeMCU (ESP8266)
Development Board**

There are various platforms for IoT system one of them is NodeMCU. It provides lower level control on devices which is known as firmware. These control runs on ESP8266 Wi-Fi SoC, which hardware is based on ESP-12 module. Esp8266 is a Wi-Fi based communication system's microchip. It uses TCP/IP protocol for communicating with internet through router. It uses 802.11b/g/n standard technology for Wi-Fi communication. It is a Tensilica L106 32-bit RISC instruction unite microprocessor with 32 KB instruction RAM, 32KB instruction Cache RAM, 80KB Data memory. ESP8266 (Pin) and Arduino Mega's (Pin) common pins are GND TX (0)-3.3V, RX (0), 3.3V. For running the ESP8266 on Arduino platform we need to install ESP8266 package in Arduino IDE. As a board is using, we need to give the additional board manager.

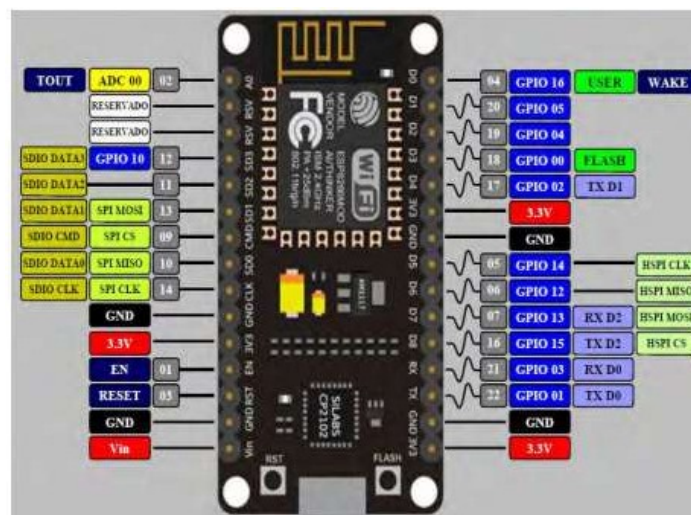


Fig 4.2: Esp8266 NodeMCU chip

As shown in Fig it has 12 GPIO pins. One ADC pin. For this project 7 GPIO pins have been used. They are D0 to D6 which delivers digital outputs.

4.2 HADWARE DETAILS

4.2.1 HUMIDITY SENSOR

Sometimes, after rains, the air feels moist. The water seems to have suspended in the air. However, in certain AC s you click some buttons and the atmosphere brightens up. How and why does it all happen? Moisture forms up in the air, resulting in humidity. However, the humidity sensor in your AC picks it up and cleans it up for you. Isn't that wonderful? Let's take a look how it does that.

A humidity sensor (or hygrometer) senses, measures and reports both moisture and air temperature. The ratio of moisture in the air to the highest amount of moisture at a particular air temperature is called relative humidity. Relative humidity becomes an important factor, when looking for comfort.

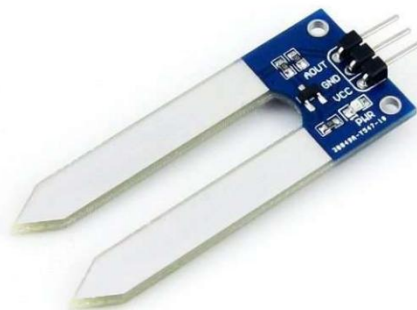


Fig 4.3 Humidity Sensor

Humidity sensors work by detecting changes that alter electrical currents or temperature in the air. There are three basic types of humidity sensors: capacitive, resistive and thermal. All three types of sensors monitor minute changes in the

atmosphere in order to calculate the humidity in the air.

4.2.2Capacitive

A capacitive humidity sensor measures relative humidity by placing a thin strip of metal oxide between two electrodes. The metal oxide's electrical capacity changes with the atmosphere's relative humidity. Weather, commercial and industries are the major application areas

4.2.3Resistive

Resistive humidity sensors utilize ions in salts to measure the electrical impedance of atoms. As humidity changes, so does the resistance of the electrodes on either side of the salt medium.

4.2.4Thermal

Two thermal sensors conduct electricity based upon the humidity of the surrounding air. One sensor is encased in dry nitrogen while the other measures ambient air. The differencebetween the two measures the humidity. The capacitive type sensors are linear and can measure relative humidity from 0% to 100%. The catch here is a complex circuit and regular calibration. However for designers this a lesser hassle over precise measurement and hence these dominate atmospheric and process measurements. These are the only types of full- range relative humidity measuring devices down to 0% relative humidity. This low temperature effect often leads to them being used over wide temperature ranges without active temperature compensation.

4.2.5Humidity Sensor working

- Coulometric: An electrolyte is formed by absorption of water resulting in a currentlevel which is proportional to the moisture content in the air.
- Gravimetric: A drying agent is exposed to moist air, resulting in weight gain by the drying agent. The increased weight corresponds to the

amount of moisture.

- Microwave/Infrared: A transmitted signal varies as the humidity increases. The attenuation is an indication of the moisture content in the medium.

Specifications

The specifications of the soil moisture sensor FC-28 are as follows

Input Voltage	3.3 – 5V
Output Voltage	0 – 4.2V
Input Current	35mA
Output Signal	Both Analog and Digital

Fig 4.4 Specification List

4.3 DHT11 Sensor and Its Working

Humidity is the measure of water vapor present in the air. The level of humidity in air affects various physical, chemical and biological processes. In industrial applications, humidity can affect the business cost of the products, health and safety of the employees. So, in semiconductor industries and control system industries measurement of humidity is very important. Humidity measurement determines the amount of moisture present in the gas that can be a mixture of water vapour, nitrogen, argon or pure gas etc... Humidity sensors are of two types based on their measurement units. They are a relative humidity sensor and Absolute humidity sensor. DHT11 is a digital temperature and humidity sensor. DHT11 is a low-cost digital sensor for sensing temperature and humidity. This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc... to measure humidity and temperature instantaneously.

DHT11 humidity and temperature sensor is available as a sensor and as a module. The difference between this sensor and module is the pull-up resistor and a power-on LED. DHT11 is a relative humidity sensor. To measure the

surrounding air this sensor uses a thermistor and a capacitive humidity sensor.

4.3.1 Working Principle of DHT11 Sensor

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form.

For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz .i.e. it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5Ma.

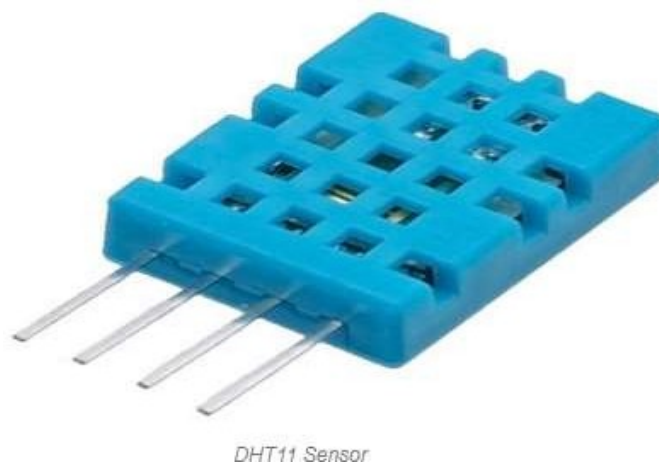


Fig 4.5 DHT11 Sensor

DHT11 sensor has four pins- VCC, GND, Data Pin and a not connected pin. A pull-

up resistor of 5k to 10k ohms is provided for communication between sensor and micro-controller.

4.4 Applications

This sensor is used in various applications such as measuring humidity and temperature values in heating, ventilation and air conditioning systems. Weather stations also use these sensors to predict weather conditions. The humidity sensor is used as a preventive measure in homes where people are affected by humidity. Offices, cars, museums, greenhouses and industries use this sensor for measuring humidity values and as a safety measure.

4.5 RELAY

4.5.1 Relay Basics

A relay is an electrical switch that uses an electromagnet to move the switch from the off to on position instead of a person moving the switch. It takes a relatively small amount of power to turn on a relay but the relay can control something that draws much more power. Ex: A relay is used to control the air conditioner in your home. The AC unit probably runs off of 220VAC at around 30A. That's 6600 Watts! The coil that controls the relay may only need a few watts to pull the contacts together.



Fig 4.6 Relay

4.5.2 Relay Basic Section

Relays (and switches) come in different configurations. The most common are shown to the right. Single Pole Single Throw (SPST) is the simplest with only two contacts. Single Pole Double Throw (SPDT) has three contacts. The contacts are usually labelled Common (COM), Normally Open (NO), and Normally Closed (NC). The Normally Closed contact will be connected to the Common contact when no power is applied to the coil. The Normally Open contact will be open (i.e. not connected) when no power is applied to the coil. When the coil is energized the Common is connected to the Normally Open contact and the Normally Closed contact is left floating. The Double Pole versions are the same as the Single Pole version except there are two switches that open and close Together. Select a relay with contacts that can handle the voltage and current requirements of the load. Keep in mind that some loads (such as motors) draw much more current when first turned on than they do at steady state. Select a relay with a coil voltage and current that you can control easily. Ex: If you want to turn on the AC unit with a 12VDC power supply get a 12VDC coil.

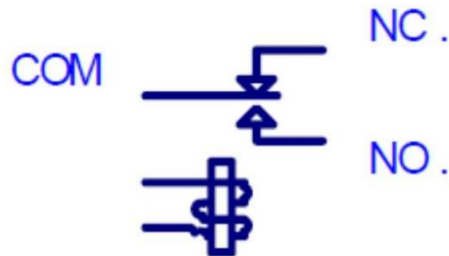


Fig 4.7 Relay Basic Section

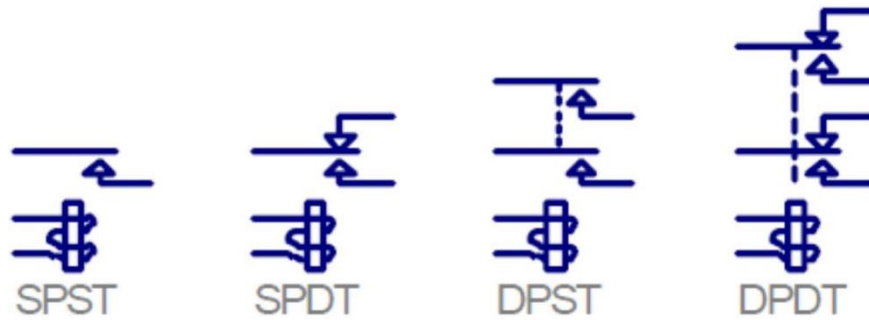


Fig 4.8 Relay Basic Section

4.5.3 Practical Considerations:

Check the relay datasheet for the expected lifetime (i.e. # of times it can open and close before failure). If the relay won't be used much (say to control the headlights on a car) a 20,000 cycle lifetime would last about 18 years if used three times a day. If the same relay was used to control a home AC unit which switches on and off much more often it would wear out in a few years. Some relays have lifetimes of over a million cycles.

4.5.4 Flyback Diode:

A relay coil is not only an electromagnet but it's also an inductor. When power is applied to the coil the current in the coil builds up and levels off at its rated current (depends on the DC resistance of the coil, $I = V/R$). Some energy is now stored in the coil's magnetic field ($E = 0.5LI^2$). When the current in the coil is turned off this stored energy has to go somewhere. The voltage across the coil quickly increases trying to keep the current in the coil flowing in the same direction ($V = Ldi/dt$). This voltage spike can reach hundreds or thousands of volts and can damage electronic parts.

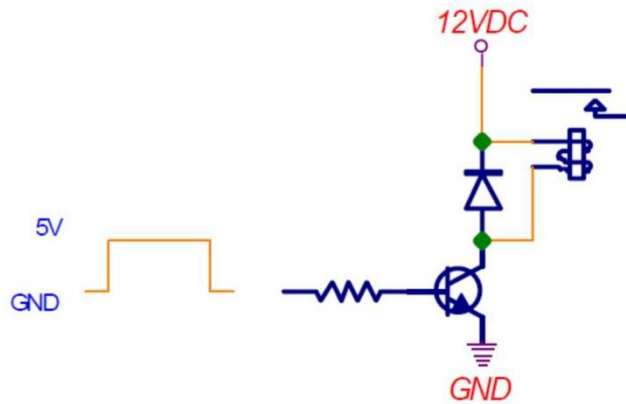


Fig 4.9 Flyback Diode Block Diagram 1

By adding a Flyback diode the current has a path to continue flowing through coil until the stored energy is used up. The diode also clamps the voltage across the coil to about 0.7V protecting the electronics. The stored energy dissipates quickly in the diode ($E = V \cdot I \cdot t$). The current stops flowing and the relay turns off. The diode should be able to handle the coil current for a short time and switch relatively fast. Note: A resistor or Zener diode can be placed in series with the diode to use up the stored energy quicker. This increases the amplitude of the voltage spike above 0.7V but the energy is used up quicker (i.e. the voltage spike won't last as long). Usually it doesn't matter if the relay takes 1ms or 100ms to turn off.

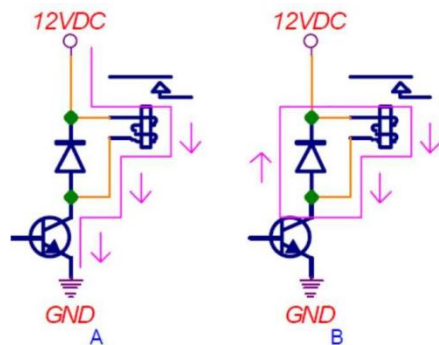


Fig 4.10 Flyback Diode Block Diagram 2

The schematic on the left illustrates how current flows in a relay. Fig A shows the current flow when the transistor is on. Fig B shows the current flow when the transistor is off. Notice how the diode completes the current loop.

4.6 SOFTWARE TOOL

4.6.1 ARDUINO SOFTWARE (IDE):

The Arduino Integrated Development Environment – or Arduino Software (IDE) – contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

Writing Sketches:

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the ArduinoSoftware (IDE), including complete error messages and other information. The bottom right- hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serialmonitor.

NB: Versions of the Arduino Software (IDE) prior to 1.0 saved sketches with the extension

.pde. It is possible to open these files with version 1.0, you will be prompted to save the sketchwith the .ino extension on save.

Verify: Checks your code for errors compiling it.

Upload: Compiles your code and uploads it to the configured board. See uploading below for details.

Note: If you are using an external programmer with your board, you can hold down the “shift” key on your computer when using this icon. The text will change to “Upload using Programmer”

New: Creates a new sketch.

Open: Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.

Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the File | Sketchbook menu instead.

Save: Saves your sketch.

Serial Monitor: Opens the serial monitor.

Additional commands are found within the five menus: File, Edit, Sketch, Tools, and Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

File:

New: Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.

Open: Allows to load a sketch file browsing through the computer drives and folders.

Open Recent: Provides a short list of the most recent sketches, ready to be opened.

Sketchbook: Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.

Examples: Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.

Close: Closes the instance of the Arduino Software from which it is clicked.

Save: Saves the sketch with the current name. If the file hasn't been named before, a name will be provided in a "Save as.." window.

Save as: Allows to save the current sketch with a different name.

Page Setup: It shows the Page Setup window for printing.

Print: Sends the current sketch to the printer according to the settings defined in Page Setup. **Preferences:** Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.

Quit: Closes all IDE windows. The same sketches open when Quit was chosen will be automatically reopened the next time you start the IDE

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS :

In this study, we developed a smart irrigation system using the Blynk app, ESP NodeMCU, and animal detection. The system aimed to automate the process of watering plants and prevent animals from destroying them.

We tested the system in a garden with various plants and monitored its performance for two months. The system successfully detected animals using a software and sent alerts to the user's smartphone through the Blynk app. The user could then take appropriate action to scare the animals away or sound alert system will be made.

The irrigation system also performed well, as it watered the plants based on the moisture level in the soil. The user could set the desired moisture level using the Blynk app, and the system would automatically turn on the water pump when the moisture level fell below the set point and shows the notification in the blynk app.

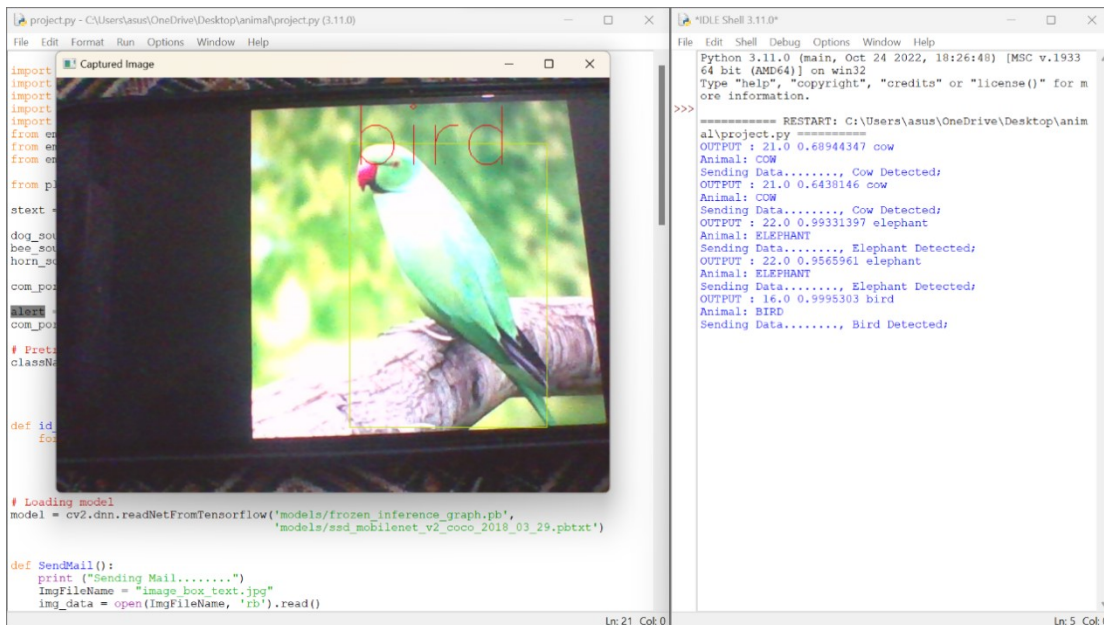


Fig 5.1 Bird Detection

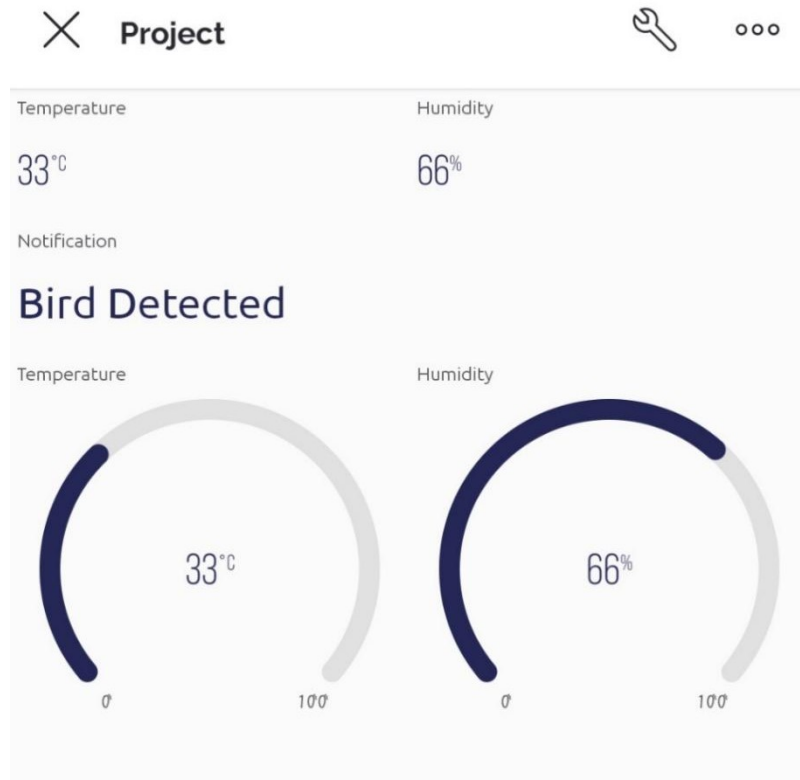


Fig 5.2 Notification in Blynk App

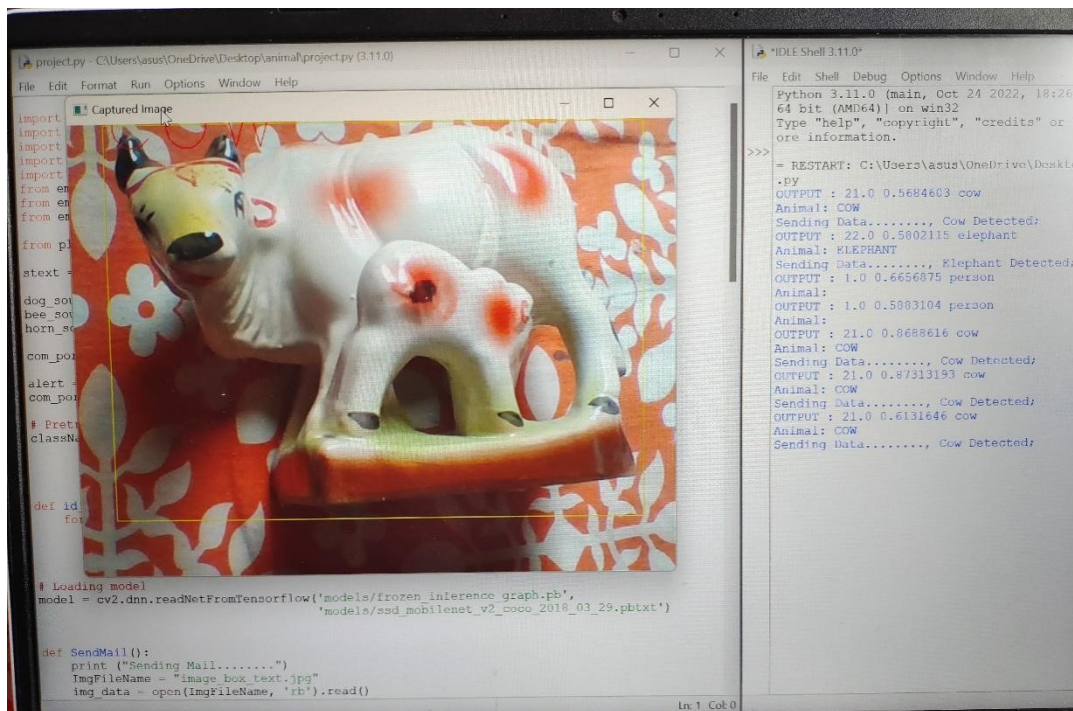


Fig 5.3 Detection OF 3D Image

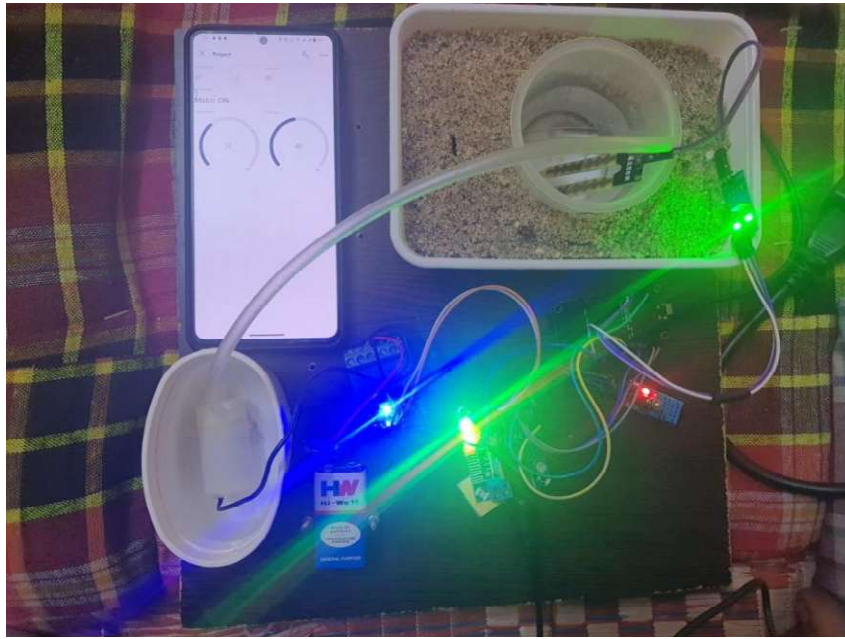


Fig 5.4 Smart Irrigation Output

5.2 DISCUSSION:

The result of this study suggests that the developed smart irrigation system with animal detection is an effective way to protect plants from animals while ensuring optimal watering. The Blynk app provides an intuitive and user-friendly interface that allows users to monitor and control the system from their smartphones.

The system's dependency on the Wi-Fi network could be one of its drawbacks. The system will not operate properly if the network is down, which could cause harm to the plants. Also, the system could not function properly in locations with weak Wi-Fi access.

Future research could investigate ways to overcome these restrictions by using different communication channels like Bluetooth or cellular networks. To provide more precise watering suggestions, the system might also be enhanced to include more sensors, including temperature and humidity sensors. Using the Blynk app, ESP NodeMCU, and a PIR motion sensor, this study showed the viability and efficacy of a smart irrigation system with animal detection. A system is a useful tool for farmers and gardeners since it offers an automated solution for animal and plant protection.

5.3 DIFFERENCE BETWEEN THE EXISTING SYSTEM AND PROPOSED SYSTEM (WHEN MICROCONTROLLER IS ESPNODE MCU VS ESPNODE MCU)

Both projects share the ESP NodeMCU development board and are concerned with smart irrigation, but they differ in terms of communication strategy and extra functionality.

GSM technology is used in this project to remotely operate and monitor the irrigation system for smart irrigation utilizing ESP NodeMCU. In order to measure the soil moisture content and other irrigation-related data, the ESP NodeMCU board interfaces with a variety of sensors, including moisture sensors, temperature sensors, and humidity sensors. To send SMS alerts or to accept commands from the user's phone to control the irrigation system, the NodeMCU board interfaces with the GSM module.

Animal recognition and smart watering using the Blynk app with ESP NodeMCU: The ESP NodeMCU development board is also used in this project to interface with sensors and measure the soil moisture content and other irrigation-related metrics. To monitor and control the irrigation system, it uses the Blynk app rather than GSM technology. The Blynk app offers a graphical interface for users to utilize on their smartphones to monitor and operate the system. A further characteristic of this project is its ability to identify the presence of animals like cows, dogs, or other livestock in a field. When this happens, the irrigation system is turned off to prevent watering those animals.

5.4 DIFFERENCE BETWEEN THE EXISTING SYSTEM AND PROPOSED SYSTEM (WHEN MICROCONTROLLER IS ARDUINO VS ESPNODE MCU)

While they both deal with smart irrigation and animal entry detection, the communication technique, hardware, and software employed in each project are different.

Using a GSM module and an Arduino microcontroller, this project automates smart irrigation and deters animal trespass. The system comprises of a PIR (Passive Infrared Sensor) to identify animals and a soil moisture sensor to track the amount of soil moisture. When there is insufficient soil moisture or animal intrusion, the Arduino board and GSM module communicate to deliver SMS notifications to the user's phone.

The user can reply by sending an SMS to activate the irrigation system or halt animal intrusion.

Animal recognition and smart watering using the Blynk app with ESP NodeMCU: The Blynk app and the ESP NodeMCU development board are used in this project to facilitate communication. The system consists of a temperature sensor, an animal detection sensor, and a soil moisture sensor. The Blynk software, which offers a graphical interface for the user to operate and monitor the system from their smartphone, interfaces with the ESP NodeMCU board to enable this functionality. The presence of animals is detected by the animal detection sensor, which then turns off the irrigation system to stop it from watering the animals.

So, despite having the same goal of creating a smart irrigation system with animal entry detection, the communication mechanism, hardware, and software employed by each project vary. The Arduino microcontroller and GSM module are used in the first project, while the ESP NodeMCU development board and the Blynk app are used in the second project.

CHAPTER - 6

SUMMARY AND CONCLUSION

Through the proposed project, our team promises to bring about a 40% to 50% reduction in the wastage of water on the agricultural field. The system being crop specific puts forth an eminent and ample water supply to every type of crop under culture. The use of wireless sensor networks aids the farmers to feasibly control and operate the irrigation system on their mobile devices reducing physical labor. Hence, the project promises reduced manpower, increasing efficiency, smart utilization of agricultural techniques therefore increasing yield, profit and gaining highly effective and safe agricultural production to bring significant impact on ensuring efficient usage of water resources as well as ensuring the efficiency and stability of agricultural production.

The problem of destruction of crops by unauthorized humans and wild animals has become a major problem in the current time. It requires crucial attention and an effective solution. The system is designed for such problems which will monitor the field using sensor and camera and captured image of intruder will be classified into authorized or unauthorized using image processing so that suitable action can be taken to prevent the farm. This system will also help them in achieving better crop yields.

We use machine learning to detect animals, entering into our farm by using deep neural network concept, a division in computer vision. In this project, we will monitor the entire farm at regular intervals through a camera which will be recording the surrounding throughout the day. With the help of a machine learning model, we detect the entry of animals and we play appropriate sounds to drive the animal away. Moisture sensor used to detect the water level or moisture of the agriculture land.

6.1 FUTURE WORK

There are several potential areas for future work for a Smart irrigation system using Blynk app with ESP Node MCU and animal detection projects, such as:

Improving the accuracy of animal detection: One area of future work is to improve the accuracy of animal detection by using more advanced algorithms and techniques such as machine learning, computer vision, and neural networks. This could help reduce false positives and improve the overall effectiveness of the system.

Integration with weather data: Another area of potential improvement is to integrate the system with weather data to adjust irrigation schedules based on the current weather conditions. This could help reduce water waste and optimize water usage.

Remote monitoring and control: It would be useful to add the ability to remotely monitor and control the irrigation system through the Blynk app. This could allow users to adjust irrigation schedules, view system status, and receive alerts if any issues arise.

Expansion to larger areas: The system could be expanded to cover larger areas by using multiple ESP Node MCU devices and sensors. This could be useful for large-scale farming operations or public parks.

Integration with other smart systems: The system could be integrated with other smart systems such as smart lighting, security, and energy management to create a more comprehensive and efficient smart environment.

6.2 RESEARCH WORK

There are several research issues that could be explored for a Smart irrigation system using Blynk app with ESP Node MCU and animal detection project, such as:

Optimization of irrigation scheduling: One research issue is to optimize the irrigation scheduling to reduce water consumption while maintaining crop yield. This could involve studying the water requirements of different crops, soil moisture monitoring, and weather forecasting to determine the most effective irrigation schedules.

Energy efficiency: Another research issue is to improve the energy efficiency of the system by optimizing the power consumption of the ESP Node MCU and sensors, and by exploring alternative energy sources such as solar or wind power.

Integration of different sensors: There is a need to integrate different types of sensors to provide a more comprehensive understanding of the environmental conditions that affect crop growth, such as temperature, humidity, and soil nutrient levels.

Animal detection algorithm: One important research issue is to improve the accuracy of the animal detection algorithm used in the system. This could involve exploring different image processing techniques, such as object detection and pattern recognition, and developing new algorithms that can better differentiate between animals and other objects.

Cost-effectiveness: A key research issue is to develop a system that is cost-effective and affordable for small-scale farmers. This could involve exploring different hardware and software options, as well as developing new business models that can help farmers access and use the technology.

6.3 IMPLEMENTATIONS ISSUES

Implementing a Smart Irrigation system using the Blynk app with an ESP NodeMCU and an animal detection project could potentially face the following implementation issues:

Hardware compatibility: Ensuring that the ESP NodeMCU board and other hardware components are compatible with the Blynk app and animal detection sensors.

Power supply: The system requires a reliable power source to function properly. Ensure that the power supply meets the voltage and current requirements of the hardware components.

Sensor calibration: The animal detection sensors need to be calibrated properly to detect animals accurately. Improper calibration could lead to false alarms or incorrect irrigation.

Data transmission: The data from the animal detection sensors and the irrigation system needs to be transmitted reliably to the Blynk app. A stable and robust internet connection is necessary to ensure uninterrupted data transmission.

User interface: The Blynk app needs to be designed with an intuitive and user-friendly interface to ensure that users can easily understand and control the irrigation system and animal detection sensors.

Maintenance and troubleshooting: Regular maintenance and troubleshooting of the system may be required to ensure proper functioning. This could include firmware updates, sensor cleaning, and hardware replacements, among others.

Environmental factors: The system needs to be designed to withstand different weather conditions and environmental factors such as humidity, temperature, and moisture

Addressing these implementation issues can help ensure the successful deployment and operation of a Smart Irrigation system using the Blynk app with an ESP NodeMCU and animal detection project.

REFERENCES

- [1] Ahonen, T., Hadid, A., Pietikainen, and M.: Face description with local binary patterns: Application to face recognition. *IEEE TPAMI* 28(12), 2037-2041 (2006).
- [2] Baojun Qu. Q.Y., "A new concentration detection system for sfn mixture gas inextra ultra high voltage power transmission systems," *IEEE Sensors Journal*, vol. 18, no. 9, pp. 3806–3812, 2018.
- [3] Burghardt, T., Calic, J.: Real-time face detection and tracking of animals. In: *Neural Network Applications in Electrical Engineering*. pp. 27{32. *IEEE* (2006).
- [4] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detectionwith discriminatively trained part-based models. *IEEE TPAMI* 32(9), 1627- 1645(2010)
- [5] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition.In:CVPR. pp. 770-778 (2016).
- [6] Kamencay, P., T. Trnovszky, M. Benco, R. Hudec, P. Sykora and A. Satnik. Accurate wild animal recognition using PCA, LDA and LBPH, In: 2016 ELEKTRO. Strbske Pleso: IEEE, 2016, pp. 62–67.
- [7] McKinion. J., J. Jenkins, D. Akins, S. Turner, J. Willers, E. Jallas, and F. Whisler, "Analysis of a precision agriculture approach to cotton production," *Computersand Electronics in Agriculture*, vol. 32, pp. 213–228, 2001.
- [8] Su. B., "Electrical anisotropic response of water conducted fractured zone in themining goaf," *IEEE Access*, vol. 4, pp. 6216–6224, 2016.
- [9] Tianhe Gong. P.C., Haiping Huang, "Secure twoparty distance computation protocol based on privacy homomorphism and scalar product in wirelesssensor networks," *Tsinghua Science and Technology (IEEE)*, vol. 21, no. 4, pp. 385–396, 2016.

- [10] Tiber Trnovszky, Patrik Kamency, Richard Orjesek, Miroslav Benco, Peter Sykro. Animal recognition system based on convolution neural network.
- [11] WU, J. L. and W. Y. MA. A Deep Learning Framework for Coreference Resolution Basedon Convolutional Neural Network. In: 2017 IEEE 11th International Conference on Semantic Computing (ICSC). San Diego: IEEE, 2017, pp. 61–64.
- [12] Xie, Z., A. Singh, J. Uang, K.S. Narayan and P.Abbeel. Multimodal blending for high-accuracy instance recognition. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. Tokyo: IEEE 2013, pp. 2214- 2221.
- [13] Yu Bie. M., “Effect of phase transition temperature and thermal conductivity on the performance of latent heat storage system,” Applied Thermal Engineering, vol. 135, pp.21

APPENDIX

A. SOURCE CODE

PYTHON CODE FOR ANIMAL DETECTION

```
import cv2
import time
import os
import serial
import smtplib
from email.mime.text import MIMEText
from email.mime.image import MIMEImage
from email.mime.multipart import MIMEMultipart

from playsound import playsound

stext = ""

dog_sound = 'dog10.mp3'
bee_sound = 'bee10.mp3'
horn_sound = 'horn10.mp3'

com_port = serial.Serial("COM5", baudrate=9600, timeout=1)

alert = False
com_port_data = ""

# Pretrained classes in the model
classNames = {0: 'background',
              1: 'person', 16: 'bird', 17: 'cat', 18: 'dog', 19: 'horse', 20: 'sheep',
              21: 'cow', 22: 'elephant', 23: 'bear', 24: 'zebra', 25: 'giraffe'}

def id_class_name(class_id, classes):
    for key, value in classes.items():
        if class_id == key:
            return value

# Loading model
model = cv2.dnn.readNetFromTensorflow('models/frozen_inference_graph.pb',
                                       'models/ssd_mobilenet_v2_coco_2018_03_29.pbtxt')

def SendMail():
    print ("Sending Mail.....")
    ImgFileName = "image_box_text.jpg"
```

```

img_data = open(ImgFileName, 'rb').read()
msg = MIMEMultipart()
msg['Subject'] = 'Animal Detected'
msg['From'] = 'embeddedproject2021@gmail.com'
msg['To'] = 'embeddedproject2021@gmail.com'
text2 = MIMEText("Animal detected :"+text)
print ("Mail message :",text2)
msg.attach(text2)
image = MIMEImage(img_data, name=os.path.basename(ImgFileName))
msg.attach(image)
s = smtplib.SMTP('smtp.gmail.com', 587)
s.ehlo()
s.starttls()
s.ehlo()
s.login('embeddedproject2021', 'Admin@123')
From='embeddedproject2021@gmail.com'
To=['embeddedproject2021@gmail.com', 'embeddedproject2021@gmail.com']
s.sendmail(From, To, msg.as_string())
s.quit()
print ("Mail Sent.")

def sendCOM(message):
    print ("Sending Data.....",message)
    com_port.write(str.encode(message +'\r\n'))
    time.sleep(3)

while(1):
    cap = cv2.VideoCapture(0)
    check, frame = cap.read()
    check, frame = cap.read()
    check, frame = cap.read()
    frame = cv2.resize(frame, (640, 480))
    #print(check)
    image = frame

    image_height, image_width, _ = image.shape

    model.setInput(cv2.dnn.blobFromImage(image, size=(300, 300), swapRB=True))
    output = model.forward()

    text = ""
    for detection in output[0, 0, :, :]:
        confidence = detection[2]
        if confidence > .5:
            class_id = detection[1]
            if class_id == 0 or class_id == 1 or class_id == 16 or class_id == 17 or

```

```

class_id == 18 or class_id == 19 or class_id == 20 or class_id == 21 or class_id
== 22 or class_id == 23 or class_id == 24 or class_id == 25 :
    class_name=id_class_name(class_id,classNames)
    if( (class_id != None) and (detection[2] != None) ):
        print(str("OUTPUT : "+ str(class_id) + " " + str(detection[2]) + " " +
class_name))
        box_x = detection[3] * image_width
        box_y = detection[4] * image_height
        box_width = detection[5] * image_width
        box_height = detection[6] * image_height
        cv2.rectangle(image, (int(box_x), int(box_y)), (int(box_width),
int(box_height)), (23, 230, 210), thickness=1)
        cv2.putText(image,class_name ,(int(box_x),
int(box_y+.05*image_height)),cv2.FONT_HERSHEY_SIMPLEX,(.005*image_widt
h),(0, 0, 255))
        if class_id == 21 :
            text = "COW"
        if class_id == 22 :
            text = "ELEPHANT"
        if class_id == 23 :
            text = "BEAR"
        if class_id == 16 :
            text = "BIRD"
        if class_id == 20 :
            text = "SHEEP"
        print ("Animal:", text)

cv2.imwrite("image_box_text.jpg",image)
imS = cv2.resize(image, (640, 480))
cv2.imshow('Captured Image',imS)
cv2.moveWindow('Captured Image',50,50)

cv2.waitKey(1)

#global com_port_data
#global alert

if(text == "COW"):
    #SendMail()
    com_port_data = "Cow Detected;"
    sendCOM(com_port_data)
    playsound(dog_sound)
    text="NA"
if(text == "ELEPHANT"):
    #SendMail()
    com_port_data = "Elephant Detected;"
    sendCOM(com_port_data)

```

```

    playsound(bee_sound)
    text="NA"
if(text == "BEAR"):
    #SendMail()
    com_port_data = "Bear Detected;"
    #sendCOM(com_port_data)
    playsound(horn_sound)
    text="NA"
if(text == "BIRD"):
    #SendMail()
    com_port_data = "Bird Detected;"
    sendCOM(com_port_data)
    playsound(horn_sound)
    text="NA"
if(text == "SHEEP"):
    #SendMail()
    com_port_data = "Sheep Detected;"
    sendCOM(com_port_data)
    playsound(dog_sound)
    text="NA"

cap.release()

cv2.waitKey(0)
cv2.destroyAllWindows()

```

REQUIRED ARDUINO CODE FOR RUNNING OF THE MICROCONTROLLER

```

#define BLYNK_TEMPLATE_ID "TMPLobf3b62W"

#define BLYNK_DEVICE_NAME "Project"

#define BLYNK_AUTH_TOKEN      "jrK-X75uV6L2IWs_4qPuo06N5ptZ1aUI"

#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

char auth[] = BLYNK_AUTH_TOKEN;

char ssid[] = "embedded";

char pass[] = "embedded123";

BlynkTimer timer;

```

```

#include "DHT.h"

#define DHTTYPE DHT11

uint8_t DHTPin = D7;

DHT dht(DHTPin, DHTTYPE);

float Temperature;

float Humidity;


int Moisture;

uint8_t MoisturePin = D1;

uint8_t RelayPin = D2;


#include <SoftwareSerial.h>

SoftwareSerial mySerial(D5, D6);

String phone_number = "9944398733";


String Python_data;

bool dataEnd = false;


int pState;


BLYNK_WRITE(V0)
{
    int value = param.asInt();
}

BLYNK_CONNECTED()
{
}

void myTimerEvent()
{
    Temperature = dht.readTemperature();
}

```

```

Humidity = dht.readHumidity();

Serial.print(Temperature);
Serial.print(",");
Serial.println(Humidity);

Blynk.virtualWrite(V1, Temperature);
Blynk.virtualWrite(V2, Humidity);

Moisture = digitalRead(MoisturePin);
Serial.println(Moisture);
if (Moisture == LOW && pState == HIGH) {
  Blynk.virtualWrite(V3, "Motor OFF");
  digitalWrite(RelayPin, LOW);
  pState = LOW;
}
if (Moisture == HIGH & pState == LOW) {
  Blynk.virtualWrite(V3, "Motor ON");
  digitalWrite(RelayPin, HIGH);
  pState = HIGH;
}

readPython();
}

void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(10000L, myTimerEvent);

  pinMode(DHTPin, INPUT);

```

```

pinMode(MoisturePin, INPUT);

pinMode(RelayPin, OUTPUT);

dht.begin();

mySerial.begin(9600);

//send_msg("System started", phone_number);

Blynk.notify("System Started");
}

void loop()
{
  Blynk.run();
  timer.run();
}

void send_msg(String msg, String phone_number) {
  mySerial.flush();
  Serial.println("Sending SMS");
  Serial.println(msg);
  mySerial.println("AT+CMGF=1");
  delay(2000);
  mySerial.println("AT+CMGS=\"+91\" + String(phone_number) + \"\\r\"");
  delay(2000);
  mySerial.println(msg);
  delay(100);
  mySerial.println((char)26);
  delay(2000);
}

void readPython() {
  while (Serial.available()) {
    char c = (char)Serial.read();

    //Serial.print(c);

    if (c == ';') {
      dataEnd = true;
    }
  }
}

```

```

    Serial.print(Python_data);

    Serial.println();

    break;
}

Python_data = Python_data + c ;
}

if (dataEnd == true) {
    Blynk.virtualWrite(V3, Python_data);

    Blynk.notify(Python_data);

    //send_msg(Python_data, phone_number);

    Python_data = "";

    dataEnd = false;
}
}

String getValue(String data, char separator, int index)
{
    int found = 0;
    int strIndex[] = { 0, -1 };
    int maxIndex = data.length() - 1;

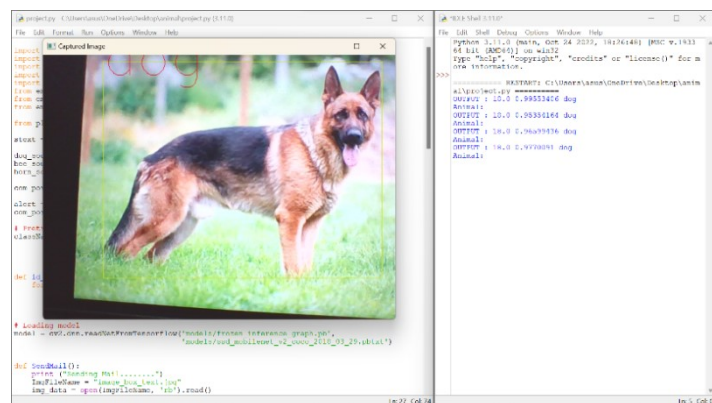
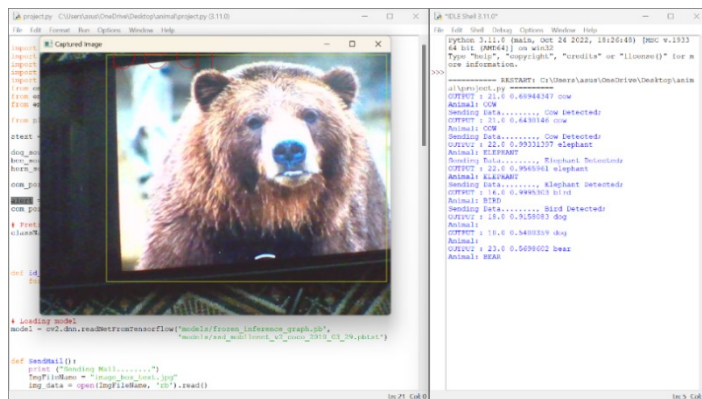
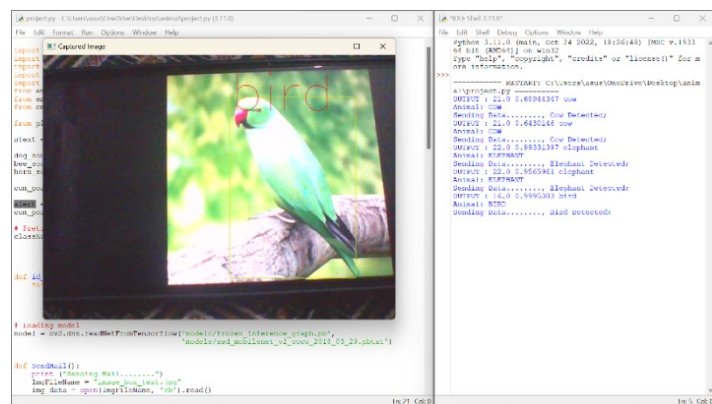
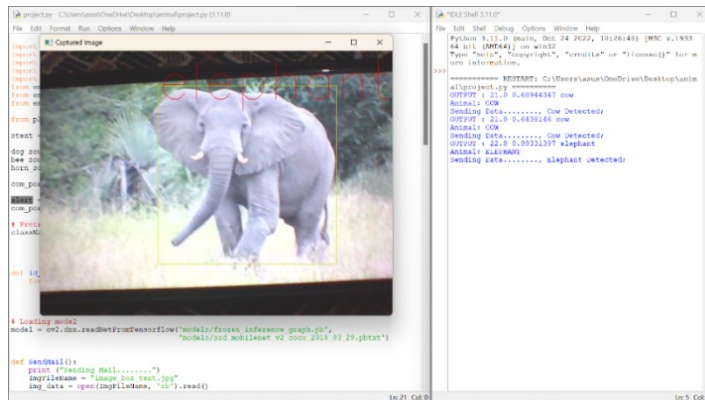
    for (int i = 0; i <= maxIndex && found <= index; i++) {
        if (data.charAt(i) == separator || i == maxIndex) {
            found++;
            strIndex[0] = strIndex[1] + 1;
            strIndex[1] = (i == maxIndex) ? i + 1 : i;
        }
    }

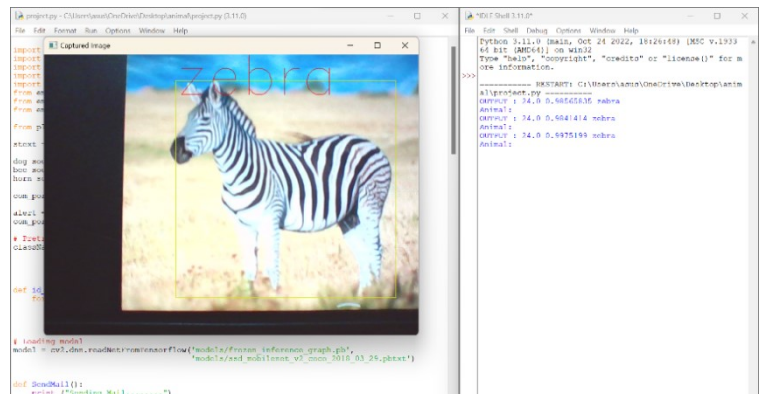
    return found > index ? data.substring(strIndex[0], strIndex[1]) : "";
}

```

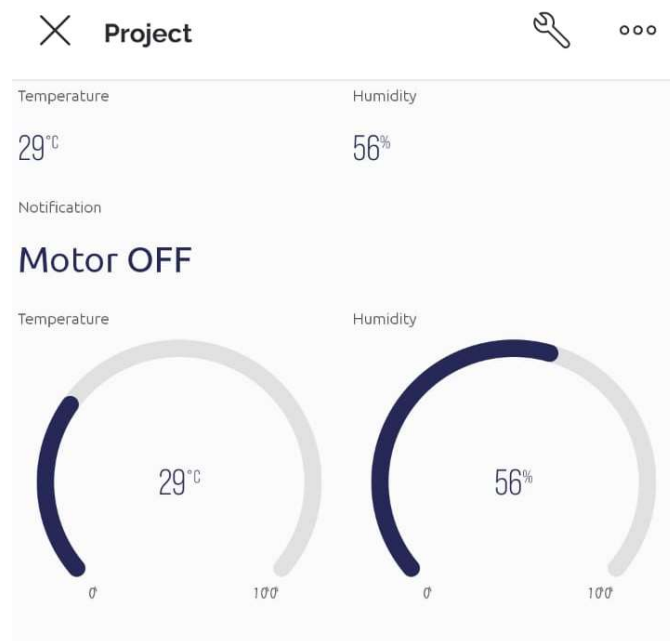
B. SCREENSHOTS

OUTPUT IMAGES OF THE ANIMAL DETECTION





OUTPUT IMAGE NOTIFICATIONS OF IRRIGATION SYSTEM



These images are the notifications shown in the blynk app when the motor is on and off during the measure of the soil moisture level

