AI BASED SCARECROW

A Mini Project Report Submitted
In partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering

by

Muda Vennela **Roll No:20N31A05F6**
Mohammad Johnny Pasha **Roll No:20N31A05F0**

Under the esteemed guidance of

B.SARITHA
Assistant Professor

Department of Computer Science and Engineering

Malla Reddy College of Engineering & Technology
(Autonomous Institution- UGC, Govt. of India)
(Affiliated to JNTUH,Hyderabad,Approved by AICTE,NBA &NAAC with'A'Grade)
Maisammaguda, Kompally, Dhulapally, Secunderabad –
500100website: www.mrcet.ac.in

2020-2024

Malla Reddy College of Engineering & Technology

(Autonomous Institution- UGC, Govt. of India)
(Affiliated to JNTUH, Hyderabad,Approved by AICTE,NBA &NAAC with'A'Grade)
Maisammaguda, Kompally, Dhulapally, Secunderabad – 500100
website: www.mrcet.ac.in

CERTIFICATE

This is to certify that this is the bonafide record of the project entitled "AI BASED SCARECROW", submitted by M.Vennela(20N31A05F6) and Md.Johnny Pasha(20N31A05F0) of B. Tech in the partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering, Department of CSE during the year 2022-2023. The results embodied in this project report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide

B.Saritha
Assistant Professor

Head of the Department

Dr. S. Shanthi
Professor

External Examiner
Dr. U Mohan Srinivas

DECLARATION

We hereby declare that the project titled "**AI BASED SCARECROW**" submitted to Malla Reddy College of Engineering and Technology (UGC Autonomous), affiliated to Jawaharlal Nehru Technological University Hyderabad (JNTUH) for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a result of original research carried-out in this thesis. It is further declared that the project report or any part thereof has not been previously submitted to any University or Institute for the award of degree.

M. Vennela (20N31A05F6)
Md. Johnny Pasha (20N31A05F0)

ACKNOWLEDGEMENT

We feel ourselves honored and privileged to place our warm salutation to our college Malla Reddy College of Engineering and Technology (UGC-Autonomous) and our principal

Dr. S. Srinivasa Rao who gave us the opportunity to have experience in engineering and profound technical knowledge.

We express our heartiest thanks to our Head of the Department **Dr. S. Shanthi** for encouraging us in every aspect of our project and helping us realize our full potential.

We would like to thank our internal guide **Ms.B. Saritha**, Assistant Professor and Project Coordinator **Mr. Sandeep Agarwalla**, Associate Professor for their regular guidance and constant encouragement. We are extremely grateful to her valuable suggestions and unflinching co- operation throughout project work.

We would like to thank our Class Incharge **G .Ravi** who in spite of being busy with his duties took time to guide and keep us on the correct path.

We would also like to thank all the supporting staff of the Department of CSE and all other departments who have been helpful directly or indirectly in making our project a success.

We are extremely grateful to our parents for their blessings and prayers for the completion of our project that gave us strength to do our project.

M.VENNELA (20N31A05F6)
MD.JOHNNY PASHA (20N31A05F0)

ABSTRACT

Crop damage caused by animal attacks is one of the major threats in reducing the crop yield. Crops in farms are many times ravaged by local animals like buffalos, cows, goats, birds etc. This leads to huge losses for the farmers. It is not possible for farmers to barricade entire fields or stay on field 24 hours and guard it. The existing systems mainly provide the surveillance functionality. They also need to take actions based on the type of animal that tries to enter the area, as different animals from entering such restricted areas. The other commonly used methods by the farmers in order to prevent the crop vandalization by animalsinclude building physical barriers, use of electric fences and manual surveillance and various such exhaustive and dangerous methods. Also, the farmers resort to the other methods by erecting human puppets and effigies in their farms, which is ineffective in warding off the wild animals, though is useful to some extent to ward off birds. So here we propose an AI based Scarecrow that protects the crops from wild animals with the help of scanning using camera, it detects the stray animals or birds and when it detects the stray animals or birds thenit produces a sound of animal extermination. We make a program with the help of live video detecting object using yolov3, coco names, cv2 modules. This, ensures complete safety of crops from animals causing damage to it.

Keywords:

YoloV3, Python, NumPy, OpenCV, PlaySound, SqLite

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF OUTPUTS

1. INTRODUCTION

This chapter gives an overview about the purpose, aim, objectives, background and operation environment of the system.

1.1 PURPOSE, AIM AND OBJECTIVES:

Purpose:

In agriculture one of the major social Problems that is existing in the present is the damaging of the crops by the wild animals. Some of the animals in South India that act as a threat to crops are deer, monkey, elephant and others. This problem must be attended immediately and an effective solution must be created and accomplished. Thus, this project aims to address this problem.

Animal attacks in India are a common story nowadays. Due to the unavailability of any detection system these attacks destroy their crops. Due to lack of proper safety measures, these villagers are left helpless to their fate. Also, the crops of villagers are destroyed due to frequent interference of animals. The crops and paddy fields cannot be always fenced. So the possibility of crops being eaten away by cows and goats are very much present. This could result in huge wastage of crops produced by the farmers. Animals such as deer, wild boars, rabbits, moles, elephants, monkeys, and many others may cause serious damage to crops. They can damage the plants by feeding on plant parts or simply by running over the field and trampling over the crops.

The foraging activities of cropland bird species like House Crow have caused more damage to wheat, while pigeons and doves cause damage to pearl millet and sunflower. Also, the parakeets and crows were found to inflict more damage to the crops than what they actually consumed.

Aim:

The main aim of the project is to detect and recognize the animals that we have stored priorlyin the database.

Objectives:

Our objective is to build a crop protection from animals using Artificial Intelligence. This system is called “AI Based Scarecrow”. This system detects the animals entering into the crop field and plays an extermination sound that is most feared by the animal. We are using YoloV3 Algorithm that is a Real time live object detection algorithm that detects the animals in the live video. cv2 module is used to capture video. PlaySound module is used to play extermination sound. We are using coco.names data set for identifying animals and their names if detected by YoloV3 Algorithm. Its accuracy is more compared to existing systems.

1.2 BACKGROUND OF PROJECT:

Manual way such as constructing different kinds of fences and using natural repellents are effective but they are not cost efficient. It is also not possible to increase the man power. So, initial projects were taken up to drive away the animals automatically by using hardware components like controllers and sensors. One such approaches camera interfaced to the Raspberry pi module. Camera is used to capture an image of wild animal and send captured image to the Raspberry pi module. When image can take by the Raspberry pi and compared with database image. After comparing, if the wild animal is detected then it gives commands to GSM module. GSM used to send message to the owner of the farm. This, system uses a motion sensor to detect wild animals approaching near the field. In such a case the sensor signals the microcontroller to take action. The microcontroller now sounds an alarm to drive away animals from the field as well as SMS send to the farmer so that the farmer may know about the issue. In recent times, researches are taken to solve this problem using Artificial Intelligence.

1.3 SCOPE OF PROJECT:

Our system mainly uses motion sensor to detect wild animals approaching near the field. In such a case the sensor signals the microcontroller to take action. The microcontroller now sounds an alarm to drive away animals from the field as well as SMS send to the farmer so that the farmer may know about the issue. In recent times, researches are taken to solve this problem using Artificial Intelligence.

1.4 MODULES DESCRIPTION:

This project is composed of three main modules which also include sub modules:

1. **OpenCV**: OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. OpenCV will help you know about the Image-processing from Basics to Advance, like operations on Images, Videos using a huge set of OpenCV programs and projects. The CV component contains mainly the basic image processing and higher-level computer vision algorithms; MLL the machine learning library includes many statistical classifiers as well as clustering tools.
2. **Numpy**: Numpy is core library of scientific computing in python. It provides powerful tools to deal with various multi-dimensional arrays in python. NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. NumPy is open-source software and has many contributors. Mathematical algorithms are written for this version of Python often run much slower than compiled equivalents. NumPy addresses the slowness problem partly by providing multidimensional arrays and functions and operators that operate efficiently on arrays, requiring rewriting some code, mostly inner loops using NumPy. They allow the user to write fast programs if most operation work on arrays or matrices instead of scalars. The NumPy array as a universal data structure in OpenCV for images, extracted feature points, filter kernels and many more vastly simplifies the programming workflow and debugging.
3. **PlaySound**: The playsound module contains – the function playsound. It requires one argument – the path to the file with the sound you'd like to play. This may be a local file, or a URL. We can set it to False for making the function run asynchronously. There's an optional second argument, block, which is set to True by default. Setting it to False makes the function run asynchronously. On Windows, uses WAVE and MP3 have been tested and are known to work. Other file formats may work as well.

2. SYSTEM ANALYSIS

In this chapter, we will discuss and analyze about the developing process of AI Based Scarecrow including software requirement specification (SRS) and comparison between existing and proposed system. The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs. proposed provides a view of how the proposed system will be more efficient than the existing one.

2.1 HARDWARE AND SOFTWARE REQUIREMENTS

2.1.1 HARDWARE REQUIREMENTS:

Monitor	**:**14’ color monitor
Mouse	**:** Optical Mouse
RAM	**:** 4 GB
Tools Used	**:** Integrated camera
System	**:** Intel Core i5 7th gen

2.1.2 SOFTWARE REQUIREMENTS:

Operating System	**:** Windows 10
Coding Language	**:** Python
IDE Tool	**:** PyCharm
Packages	**:** OpenCV, NumPy, PlaySound

2.2 SOFTWARE REQUIREMENT SPECIFICATION:

2.2.1 SRS:

Software Requirement Specification (SRS) is the starting point of the software developing activity. As system grew more complex it became evident that the goal of the entire system cannot be easily comprehended. Hence the need for the requirement phase arose. The software project is initiated by the client needs. The SRS is the means of translating the ideas of the minds of clients (the input) into a formal document (the output of the requirement phase.)

The SRS phase consists of two basic activities:

1) **Problem/Requirement Analysis:**

The process is order and more nebulous of the two, deals with understand the problem, the goal and constraints.

Requirement Specification:

Here, the focus is on specifying what has been found giving analysis such as representation, specification languages and tools, and checking the specifications are addressed during this activity.

The Requirement phase terminates with the production of the validate SRS document. Producing the SRS document is the basic goal of this phase.

2.2.2 ROLE OF SRS:

The purpose of the Software Requirement Specification is to reduce the communication gap between the clients and the developers. Software Requirement Specification is the medium through which the client and user needs are accurately specified. It forms the basis of software development. A good SRS should satisfy all the parties involved in the system.

2.2.3 SCOPE:

This document is the only one that describes the requirements of the system. It is meant for the use by the developers, and will also be the basis for validating the final delivered system. Any changes made to the requirements in the future will have to go through a formal change approval process. The developer is responsible for asking for clarifications, where necessary, and will not make any alterations without the permission of the client.

2.2.4 EXISTING SYSTEM:

Manual way such as constructing different kinds of fences and using natural repellents and effective but they are not cost efficient. It is also not possible to increase the man power. So, initial projects were taken up to drive away the animals automatically by using hardware components like controllers and sensors.

One such approaches camera interfaced to the Raspberry pi module. Camera is used to capture an image of wild animal and send captured image to the Raspberry pi module. When image can take by the Raspberry pi and compared with database image. After comparing, if the wild animal is detected then it gives commands to GSM module. GSM used to send message to the owner of the farm. The problem here is there is more of hardware componentswhich is not cost efficient and its maintenance is also hard.

2.2.4.1 DRAWBACKS OF EXISTING SYSTEM:

- Electric fences are dangerous to animals and humans.
- IOT based Sensor monitoring doesn't provide accurate results

2.2.5 PROPOSED SYSTEM:

Our goal is to build a crop protection from animals using Artificial Intelligence. This system is called "AI Based Scarecrow". This system detects the animals entering into the crop field and plays an extermination sound that is most feared by the animal. We are using YoloV3 Algorithm that is a Real time live object detection algorithm that detects the animals in the live video. cv2 module is used to capture video. Playsound module is used to play extermination sound. We are using coco.names data set for identifying animals and their names if detected by YoloV3 Algorithm. Its accuracy is more compared to existing systems.

3. TECHNOLOGIES USED

3.1 PYTHON:

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

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging. Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace**.**

A source level debugger allows inspection of local and global variables, evaluation ofarbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

3.2 MACHINE LEARNING:

Machine learning is a branch of Artificial Intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. One of its own, Arthur Samuel, is credited for coining the term, "machine learning" with his research around the game of checkers. Robert Nealey, the self-proclaimed checkers master, played the game on an IBM 7094 computer in 1962, and he lost to the computer. Compared to what can be done today, this feat almost seems trivial, but it's considered a major milestone within the field of artificial intelligence. Over the next couple of decades, the technological developments around storage and processing power will enable some innovative products that we know and love today, such as Netflix's recommendation engine or self-driving cars. Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data mining projects. These insights subsequently drive decision making within applications and businesses, ideally impacting key growth metrics. As big data continues to expand and grow, the market demand for data scientists will increase, requiring them to assist in the identification of the most relevant business questions and subsequently the data to answer them.

3.2.1. MACHINE LEARNING VS DEEP LEARNING VS NEURAL NETWORKS:

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

The way in which deep learning and machine learning differ is in how each algorithm learns. Deep learning automates much of the feature extraction piece of the process, eliminating some of the manual human intervention required and enabling the use of larger data sets.

"Deep" machine learning can leverage labeled datasets, also known as supervised learning, to inform its algorithm, but it doesn't necessarily require a labeled dataset. It can ingest unstructured data in its raw form (e.g., text, images), and it can automatically determine the set of features which distinguish different categories of data from one another. Unlike machine learning, it doesn't require human intervention to process data, allowing us to scale machine learning in more interesting ways. Deep learning and neural networks are primarily credited with accelerating progress in areas, such as computer vision, natural language processing, and speech recognition.

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

3.2.3 MACHINE LEARNING METHODS?

Machine learning classifiers fall into three primary categories.

Supervised machine learning

Supervised Learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately. This occurs as part of the cross validation process to ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve for a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, support vector machine (SVM), and more.

Unsupervised machine learning

Unsupervised Learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction; principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, probabilistic clustering methods, and more.

Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data (or not being able to afford to label enough data) to train a supervised learning algorithm.

OpenCV

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

SQLite

SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects.

SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format. SQLite database files are a recommended storage format by the US Library of Congress. Think of SQLite not as a replacement for Oracle but as a replacement for fopen()

4. SYSTEM DESIGN & UML DIAGRAMS

System design is transition from a user-oriented document to programmers or data base personnel. The design is a solution, how to approach to the creation of a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Designing goes through logical and physical stages of development, logical design reviews the present physical system, prepare input and output specification, details of implementation plan and prepare a logical design walkthrough.

4.1 SOFTWARE DESIGN:

In designing the software following principles are followed:

1) **Modularity and partitioning**: Software is designed such that, each system shouldconsist of hierarchy of modules and serve to partition into separate function.
2) **Coupling:** Modules should have little dependence on other modules of a system.
3) **Cohesion:** Modules should carry out in a single processing function.
4) **Shared use:** Avoid duplication by allowing a single module be called by other thatneed the function it provides.

4.2 ARCHITECTURE:

Architecture diagram is a diagram of a system, in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. The block diagram is typically used for a higher level, less detailed description aimedmore at understanding the overall concepts and less at understanding the details of implementation.

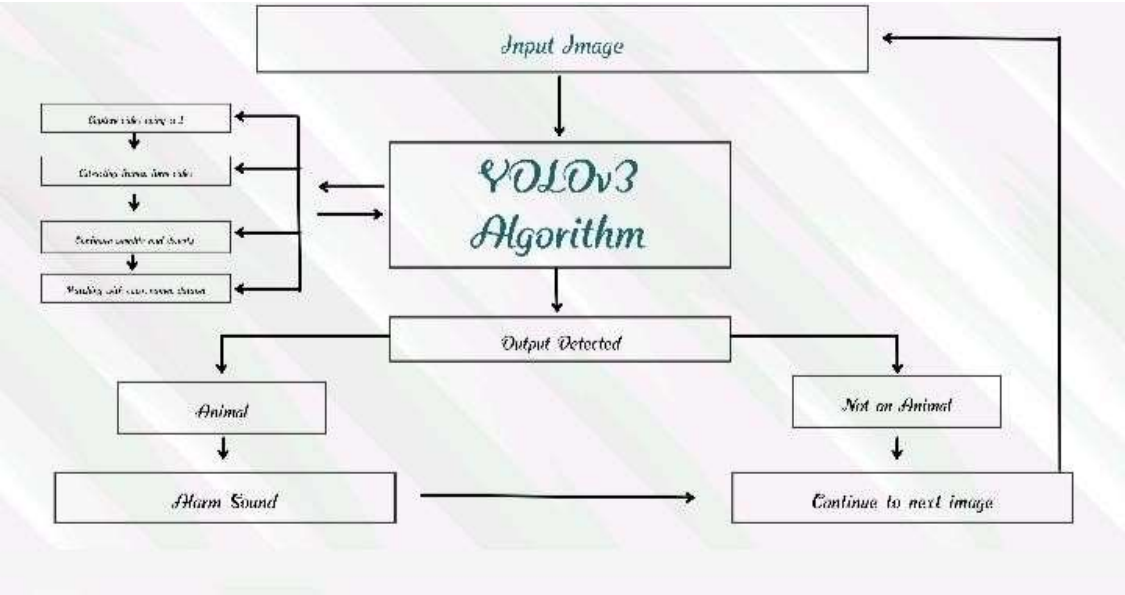

FIGURE 4.2: ARCHITECTURE DIAGRAM

4.3 UNIFIED MODELING LANGUAGE (UML) :

The unified modeling is a standard language for specifying, visualizing, constructing and documenting the system and its components is a graphical language which provides a vocabulary and set of semantics and rules. The UML focuses on the conceptual and physical representation of the system. It captures the decisions and understandings about systems that must be constructed. It is used to understand, design, configure and control information aboutthe systems.

Depending on the development culture, some of these artifacts are treated more or less formally than others. Such artifacts are not only the deliverables of a project; they are also critical in controlling, measuring, and communicating about a system during its development and after its deployment.

The UML addresses the documentation of a system's architecture and all of its details.The UML also provides a language for expressing requirements and for tests. Finally, the UML provides a language for modeling the activities of project planning and release management.

4.3.1 BUILDING BLOCKS OF UML:

The vocabulary of the UML encompasses three kinds of building blocks:

- ✓ Things.
- ✓ Relationships.
- ✓ Diagrams.

4.3.1.1 Things in the UML:

Things are the abstractions that are first-class citizens in a model; relationships tie these things together; diagrams group interesting collections of things.

There are four kinds of things in the UML:

- ✓ Structural things.
- ✓ Behavioral things.
- ✓ Grouping things.
- ✓ Annotational things.

Structural things are the nouns of UML models. The structural things used in the project design are:

- ✓ First, a **class** is a description of a set of objects that share the same attributes,operations, relationships and semantics.

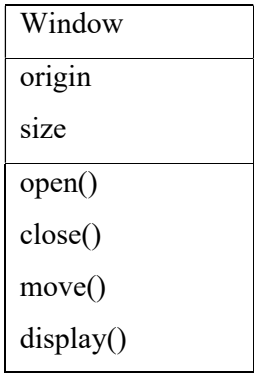

Fig: Classes

- ✓ Second, a **use case** is a description of set of sequence of actions that a systemperforms that yields an observable result of value to particular actor.

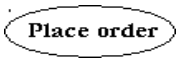

Fig: Use Cases

- ✓ Third, a node is a physical element that exists at runtime and represents a computational resource, generally having at least some memory and often processing capability.

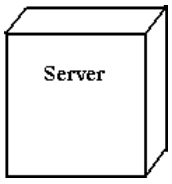

Fig: Nodes

1. **Behavioral things** are the dynamic parts of UML models. The behavioral thing usedis:
   - ✓ Interaction: An interaction is a behavior that comprises a set of messagesexchanged among a set of objects within a particular context to accomplish a

specific purpose. An interaction involves a number of other elements, including messages, action sequences (the behavior invoked by a message, andlinks (the connection between objects).

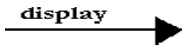

Fig: Messages

4.3.1.2 Relationships in the UML:

There are four kinds of relationships in the UML:

- ✓ Dependency.
- ✓ Association.
- ✓ Generalization.
- ✓ Realization.

✓ A **dependency** is a semantic relationship between two things in which a change to one thing may affect the semantics of the other thing (the dependent thing).

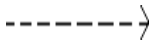

Fig: Dependencies

✓ An **association** is a structural relationship that describes a set links, a link being a connection among objects. Aggregation is a special kind of association, representing astructural relationship between a whole and its parts.

Fig: Association

✓ A **generalization** is a specialization/ generalization relationship in which objects of the specialized element (the child) are substitutable for objects of the generalized element (the parent).

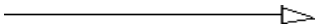

Fig: Generalization

- ✓ A **realization** is a semantic relationship between classifiers, where in one classifierspecifies a contract that another classifier guarantees to carry out.

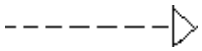

Fig: Realization

UML DIAGRAMS:

4.3.2.1 USE CASE DIAGRAM:

A use case diagram is a graph of actors set of use cases enclosed by a system boundary, communication associations between the actors and users and generalization among use cases. The use case model defines the outside (actors) and inside (use case) of the system's behavior.

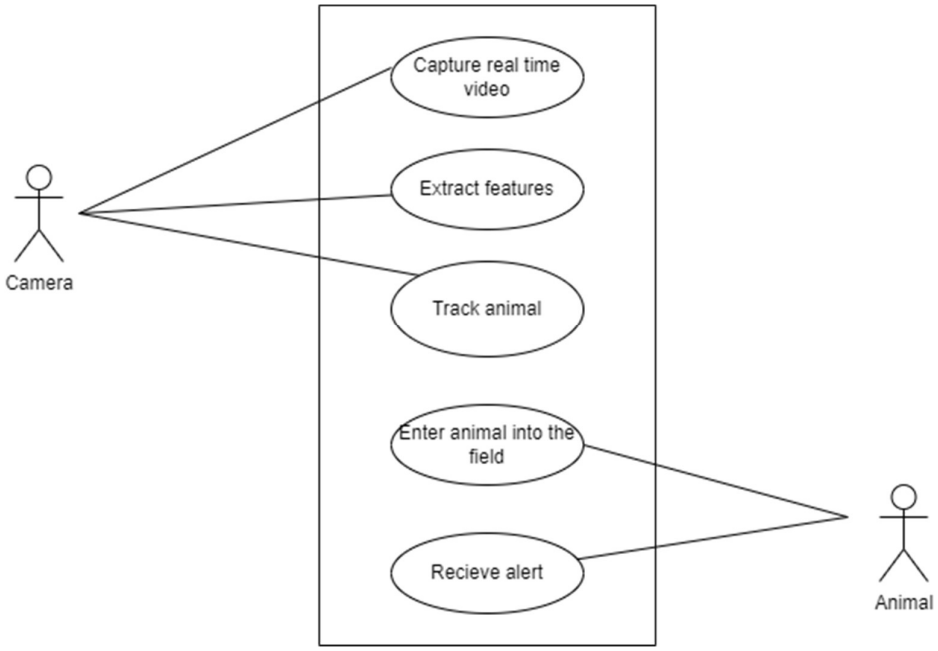

FIGURE 4.3.2.1: USE CASE DIAGRAM

4.3.2.2 SEQUENCE DIAGRAM:

Sequence diagram are used to represent the flow of messages, events and actions between the objects or components of a system. Time is represented in the vertical direction showing the sequence of interactions of the header elements, which are displayed horizontally at the top of the diagram.

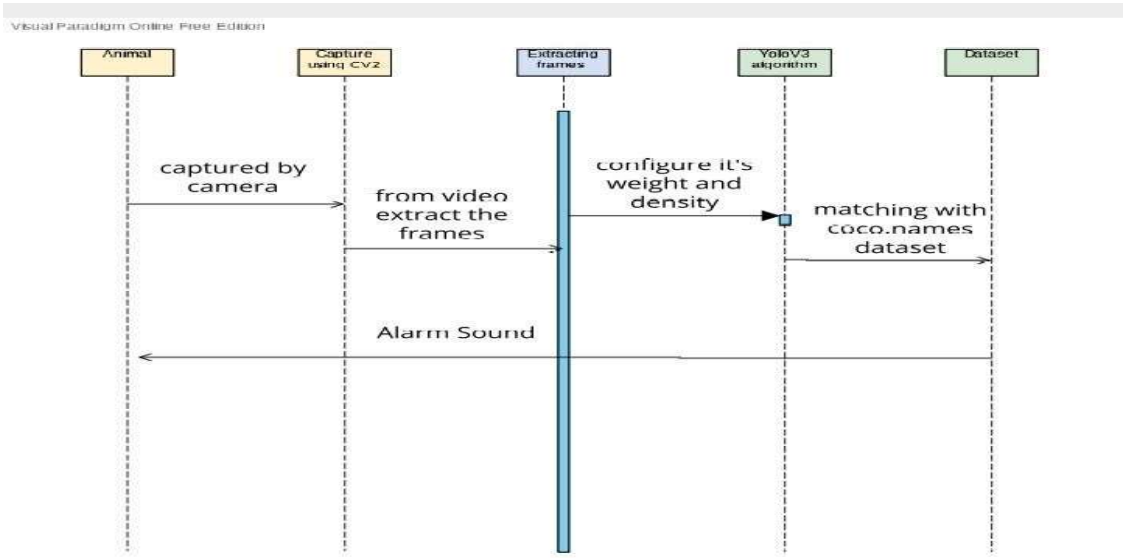

FIGURE 4.3.2.2: SEQUENCE DIAGRAM

4.3.2.4 ACTIVITY DIAGRAM:

Activity diagram represent the business and operational workflows of a system. An Activity diagram is a dynamic diagram that shows the activity and the event that causes the object to be in the particular state.

So, what is the importance of an Activity diagram, as opposed to a State diagram? A State diagram shows the different states an object is in during the lifecycle of its existence in the system, and the transitions in the states of the objects. These transitions depict the activities causing these transitions, shown by arrows.

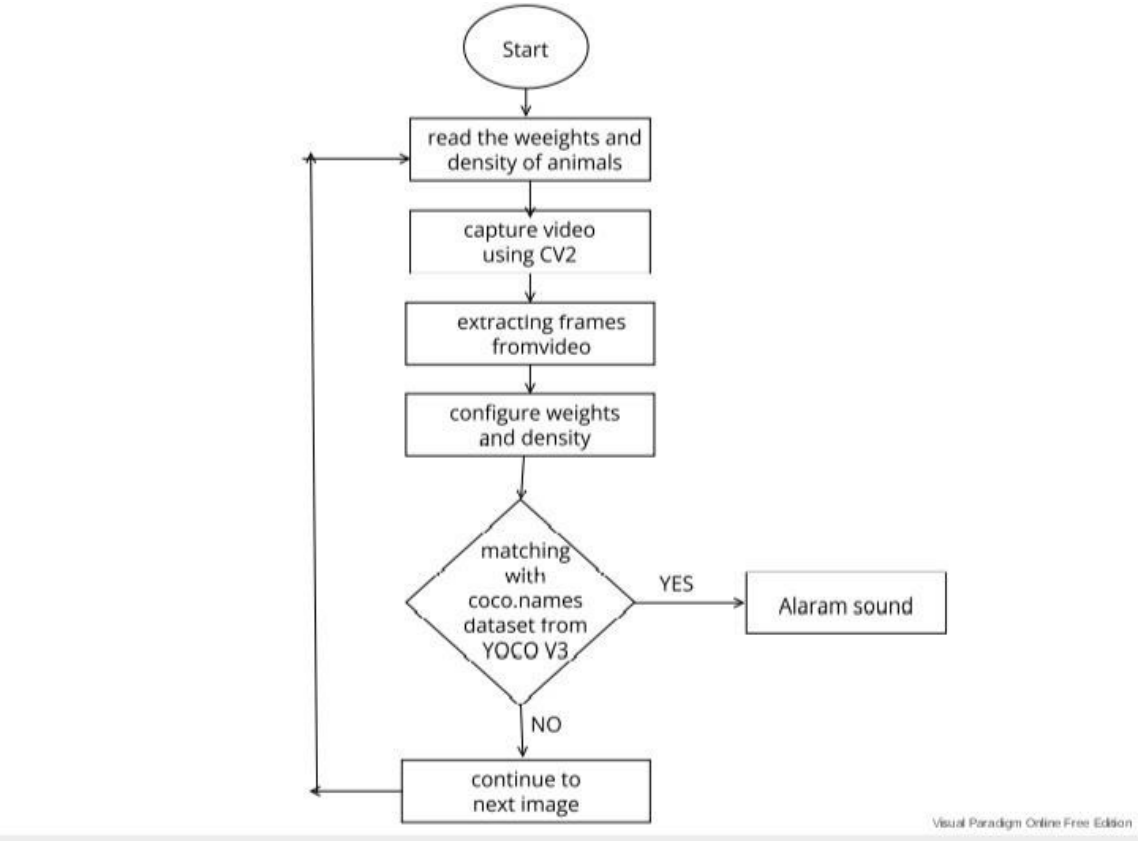

FIGURE 4.3.2.4: ACTIVITY DIAGRAM

4.3.2.5 STATE CHART DIAGRAM:

State chart diagram is used to describe the states of different objects in its life cycle. So the emphasis is given on the state changes upon some internal or external events. These states of objects are important to analyse and implement them accurately. State chart diagrams are very important for describing the states. States can be identified as the conditionof objects when a particular event occurs.

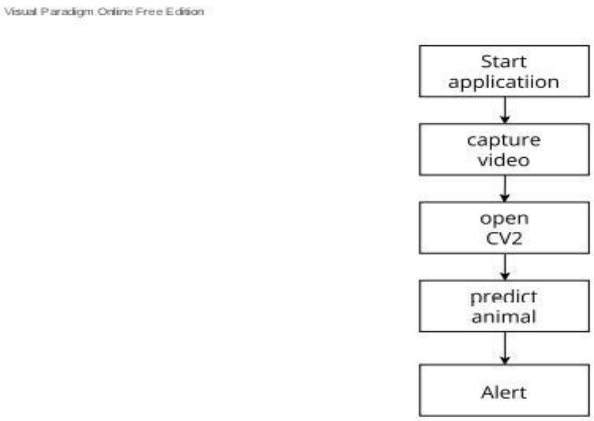

FIGURE 4.3.2.5: STATE CHART DIAGRAM

.

5. INPUT/OUTPUT DESIGN

5.1 INPUT DESIGN:

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations. Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

5.2 OUTPUT DESIGN:

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself.

6. IMPLEMENTATION

```
import
pygameimport
cv2
import numpy as np
import pygame
from pygame import mixer
from playsound import
playsoundimport os

cap =
cv2.VideoCapture(0)wht
= 320
confThreshold = 0.5
nmsThreshold = 0.3

classesFile =
'coco.names'classNames
= []
with open(classesFile,'rt') as f:
    classNames =
f.read().rstrip('\n').split('\n')modelConfi =
'yolov3.cfg'
modelwei = 'yolov3.weights'
net = cv2.dnn.readNetFromDarknet(modelConfi,modelwei)
net.setPreferableBackend(cv2.dnn.DNN_BACKEND_OPENCV)
net.setPreferableTarget(cv2.dnn.DNN_TARGET_CPU)

def
    findobj(outputs,img):
    ht,wt,ct= img.shape
    bbox=[]
    classIds=[
    ]confs=[]

    for output in
```

```
outputs:for det in
output:
    scores = det[5:]
```

```
            classId =
            np.argmax(scores)
            confidence
            =scores[classId]
            if confidence > confThreshold:
                w,h=int(det[2]*wt),int(det[3]*ht)
                x,y = int((det[0]*wt)-w/2),int((det[1]*ht)-
                h/2)bbox.append([x,y,w,h])
                classIds.append(classId)
                confs.append(float(confidence))
    indices = cv2.dnn.NMSBoxes(bbox,confs,confThreshold,nmsThreshold)
    #print(indices)
    for i in indices:
        box = bbox[i]
        x,y,w,h = box[0],box[1],box[2],box[3]
        cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,0),2)
        cv2.putText(img,f'{classNames[classIds[i]].upper()} {int(confs[i]*100)}%',(x,y-
10),cv2.FONT_HERSHEY_SIMPLEX,0.9,(255,255,255),3)
        x=classNames[classIds[i]]
        if x=="horse" or x=="elephant" or x=="zebra" or x=="giraffe":

os.system('C:\\Users\\jeswa\\PycharmProjects\\Scarecrow\\sounds\\CrackersSound.mp3')
        elif x=="cat" or x=="dog" or x=="sheep":

os.system('C:\\Users\\jeswa\\PycharmProjects\\Scarecrow\\sounds\\LionRoarSound.mp3')
        elif x=="bird":
            os.system('C:\\Users\\jeswa\\PycharmProjects\\Scarecrow\\sounds\\OwlSound.mp3')
        elif x=="cow":

os.system('C:\\Users\\jeswa\\PycharmProjects\\Scarecrow\\sounds\\RunningSound.mp3')
        elif x=="bear":
            os.system('C:\\Users\\jeswa\\PycharmProjects\\Scarecrow\\sounds\\PanSound.mp3')
while True:
    outputnames = []
    success, img =
    cap.read()
```

```
blob=cv2.dnn.blobFromImage(img,1/255,(wht,wht),[0,0,0],1,crop=False)
```

```
    net.setInput(blob)

    layer_names = net.getLayerNames()
    #print(layer_names)
    #print(net.getUnconnectedOutLayers())
    outputnames = [layer_names[i - 1] for i in
    net.getUnconnectedOutLayers()]outputs = net.forward(outputnames)
    findobj(outputs,img)



    cv2.imshow('image',img)
    cv2.waitKey(1)
cap.release()
cv2.dertroyAllWindows(
)
```

7. TESTING

Software testing is a critical element of software quality assurance and represents theultimate review of specification, design and code generation.

7.1 TESTING OBJECTIVES:

- To ensure that during operation the system will perform as per specification.
- To make sure that system meets the user requirements during operation
- To make sure that during the operation, incorrect input, processing and output will bedetected
- To see that when correct inputs are fed to the system the outputs are correct
- To verify that the controls incorporated in the same system as intended
- Testing is a process of executing a program with the intent of finding an error
- A good test case is one that has a high probability of finding an as yet undiscoverederror

7.2 TESTING METHODOLOGIES:

- ✓ White box testing.
- ✓ Black box testing.
- ✓ Unit testing.
- ✓ Integration testing.
- ✓ User acceptance testing.
- ✓ Output testing.
- ✓ Validation testing.
- ✓ System testing.

1) White Box Testing:

White box testing is a testing case design method that uses the control structure of the procedure design to derive test cases. All independents path in a module are exercised at leastonce, all logical decisions are exercised at once, execute all loops at boundaries.

2) Black Box Testing:

Black Box Testing attempts to find errors in following areas or categories, incorrect or missing functions, interface error, errors in data structures, performance error and initialization and termination error. Here all the input data must match the data type to become a valid entry.

3)Unit Testing:

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

4)Integration Testing:

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

- ✓ **Top Down Integration:**

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module.

- ✓ **Bottom Up Integration:**

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs iseliminated.

5)User acceptance Testing:

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

6)Output Testing:

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

7)Validation Testing:

Validation testing is generally performed on the following fields:

- ✓ **Text Field:**

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

- ✓ **Numeric Field:**

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error message. The individual modules are checked for accuracy and what it has to perform.

- ✓ **Preparation of Test Data:**

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncoveredand corrected by using above testing steps and corrections are also noted for future use.

- ✓ **Using Live Test Data:**

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of datafrom their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the filesand have them entered themselves.

- ✓ **Using Artificial Test Data:**

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

7.3 USER TRAINING:

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

7.4 MAINTAINENCE:

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which willmake maintenance easier.

7.5 TESTING STRATEGY :

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements. Software testing is a critical element of software quality assurance and represents the ultimatereview of specification design and coding.

7.5.1 SYSTEM TESTING:

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

7.5.2 UNIT TESTING:

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

7.6 TEST CASES:

Test Case 1: Live animal recognition	**Priority (H, L):** High
Test Objective: To Recognize the animal and give alert	
Test Description: Detecting and extracting the features from the animal	
Requirements Verified: Yes	
Test Environment: Application can be used in the laptop or any system containing camera	
Test Setup/Pre-Conditions:	
Actions	Expected Results
The animal image is recognized	Successfully Saved
Pass: Yes **Conditions pass: Yes** **Fail**: No	
Problems / Issues: NIL	
Notes: Successfully Executed	

TABLE 7.6.1: ANIMAL RECOGNITION TEST CASE 1

Test Case 2: Live Animal recognition	**Priority (H, L):** High
Test Objective: To Recognize the animal and give alert	
Test Description: Detecting and extracting the features from the animal	
Requirements Verified: Yes	
Test Environment: Application can be used in the laptop or any system containing camera	
Test Setup/Pre-Conditions:	
Actions	Expected Results
The animal image is recognized	Successfully Saved
Pass: Yes **Conditions pass: Yes** **Fail**: No	
Problems / Issues: NIL	

TABLE 7.6.2: ANIMAL RECOGNITION TEST CASE 2

8. OUTPUT SCREENS

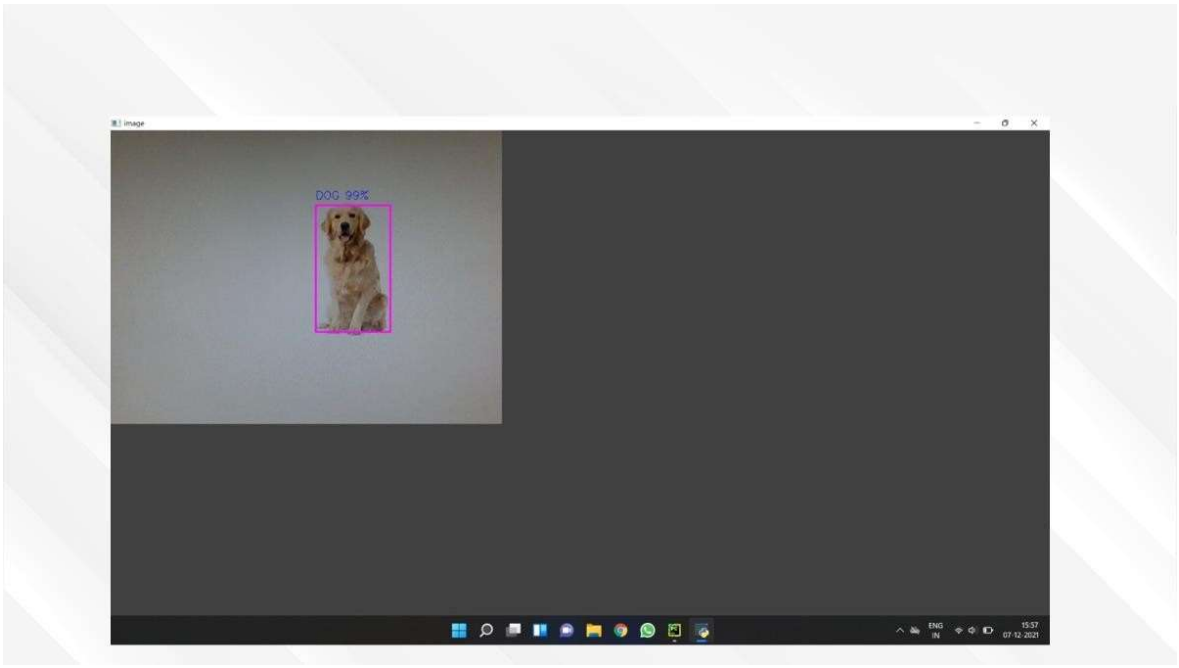

FIGURE 8.1.1:Animal Recognization1

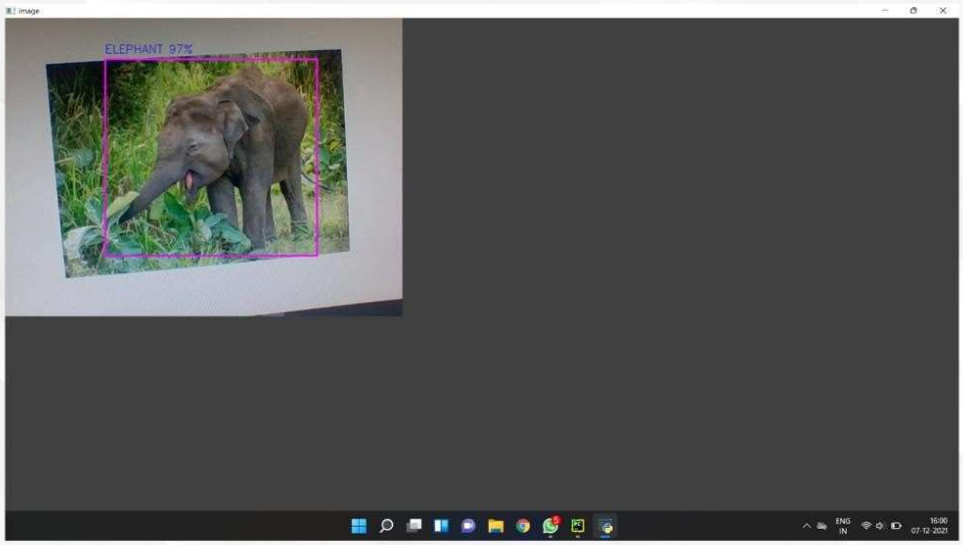

FIGURE 8.1.2:Animal Recogniozation2

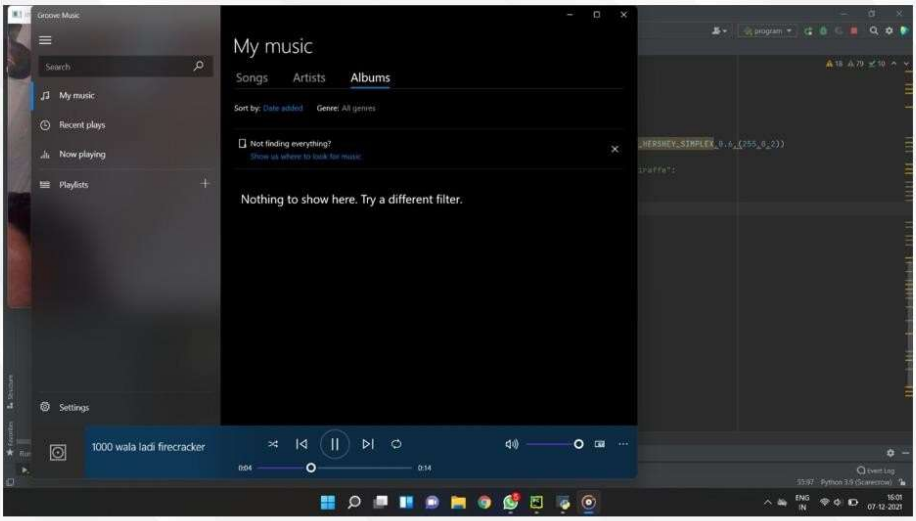

FIGURE 8.1.3:Play Sound Module

9. CONCLUSION & FUTURE SCOPE

9.1 CONCLUSION:

AI Based Scarecrow is a system that repels the wild animals that are trying to enter the field and exterminates them by playing the sound that they fear off. So, it can be concluded that we can recognize and reverse the animals before they enter the field by playing various repelling sounds. The problem of crop vandalization by wild animals has become a major social problem in current time. In other words, while utilizing his/her crop production, every farmer should be aware and take into consideration the fact that animals are living beings and need to be protected from any potential suffering. It requires urgent attention and an effective solution. By doing so, we reduce the crop loss and man power. This project is very useful andaffordable to the farmer. The module will not be dangerous to animal and human being, and it protects farm. Thus, this project carries a great social relevance as it will help farmers in protecting their fields and save them from significant financial losses and will save them from the unproductive efforts that they endure for the protection of their fields. This ensures complete safety of crops from animals causing damage to it.

9.2 FUTURE SCOPE:

We our using an integrative approach in the field of Artificial Intelligence to overcome this. The goal of this work is to provide a repelling and monitoring system for crop protection against animal attacks. In our future work, we will extend the current functionalities of a model like increasing the dataset so as to achieve high accuracy and investigate the chance of incorporating the future of the model to other sectors. It can be made into a robot and do the needful actions like moving hands and make it scare animals. It can be made to detect Human intrusion in order to stop robbery of crops

10. BIBLIOGRAPHY

10.1 WEBSITES:

- ✓ https://en.wikipedia.org/wiki/TensorFlow
- ✓ https://en.wikipedia.org/wiki/Convolutional_neural_nework
- ✓ Geeks for Geeks
- ✓ http://hunspell.github.io/
- ✓ http://keras.io/
- ✓ https://opencv.org/

10.2 REFERENCES:

- ✓ T. Yang, Y. Xu, and "A., Hidden Markov Model for Gesture Recognition", CMU-RI-TR-94 10, Robotics Institute, Carnegie Mellon Univ., Pittsburgh, PA, May 1994.
- ✓ Pujan Ziaie, Thomas M uller, Mary Ellen Foster, and Alois Knoll "A Na¨ive Bayes Munich, Dept. of Informatics VI, Robotics and Embedded Systems, Boltzmannstr. 3,DE-85748 Garching, Germany.
- ✓ https://docs.opencv.org/2.4/doc/tutorials/imgproc/gausian_median_blur_bilateral_filter/gausian_median_blur_bilateral_filter.html
- ✓ Mohammed Waleed Kalous, Machine recognition of Auslan signs usingPowerGloves: Towards large-lexicon recognition of sign language.
- ✓ aeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural Networks-Part-2/
- ✓ http://www-i6.informatik.rwth-aachen.de/~dreuw/database.php
- ✓ Pigou L., Dieleman S., Kindermans PJ., Schrauwen B. (2015) Sign Language Recognition Using Convolutional Neural Networks. In: Agapito L., Bronstein M.,Rother C. (eds) Computer Vision - ECCV 2014 Workshops. ECCV 2014. LectureNotes in Computer Science, vol 8925. Springer, Cham
- ✓ Zaki, M.M., Shaheen, S.I.: Sign language recognition using a combination of newvision-based features. Pattern Recognition Letters 32(4), 572–577 (2011).