Vellore Institute of Technology, Vellore

Embedded Systems (BCSE305L)

Digital Assignment 2

Faculty: Prof. Arunkumar T

Topic:

Camera Trap IoT System for Wildlife Monitoring

NAME: SATYAM KUMAR
REG NO: 22BCE3070

Index:

1.Introduction

This project focuses on developing an IoT-based camera trap system using **Raspberry Pi, a PIR sensor, and YOLOv8 for object detection**. The system detects motion using a PIR sensor, captures images and live video, processes them for object identification, and sends alerts via Telegram when specific objects (humans or animals) are detected.

2. System Components

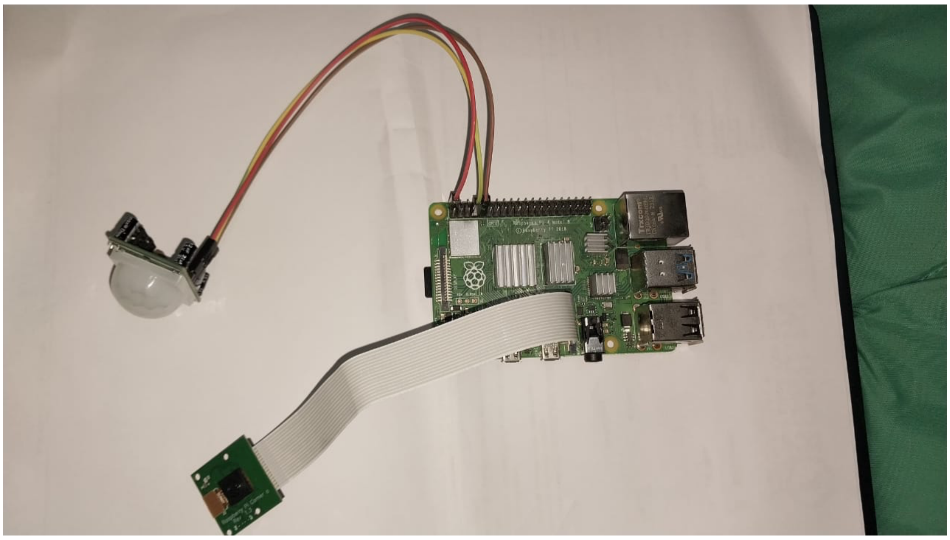
Hardware Requirements:

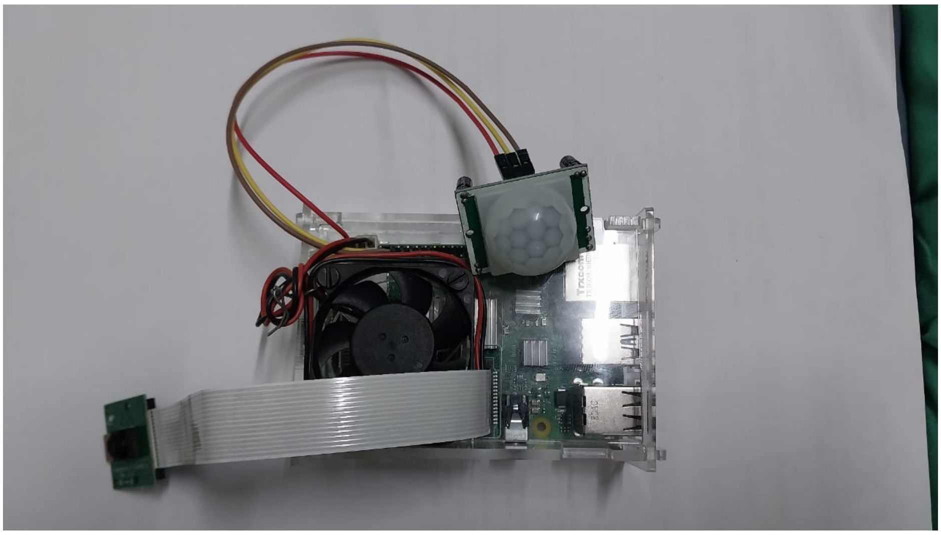
- Raspberry Pi (Model 4B)
- Raspberry Pi Camera Module
- PIR Motion Sensor (Connected to GPIO 4)
- Power Supply
- SD Card with Raspberry Pi OS

Software Requirements:

- Python
- OpenCV
- YOLOv8 (Ultralytics)
- Telegram API for alerts
- Libcamera for video streaming
- GPIO library for PIR sensor

3. Circuit diagram





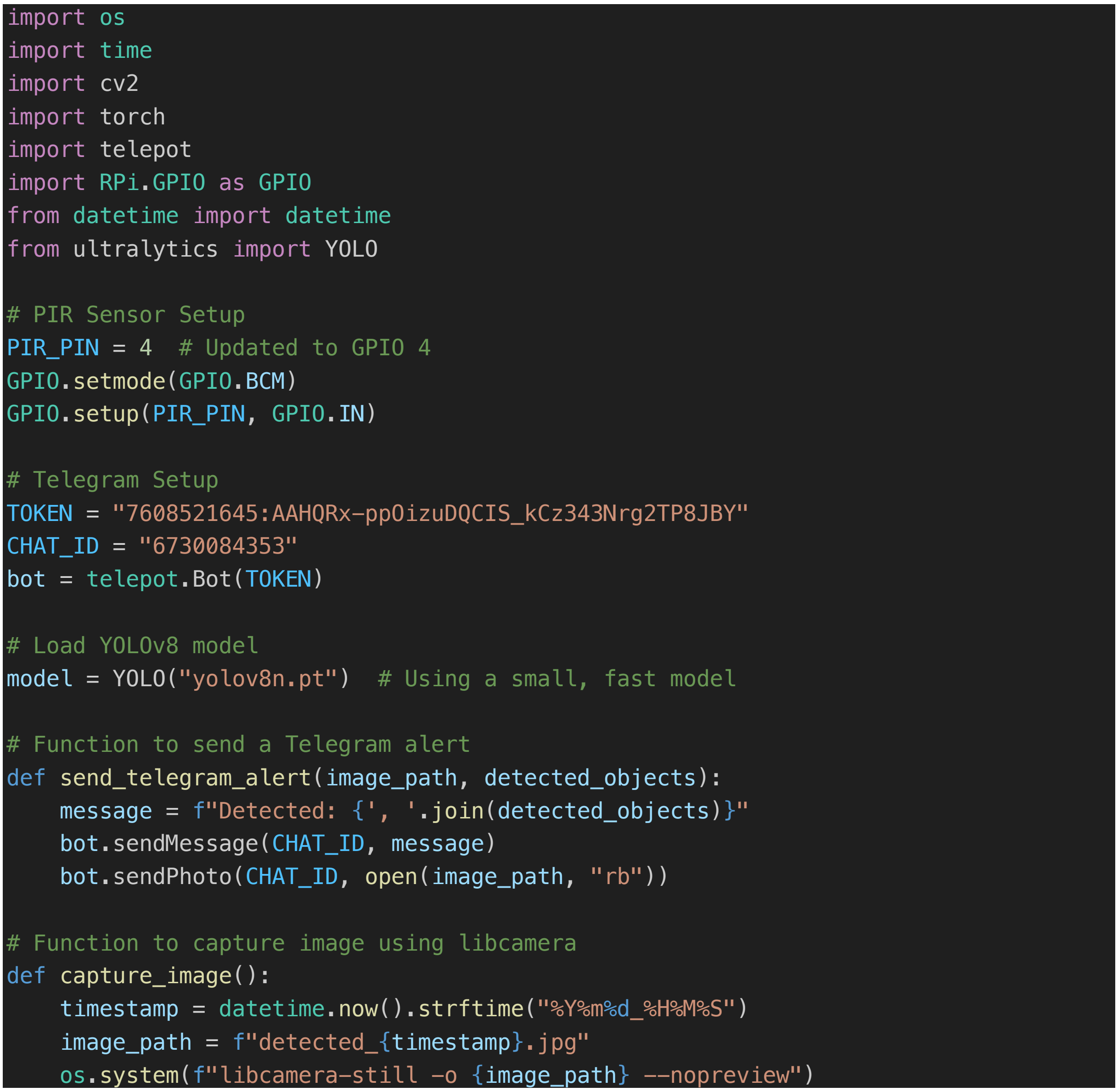
1. Connect the PIR sensor:
   - **VCC** to Raspberry Pi **5V**
   - **GND** to Raspberry Pi **GND**
   - **OUT** to **GPIO 4**
2. Attach the Raspberry Pi Camera Module.
3. Ensure the Raspberry Pi has an active internet connection for Telegram alerts.

4. Code Implementation

Motion Detection & Object Identification Code:

- The system continuously monitors motion using the PIR sensor.
- Upon detecting motion, it captures an image using **libcamera**.
- The captured image is processed using **YOLOv8** for object identification.
- If an animal or human is detected, an image is saved, and an alert is sent to Telegram with the identified object type.

Project Source Code



```python
import os
import time
import cv2
import torch
import telepot
import RPi.GPIO as GPIO
from datetime import datetime
from ultralytics import YOLO

# PIR Sensor Setup
PIR_PIN = 4  # Updated to GPIO 4
GPIO.setmode(GPIO.BCM)
GPIO.setup(PIR_PIN, GPIO.IN)

# Telegram Setup
TOKEN = "7608521645:AAHQRx-ppOizuDQCIS_kCz343Nrg2TP8JBY"
CHAT_ID = "6730084353"
bot = telepot.Bot(TOKEN)

# Load YOLOv8 model
model = YOLO("yolov8n.pt")  # Using a small, fast model

# Function to send a Telegram alert
def send_telegram_alert(image_path, detected_objects):
    message = f"Detected: {', '.join(detected_objects)}"
    bot.sendMessage(CHAT_ID, message)
    bot.sendPhoto(CHAT_ID, open(image_path, "rb"))

# Function to capture image using libcamera
def capture_image():
    timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
    image_path = f"detected_{timestamp}.jpg"
    os.system(f"libcamera-still -o {image_path} --nopreview")
    return image_path

# Main loop
print("Waiting for motion...")
while True:
    if GPIO.input(PIR_PIN):  # Motion detected
        print("Motion detected! Capturing image...")
        image_path = capture_image()

        # Run YOLO detection
        results = model(image_path)
        detected_objects = [model.names[int(obj)] for obj in results[0].boxes.cls]

        # Filter for animals and humans
        relevant_objects = [obj for obj in detected_objects if obj in ["person",
"dog", "cat", "cow", "horse", "sheep", "elephant"]]

        if relevant_objects:
            print(f"Detected: {', '.join(relevant_objects)}")
            send_telegram_alert(image_path, relevant_objects)
        else:
            print("No relevant object detected.")

        time.sleep(2)  # Avoid rapid triggering

GPIO.cleanup()
```

5. Test Case Record

Testing Process:

1. **Motion Detection:** Verified PIR sensor triggers correctly.
2. **Camera Capture:** Ensured images are captured and saved properly.
3. **Object Detection:** Tested YOLOv8 with different animals and humans.
4. **Telegram Alerts:** Checked if messages with detected objects and images were sent.
5. **Live Video Feed:** Confirmed real-time display on the Raspberry Pi screen.

6. Outcome Evaluation

Expected Results:

- Motion detection using the PIR sensor.
- Accurate object identification using YOLOv8.
- Immediate Telegram notifications with images of detected humans/animals.
- Live video feed display on the Raspberry Pi screen.

Actual Results:

- The system successfully detected motion and captured images.
- Object detection using YOLOv8 was **fast and accurate**.
- Telegram notifications were received with object classification.
- The real-time video stream was displayed without lag.

7. Future Improvements

1. **Low-Power Optimization:** Improve energy efficiency for outdoor usage.
2. **Cloud Integration:** Store detected images in Google Drive or AWS.
3. **Night Vision Support:** Use an IR camera for nighttime detection.
4. **Multi-Sensor Integration:** Add additional motion sensors for wider coverage.
5. **AI Model Enhancement:** Fine-tune YOLOv8 for better accuracy on local wildlife.
6. **SMS or Call Alerts:** Implement Twilio API for phone notifications.
7. **GPS Integration:** Add GPS tracking to log the location of detections.

8. Conclusion

This project successfully implemented an IoT-based camera trap system with real-time motion detection, object identification, and alert notifications. It demonstrates an efficient approach for wildlife monitoring and security applications. Future upgrades can enhance its usability and accuracy, making it more robust for real-world applications.