Design And Development of An AI-Driven Solution For Rhesus Macaque

[1*]Santosh Giri, [2]Bikal Adhikari, [3]Binaya Basnet, [4]Binit K.C., [5]Rajesh Subedi

[1,2,3,4,5]Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, TU

Email: [1]santoshgiri@pcampus.edu.np, [2]bikaladhikari@pcampus.edu.np, [3]077bei016.binaya@pcampus.edu.np, [4]077bei017.binit@pcampus.edu.np, [5]077bei032.rajesh@pcampus.edu.np

DOI: 10.3126/jacem.v12i01.93908

Abstract

In much of the farming land of rural Nepal, wild animals, most notably the Rhesus Macaque, a widely distributed monkey species found in Nepal, pose a significant issue. These animals frequently find their way into the farmland and destroy the farm crops, endangering the efforts and earnings of the farm- ers. The research aims to create a smart solution that uses Artificial Intelligence-based detection and ultrasonic-based repellents. The system enables the use of the AI-powered cameras to scan and monitor the target area and identify the Rhesus Macaque in real-time using the YOLO algorithm, trained using a custom dataset of monkey images, and upon detection, use an ultrasonic deterrent system to repel the Rhesus Macaque. The detection model was trained on the dataset collected from the Pashupatinath and Swayambhunath area, as the farmland shares similar species, to enhance the detection performance of the model under local settings. Upon identification of the species, the detection model activates the ultrasonic deterrent subsystem. The YOLO based detection model achieved a precision of up to 1.00, an F1-score of 0.97, and a recall of 0.99. The 555 Timer IC-Based ultrasonic generator was validated at the prototype stage, with its generated output frequency measured and verified using an oscilloscope.

Keywords—*Rhesus Macaque, Artificial Intelligence, Real-time object detection, YOLO algorithm, Frequency repellent*

1. INTRODUCTION

It cannot be debated that agriculture continues to occupy a significant role in the Nepalese economy (roughly 23.9 % of the country's GDP) [1], as it has been using farming as their major source of income for most of the people residing in rural areas. Over the years, agriculture has supported these people with food and financial income through the sales of crops, which have been grown and harvested. However, in the recent past, wild animals, especially monkeys, have posed a threat to farming activities and the farm- ing industry more specifically [2][3]. Monkeys are always a big threat to farmers and their production since they intrude on farms in the process, causing a lot of damage to plantation crops [4][5], which ends up making many farmers get huge losses since they depend so much on what they harvest for their food. According to the study [6], done on Budhigandaki river basin lying on Dhading and Gorkha districts of central Nepal, Crop raiding data, collected via questionnaire, survey method to local households in the nearby villages and also through direct observation showed that maize (58.43%) was the highest raided, followed by rice (11.34%), lentil (8.74%), peanut (4.35%), soyabean (4.18%), wheat (3.22%), fruits (2.97%), black pulses (1.87%), potato (1.67%), sesham (0.92%), tomato (0.79%), millet (0.67%), mustard (0.36%), broad beans (0.25%), brown lentil (0.18%) and pumpkin (0.06%).

Another factor that goes a long way in increasing the severity of this phenomenon is the fact that there are no efficient deterrent measures that can be employed at the current moment. In most circumstances, growers use retributive measures that entail beating the animals or taking retributive action against them, which is unethical. Some farmers use the 'Watchman' model [7][8], where people have to sit for many hours to protect the fields. However, this method takes time, and in most cases, it will not be effec- tive enough to protect crops from destruction, and in the process, will take more effort from the already stretched rural workers. Since agriculture is an essential aspect of the economy in these areas, there is a need to come up with better measures of controlling the threat posed by wildlife. The continued use of inefficient and labor-intensive practices not only puts the income of farmers at risk but also poses a threat to the disruption of ecological balance through practices that may be detrimental to wild animals. There is a dire need to come up with new technologies that can help in protecting plants and crops from being affected, while not affecting the wildlife and putting much pressure on the rural farmers. Many such solutions would maintain the fiscal integrity of rural farming economics and would also not harm the rich biological diversity of Nepal.

Various techniques have been used for monkey repellent. However, the techniques used are still tradi- tional. The most common method most of the Farmers have used is attaching scarecrows in different parts of the field and using a catapult to deter monkeys. But this method is ineffective. As reported by The Economic Times [9], Ahmedabad airport officials were directed to dress up in bear costumes to scare away langurs and monkeys running riot at the airport. In addition, according to The Rising Nepal [10], the farmers of Illam, fed up with safeguarding their corn from the monkeys in the field all day, began to make and use devices that produce loud and explosive sounds to scare the apes away. These manual techniques are helpful, but are time-consuming and need constant physical monitoring of the field to repel monkeys. So, an automated monkey repelling technique has to be developed, which uses current technology for constant monitoring as well as repelling monkeys.

Hence, this research proposes an AI-driven system specially designed to drive away Rhesus macaques with the help of real-time tracking and ultrasonic sound-based repellent. The system integrates real-time object detection using the YOLO model and an ultrasonic sound emission mechanism. The system iden- tifies the presence of the macaques using the camera module and sends specific high-frequency sound waves to discourage the animal from entering or remaining in the protected area. The system is specifi- cally designed for the environmental and agricultural conditions of Nepal, with the intention of minimiz- ing crop damage and promoting wildlife-friendly management methods.

2. LITERATURE REVIEW

Agriculture plays a vital role in sustaining the livelihoods of millions of people worldwide, but it is in- creasingly under threat from wildlife. Among the animals contributing to this issue, monkeys, particularly the Rhesus macaque, have become a significant nuisance. As reported by The Kathmandu Post [11], Mon- keys have become a real threat to the locals, attacking them, and at least eight to 10 people are injured in monkey attacks every year in local level. Not only have they impacted human beings, but they have also impacted the agricultural sector, resulting in damage to crops and, in turn, financial losses for farmers and food insecurity in the affected regions. The authors in [12], proposed the Embedded Edge-AI-based Intelligent Animal Repelling System (EEAIRS) with attributes like real-time animal detection, LoRa communication, and solar power. The experimen- tal outcomes revealed that the proposed system obtained an accuracy of 87% in animal detection and a 30% reduction in false alarms. Moreover, the system successfully repelled animals in 92% of cases and minimized power consumption by 15%. The authors in [13] suggested a relay device that employed the passive infrared sensor (PIR), which turns on the driver responsible for the production of ultrasound. The sound produced by the device is 120dB at a distance of approximately 1 m and has a very wide band of 20kHz-40kHz, which can be tuned to the animal to be repelled. Also, regarding the health issues caused by ultrasonic sound frequency, the authors in [14] stated through research and observation that 8-hour exposure to 110 dB at 20 kHz in the third-octave band at 20 kHz, 25 kHz, and 31.5 kHz did not cause hearing loss in audible frequencies. However, people exposed for 15 min to 150 dB at 20 kHz frequency did cause damage to the audible frequencies. The authors in [15] validated the innovative sound device that repels monkeys using three different practices in Ramkot and Bhool district of India. The validation test was performed in three places at peak hour at an interval of 30 minutes. The farmers' response to beating drums and fireworks was good, with few monkeys found in the field. But for the monkey repellent sound device that mimics baboon barking, the total number of monkeys seen was almost zero. According to The Times of India [16], Telangana State Agriculture University has devised a solution to address the monkey problem plaguing Secunderabad and other areas. The university's vertebrate pest management division developed an agri cannon, a device that emits a loud sound when fired to scare off monkeys. Officials stated that monkeys caused about 60 to 70 percent crop damage roughly 20 days before harvest, but with the introduction of the cannon, the damage has been reduced to 20 percent. The authors in [17] propose an Arduino-based smart monkey repellent circuit that projects high-intensity ultrasonic sound that annoys and makes monkeys leave the surrounding area without causing harm to both animals and humans. For the detection purpose, we use an object detection technique that uses neural networks to localize and classify objects in images. Object detection algorithms are broadly categorized into one-stage and two- stage models. The two-stage detection model comprises region proposal, followed by the classification of those regions and refinement of location predictions. In contrast, single-shot detection skips the region proposal stage, providing final localization and content prediction in a single step [18]. Two-stage algo- rithms include R-CNN, Fast R-CNN, and Faster R-CNN [19]. One-stage algorithms include YOLO(You Only Look Once), SSD(Single Shot Detector), RetinaNet, CornerNet, and FCOS(Fully Convolutional One-Stage Object Detection) [20]. Among one-stage methods, YOLO [21] is a classic example. The YOLO series has seen many versions, from v1 to v8. The YOLO series focuses on real-time and high classification accuracy, using optimal computational parameters for fast detection, high accuracy, and edge devices [22]. The development of the YOLO network from V1 to V5 has seen the addition of many improvements, including grid division (v1), anchors and two-stage training (v2), multi-scale detec- tion with FPN (v3), SPP and GIOU loss (v4), and models with variable sizes and Hardswish activation (v5) [23]. YOLO has many applications in different areas, as shown in a research paper on plant dis- ease detection using deep learning [24]. The authors employed convolutional neural network algorithms, specifically YOLO-v5 and YOLO-v8, to detect infected corn leaves, comparing mean average precision (mAP) at 50 and 50-95. YOLO-v8 exhibited superior accuracy and a higher detection rate. After a com- prehensive examination of various object detection algorithms, our research identifies the YOLO (You Only Look Once) algorithm as the most fitting and effective solution for our objectives in object detection, especially in real-time [25] [26] [27] [28] .

3. METHODOLOGY

This study proposes an integrated system that has two primary parts: (1) a YOLO Model-based model for real-time Rhesus macaques detection and (2) an ultrasonic repellent model. The detailed architecture is presented in the Figure 1.

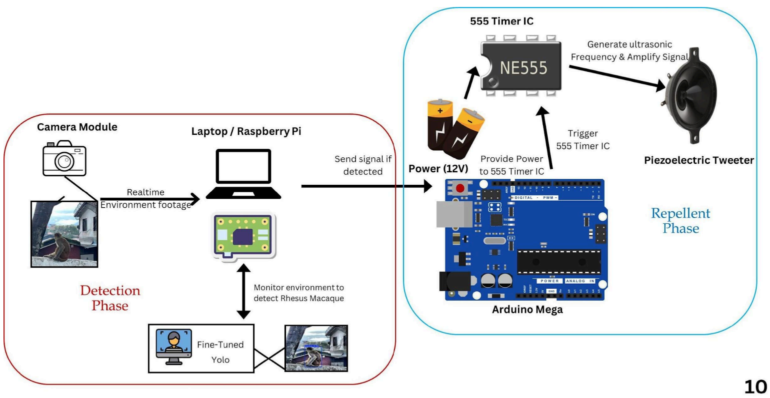


Figure 1: Overall Methodology of YOLO-based detection and ultrasonic-based repellent module

A. Detection Model Design

a. Data Collection and Pre-processing

The required images of the rhesus macaque were taken from the Swayambhunath and Pashupatinath area, as the farmland and surrounding areas share a similar species composition. Approximately 400 images were taken using a mobile phone camera in .jpg format. After data collection, the images were then annotated using a labeling tool. We used Roboflow for image labeling. then we have applied the pre-processing/augmentation techniques like (1) Auto-Orientation, (2) Resizing: Stretch image to 640x640 pixels, (3) Flip: Horizontal, (4) 90° Rotate: Clockwise, Counter-Clockwise, (5) Rotation: Between -15° and 15°, (6) Blur: upto 1.8px. This was done as these pre-processing and augmentation techniques help to transform existing images to create new variations and increase the number of images in the dataset. Which makes models more accurate across a broader range of use cases, and it creates new training examples for the model to learn from by generating augmented versions of each image in your training set. By doing so, our dataset images increased from 400 images to 950 images altogether.

b. Dataset Splitting

The dataset is then split into three parts: 75% training data, 20% validation data, and 5% testing data. The test dataset was selected to the bare minimum due to the limited dataset size, as the data collection is still ongoing. Training data is the subset of the dataset used to train the machine learning model and learn patterns. This helps to train our model how to perform. Validation data is used to evaluate the model during training. Once the model is built, previously unseen data is fed into the model to check its performance. This dataset is known as testing data and helps to evaluate the progress of the model in terms of accuracy and precision, and helps to optimize it for future use. Among 950 images altogether, we split our data as follows: (1) Training Set: 715 images, (2) Valid Set: 190 images, (3) Test Set: 45 images.

c. Base Model Selection

The detection model is based on YOLO. We have selected the YOLOv8[29] architecture, more precisely YOLOv8n, a nano version of YOLOv8, as the base model. The reason for selecting YOLOv8 as the base model is due to its balance between detection accuracy and computational efficiency in our case.

d. Model Hyperparameters Configuration and Training

To build and train the detection model, we imported YOLOv8n from the ultralytics library and fine-tuned using the prepared Rhesus Macaque dataset. The YOLOv8 architecture was configured for training with an input image of 640X640, a batch size of 16, learning rate of 0.01. In order to effectively utilize comput- ing resources, we made use of Kaggle's GPU. By using Kaggle's GPU resources, we were able to speed up the training process and improve the model's performance through repeated parameter refinement.

As shown in Figure 2. Initially, the training epochs were set to 300 for our YOLOv8n-based model; however, the process was conducted over 175 epochs only. This early stopping was triggered because no improvements were observed in the last 100 epochs, after 175 epochs, and the best model was saved at epoch 175. The model at this epoch ensured optimal performance without overfitting and also demonstrated strong performance metrics for the effective detection task.

The loss for validation was slightly higher at the beginning; however, during the first 50 epochs, the training and validation loss decreased effectively, indicating effective learning and correct parameter up- dates. These metrics highlight the model's accuracy and reliability in detecting rhesus macaques based on training data.

e. Real-Time Inference and Trigger Mechanism

The trained YOLOv8 model is included in our Rhesus Macaque Repellent System for practical use throughout the model deployment phase. The training process results in the creation of the best.pt file, which contains the learnt weights and optimized parameters. This file is the foundation upon which we build our model deployment, enabling us to use it with ease to detect Rhesus Macaque in realtime envi- ronment. This real-time monitoring is then analyzed by a computing unit, such as a Raspberry Pi, with a lightweight YOLOv8n algorithm designed to detect rhesus macaques. When a macaque is detected by the detection model, it activates an Arduino Mega to turn on a 555 Timer IC-based deterrent model to emit ultrasonic sound waves through a piezoelectric tweeter.

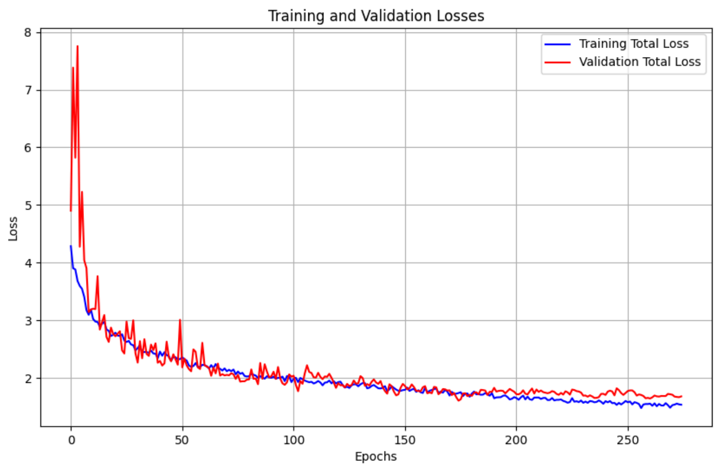


Figure 2: Training and validation Loss curve of our detection model

f. Repellent System Design and Implementation

The hardware components used for the repellent system are: Arduino Mega 2560, resistors (1×1kΩ and 1×12kΩ), a 2.5nF capacitor, a 555 Timer, an IRFZ44N MOSFET, a Kenwood tweeter (8Ω impedance, 50W peak power), and connecting wires. Our major focus on the repellent system design is generating an ultrasonic frequency which varies from 20KHz to 50KHz. This is the range that is not audile by human but irritates monkeys. After the yolo model detects a monkey in a video frame, a signal is sent to Arduino to turn on the 555 timer IC. The signal to Arduino is sent using pyserial which is a library of Python that encapsulates access to the serial port. To initialize Arduino to receive a signal through the serial port, we uploaded code to Arduino: String command;

```
void setup(){
Serial.begin(9600);pinMode(2,OUTPUT);
digitalWrite(2,LOW);
}

voidloop(){
if(Serial.available()>0){
command=Serial.readStringUntil('\n');command.trim();
if(command=="ON"){
digitalWrite (2,HIGH);
}elseif(command=="OFF"){
            digitalWrite(2,LOW);
        }
    }
}
```

This Arduino code monitors serial commands that trigger the control of digital output on pin 2. The setup() function starts serial communication at a baud rate of 9600 before setting pin 2 as output to ini- tially switch off with the LOW state. The loop() function maintains constant evaluation of serial port data reception. The available data initializes its read of incoming text strings until a newline character stops reception, then performs whitespace trimming before initiating the comparison process. The ”ON” com- mand activates the device connected to pin 2 by enabling HIGH voltage through it. When the command reads ”OFF”, the code operates pin 2 to set a LOW state for the device powering off. The code enables external devices, together with software elements (serial monitor or Bluetooth module), to control output pins through basic text messages.

After uploading the code to Arduino, we need to setup python code to communicate with the Arduino.

```
import serial
import serial.tools.list_ports

# Initialize Arduino
serialInst = serial.Serial()serialInst.baudrate = 9600
serialInst.port = "COM6" serialInst.open()

ON = "ON\n".encode('utf-8') OFF = "OFF\n".encode('utf-8')

if monkey_detected:
serialInst.write(ON) serialInst.flush()
else:
serialInst.write(OFF) serialInst.flush()
```

The program uses Python to connect Python with an Arduino board through serial lines for directing electronic devices. The code implements the PySerial library, which enables it to transmit commands to Arduino devices through the designated COM port. The first step of code sets the baud rate to 9600 according to Arduino serial settings while selecting COM6(or any port) as the port connection. Next, the program establishes communication through the opened port. Before sending data through serial commu- nication, the script creates and converts "ON" and "OFF" commands to UTF-8 byte format. The program- ming system verifies the Boolean condition of the monkey_detected variable. When monkey_detected becomes True, the program sends the ON command to the Arduino to activate a connected device. The "OFF" command is transmitted based on the False condition of monkey_detected. Every one of these commands must be followed by a serialInst.flush() call to guarantee complete transmission of the buffer data.

After initialization of the communication between Python and Arduino, the host device can now sent sig- nal based on the monkey detection to Arduino. Since port 2 of Arduino is used here to output the signal to the Vcc of the 555 Timer. The 555 Timer is used here to generate output frequency in a varying range of 20KHz to 50KHz. The frequency and duty cycle of the output signal of the timer IC can be varied by changing the resistor and capacitor values, according to the result required.

$$f = \frac{1.44}{(R1 + 2 \cdot R2) \cdot C}$$

$$D = \frac{R_1 + R_2}{R_1 + 2 \times R_2}$$

We have used R1=1K, R2=12K and C=2.5uF. These values of the component yields output frequency of 28Khz and a duty cycle of 50%. However, we need to consider the input voltage according to the output voltage and power requirements. We have used a 5V input to the 555 Timer, which is provided by Arduino. The circuit uses an astable configuration of a 555 timer IC, which receives power from either a 12V/5V DC supply based on component specifications. The circuit creates an uninterrupted sequence of square wave signals that appear at pin 3 and runs independently from any triggering input. The output wave- form obtains its frequency and duty cycle characteristics through the operations of timing components R1 (1kΩ), R2 (12kΩ), and C1 (1.2nF). The pulse signal sent to visualizers for waveform analysis also flows into digital counting equipment that performs real-time frequency counting. The counter automat- ically rises in value during each clock cycle and thus monitors the output signals from the timer oscillator.

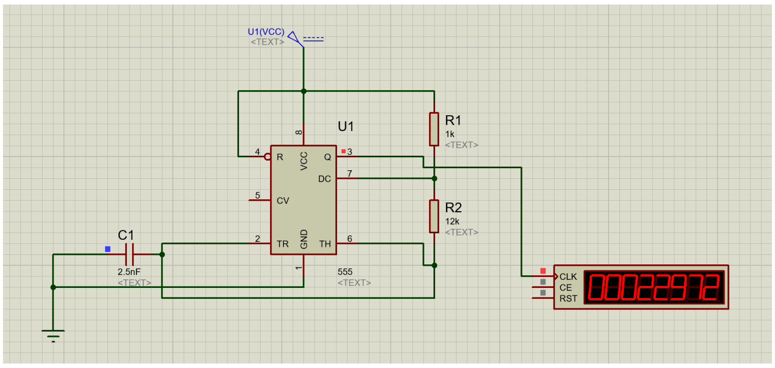


Figure 3: Circuit Diagram Simulation

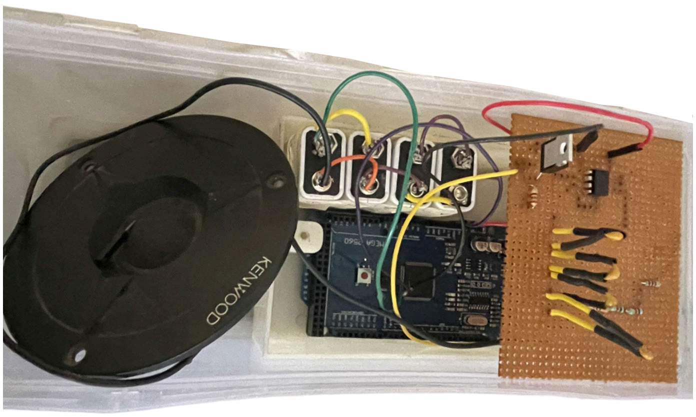


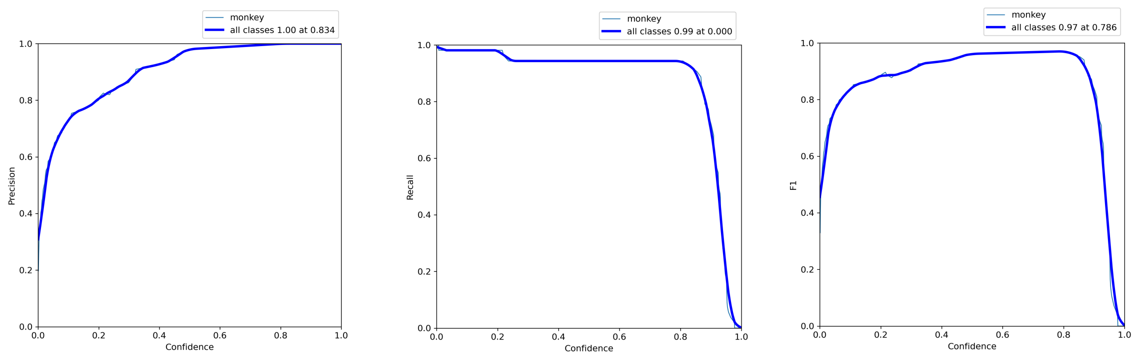
Figure 4: Hardware Implementation

The square wave output signal amplified using the IRFZ44N IC, which is an N-channel MODFET, can be transmitted to a speaker system for producing audible sound. A Kenwood tweeter speaker rated at 8Ω impedance with 150W peak power replaces the traditional buzzer to achieve higher audio clarity and loudness. The high-power speaker needs an external amplifier circuit because the 555 timer lacks the capability to operate it directly. Both the amplifier and speaker can be powered through a 12V power supply, and 5V power is suitable for driving the 555 timer and digital logic components. The imple- mented design allows for clear audio functions when producing pulse signals.

4. RESULTS AND DISCUSSION

A. Precision Confidence Curve of detection model

Figure 5(a) demonstrates the Precision Confidence Curve of the YOLOv8n model. Confidence is posi- tively related to precision, and the confidence is 1.00 when there is a threshold of 0.834, which means that the predictions are highly accurate when there is a high confidence. Figure 5(b) is the Recall Confidence Curve. It is observed that Recall is nearer to 1.0 when the confidence level is low, but is declining rapidly after 0.8, which indicates that increasing the level of confidence makes the model selective. The recall at 1.1 confidence is 0.99 in all classes. Figure 5(c) represents the F1 Confidence Curve. The highest F1 score of 0.97 is achieved at the confidence level of 0.786, which is the best tradeoff between precision and recall. At this stage, F1 reduces because of low recall.



(a) Precision Confidence Curve (b) Recall Confidence Curve (c) F1 Curve

Figure 5: Precision, Recall and F1 Confidence Curves

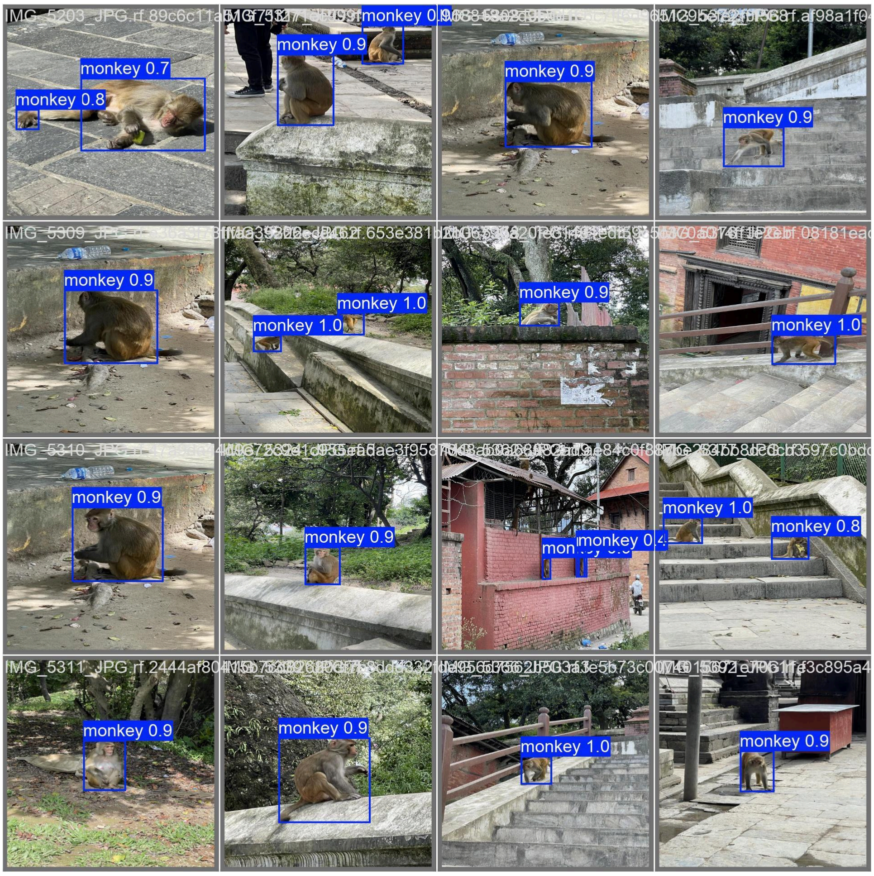


Figure 6: Output images prediction

B. Qualitative Detection Results

The Figure 6 shows sample detection results in a real-time environment. The detection model identifies rhesus macaques in diverse real-time environments, including staircases, temple-premises, pavements, brick walls, etc. The bounding boxes indicate detected objects classified under the ”monkey” category.

C. Repellent System Prototype Validation

The repellent sub-system is currently in the prototype development stage and has not been field tested in live environment settings. At this phase, the validation of the ultrasonic frequency output of the repellent model was measured and verified using an oscilloscope. During measurement and validation, the different parameters are calculated as follows:

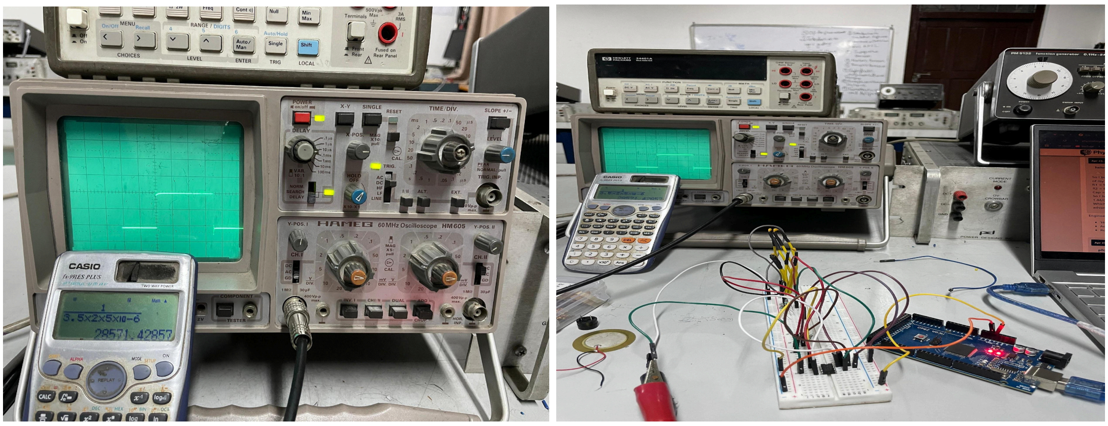


Figure 7: Checking Frequency in Oscilloscope

Duty Cycle:

$$D = \frac{R_1 + R_2}{R_1 + 2\times R_2} = \frac{1000 + 1000}{1000 + 2\times 1000} = \frac{2000}{3000} = 0.51 \approx 50\%$$

From the oscilloscope, Time period for 50% duty cycle =3.5. So, the time period for 1 pulse:

$$T = 3.5 \times 5 \times 2 \times 10^{-6} = 3.5 \times 10^{-5} \text{ seconds}$$

Frequency:

$$\frac{1}{T} \approx 28571\, Hz = 28.57\, kHz$$

Voltage measurement:

Volt/div = 2V and Output voltage = 2 x 2.1 = 4.2V

4.2V is provided by the output of the 555 timer when the input from another IC is provided as 5V.

5. CONCLUSION

The integration of the YOLOv8n model with the ultrasonic deterrent circuitry implemented the real-time monkey detection system, with valid alert and activation capabilities. As the size of the dataset increased, the generalization ability of the model and the accuracy of detection enhanced considerably. A set of performance optimization methods increased the stability of training, minimized the variance of predic- tions, and increased the convergence rate. Based on comparative analysis, it was observed that YOLOv8n could achieve more stable training outcomes, making it a preferred model due to its higher computational efficiency and deployment capability. The deterrent subsystem uses a 555 Timer IC-based ultrasonic gen- erator with a combination of an Arduino Mega 2560, passive timing elements, an IRFZ44N MOSFET driver, and a high-frequency tweeter. The circuit generates ultrasonic waves between 20 and 50 kHz, which are not audible to human beings but leave monkeys agitated. Once the monkey has been spotted by the YOLOv8n model, a serial command to the host will trigger the Arduino, and this causes the 555 timer to enter astable mode and create the necessary waveform. The produced ultrasonic frequency was experimentally confirmed on an oscilloscope at the hardware level, where it was observed that the ultra- sonic frequency remained stable at about 28 kHz, which verifies the functional validity of the prototype circuit of the repellent model. A MOSFET amplification of the output signal is used to power the tweeter satisfactorily so that the sound propagates sufficiently. The future work will focus on controlled field test- ing of the repellent model to analyze the effectiveness of the proposed repellent model under real-world conditions.

ACKNOWLEDGMENT

This research was supported by the University Grants Commission, Nepal, under Faculty Research Grant (FRG.81/82 Engg-05). The authors would like to acknowledge the University Grants Commission, Nepal, for the financial support.

REFERENCES

[1] Nepal Rastra Bank. Current macroeconomic and financial situation of nepal. https://www.nrb.org.np/contents/uploads/2022/08/Current-Macroeconomic-and-Financial-Situation-English-Based-on-Annual-data-of-2021.22-2.pdf. Accessed: January, 2026.

[2] Dialogue Earth. In nepal, debate rages on crop-raiding monkeys. https://dialogue.earth/en/nature/in- nepal-debate-rages-on-crop-raiding-monkeys/. Accessed: January, 2026.

[3] fiscalnepal. Nepal declares monkeys as ‘agriculture-harmful wildlife’, farmers can act without penalty. https://www.fiscalnepal.com/2026/02/16/24508/nepal-declares-monkeys-as-agriculture-harmful-wildlife-farmers-can-act-without-penalty/. Accessed: January, 2026.

[4] Kathmandu Post. Monkey, maize and man. https://kathmandupost.com/investigations/2022/03/26/monkey- maize-and-man. Accessed: January, 2026.

[5] mongabay. Monkeys, porcupines team up to destroy crops, nepal’s farmers say. https://news.mongabay.com/2022/04/monkeys-porcupines-team-up-to-destroy-crops-nepals-farmers-say/. Accessed: January, 2026.

[6] Suvas Chandra Ghimire and Mukesh Kumar Chalise. Status of crop raiding by assamese mon- keys (macaca assamensis) along the budhigandaki river, central nepal. *Journal of Natural History Museum*, 30:294–305, 2018.

[7] kathmandupost. Troubled villagers hire watchmen to protect crops from monkeys. https://kathmandupost.com/karnali-province/2023/08/21/troubled-villagers-hire-watchmen-to-protect-crops-from-monkeys. Accessed: January, 2026.

[8] The Rising Nepal. Resunga hires 25 watchmen to drive away monkeys. https://risingnepaldaily.com/news/47801. Accessed: January, 2026.

[9] Economic Times. Ahmedabad employs bears for its airport. https://economictimes.indiatimes.com/industry/transportation/airlines-/-aviation/ahmedabad-employs-bears-for-its-airport/grin-and-bear-it/slideshow/74004272.cms. Accessed: January, 2026.

[10] The Rising Nepal. Farmers learning new methods to deter monkeys. https://risingnepaldaily.com/news/14007, 2022. Accessed: January, 2026.

[11] kathmandupost. Authorities scramble to tackle monkey menace in gandaki. https://kathmandupost.com/gandaki-province/2024/02/09/authorities-scramble-to-tackle-monkey- menace-in-gandaki. Accessed: January, 2026.

[12] Davide Adami, Mike O. Ojo, and Stefano Giordano. Design, development and evaluation of an intelligent animal repelling system for crop protection based on embedded edge-ai. *IEEE Access*, 9:132125–132139, 2021.

[13] Stefano Giordano, Ilias Seitanidis, Mike Ojo, Davide Adami, and Fabio Vignoli. Iot solutions for crop protection against wild animal attacks. In *2018 IEEE international conference on Environmen- tal Engineering (EE)*, pages 1–5. IEEE, 2018.

[14] González-Lezcano RA Moyano DB, Paraiso DA. Possible effects on health of ultrasound exposure, risk factors in the work environment and occupational safety review, 2022. Accessed: September 2, 2024.

[15] RK Gupta. Innovative sound device that repel monkeys and its scientific validation in jammu, 2019.

Accessed: September 8, 2024.

[16] Times of India. Now, special gun to scare monkeys, 2019. Accessed: September 9, 2024.

[17] V Janani and C Shanthi. Human-animal conflict analysis and management-a critical survey. In *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*, pages 1003–1007. IEEE, 2022.

[18] Chinmay U Parab, Canicius Mwitta, Miller Hayes, Jason M Schmidt, David Riley, Kadeghe Fue, Suchendra Bhandarkar, and Glen C Rains. Comparison of single-shot and two-shot deep neural network models for whitefly detection in iot web application. *AgriEngineering*, 4(2):507–522, 2022.

[19] O Hmidani and EM Ismaili Alaoui. A comprehensive survey of the r-cnn family for object detection. In *2022 5th International Conference on Advanced Communication Technologies and Networking (CommNet)*, pages 1–6. IEEE, 2022.

[20] Hang Zhang and Rayan S Cloutier. Review on one-stage object detection based on deep learning.

EAI Endorsed Transactions on e-Learning, 7(23):e5–e5, 2021.

[21] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[22] Muhammad Hussain. Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection. *Machines*, 11(7):677, 2023.

[23] Peiyuan Jiang, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. A review of yolo algorithm develop- ments. *Procedia Computer Science*, 199:1066–1073, 2022.

[24] Nidya ChitraningrumLies BanowatiDina Herdiana. Comparison study of corn leaf disease detection based on deep learning yolo-v5 and yolo-v8. *Journal of Engineering and Technological Sciences*, pages 1034–1097, 2024.

[25] Gudala Lavanya and Sagar Dhanraj Pande. Enhancing real-time object detection with yolo algo- rithm. *EAI Endorsed Transactions on Internet of Things*, 10, 2024.

[26] Chang Ho Kang and Sun Young Kim. Real-time object detection and segmentation technology: an analysis of the yolo algorithm. *JMST Advances*, 5(2):69–76, 2023.

[27] U Sirisha, S Phani Praveen, Parvathaneni Naga Srinivasu, Paolo Barsocchi, and Akash Kumar Bhoi. Statistical analysis of design aspects of various yolo-based deep learning models for object detection. *International Journal of Computational Intelligence Systems*, 16(1):126, 2023.

[28] Lu Tan, Tianran Huangfu, Liyao Wu, and Wenying Chen. Comparison of yolo v3, faster r-cnn, and ssd for real-time pill identification. 2021.

[29] Ultralytics. Ultralytics yolov8. https://github.com/ultralytics/ultralytics, 2023. Accessed: February 2026.