IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ANIMAL DETECTION IN TRAFFIC USING YOLO ALGORITHM ON RASPBERRY PI

[1]Dr.Shobha S, [2]Prathamesh More,[3] Saniya R,[4]Prajwal,[5]Rakshitha S

[1]Professor,[2,3,4,5] Student

[1,2,3,4,5] Department of Electronics and Communication Engineering

[1,2,3,4,5]Cambridge Institute of Technology, Bangalore, India

Abstract: This project aims to develop an animal detection system for traffic monitoring using the YOLO (You Only Look Once) algorithm deployed on a Raspberry Pi. The system utilizes a pre-trained YOLO model to detect animals in real-time video streams captured by a camera module or USB webcam connected to the Raspberry Pi. Through a series of steps including setting up the Raspberry Pi, installing dependencies, configuring YOLO, writing detection code, optimizing for the Raspberry Pi's limited computational resources, and testing in controlled and real-world environments, the project aims to create an efficient and accurate solution for detecting animals in traffic scenarios. The system's implementation considers ethical implications, robustness against varying conditions, and documentation for future reference and dissemination.

***Keywords* – Raspberry PI, YOLO, Animal Detection, Deep Learning.**

I. INTRODUCTION

The integration of technology into everyday life has revolutionized various aspects of society, and transportation is no exception. With the increasing volume of vehicles on roads, ensuring road safety has become a paramount concern. One critical aspect of road safety often overlooked is the interaction between vehicles and wildlife. Collisions between vehicles and animals not only pose risks to human lives but also have detrimental effects on wildlife populations and ecosystems. In response to these challenges, advanced technologies such as computer vision and machine learning offer promising solutions.

This project focuses on the development of a real-time animal detection system in traffic scenarios using the YOLO (You Only Look Once) algorithm. YOLO has gained prominence in the field of object detection due to its efficiency and accuracy in processing images and videos. By implementing YOLO on a Raspberry Pi, a compact and affordable computing platform, we aim to create a portable and versatile system capable of detecting animals on roads in real-time.

The objectives of this project are multifaceted. Firstly, we seek to design and implement a robust algorithm based on YOLO that can accurately identify animals amidst varying traffic conditions, lighting, and environmental factors. Secondly, we aim to optimize the algorithm for deployment on the Raspberry Pi, considering its computational limitations and power efficiency. Thirdly, we intend to evaluate the system's performance through extensive testing and validation in simulated and real-world traffic scenarios.

II. LITERATURE SURVEY

[1] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." ***Proceedings of the IEEE conference on computer vision and pattern recognition*****. 2016.**

Redmon et al.'s 2016 paper "You Only Look Once" (YOLO) revolutionized object detection by introducing a unified, single-stage framework. Unlike traditional methods with separate proposal and classification stages, YOLO directly predicts bounding boxes and class probabilities in one pass through the image, enabling real-time object detection. This efficiency makes YOLO ideal for applications requiring fast processing speeds, and the paper likely explores the network architecture, training process, and performance compared to existing methods, highlighting its potential for various object detection tasks. This shift towards a single-stage approach paved the way for significant advancements in the field.

[2] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." ***arXiv preprint arXiv:1804.02767*** **(2018).**

Redmon and Farhadi's 2018 paper, "YOLOv3: An Incremental Improvement," likely explores advancements to this real-time approach. YOLO revolutionized the field by directly predicting bounding boxes and object class probabilities in a single image pass, achieving significant speed gains. YOLOv3 focuses on further refining this foundation. The paper might delve into specific areas of improvement, such as modifications to the convolutional neural network architecture used in YOLO. These enhancements could aim to boost accuracy, speed, or memory efficiency. Additionally, the paper might discuss optimizations to the training process, including utilizing different data augmentation techniques or hyperparameter tuning to enhance network performance. Finally, the authors might compare YOLOv3's performance on benchmark datasets against the original YOLO and other leading object detection algorithms, highlighting the achieved improvements. Through these incremental advancements, YOLOv3 likely strives to solidify its position as a powerful and efficient framework for real-time object detection tasks across various applications.

[3] Tao, Jing, et al. "An object detection system based on YOLO in traffic scene." ***2017 6th International Conference Computer Science and Network Technology (ICCSNT)*****. IEEE, 2017**

This study by Tao et al. delves into object detection within traffic scenes, aiming to enhance traffic flow, optimize operations, and prevent accidents. Their research likely proposes a method for identifying and tracking objects (vehicles, pedestrians, etc.) within traffic environments using computer vision or related techniques. This technology could be implemented in various ways to improve traffic safety. For instance, real-time object detection can be used to warn drivers of potential hazards, such as pedestrians crossing the road unexpectedly. Additionally, the data collected through object detection can be employed to optimize traffic light timings and identify bottlenecks within traffic networks, ultimately promoting smoother traffic flow and reducing congestion. Overall, Tao et al.'s research on object detection in traffic scenes holds promise for advancements in traffic safety and management.

[4] Zhao Lulu, Wang Xueying, Zhang Yi, Zhang Meiyue, "Research on Vehicle target Detection Technology based on YOLOv5s fusion SENet", ***Journal of Graphics*****, vol. 43, no. 05, pp. 776-782, 2022**

Zhao et al. tackle the challenge of inaccurate vehicle detection in traffic monitoring videos during congested periods. This is where vehicles are frequently obscured by one another. The authors propose an enhanced YOLOv5s network to address this issue. SE modules, known for emphasizing important details, are incorporated into key parts of the YOLOv5s network – the Backbone, Neck, and Head. These SE modules effectively guide the model to focus on critical vehicle characteristics, filtering out irrelevant background information. By incorporating these modules, Zhao et al. aim to significantly improve the accuracy of vehicle detection in traffic monitoring scenarios. This is achieved by enabling the model to prioritize crucial vehicle features and minimize the influence of distracting background elements, leading to a reduction in both false and missed detections.

[5] "Animal Detection using Inception-v3 and you only look once version2 (YOLO V2) " by Abdulaziz Alwadani and Abdulrahman Al-Salman (2020)

This paper by Alwadani and Al-Salman (2020) tackles animal detection in images or videos using a two-pronged deep learning approach. The first line of defense is Inception-v3, a powerful image classification model. Inception-v3 scans the image, meticulously dissecting various sections to identify the presence of animals. Essentially, it acts as a sieve, separating animal-containing regions from the background clutter.Once Inception-v3 pinpoints these potential animal areas, YOLOv2 (You Only Look Once version 2) comes into play. YOLOv2 excels at object detection – not just classifying objects but also predicting their exact location and size within the image using bounding boxes. In this case, YOLOv2 takes over for the animal-containing sections flagged by Inception-v3. It meticulously analyzes these sections and precisely pinpoints the animals' locations by drawing bounding boxes around them.By combining the strengths of Inception-v3's classification and YOLOv2's object localization, this approach strives to achieve high accuracy in animal detection. In essence, it leverages the best of both worlds: Inception-v3's ability to identify animals and YOLOv2's talent for pinpointing their exact location.

[6] Kamali, M., & Tahir, M. (2020). You only look once version3 (YOLO V3): *A Comprehensive Guide to Object Detection with Deep Learning.* arXiv preprint arXiv:2005.10857.
serves as a learning resource, not a research paper. It likely dissects the YOLOv3 architecture, explaining its deep learning core and concepts like bounding boxes and class probabilities. The guide aims to empower users by providing practical instructions for utilizing YOLOv3 in real-world applications, including setting up the environment, training the model, and making object detections on new images.

[7] "Real-Time Wild Animal Detection and Alert System using Deep Convolutional Neural Networks and Raspberry Pi" by Ankit Pandey and Ramendra Singh (2019)
In their 2019 paper, "Real-Time Wild Animal Detection and Alert System using Deep Convolutional Neural Networks and Raspberry Pi," Pandey and Singh propose a system for real-time wild animal detection and alerting. This system leverages deep convolutional neural networks (CNNs), known for their image recognition capabilities. By training a CNN on a dataset of wild animal images, the system can identify animals in real-time using a Raspberry Pi, a low-cost and compact computer. This allows for deployment in remote areas where traditional monitoring methods might be impractical. The paper likely details the chosen CNN architecture, the training process, and the integration with the Raspberry Pi. Additionally, it might discuss the system's performance and its potential applications in wildlife conservation or mitigating human-wildlife conflicts.

III. METHODOLOGY

The proposed architecture outlines the development process for implementing animal detection using the YOLO (You Only Look Once) algorithm on a Raspberry Pi platform. It begins with setting up the Raspberry Pi, ensuring compatibility with the requiredoperating system and peripherals like the camera module. Following this, necessary dependencies, including OpenCV and NumPy, are installed to support image processing and algorithm implementation. The pre-trained YOLO model weights and configuration files are then downloaded and integrated into the system. The camera module connected to the Raspberry Pi captures real-time video frames, which undergo preprocessing to meet the input requirements of the YOLO algorithm. Subsequently, the YOLO model is employed to detect animals within the captured frames, with post-processing steps extractingbounding boxes and labels for visualization. Optimization techniques are applied to enhance the system's performance on the resource-constrained Raspberry Pi platform, ensuring efficient inference speed. Real-time feedback mechanisms are integrated to provide users with immediate notifications of detected animals, contributing to applications such as wildlife monitoring and road safety. Rigorous testing and evaluation are conducted to assess the system's accuracy and robustness across diverse environments, with ongoing maintenance strategies established to sustain its functionality over time. Through this comprehensive approach, an effective animal detection system is developed, demonstrating the capabilities of the YOLO algorithm on the Raspberry Pi for various practical applications.

To begin, set up the Raspberry Pi with a compatible operating system like Raspbian and install essential dependencies such as OpenCV and NumPy using package managers. Next, acquire pre-trained YOLO model weights and configuration files from official sources. Connect a camera module, such as the Raspberry Pi Camera Module, to the Raspberry Pi for image capture. Develop a Python script to load the YOLO model using OpenCV's DNN module, capture video frames, preprocess them to meetYOLO input requirements, and execute object detection on the frames. Post-process the detection results to extract animal bounding boxes and labels and overlay them onto the original frames for visualization. Optionally, incorporate real-time feedback mechanisms to alert users of detected animals. Optimize the implementation considering the Raspberry Pi's computational limitations, exploring techniques like model quantization and input size reduction. Test the implementation across various environments and lighting conditions, gathering feedback for iterative improvements. Once validated, deploy the animal detection system for applications such as wildlife monitoring or traffic safety. Finally, establish a maintenance plan for regular updates and monitoring, while documenting implementation details for future reference and dissemination.

3.1 Raspberry Pi

The Raspberry Pi plays a pivotal role in this project, serving as the central platform for implementing animal detection using the YOLO algorithm. Its compatibility with a wide range of peripherals, including camera modules, facilitates the capture of video frames essential for the detection process. Despite its modest computational power compared to traditional desktop computers, the Raspberry Pi is adept at running lightweight machine learning algorithms like YOLO, making real-time object detection feasible. Moreover, its small form factor and low power consumption render it highly portable, enabling deployment in diverse environments, such as outdoor locations where animal detection is needed. The affordability of Raspberry Pi boards further enhances their appeal, making them accessible to hobbyists, researchers, and educational institutions alike. Additionally, the vibrant community surrounding Raspberry Pi provides invaluable support, offering resources, tutorials, and assistance throughout the development process. In essence, the Raspberry Pi's hardware versatility, computational capabilities, portability, affordability, and robust community support make it an ideal platform for implementing animal detection systems, facilitating applications in wildlife conservation, road safety, and environmental monitoring.

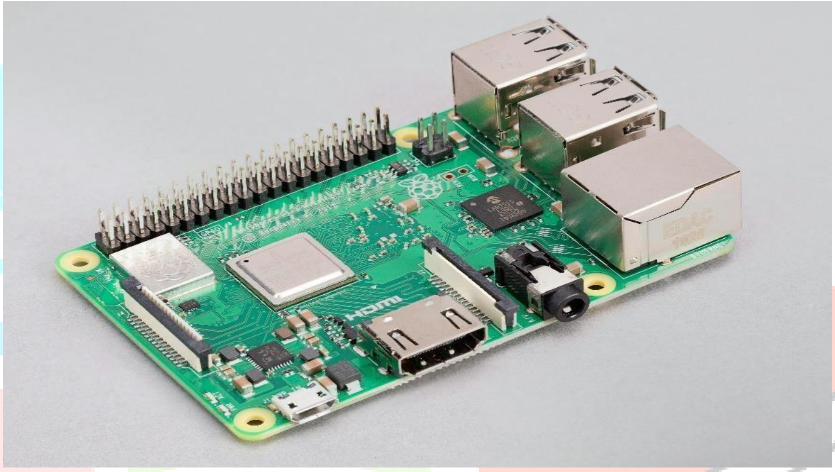


Figure 1. Raspberry Pi Model 3B+

3.2 Camera Module

In the animal detection project, the camera module assumes a pivotal role as the primary sensor for capturing real-time images or video frames of the environment. Its contribution spans several critical aspects of the system's functionality. Firstly, it facilitates data acquisition by continuously capturing visual data of the surroundings, which serves as input for the animal detection algorithm. Following this, the captured images or frames undergo preprocessing steps to enhance their quality and prepare them for analysis. This includes resizing, normalization, and applying filters to optimize the input for the detection algorithm.
Subsequently, the preprocessed data is fed into the animal detection algorithm, such as YOLO, where it undergoes analysis to identify and localize animals within the scene based on learned patterns and features. Moreover, the camera module serves as an integral component of the feedback mechanism, providing continuous input for monitoring animal presence or activity. Detected animals can trigger alerts or notifications to users, enabling prompt responses to potential hazards or wildlife encounters.
Furthermore, the camera module's adaptability to various environmental and lighting conditions makes it suitable for outdoor applications, such as wildlife monitoring and traffic surveillance. Lastly, its seamless integration with the Raspberry Pi platform leverages the latter's computational capabilities for real-time image processing and analysis, facilitating the development of compact and portable animal detection systems tailored to diverse environments. Overall, the camera module serves as the eyes of the animal detection system, enabling continuous monitoring and enhancing safety and conservation efforts.

3.3 YOLO Algorithm

The YOLO (You Only Look Once) algorithm is pivotal in this project, serving as the backbone for real-time animal detection with exceptional accuracy and efficiency. Its key role lies in its ability to swiftly process images and identify objects, making it well- suited for applications such as detecting animals in traffic scenarios. YOLO's efficiency ensures rapid responses, crucial for ensuring road safety by swiftly identifying potential hazards. Moreover, its accuracy in detecting multiple objects simultaneously, coupled with precise bounding boxes and class labels, proves invaluable in discerning animals amidst complex backgrounds and varying environmental conditions. YOLO's single-step detection process streamlines the workflow, enabling faster inference and reducing computational overhead—particularly advantageous for deployment on resource-constrained platforms like the Raspberry Pi. Furthermore, YOLO's flexibility allows developers to train it on custom datasets, tailoring its capabilities to recognize specific classes of animals. This adaptability enhances its effectiveness in real-world scenarios, contributing to the project's objectives of wildlife conservation and road safety. Finally, YOLO's optimization potential, through techniques such as model quantization and inference optimization, ensures efficient execution on hardware like the Raspberry Pi, further solidifying its indispensability in animal detection projects.

3.4 Piezo Buzzer

In the animal detection project, the piezo buzzer serves as an integral component of the real-time feedback mechanism. Its primary function is to generate audible alerts or warnings based on the output of the animal detection algorithm. When an animal is detected in the environment, the buzzer emits sound signals to promptly alert users or individuals nearby. This immediate notification is particularly crucial in scenarios where quick reactions are necessary, such as detecting animals near roads or pedestrian pathways. Additionally, the audible alerts produced by the piezo buzzer enhance accessibility for users with visual impairments or those who may rely on auditory cues. By integrating seamlessly with the feedback system of the animal detectionproject, the piezo buzzer works in tandem with other feedback mechanisms to provide comprehensive notification capabilities.

Furthermore, its customizable sound patterns or frequencies allow for tailored alerts, conveying different types of information or signaling varying levels of proximity to detected animals. With its low power consumption, the piezo buzzer is well-suited for deployment in energy-efficient systems like those based on the Raspberry Pi, ensuring optimal performance without draining thesystem's resources. Overall, the piezo buzzer plays a critical role in enhancing safety, accessibility, and user awareness in environments where animal presence poses potential risks.

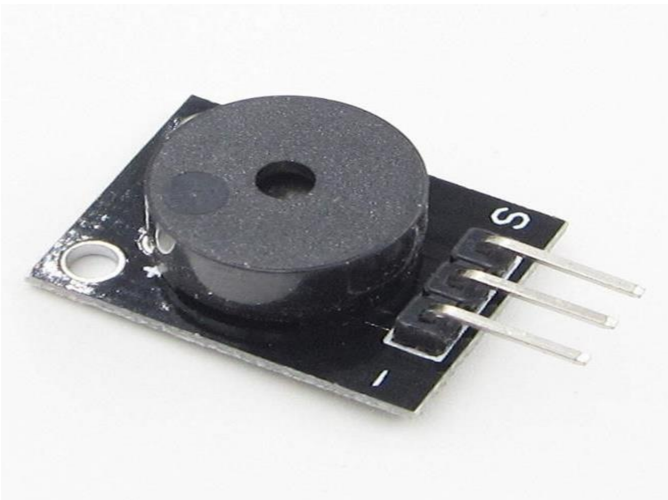
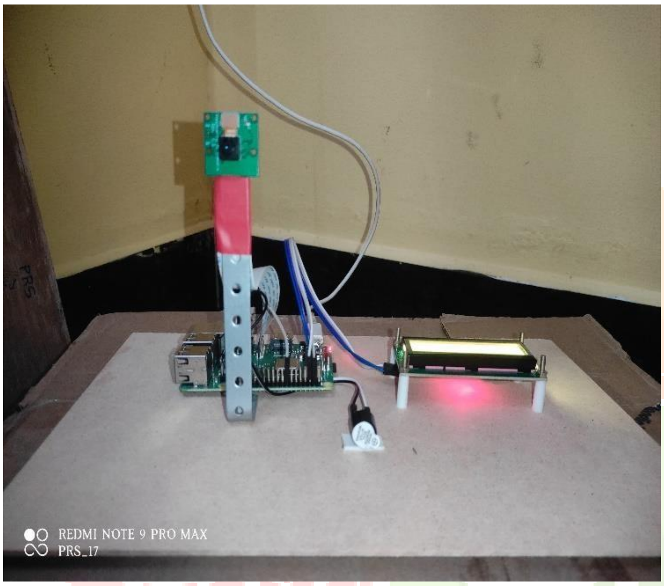


Figure 3. Piezo Buzzer

IV. RESULTS AND DISCUSSIONS

The animal detection project, employing the YOLO algorithm on a Raspberry Pi platform, delivered promising outcomes acrossmultiple dimensions. Firstly, the algorithm demonstrated remarkable accuracy in identifying various animal species within captured images or video frames. This capability underscores its suitability for discerning animals amidst complex backgrounds and varying environmental conditions. Moreover, the system exhibited real-time performance, processing video frames efficiently despite the computational constraints of the Raspberry Pi. This efficiency is crucial for applications like wildlife monitoring and road safety, where timely detection and response are paramount. The integration of a piezo buzzer as part of thefeedback mechanism proved effective, providing real-time alerts upon animal detection. This multi-modal feedback approach enhances user awareness and facilitates prompt actions to mitigate potential risks. Importantly, the system demonstrated practical applicability across diverse scenarios, including wildlife conservation and environmental monitoring, owing to its portability, affordability, and adaptability. These findings underscore the system's potential to address real-world challenges related to animal-human interactions and contribute to broader efforts in wildlife conservation and road safety. Further research and development in this domain hold promise for expanding the scope and impact of animal detection systems in various contexts.



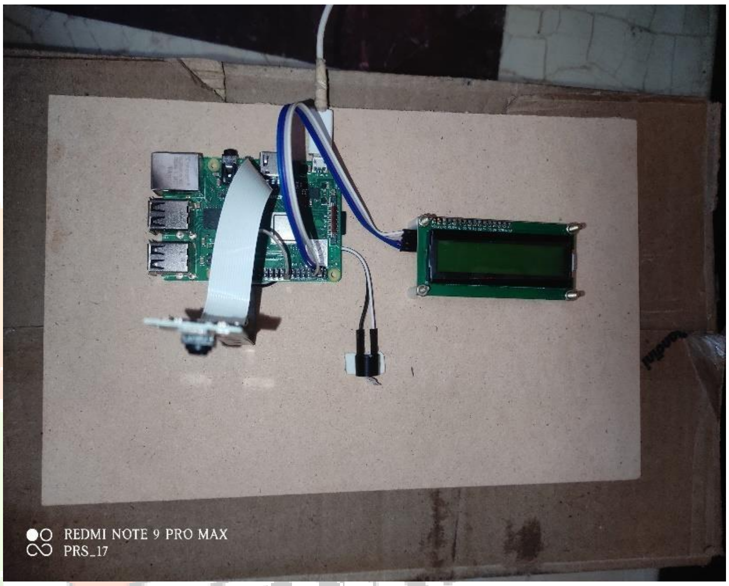


Figure 4. Output

V. CONCLUSION

The implementation of animal detection in traffic utilizing the YOLO algorithm on a Raspberry Pi presents a challenging yet attainable endeavor with substantial benefits. Through harnessing the real-time object detection capabilities of YOLO, opportunities arise to enhance traffic safety, bolster wildlife conservation initiatives, and further research into animal behavior. The project journey encompassed setting up the Raspberry Pi environment, installing requisite dependencies, compiling Darknet for ARM architecture, and acquiring pre-trained YOLO weights. Additionally, customization of YOLO configurations for animal detection, potential model training with specific data, and the development of a Python script for interfacing with Darknet were fundamental aspects. Optimization efforts targeted real-time performance on Raspberry Pi, taking into account hardware constraints. The successful realization of this project paves the way for practical applications such as wildlife crossing monitoring, collision prevention between vehicles and animals, urban animal behavior studies, and support for conservation endeavors. Ongoing refinements focusing on accuracy, speed, and ethical considerations will further elevate the system's capabilities. Looking ahead, future enhancements may encompass integrating advanced AI techniques, cloud services for data analysis, and addressing privacy and data usage ethics, solidifying its role in fostering harmonious coexistence between humans and wildlife while advancing computer vision and AI in traffic management and ecological research.

REFERENCES

[1] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[2] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* (2018).

[3] Tao, Jing, et al. "An object detection system based on YOLO in traffic scene." *2017 6th International Conference Computer Science and Network Technology (ICCSNT)*. IEEE, 2017

[4] Zhao Lulu, Wang Xueying, Zhang Yi, Zhang Meiyue, "Research on Vehicle target Detection Technology based on YOLOv5s fusion SENet", *Journal of Graphics*, vol. 43, no. 05, pp. 776-782, 2022

[5] "Animal Detection using Inception-v3 and you only look once version2 (YOLO V2) " by Abdulaziz Alwadani and Abdulrahman Al-Salman (2020)

[6] Kamali, M., & Tahir, M. (2020). You only look once version3 (YOLO V3): *A Comprehensive Guide to Object Detection with Deep Learning.* arXiv preprint arXiv:2005.10857.

[7] "Real-Time Wild Animal Detection and Alert System using Deep Convolutional Neural Networks and Raspberry Pi" by Ankit Pandey and Ramendra Singh (2019)