IoT Project: TrueDetect Doorbell

Faculty of Engineering and Applied Science

Design and Analysis of IoT Software Systems SOFE-4610U

Mithusan Arulampalam - 100787123

Mohammad Mohammadkhani - 100798165

Date: December 2nd, 2024

GitHub Link: https://github.com/Mithusan/IoTFinal-TrueDetectDoorbell

Abstract

The TrueDetect Smart Doorbell is designed to improve home security by combining motion detection with advanced machine learning techniques for more accurate identification of people and objects. Unlike traditional motion sensors that simply detect any movement, TrueDetect uses real-time object detection to distinguish between humans and other types of movement, significantly reducing the number of false alarms. By incorporating the YOLO3 (You Only Look Once) model for detecting objects and OpenCV's Haar Cascade for facial recognition, the system can tell the difference between familiar faces and strangers. This allows it to send precise notifications to users, keeping them informed only when necessary. The doorbell also streams live video to a mobile app via the Blynk platform, so users can see what's happening in real time, wherever they are. The main goal of this project is to create a smart doorbell camera that not only detects human presence but also enhances security by providing accurate, timely notifications and live video feeds directly to a user's phone. With this system, users can have greater peace of mind, knowing they are always connected to what's happening at their front door.

Introduction

Current smart doorbells, while equipped with motion detection sensors and basic object recognition, often struggle to identify the source of an alert accurately. These systems frequently send false alarms triggered by factors such as pets, moving branches, shifting shadows, or any other non-important object. The TrueDetect Doorbell overcomes these challenges by using advanced image recognition and machine learning algorithms to distinguish between human presence and other types of motion, as well as being able to recognize faces.

Using YOLO (You Only Look Once) for object recognition and OpenCV for facial recognition, TrueDetect provides a smarter and more reliable approach to home security. The system not only detects a person at the door but also tells the difference between an actual visitor and harmless disturbances, meaning TrueDetect can distinguish familiar faces, such as family members or residents, from unknown people and correctly alert when a stranger is at the door.

TrueDetect also allows users to stream live video directly to their mobile devices, enabling real-time monitoring of the front door for a more responsive and convenient security solution.

The central question addressed is: How can image recognition and machine learning improve the accuracy and user experience of motion detection in smart doorbells?

Proposed answer: *using strong image recognition models such as YOLO3 and OpenCV will help with narrowing down detection and making sure systems only alert users when it is necessary, this will ultimately eliminate false alerts with adequate conditions.*

Literature Review

While current research demonstrates the effectiveness of YOLO for real-time object detection and OpenCV's Haar Cascade for facial recognition, there is still a gap in how these technologies can be seamlessly integrated to perform well in real-world situations. The ideal solution—a smart doorbell that offers high accuracy, minimal false alarms, and quick response times—has yet to be achieved in a practical, cost-effective system. More research is needed to refine these models, especially to improve their performance in environments with changing lighting conditions and different levels of computational power.

Although YOLO and Haar Cascade have been tested extensively in controlled environments, there is limited understanding of how these models will fare in the unpredictable conditions of a home setting. Factors such as fluctuating light levels, background distractions, and seasonal changes, like different sunlight angles, can affect detection accuracy. Additionally, the processing power of devices like Raspberry Pi, which are commonly used in smart doorbells, must be carefully considered to ensure that object detection and facial recognition happen without delays or compromised accuracy.

This project aims to fill these gaps by developing a smart doorbell system that combines real-time object detection and facial recognition to reduce false alarms and improve response time. By leveraging YOLO for object detection and Haar Cascade for facial recognition, the system will aim to replace traditional motion sensors, which are often unreliable, and provide more accurate visitor identification. The goal is to optimize these technologies for various real-world conditions, ensuring that the system is both fast and precise.

The project's significance is its potential to enhance the effectiveness of smart home security systems. By minimizing false alarms and improving the accuracy of identification, particularly in challenging lighting conditions, this project could lead to smarter, more reliable security solutions.

Research Objectives

Primary Objective: To develop a smart doorbell that accurately detects human presence using motion detection combined with machine learning-based object and facial recognition.
Secondary Objectives:

- Send notifications to a mobile application upon detecting people at the door.
- Provide real-time streaming of the doorbell camera through the mobile application.
- Implement facial recognition suppression to avoid alerts for known faces.

Methodology

The core design of the TrueDetect Smart Doorbell combines several hardware and software components to ensure seamless functionality and efficient performance. The system incorporates the following:

- ***Camera:*** The doorbell features a **Raspberry Pi Camera Module 2**, providing high-definition image capture capabilities. The camera enables the system to record images of visitors and detect both human and non-human objects using machine learning models.
- ***Microcontroller & Wi-Fi Module:*** A **Raspberry Pi 3** serves as the central processing hub. Equipped with both processing power and a **Wi-Fi module**, the Raspberry Pi allows for efficient data transfer, internet connectivity, and communication with the mobile app for live streaming and notifications.
- ***Software Framework:*** The doorbell's software is developed using **Python**, chosen for its simplicity, versatility, and strong community support, especially in the domain of machine learning. Python allows easy integration of machine learning libraries like **YOLO** for object detection and **OpenCV** for facial recognition. The Raspberry Pi can host the video and stream on a URL, which is then recognized and shown on the **Blynk** app.
- ***Machine Learning Model:***
    - *YOLO:* For object detection, YOLOv3 will be employed. YOLO is a Convolutional Neural Network (CNN) optimized for real-time object detection. The model is pre-trained on large datasets, such as COCO and ImageNet, containing a wide variety of labeled images. By leveraging this pre-trained model, the TrueDetect system can effectively identify objects in real-time, focusing on distinguishing between people and non-people (e.g., animals or packages).
    - *Facial Recognition:* For personal identification, OpenCV Haar Cascade classifiers will be used for facial recognition. Haar Cascades are machine learning-based classifiers used to detect objects (in this case, faces) in images and video streams. While the primary role of YOLO is to detect general objects, Haar Cascade focuses specifically on facial detection, allowing the system to identify known individuals.
- ***Live Streaming and Notification System:*** The system provides live streaming through a web server, accessible via a mobile app using **Blynk** for easy interface and remote monitoring. Notifications are triggered based on object or facial detection, and the system communicates with the mobile app to send alerts in case an unknown person is detected at the door. The Blynk app is the UI for the owners of the device, they will get notifications and alerts from the Blynk app and can access settings to whitelist familiar faces from notifications.

Performance Evaluation and Discussions

Object Recognition Accuracy:

TrueDetect uses YOLO3 for object detection, effectively distinguishing between people and non-people. While factors like lighting and background clutter can reduce accuracy, increasing image contrast in low light significantly improves performance. Based on multiple tests the system achieves 90% accuracy in various environments.

Facial Recognition Accuracy:

Using OpenCV's Haar Cascade, known individuals were identified with about 85% accuracy. Recognition accuracy drops in low-light scenarios but can be improved with better lighting around the camera.

Streaming Latency and Quality:

Streaming latency is low, ranging from (200ms-800ms), with video quality up to 720p at 60 fps, ensuring smooth monitoring. However, the Raspberry Pi 3's hardware and network conditions may cause occasional buffering and reduced performance in poor network environments.

Notification Timing:

The PI sends notifications to the Blynk app within 1–3 seconds of detecting an object or face. While this target should be met normally, network speed and device load may cause slight delays. The project will continually refine processes to ensure the fastest possible response time. Users can enable a whitelisting feature to prevent notifications when known individuals are detected, streamlining the experience.

Conclusion

In Conclusion, TrueDetect integrates machine learning into home security, using already available and well-researched libraries and ML models to enhance homeowners' experiences and provide safety and security. Real-time streaming, notifications, and privacy-conscious local processing make TrueDetect an effective and user-friendly tool for modern home security.

References

1. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
2. OpenCV. (2024). Haar Cascade Classifier. *OpenCV Documentation*. https://docs.opencv.org/2.4/doc/tutorials/objdetect/cascade_classifier/cascade_classifier.html
3. Blynk. (2024). Blynk IoT Platform. *Blynk Documentation*. https://docs.blynk.io