

Front Porch Buddy

A modern home camera notification system for families with neighbors.

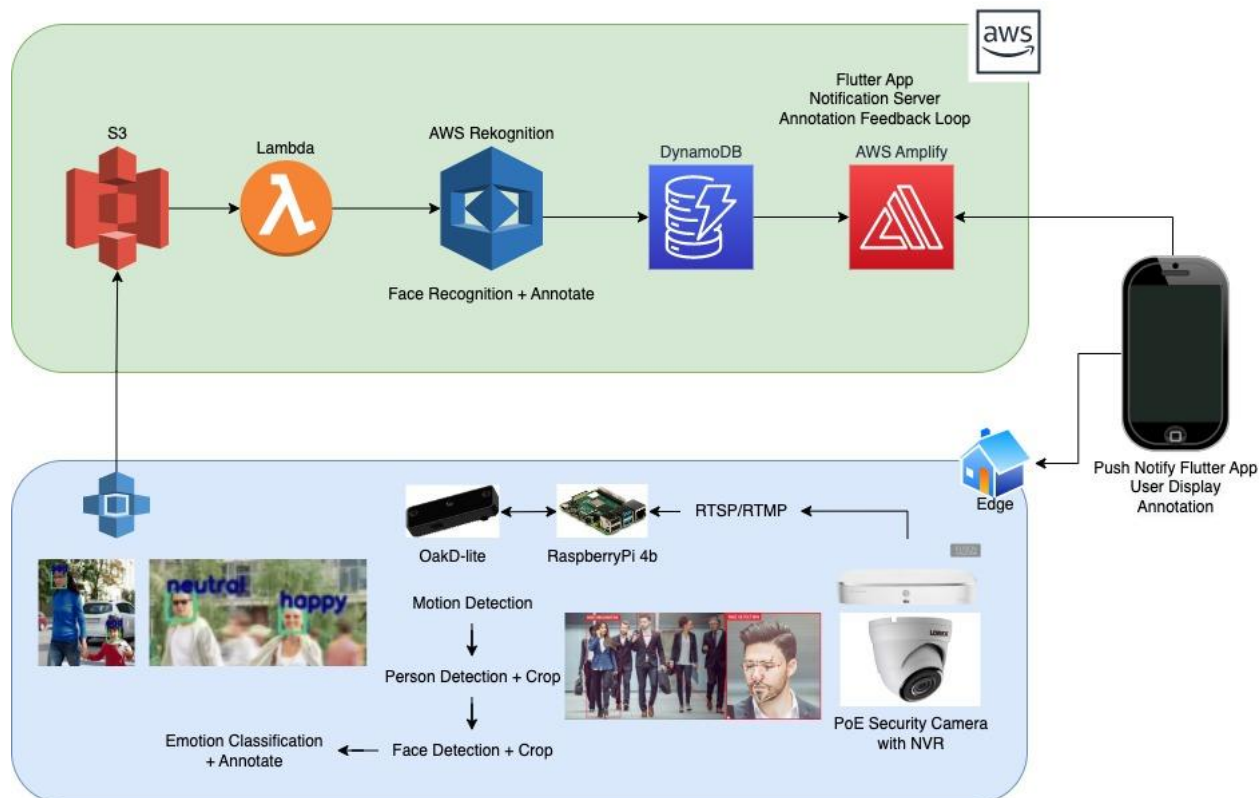
Motivation

In today's fast-paced world, there are concerns about missing out on significant events at the front door when individuals are away from home. These concerns are particularly important when it comes to the safety and well-being of children and the elderly. Installing a video doorbell or smart security camera can provide peace of mind and help individuals stay connected to their homes while they are away. The motivation of this assignment is to make our security camera smarter.

This capstone project explores the development of a system that utilizes an edge-deployed OakD-Lite camera, Raspberry Pi 4b, PoE Security Camera feed, AWS Cloud services, and a Flutter iOS app to create a smart monitoring system for the front door. The system leverages motion detection, object detection, emotion detection, and facial recognition to filter out noise and provide real-time notifications when relevant events occur. This project not only demonstrates the feasibility of such a system, but also its potential to reduce the stress of being unaware of significant events that may occur at the front door.

Proposed Architecture

Below is proposed architecture that shows the end to end flow of a user journey -



Hardware:

- PoE security camera for video capture
- OakD-Lite camera for video capture and detection models
- Raspberry Pi 4b to coordinate edge pipeline
- iOS device for user notification and data labeling

Software:

- OpenCV
- TensorFlow Lite
- Flutter

Machine Learning Models:

- Person detection
- Face segmentation
- Emotion classification
- Face instance recognition model
- (optional) Motion detection

Protocols:

- RTSP / RTMP (or reuse existing sequences from NVR)
- HTTP (required) and MQTT (optional)
- (existing installation) PoE for powering the cameras

Web Services:

- AWS Cloud for ingesting annotated data and further analysis
- AWS Rekognition or SageMaker (optional)
- Flutter app for user notification and annotation
- Greengrass and AWS IOT (optional)

Timeline

Checkpoint t Date	Edge Tasks (Rahul)	Cloud Tasks (Anand)

March 19	<ul style="list-style-type: none"> • Setup OakD-Lite camera and Raspberry Pi 4b with PoE security camera. • Filter stream into sequences with motion detection. • Person detection model with the motion detection algorithm to detect objects only when motion is detected. 	<ul style="list-style-type: none"> • Set up the cloud-based AWS services for ingesting and analyzing the annotated data. • Develop the user notification and labeling system using the Flutter iOS app. • Develop the machine learning models for object detection and facial recognition.
April 2	<ul style="list-style-type: none"> • Develop the face crop algorithm to extract the facial features of any person detected. • Prepare the system for integration with the cloud. 	<ul style="list-style-type: none"> • Develop the user interface for the iOS app. • Develop the communication protocols between the edge and cloud, such as HTTP and MQTT.
April 16	<ul style="list-style-type: none"> • Test the edge computing system and make any necessary modifications. • Enhance the features of the edge computing system, if necessary. 	<ul style="list-style-type: none"> • Test and deploy the cloud-based web services and iOS app. • Enhance the features of the cloud-based web services and iOS app, if necessary.
April 30	<ul style="list-style-type: none"> • Write up the report summarizing the project's goals, technical components, and results. • Develop a video summary of the project that highlights the system's features and capabilities. • Present the final deliverables to stakeholders, including a demo of the system in action. 	

Group

Rahul Singhal (NetID: rahuls11)

Anand Todkar (NetID: atodkar2)