VigilEye

Home Security Automation using Face Recognition and Suspicious Activity Detection

Table of contents

Problem Definition

- Home security remains a major concern due to increasing threats like burglary and unauthorized access.
- Many traditional security systems rely on basic motion detection, leading to false alarms and difficulty distinguishing between authorized and unauthorized people.
- Current systems do not effectively balance ease of use, ID verification, and breathing comfort, affecting overall security integration.
- Elderly individuals often find it difficult to interact with these systems due to poor user interfaces.

Objective

- Profile-based restrictions for visitors to personalize access control.
- Automatically detect unusual activities using CCTV or hidden cameras
- Provide a simple and intuitive user interface for easy operation
- Ensure accessibility for elderly people with user-friendly features.

VigilEye

- VigilEye is an evolved home security system combining three essential units working as a whole
  - mobile application
  - face recognition tool
  - surveillance camera
- The mobile app allows users to create and manage their profiles, while the face recognition system ensures accurate identification of individuals.
- The discreetly placed surveillance camera captures and analyzes activities in real-time, detecting suspicious behavior.
- Additionally, VigilEye offers profile-based access restrictions, automatic detection of unusual activities, and an elder-friendly interface with voice control to ensure easy operation for seniors, making it a comprehensive and accessible security solution.

Ideas

Suspicious Activity Detection

prepared dataset for said model

Profile-Based Restrictions

Add, update and maintain a repository of faces

Voice Control for Elders

Enable elderly people to use voice commands for easier access and operation.

Tech Stack

DeepFace

Face Recognition

Yolo v8

suspicious activity detection

Flask

RestAPI

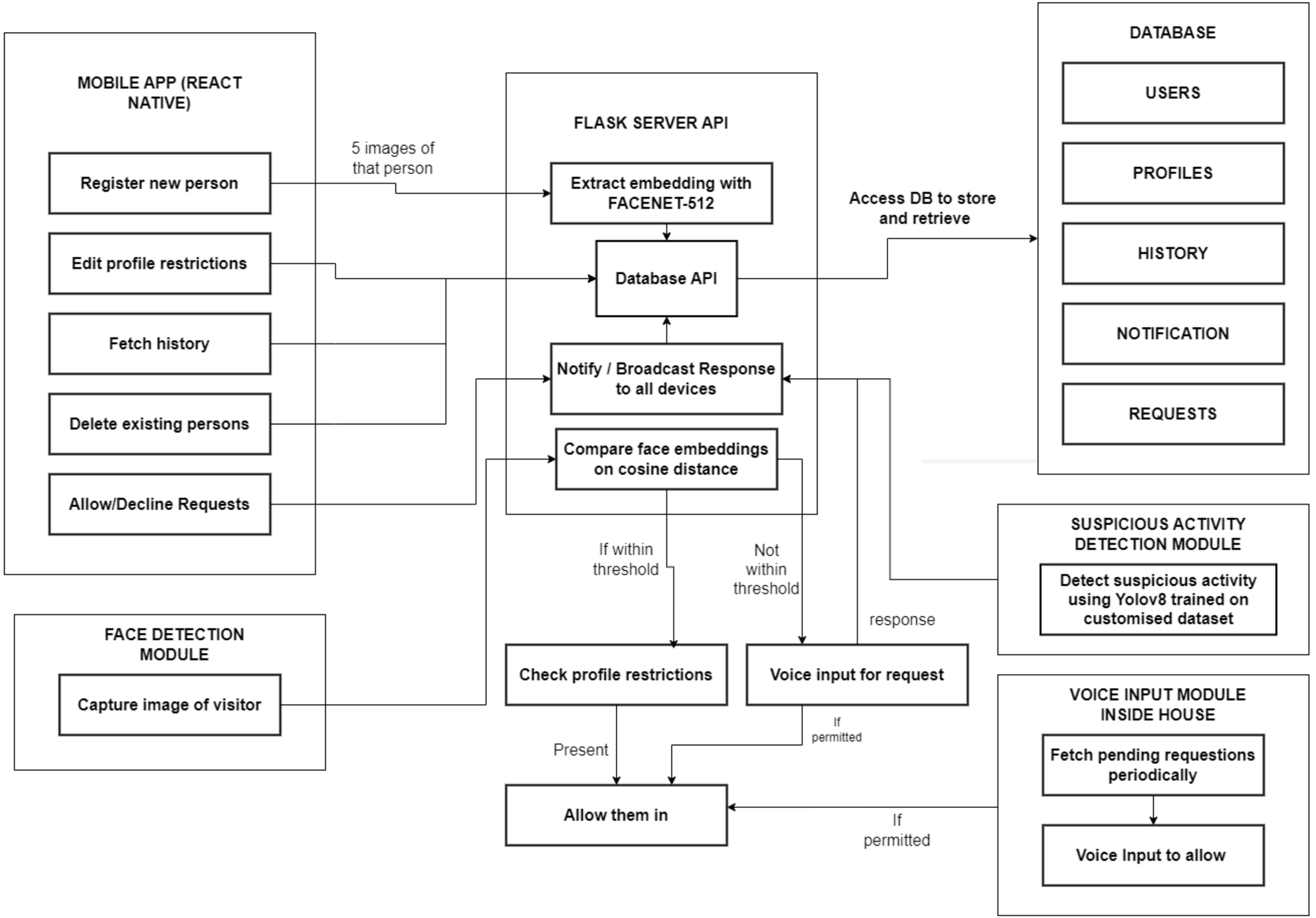
React Native - Expo

Android development

React Native Paper

UI Component Library

System Architecture

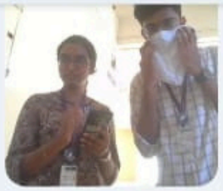


Detailed Design

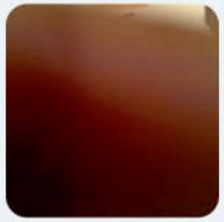
- MongoDB was selected to handle unstructured data, such as facial embeddings, due to its flexibility and scalability.
- API to interact with the DB and models was built using Flask as python gives access to ML libraries such as scikit learn, PyTorch and numPy.
- The mobile app was developed using React Native due to its platform independent framework with high developer velocity
- For the UI, React Native Paper was used to align with Google's Material Design principles.
- Suspicious activity detection is trained using YOLOv8 for building state-of-the-art Image detection CNNs

Screenshots

A masked person found
16:14:05



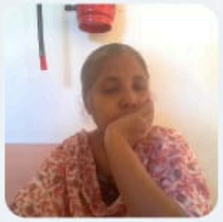
Something blocked the camera!
16:12:44



Salai
16:11:41

Unknown person
16:11:03

Unknown person
16:10:11



Many people found in front of your house!
15:14:53

VigilEye

Add a new Person

Name
Lakshmi

Relation
Sister

Pick Images

Tap on an image to remove it

Suspicious activity detected! at 22/10/2024, 11:55:49

Suspicious activity detected at 22/10/2024, 12:02:50

Permissions

Salai is at the door

Paper delivery

Open door

Deny

Current profile - 2

F
Friend

O
Old People

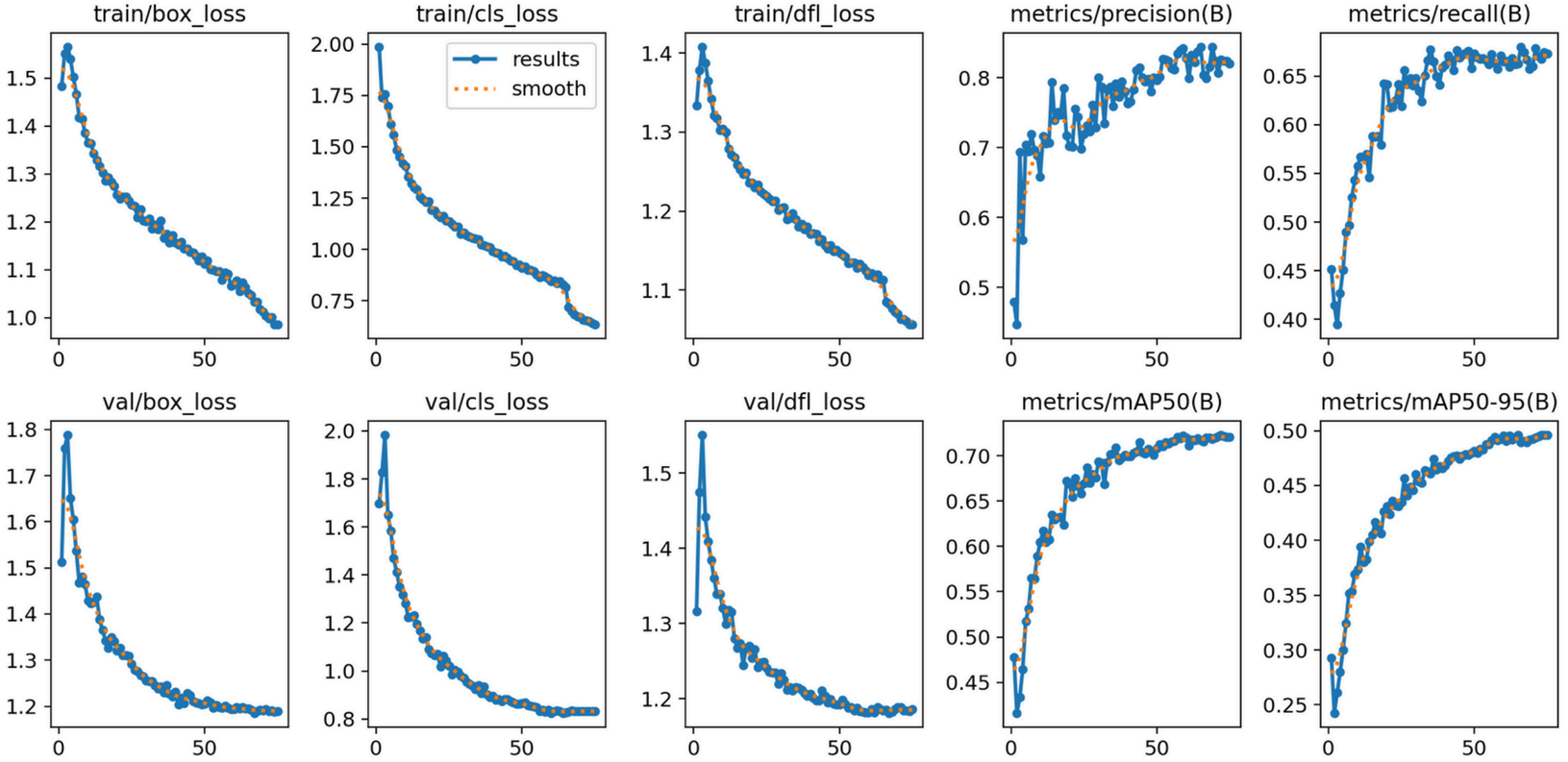
C
Children

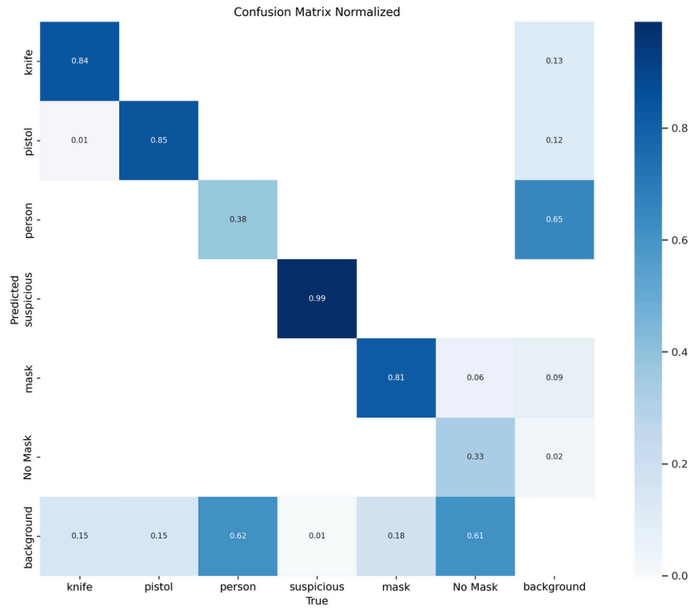
D

P

Suspicious Activity Detection

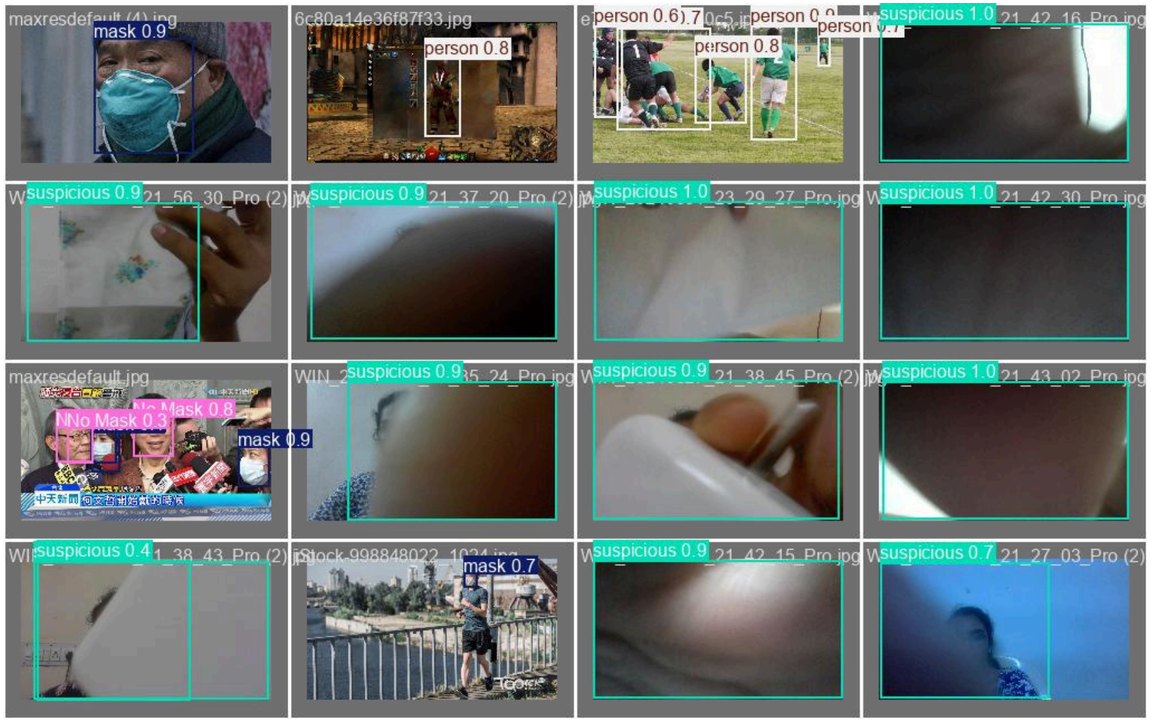
Results





Confusion matrix

Suspicious Activity Detection



Predictions

Meet The Team

1) Varun Karthik T - 2022115060

2) Ezhil Dhiraviya J - 2022115077

3) Salai Kowshikan S - 2022115081

Mentored by:

Dr. K Vani,

Professor, Department of IST