UNIVERSITY OF SRI JAYEWARDENEPURA

Faculty of Engineering

CO4002 Engineering Project

PROJECT PROPOSAL REPORT

Group No	: 07
Supervised By	: Dr. Sampath Edirisinghe
Project Name	: WLAN EDGE COMPUTERS
Index No	: 18/ENG/112 - Pasindu Uduwela
	: 18/ENG/ - Kesara Premabandhu
	: 18/ENG/ - Thisari Madurapperuma
	: 18/ENG/ - Charuka Jayamali

Contents

List Of Figures

1. Abstract

Edge computing enables the processing of data close to the network edge. Hence, edge computing provides low latency, reduces the data volume transported in the networks, reduces the cost of operation and increases privacy of data. A WLAN edge computer adds computational and storage capabilities to the household or industrial WLANs. Thereby, services such as surveillance video processing, IoT data processing, industrial control can operate with minimum latency.

This project would involve a design of a WLAN edge computer that directly connects to a Wi-Fi router. The devices in the network will connect to the WLAN edge computer, where server applications would process the data of these applications.

The objective of this project is to implement a WLAN Edge Computer that can be used in a home environment with the features, Surveillance video processing, human recognition, pose detection, home companion chat bot, smart applications and a home media server.

2. Background

Wireless local area networks (WLANs) and edge computing are two separate technologies that are often used together to improve the performance and efficiency of networked systems.

WLANs are wireless networks that use radio waves to provide wireless high-speed Internet and network connections. They can be used to connect devices such as laptops, smartphones, and tablets to the Internet, as well as to connect devices within a local area network (LAN).

Edge computing is a distributed computing paradigm that brings computation and data storage closer to the sources of data, such as sensors, cameras, and other devices. This allows for faster processing and analysis of data, as well as reduced latency and bandwidth requirements.

WLAN edge computing can be used to improve the performance and efficiency of surveillance video processing systems. One example of this is using edge devices such as gateways or cameras with built-in edge computing capabilities, to perform tasks such as video compression, object detection, and facial recognition. This can help to reduce the amount of data that needs to be transmitted over the network to a central location for further processing, and also reduce the amount of data stored and processed in the cloud.

Another example is using edge computing to perform real-time analytics on the surveillance video, such as identifying and tracking objects, detecting suspicious behavior, and raising alarms. This can help to improve the effectiveness of the surveillance system, and also reduce the amount of data stored and processed in the cloud.

Overall, WLAN edge computing can be used to improve the performance and efficiency of surveillance video processing systems by performing video analysis and analytics at the edge of the network, close to the cameras and other sensors, reducing the amount of data that needs to be transmitted over the network, as well as reducing latency and bandwidth requirements.

WLAN edge computing also can be used to improve the performance and efficiency of smart home applications. The edge devices can collect data from various sensors, such as temperature and motion sensors, and perform tasks such as controlling lights, thermostats, and security cameras.

3. Related work

There have been several research efforts and projects in the area of WLAN edge computing, and many of them have been published in scientific journals and conferences.

In contrast, the paper[1] presents a smart video surveillance system executing AI algorithms in low power consumption embedded devices. The computer vision algorithm, typical for surveillance applications, aims to detect, count and track people's movements in the area. This application requires a distributed smart camera system.

The proposed architecture in paper [2] consists of a Convolutional Neural Network (CNN) enabled in an edge device, with reduced computational complexity, which classifies various actions like Pulling, pushing and other hand movements and locates the identified activities in the image frame using bounding boxes. The proposed architecture gives an alert whenever a suspicious activity is detected.

The main contribution of the paper is to build a smart camera system capable of processing videos locally, at the edge nodes, using embedded devices. For this reason, we need to review the design and capabilities of constrained devices to analyze if they can execute real-time artificial intelligence processing. Shi [3] defines edge computing as "enabling technologies that allow computing to be performed at the edge of the network, on downstream data on behalf of cloud services and on upstream data on behalf of IoT services". This allows using the full potential offered by artificial intelligence, without losing computing capacity in the processor.

In paper [4], researchers face up security and availability issues in smart homes and propose an edge-of-things solution that focuses on putting the management of the home at the edge. The management is controlled by the network operator in a similar way as occurs with current set-top-boxes for multimedia streaming at home. They propose an architecture for this system, implement the necessary modules and test it from the point of view of security and availability. The results show that the proposed edge-of-things solution is able to solve many of the challenges that current smart home applications present.

4. Problem Statement

Applications like surveillance video processing requires high performance and expensive hardware to process. So usually cloud computing is used for computing for this type of applications. Cloud computing is a model of delivering computing resources and services over the internet. It allows users to access and use shared computing resources, such as servers, storage, and applications, on a pay-per-use basis. This eliminates the need for organizations to invest in and maintain their own physical hardware, reducing costs and increasing scalability. How ever the cloud computing is having some limitations for using them for all applications. The limitations of cloud computing for surveillance video processing include:

- Latency: The transmission of large video files over the internet can result in significant delays, making real-time processing challenging.
- Bandwidth Requirements: Processing high-resolution video requires a large amount of bandwidth, which can be expensive and limit the scalability of cloud-based systems.
- Security: The transmission of sensitive video data over the internet raises security concerns, as there is a risk of data interception or theft.
- Reliability: Cloud systems can be prone to outages or slowdowns, affecting the reliability of video processing.
- Cost: Storing and processing large amounts of video data in the cloud can be expensive, especially for organizations with limited budgets.
- Compliance: Some industries have specific regulations around the storage and processing of sensitive video data, making cloud computing a challenge for compliance.
- Technical Expertise: Cloud computing requires technical expertise and specialized skills to set up and maintain, making it difficult for organizations without IT resources.

These limitations can be addressed through WLAN Edge computing concept by bringing the computational part inside to the Wireless LAN. WLAN Edge computing is a technology that allows processing and storage of data at the edge of a network, closer to the source of the data. It is beneficial for the following reasons:

- Lower Latency: Data processing at the edge reduces the time it takes for data to travel to the cloud and back, providing faster response times.
- Improved Reliability: By processing data locally, edge computing can improve reliability by reducing the dependence on a remote server and ensuring data is processed even if there is an internet outage.
- Reduced Network Bandwidth: By processing data at the edge, the amount of data transmitted to the cloud is reduced, reducing the demand on network bandwidth.
- Enhanced Security: By processing sensitive data locally, the risk of data being intercepted in transit is reduced.
- Cost Savings: By reducing the need for cloud computing resources, WLAN Edge computing can help to lower costs for data processing and storage.

Apart from the surveillance video processing, edge computer can be used to host several other services integrated. Smart home appliances can be controlled using the same edge computer without investing on a separated system. Personalized media server, Home companion chat bot are few other services available in the proposed WLAN edge computer.

5. Proposed design/objectives

In the project are two main applications that run on the WLAN. Those are surveillance video processing (SVP) and Smart Home Appliance (SHA) applications. The current video surveillance application can be worked in any environment according to the conditions. Currently for the proposed SVP we have planned to include the following features.

- Face recognition and face identification algorithms will be implemented.
- Theft and some vulnerable items like knives and guns will be detected.
- All the detection history will be
- If the above functionality is invoked by the system those data will be stored in a database and email or a message will be sent to the user or the admin of this system.

In the SHA application currently, the following features will be implemented. This application will consist of both hardware and software.

- A prototype device will be implemented to show that the user can control his or her household equipment using the proposed application.
- Energy forecasting algorithm will be implemented using LSTM.
- Forecasted and current energy usage details will be shown using an attractive dashboard.

In the project instead of using CCTV cameras we currently make Our Own WIFI camera system using web cameras. In the raspberry pi we are running a python program to the video streams that are getting through multiple cameras. For that we will be used vidgear high performance video processing python library to built the video streaming system. Currently we didn’t observed any latency issue using this library and tested with some dummy pose estimation module with real-time. Instead of CC TV cameras we will be using HIKVISON web camera that support 1080p video streams and as the main micro controller board we will be use raspberry pi 3b+ which has 1GB of memory.

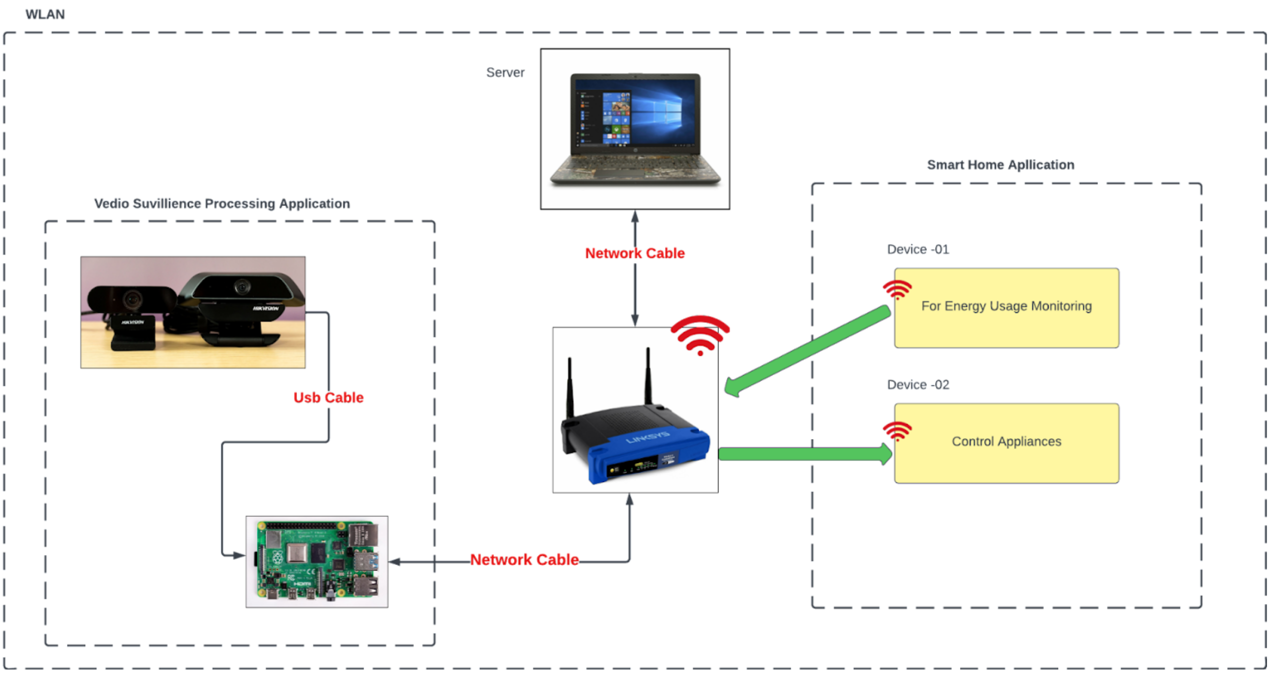


Figure 1: Illustration of the system

For smart home Application following Design was proposed.

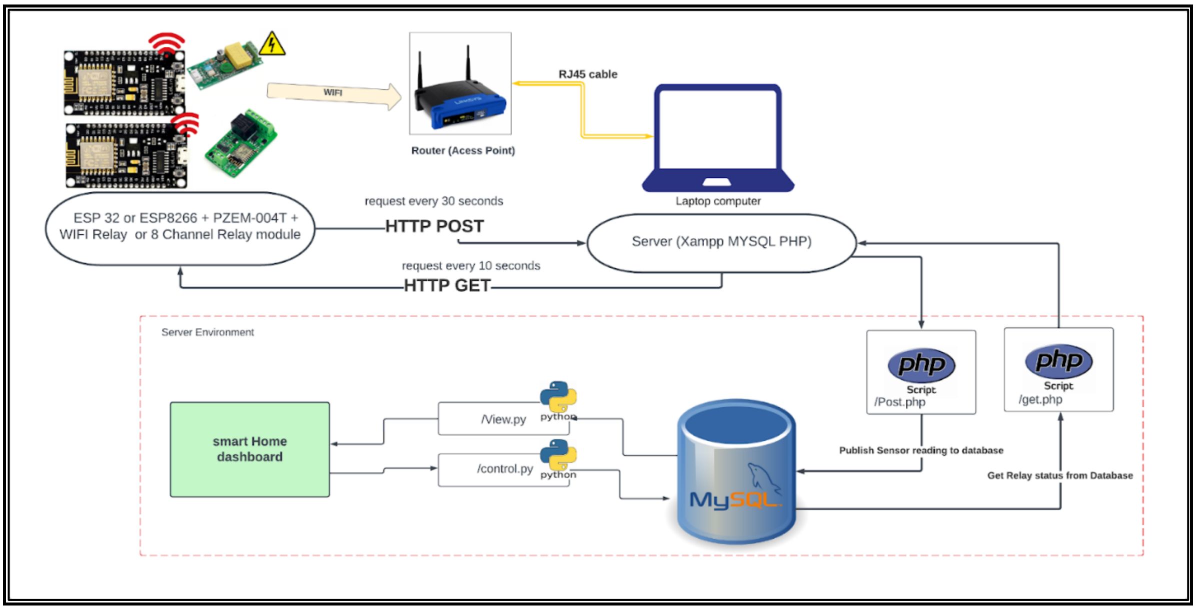


Figure 2 : proposed design for smart home controlling and monitoring system

The prototyped device will contain following microcontrollers and sensors. The circuitry will be as follows-

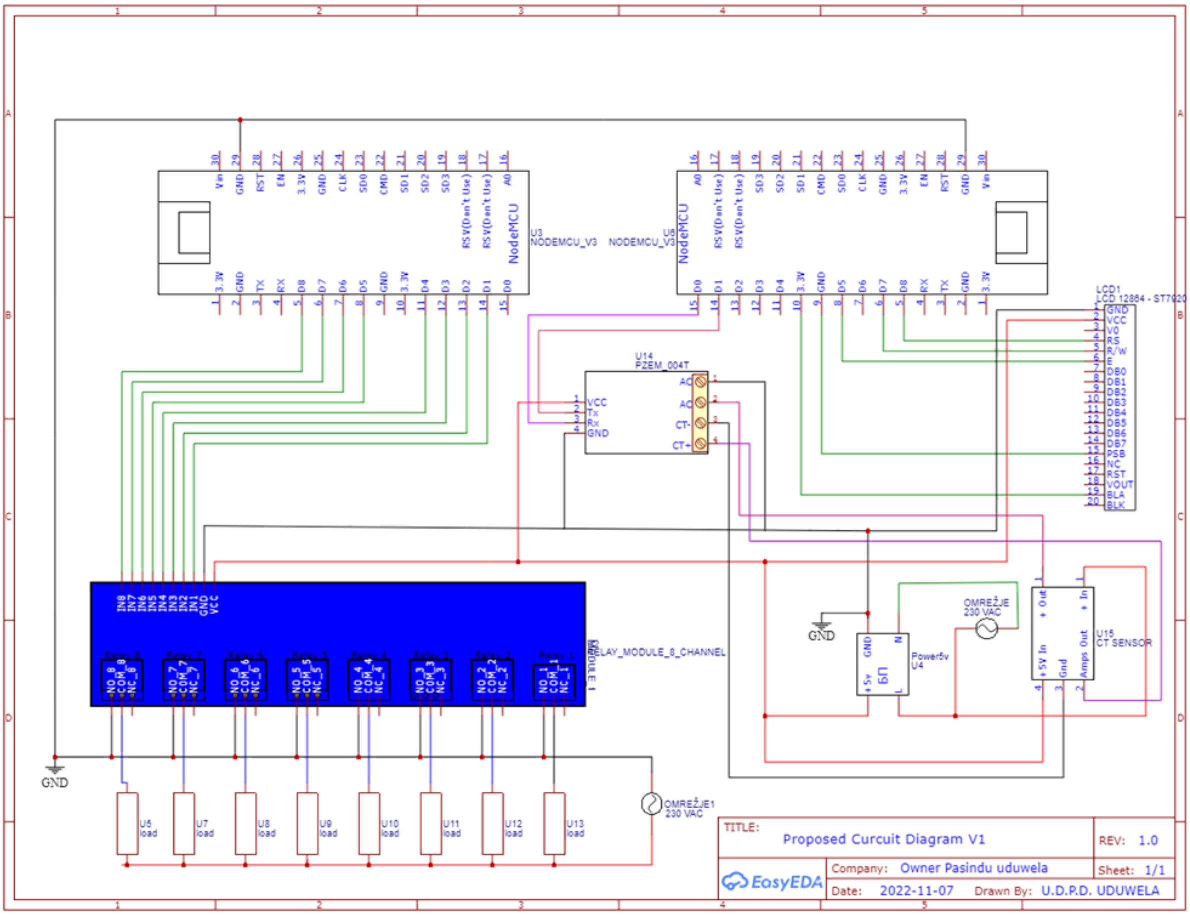


Figure 3 : Circuit diagram of the proposed device

The above shown circuit diagram will not be the final circuit design of the system. According to the current market available sensors and modules the final prototype device will be changed but the mentioned functionalities will be same. Using the above device, we will be able to measure the individual home electricity usage and control home appliances that are connect to the home Wi-Fi router. For that we need a free WIFI access point .If there are many free WIFI access points available it will connect to the high signal strength device. For that we have to modified the nodemcu code to get a Ip address from a keypad or to read a string from database at the device beginning. At the moment to avoid these issues we will be pre-configured our protype device to connect to a known WIFI network.

5.1 Surveillance video processing

It can be applied to new surveillance video to classify the poses of individuals in the video. This can be useful for identifying specific actions or behaviors, such as suspicious activity, in a surveillance setting. Additionally, this technology can be used in video analytics for better understanding of crowd behavior, traffic analysis, and many other use cases. In this case we use that pose classification. These are the steps following to the train the model.

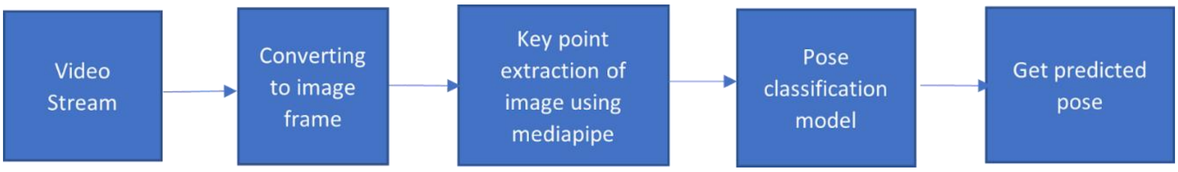


Figure 4 : Work flow of the surveillance video processing

5.2 Facial recognition for surveillance video processing

Facial recognition technology is used in surveillance video processing to automatically identify individuals by analyzing their facial features. This can be used to track and monitor individuals as they move through a specific area or building, and can also be used to compare images of unknown individuals to a database of known individuals in order to identify them.

6.0 Implementation Plan

6.1 Use Case Diagram

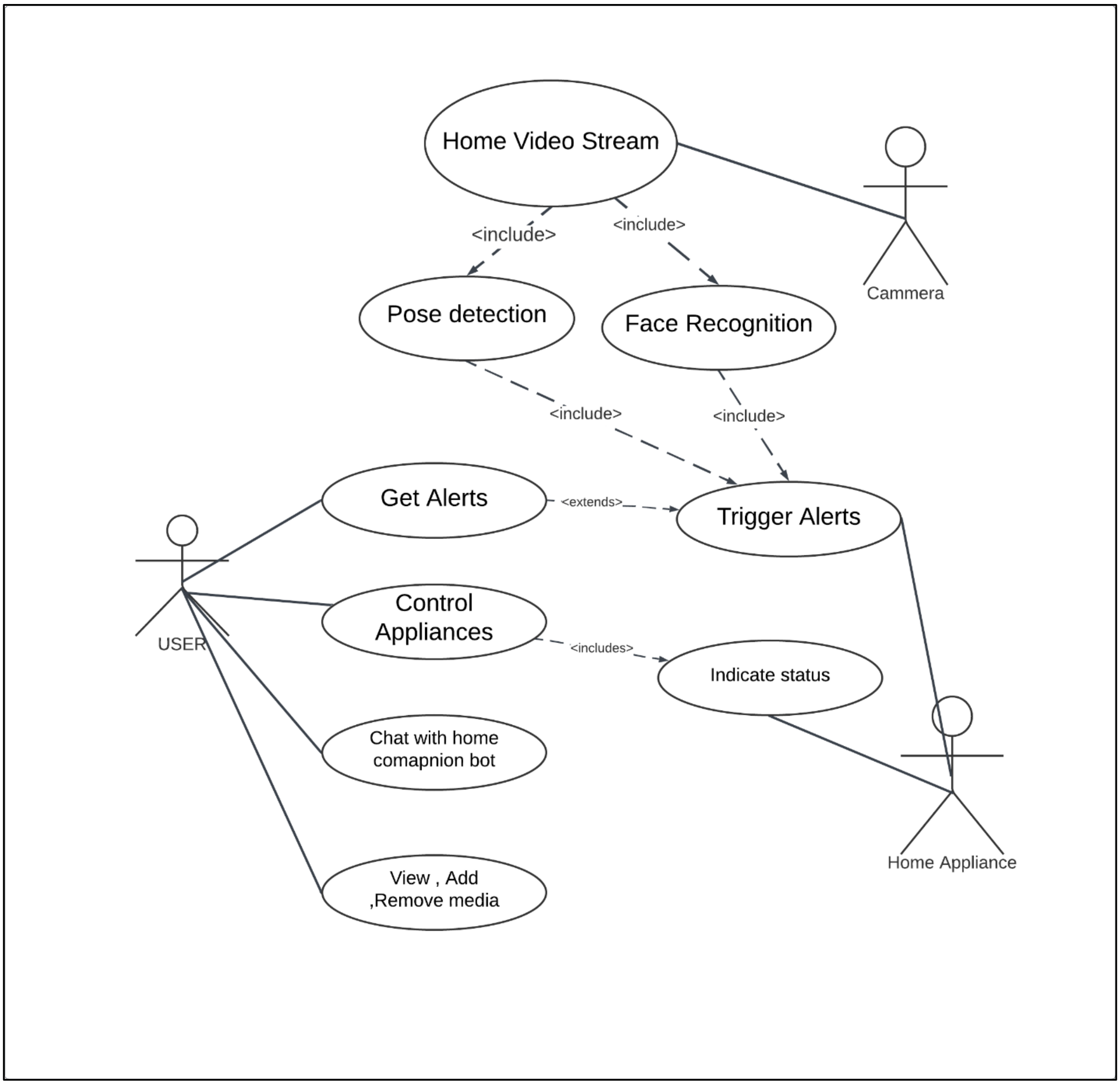


Figure 5 : Use Case diagram of the system

6.2 Core layer

1. Building and training the machine learning models for surveillance video processing system.

➢ **Pose classification Service**

- Collect and prepare a dataset of images or videos that contain the poses we want to classify.
- Install MediaPipe and its dependencies on the machine.
- Use MediaPipe's Pose Detection or Pose Estimation models to extract keypoint data from the dataset.
- Preprocess the keypoint data, if necessary, to format it for use in the classification model.
- Train a classification model using the keypoint data and desired classification labels.
- Use the trained classification model to make predictions on new data.
- Evaluate the performance of the model using metrics such as accuracy or F1 score.
- Optimize the model, if necessary, by adjusting the parameters, collecting more data, or using a different classification algorithm.
- Once the model is performing well, deploy it in an application using MediaPipe's framework.
- Before deploying the model, it should be converted to a docker image. Docker images can be run on any machine that has Docker installed, regardless of the operating system or dependencies. This makes it easy to deploy models across different environments, such as development, testing, and production.

➢ **Facial recognition service**

- **Data collection**: Collect a dataset of images and videos that will be used to train and test the facial recognition model. This dataset should include a variety of images of different individuals, with different lighting conditions, angles, and expressions.
- **Annotation**: Label the images and videos in the dataset by identifying the facial regions of interest, such as the eyes, nose, and mouth. This step is usually done manually, but can also be done using semi-automatic tools.
- **Model training**: Use the annotated dataset to train a facial recognition model using MediaPipe. This involves feeding the images and videos into the model, along with

the corresponding labels, and adjusting the model's parameters to optimize its performance.

- **Model evaluation**: Evaluate the trained model on a separate test dataset to measure its accuracy and performance. This allows you to identify any issues with the model and make adjustments as needed.
- **Deployment**: Once the model is trained and tested, it can be deployed in a live system, such as a surveillance camera, for real-time facial recognition. The model takes the video feed as input and outputs the facial features and the identity of the person.
- **Monitoring and Maintenance**: After deployment, it is important to monitor the performance of the model and the system, and make any necessary adjustments or updates to ensure optimal performance over time.

2. Building and training the machine learning model for Smart Home Application

➢ **<u>Energy Forecasting Service</u>**

- Build Energy forecasting Model using LSTM.
- Finally saved the model and use the model to do predictions on real-time data.

➢ **<u>Control appliances service</u>**

- Control the relay modules manually.

3. Setting up database system to store important data gathering from above three models. currently we are planning to keep separate database for both applications to store important data obtained from the models. But if time permits we will be make relational database that connect with both application.

6.3 Application layer

1. Web dashboard development for both applications using Flask. The web application will be built using python flask and it will e responsive to use with mobile screen.

7. Architectural Plan

For the entire application we are going to use monolithic architecture. We will try to run machine learning models separately as modules or services. Because some parts of the system might not functioned for very small datasets. Specially forecasting part would not functioned for out device .For that we will planed to used online available dataset to show the model is working correctly. on the other hand the ML model will converted to docker file for deployment easiness.

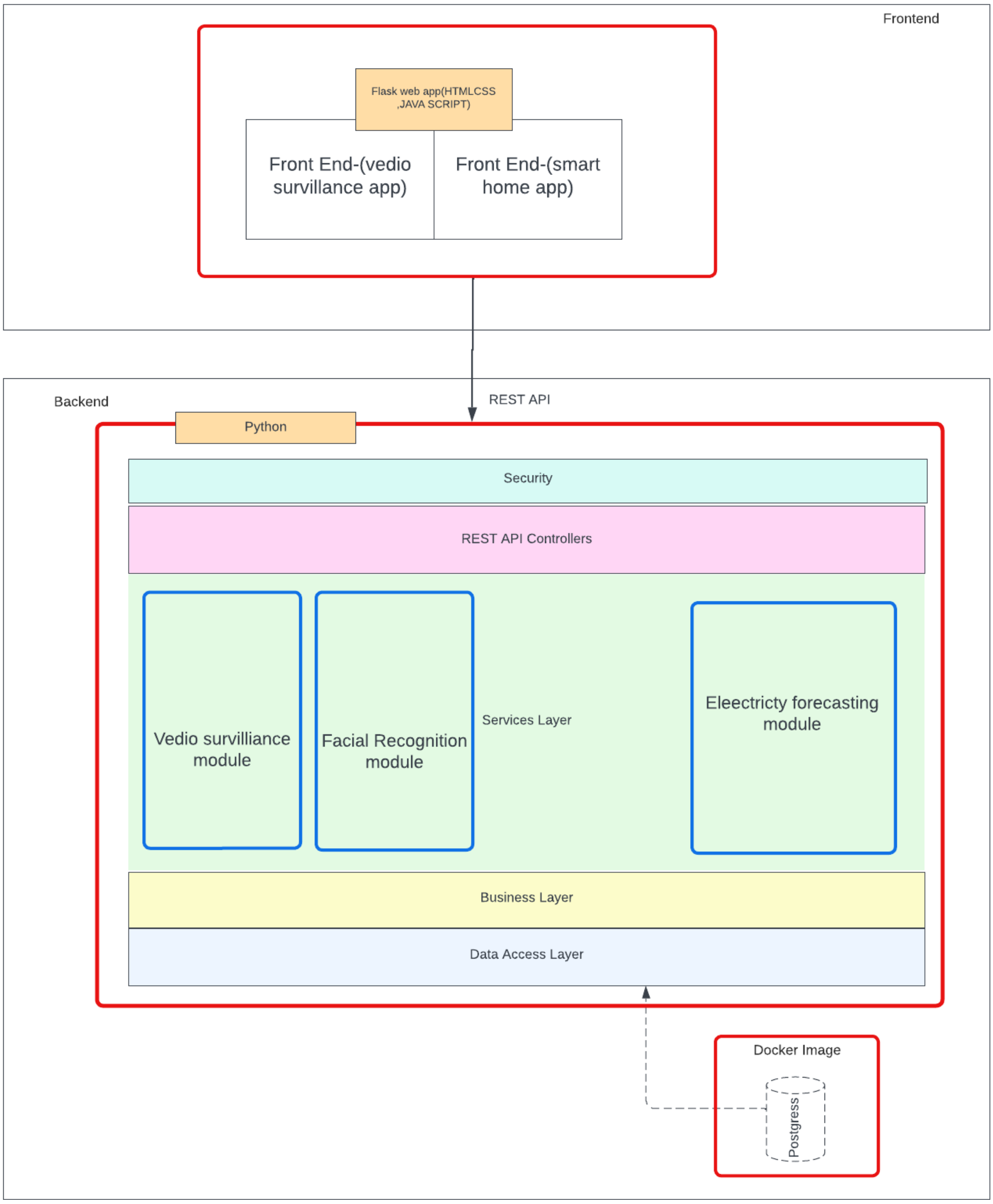


Figure 6-Propose system architecture

8. Milestones

- Background research and literature survey
- Building the client application for the camera setup (CC TV camera)
- Building a prototype device for electricity monitoring and controlling
- Build a model for electricity forecasting(daily , weekly and monthly forecasting)
- Building machine learning models for facial recognition and pose classification
- Building the server-side application using Microservices and REST APIs
- Create a dummy ML algorithm and add as a module or service .
- Implementing voice recognition module
- containerized all the services and add them to the back-end server
- manage our docker containers according to the system functionalities
- System Integration system testing and debugging

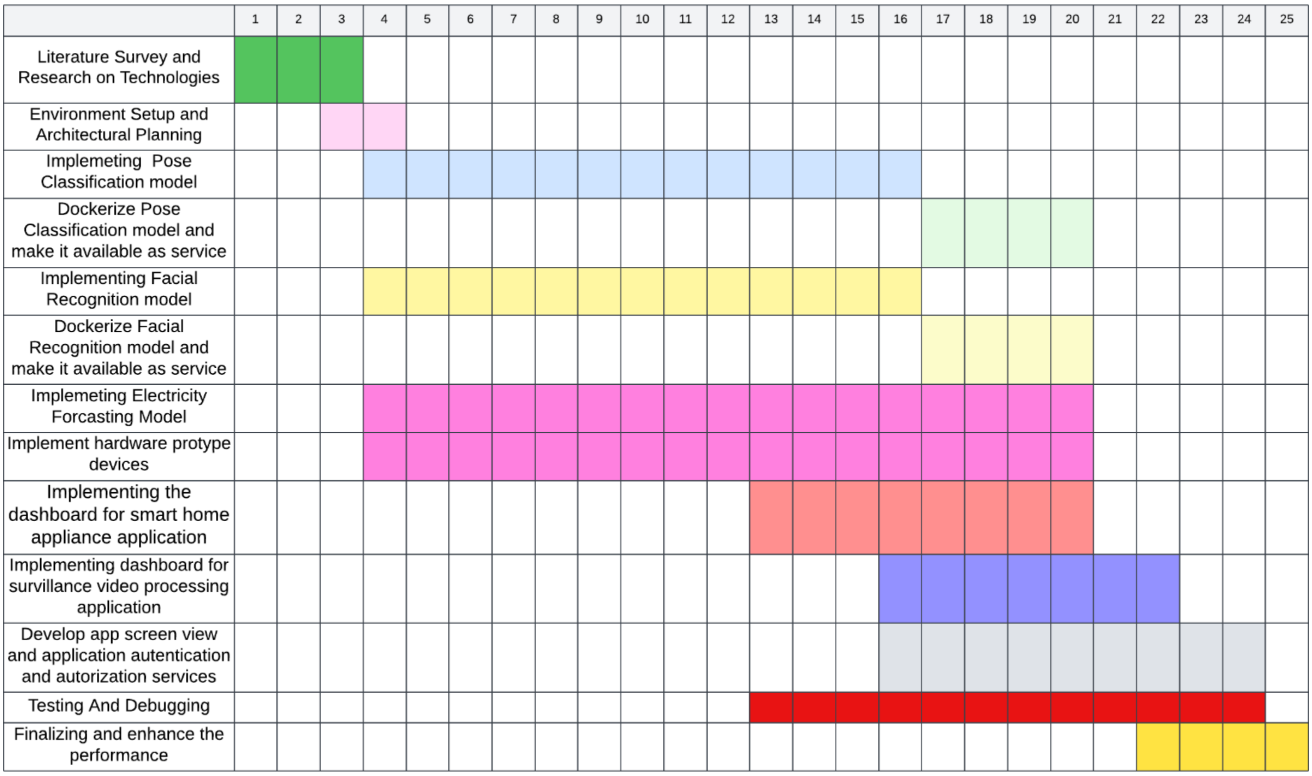
9. Testing Approach

1. Testing model performance and accuracy.
2. Test the porotype devices works correctly.
3. Furthermore we will do statistical analysis and performance analysis tests.
4. The prototype device and its data will be validated manually.

10. Challenges

- Major challenge encountered in this project is acquiring hardware resource needed for implementing the surveillance video processing feature. For implementing this feature, it is required to have CC TV cameras, which are capable of transmitting video stream over Wi-Fi network. To address this problem, we came up with an idea to make prototype of a CC TV camera with this capability, interfacing a USB web camera with a Raspberry Pi microprocessor. So, the Raspberry Pi itself work as a server and streams video and server computer work as a client in this scenario to capture each video stream from cameras.
- When implementing the face recognition feature, it is not giving a reliable output due to the inability of capturing a better frame for processing. This happens because, usually CC TV cameras are fixed in a higher position from the floor level so that all the faces will be captured in a view from a higher level. But the data sets which are available are not included of images in like this for training the model. The other problem is the faces in captured frames are not much clear for processing due to the depth.
- Video processing requires high performance hardware to give quick responses. So, this is a challenge when implementing the WLAN Edge Computer for a home environment to be used in a typical home computer with low spec performance. So, the video surveillance feature must be optimized to work smooth on low spec computers by reducing the frame processing rate.
- To do the energy prediction more accurate the model accuracy score should be in higher level. But using one model we cannot give proper comments about the model. To avoid that we have to go for an ensemble model+ hyper parameter optimization techniques instead of relying on one basic Algorithm. Due to the time and the project complexity designing an ensemble type energy forecasting model is not feasible to develop.
- Smart Home application included in the WLAN Edge Computer, is capable of controlling the home appliances like bulbs, fans and other electrical appliances through the web interface. And also, some bulbs will be automatically switched on based on the human detections of video surveillance application. Integrating these separate systems to form the final solution is a huge challenge in this project.

11. Timeline


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
Literature Survey and Research on Technologies
Environment Setup and Architectural Planning
Implemeting Pose Classification model
Dockerize Pose Classification model and make it available as service
Implementing Facial Recognition model
Dockerize Facial Recognition model and make it available as service
Implemeting Electricity Forcasting Model
Implement hardware protype devices
Implementing the dashboard for smart home appliance application
Implementing dashboard for survillance video processing application
Develop app screen view and application autentication and autorization services
Testing And Debugging
Finalizing and enhance the performance

References

[1] Smart Video Surveillance System based on Edge Computing (no date). Available at: https://www.researchgate.net/publication/351252277_Smart_Video_Surveillance_System_Based_on_Edge_Computing (Accessed: January 20, 2023).

[2] panelAishwaryaD.PersonEnvelopeMinuR.I.Envelope, A.links open overlay et al. (2021) Edge computing based surveillance framework for Real Time Activity Recognition, ICT Express. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S2405959521000540 (Accessed: January 20, 2023).

[3] Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge computing: Vision and challenges. IEEE Internet Things J.2016,3, 637–646

[4] Batalla, J.M. and Gonciarz, F. (2018) Deployment of Smart Home Management System at the edge: Mechanisms and protocols - neural computing and applications, SpringerLink. Springer London. Available at: https://link.springer.com/article/10.1007/s00521-018-3545-7 (Accessed: January 20, 2023).

[5]. A. Fortes, "Deep Learning based human pose estimation using opencv and MediaPipe," Medium, 04-Jun-2021. [Online]. Available: https://medium.com/nerd-for-tech/deep-learning-based-human-pose-estimation using-opencv-and-mediapipe-d0be7a834076. [Accessed: 01-Nov-2022].