Raspberry Pi-based Face Recognition Door Lock System

Name: Seifeldin Sherif Elnozahy

Student ID: 1191102388

Supervisor: DR. Chinnaiyan Senthilpari

Modrator: DR. Lee Chu Liang

Date: 18-JUN-2024

Multimedia University

Table Of Contents

Introduction

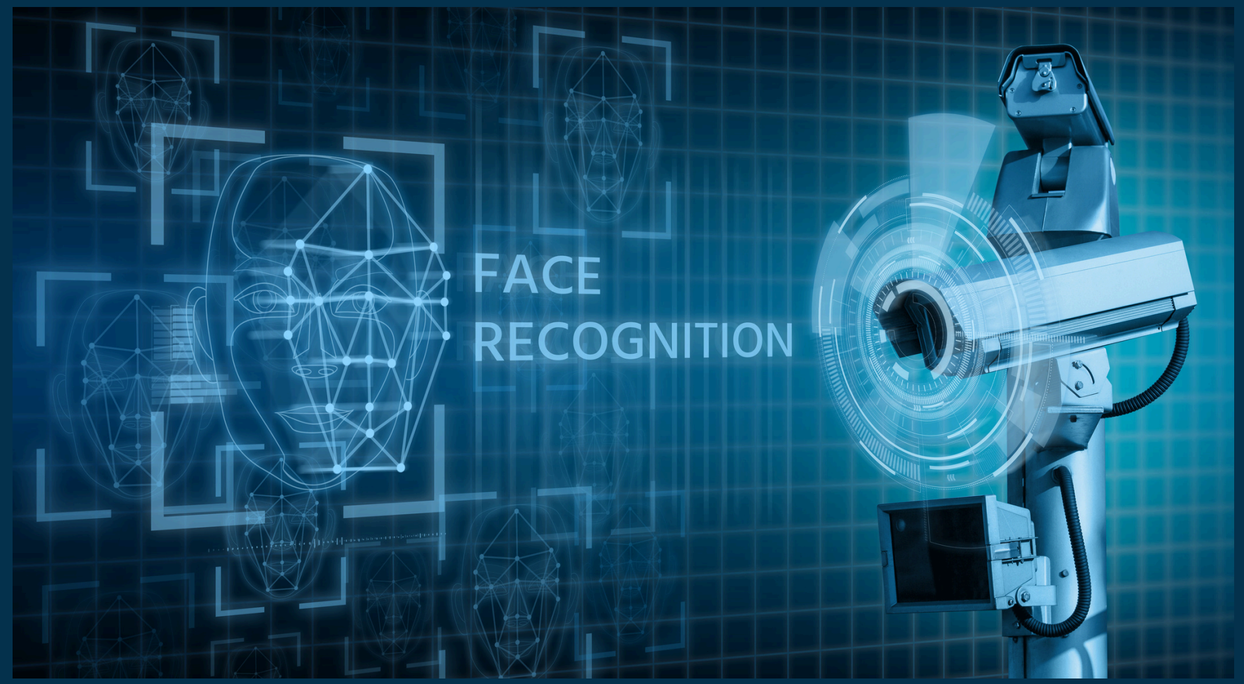
In recent years, security has become a crucial aspect of our daily lives. Traditional door locks, despite their long history, have several limitations such as the risk of losing keys and unauthorized duplication. These issues highlight the need for more advanced and reliable security solutions. Facial recognition technology, leveraging AI and machine learning advancements, offers a secure and user-friendly alternative.

Facial recognition technology eliminates the need for physical keys, reducing the risk of unauthorized access and enhancing user convenience. By accurately identifying individuals based on their unique facial features, it provides a non-intrusive and efficient security solution. This project aims to develop a Raspberry Pi-based face recognition door lock system, providing a scalable and affordable solution to modern security challenges.

Problem Statement

Traditional key-based door locks pose significant security risks, such as lost or stolen keys, unauthorized key copying, and the inconvenience of managing physical keys. These issues compromise the safety of homes and businesses and add user frustration. Therefore, there is a pressing need for advanced, reliable, and user-friendly security solutions that effectively address these vulnerabilities and provide enhanced protection.

WHY
THIS PROJECT?



Motivations:

- Enhance security by reducing vulnerabilities associated with traditional locks.
- Improve efficiency in access control systems.
- Increase user-friendliness by eliminating the need for physical keys.
- Leverage advancements in AI and machine learning for better security solutions.

Objectives:

- Develop a facial recognition door lock system using Raspberry Pi.
- Ensure the system is scalable and can be adapted for various applications.
- Utilize open-source software to keep costs affordable.
- Create a reliable and robust system capable of real-time face recognition.
- Integrate the system with existing door lock mechanisms for practical deployment.

Literature Review

Facial recognition technology has been extensively studied for its application in enhancing security systems. Research has focused on advancing the accuracy and reliability of algorithms through innovations in deep learning and computer vision. These advancements enable systems to effectively identify individuals based on unique facial features, offering a robust alternative to traditional authentication methods. Studies also highlight practical challenges such as real-time processing limitations and the need for consistent performance across different environments. Overcoming these challenges requires ongoing research into algorithmic enhancements and practical implementations that ensure reliable operation under diverse conditions.

Beyond security enhancements, facial recognition technology has shown promise in automating identity verification processes across various domains, including smart homes, public facilities, and commercial environments. By reducing reliance on physical credentials, these systems streamline access control and operational efficiencies. This dual focus on security improvement and operational enhancement underscores the transformative potential of facial recognition technology in reshaping security protocols and access management practices.

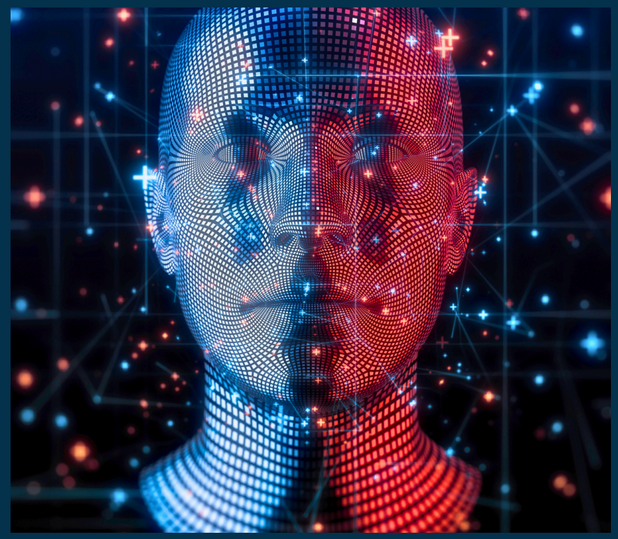
SUMMARY

Paper	Techniques/Algorithms Used	Key Findings	Challenges/Limitations
A Face Recognition Method In Machine Learning (ML) For Enhancing Security In Smart Home	CNN, CNN Mobilenet, CNN Alexnet	High accuracy in recognizing faces	Data preparation, computational complexity, privacy concerns
Raspberry Pi-Powered Door Lock with Facial Recognition	OpenCV, Haar Cascade	Effective for secure access control	Processing power and time for training, importance of reliable internet connectivity
Face Recognition Door Lock System Using Raspberry PI	Haar Cascade Classifier, OpenCV	Accurate face detection, cost-effective solution	Real-time processing limitations
Face Recognition Door Lock System Using Raspberry Pi	PCA, OpenCV	Cost-effective, low power consumption	Integration with existing security infrastructure
Home Security System Based On Facial Recognition	HOG algorithm, IoT, GSM	High precision and efficiency, remote access control	Privacy concerns, user acceptance

Project Concept

INTEGRATION OF RASPBERRY PI AND FACE RECOGNITION TECHNOLOGY

The project centers on a Raspberry Pi-based face recognition door lock system, combining affordability with advanced security features. The Raspberry Pi serves as the central computing platform, leveraging its capabilities to process images captured by a standard camera module



Project Concept

01

UTILIZATION OF AI AND MACHINE LEARNING FOR ENHANCED SECURITY

AI technologies such as OpenCV and TensorFlow are integrated to enable facial recognition capabilities. These frameworks empower the system to learn and improve accuracy over time, ensuring robust access control without the need for specialized lighting conditions or night vision.

Project Concept

02

SEAMLESS INTEGRATION WITH SMART HOME SYSTEMS

The system leverages network connectivity to seamlessly integrate with smart home environments. This enables the camera to recognize authorized faces and automatically unlock doors, ensuring a secure and convenient access experience without the need for physical keys or manual intervention.

Project Concept

03

SIMPLIFYING USER EXPERIENCE AND ENHANCING SECURITY MEASURES

By eliminating the need for physical keys or codes, our solution simplifies user experience while bolstering security. Facial recognition mitigates risks associated with traditional access methods, offering a seamless and intuitive way to control access in residential and small business environments.

Techniques and Methods Utilized

1. **Face Detection:**

Employed advanced face detection techniques using the Haar Cascade classifier provided by OpenCV. This classifier effectively identifies facial features based on learned patterns, enabling precise detection even in varying environmental conditions.

2.Deep Learning for Face Recognition:

The system utilizes a sophisticated deep learning approach for accurate face recognition. Specifically, we integrate a pre-trained Convolutional Neural Network (CNN) model tailored for facial recognition tasks. This model has been fine-tuned to distinguish between authorized and unauthorized individuals with high accuracy, ensuring robust security in access control scenarios.

Libraries and Tools Used

01 OPENCV (CV2)

Purpose:

- Integral for video capturing, image processing, and robust face detection capabilities within system.

Description:

- OpenCV provides essential functionalities for real-time video processing, enabling efficient detection and tracking of faces using the Haar Cascade classifier.

02 NUMPY

Purpose:

- Essential for numerical operations and efficient manipulation of images.

Description:

- NumPy's powerful array operations facilitate rapid computations and transformations on image data, crucial for preprocessing tasks in our face recognition pipeline.

03 TENSORFLOW

Purpose:

- Utilized for loading and executing a pre-trained Convolutional Neural Network (CNN) model.

Description:

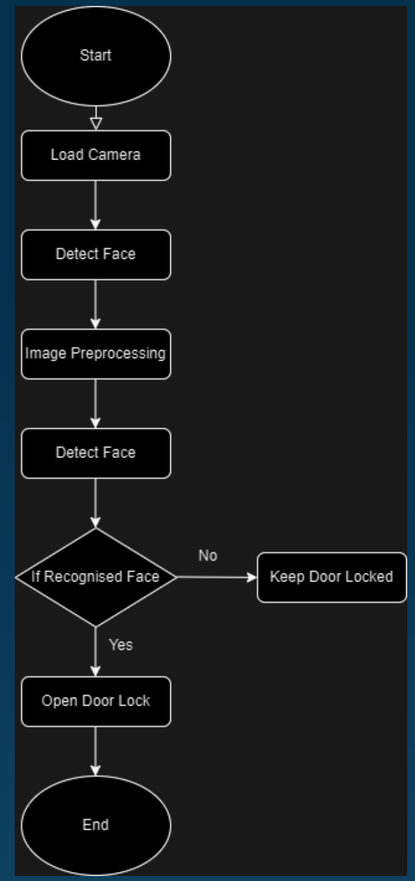
- TensorFlow supports seamless integration of deep learning models, enabling accurate face recognition through advanced neural network architectures tailored for our specific security application.

Execution Details

- **Image Preprocessing:**
  - Convert images to grayscale to reduce computation complexity.
  - Resize images to match the input size required by the CNN model (224x224 pixels).
  - Normalize pixel values to ensure consistent data input for accurate face recognition.

- **Real-time Processing:**
  - Capture frames from the webcam in real-time.
  - Perform image processing to enhance image quality and facilitate face detection.
  - Utilize the pre-trained CNN model for face recognition tasks.
  - Implement algorithms for real-time decision-making, such as unlocking a door upon face recognition.

Flowchart for the process



Model Training and Inference

- **Model Training:**
  - Utilized a Convolutional Neural Network (CNN) architecture.
  - Trained the model using a dataset consisting of:
  - 130 photos of my face.
  - 130 photos of other faces for contrast and generalization.
  - Implemented data augmentation techniques to enhance model robustness.

- **Model Inference:**
  - During operation, the trained CNN model performs inference on real-time video or image input.
  - The model predicts whether a detected face matches the identities learned during training.
  - Utilizes feature extraction and comparison techniques to achieve accurate identification.

Training the model code snippet:

```python
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D, Dropout
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceL
from tensorflow.keras.regularizers import l2

# Define paths to your data directories
train_data_dir = r'C:\Users\sseno\OneDrive\Documents\FYP\Dataset'
validation_data_dir = r'C:\Users\sseno\OneDrive\Documents\FYP\Datatest'

# Define image dimensions and other constants
img_width, img_height = 224, 224
batch_size = 16
epochs = 30

# Data Augmentation
train_datagen = ImageDataGenerator(
    rescale=1. / 255,
    shear_range=0.3,
    zoom_range=0.3,
    rotation_range=40,
    width_shift_range=0.3,
    height_shift_range=0.3,
    horizontal_flip=True,
    brightness_range=[0.8, 1.2],
    channel_shift_range=0.2,
    fill_mode='nearest'
)

test_datagen = ImageDataGenerator(rescale=1. / 255)
```

Simulation of Door Lock/Unlock

System Operation:

Unlocking:

- Initiates when a recognized face is detected with a prediction confidence > 0.5.
- Confirms identity based on stored facial features from the trained model.
- Triggers the unlocking mechanism securely and promptly upon positive recognition.

Locking:

- Activates when no recognized face is detected or the prediction confidence <= 0.5.
- Ensures the door remains securely locked to prevent unauthorized access.
- Maintains security by responding effectively to non-recognized or unidentifiable faces.

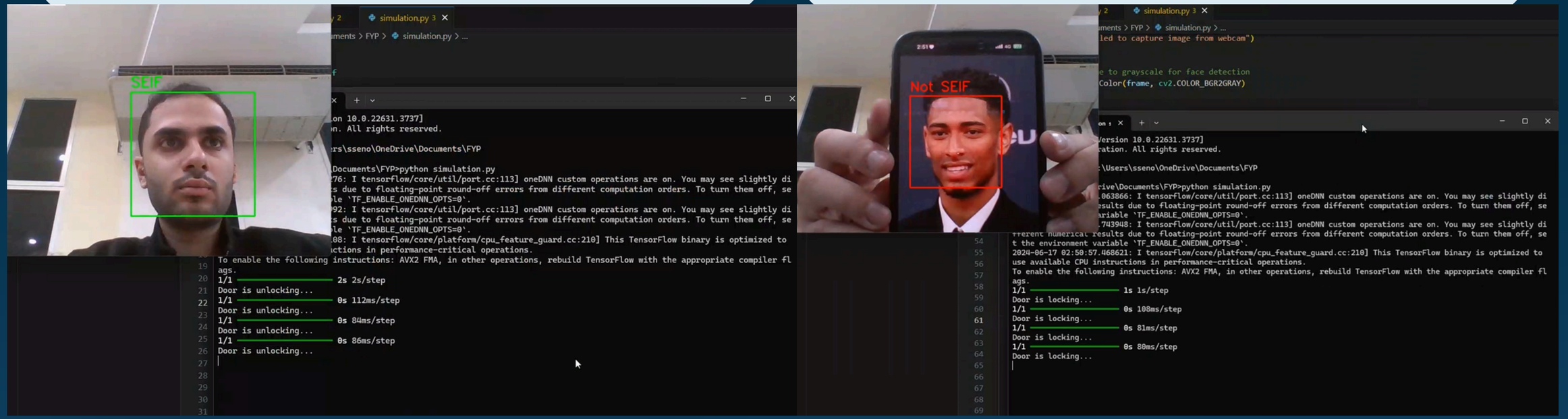
```python
# Load face cascade classifier
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')

# Initialize webcam
cap = cv2.VideoCapture(0)

# Flag to track if SEIF face is detected
seif_detected = False

def simulate_door_unlock():
    print("Door is unlocking...")
    # Simulate unlocking by printing a message or waiting
    import time
    time.sleep(2)  # Simulate unlocking delay

def simulate_door_lock():
    print("Door is locking...")
    # Simulate locking by printing a message or waiting
    import time
    time.sleep(1)  # Simulate locking delay

try:
    while True:
        # Capture frame-by-frame
        ret, frame = cap.read()

        if not ret:
            print("Failed to capture image from webcam")
            break

        # Convert frame to grayscale for face detection
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        # Detect faces
```

```python
        # Determine label based on prediction threshold
        if prediction_value > 0.5:  # Adjust threshold as needed
            label = "SEIF"
            color = (0, 255, 0)  # Green for SEIF
            seif_detected = True  # Set SEIF detected flag
        else:
            label = "Not SEIF"
            color = (0, 0, 255)  # Red for Not SEIF

        # Draw bounding box around the face
        cv2.rectangle(frame, (x, y), (x+w, y+h), color, 2)
        cv2.putText(frame, label, (x, y-10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, color, 2)

    # Check if SEIF was detected to decide door lock/unlock
    if seif_detected:
        simulate_door_unlock()  # Simulate door unlock
    else:
        simulate_door_lock()  # Simulate door lock

    # Display the resulting frame
    cv2.imshow('Face Recognition', frame)
```

RESULTS

THE SYSTEM SUCCESSFULLY RECOGNIZED MY FACE IN REAL-TIME AND SIMULATED THE DOOR UNLOCKING PROCESS

WHEN AN UNRECOGNIZED FACE WAS DETECTED, THE SYSTEM SIMULATED THE DOOR REMAINING LOCKED



Discussion

Facial recognition technology for door lock systems has been effectively demonstrated through our simulation, showcasing its potential feasibility and utility. The system successfully recognized authorized faces in real-time, illustrating its application in enhancing security and user convenience.

However, challenges such as varying lighting conditions pose significant considerations for real-world deployment. Future iterations will focus on optimizing the model to handle these conditions seamlessly, ensuring reliable performance across different environments. This discussion underscores the transformative impact of facial recognition in modern security systems, highlighting both its strengths and the necessary steps for refinement.

FYP 2 and Improvements

In FYP 2 , Implementation of the complete system with the current hardware setup on Raspberry Pi. This includes integrating the camera, relay module, and lock mechanism to enable real-world functionality. Extensive testing will be conducted to validate system performance and reliability under different scenarios.

Improvements

01 ENHANCE FACIAL RECOGNITION MODEL

Improve accuracy and robustness through additional training data and fine-tuning parameters.

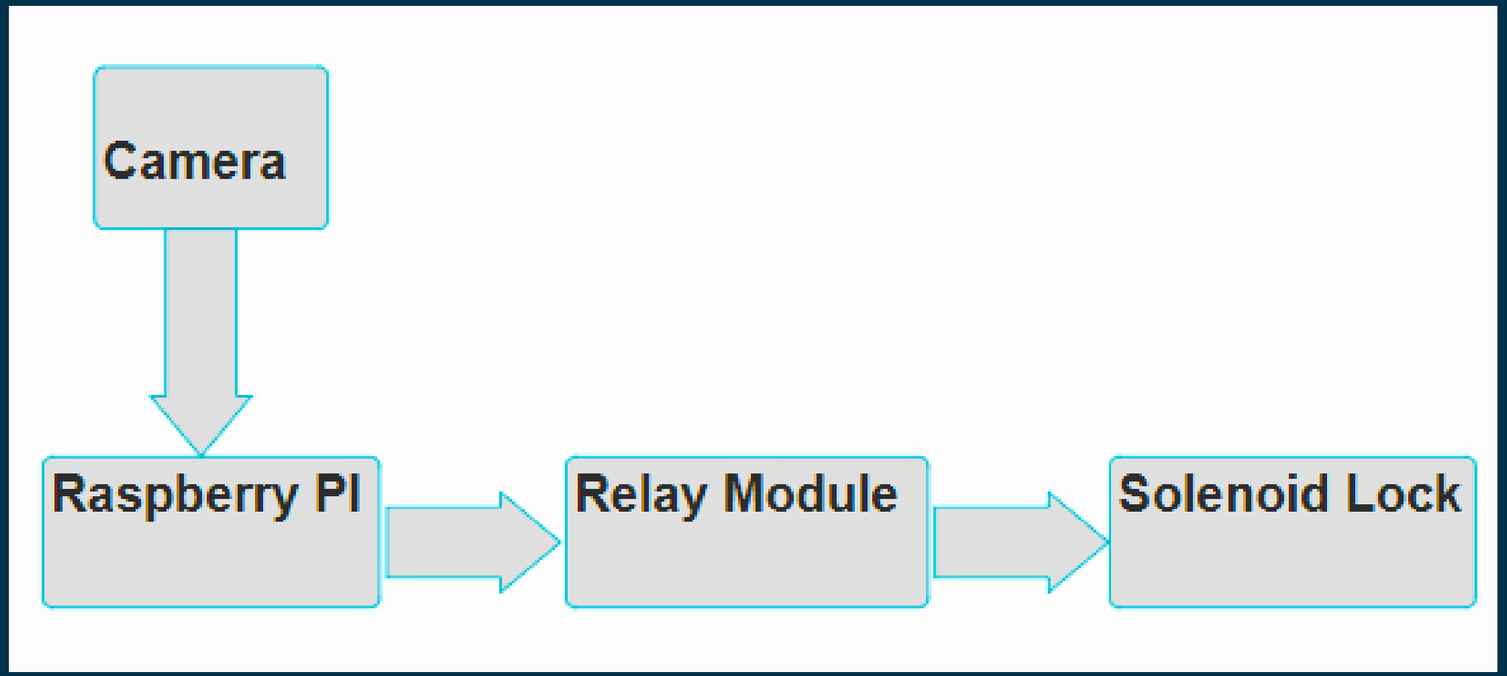
02 OPERATIONAL OPTIMIZATION

Fine-tune the system to maximize performance with the existing Raspberry Pi camera setup

03 SECURITY ENHANCEMENTS

Integrate multi-factor authentication to bolster security measures and ensure robust access control

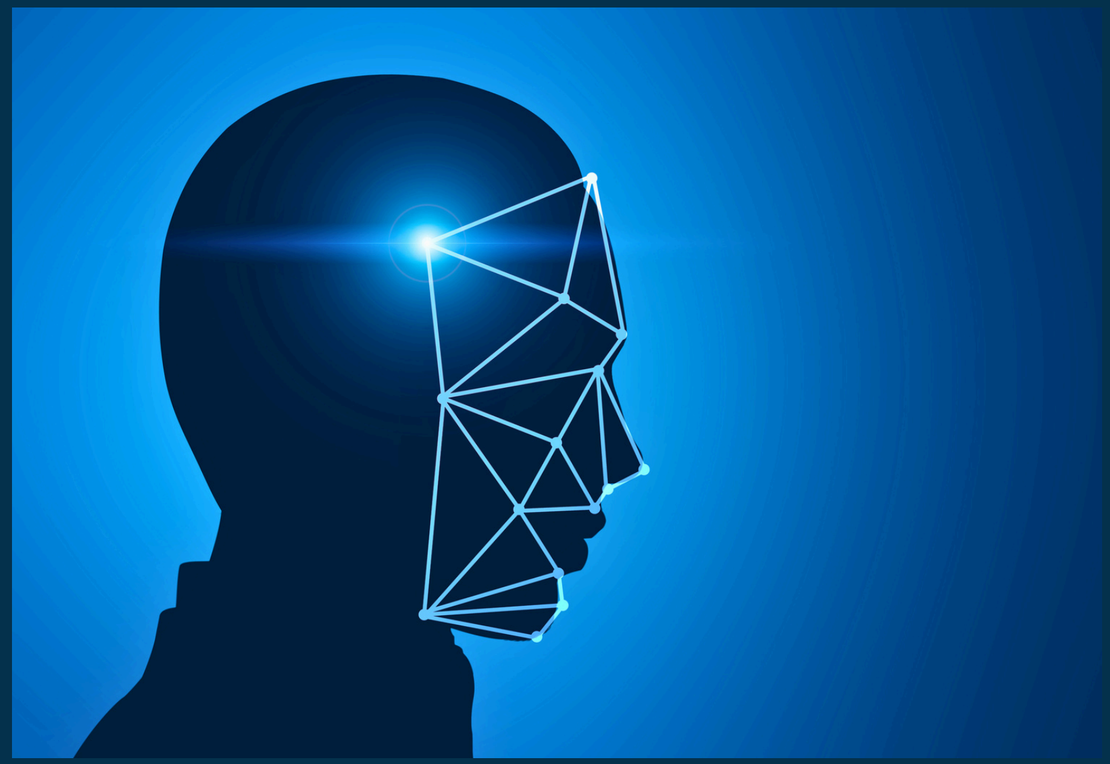
Block Diagram



CONCLUSION

This project successfully simulated a Raspberry Pi-based facial recognition door lock system, demonstrating its ability to enhance security through advanced AI integration. Using OpenCV and TensorFlow, the system effectively recognized faces in real-time, simulating door access based on recognition results.

Future steps involve integrating hardware components like the Raspberry Pi, camera module, relay mechanism, and door lock for comprehensive testing in real-world environments. Enhancements will focus on refining system robustness and potentially integrating multi-factor authentication for enhanced security. This project highlights the potential of AI-driven solutions for reliable access management in residential and commercial settings.



Demonstration Video

Please watch the demonstration video showcasing the simulation.

Video Link

Q & A

References

[1] M. Alshar'e, M. R. Al Nasar, R. Kumar, M. Sharma, Prof. D. Dharamvir, and V. Tripathi, "A Face Recognition Method In Machine Learning (ML) For Enhancing Security In Smart Home," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Muscat, Oman, Dubai, UAE, and Noida, India, 2022, pp. 1081-1084. https://doi.org/10.1109/ICACITE53722.2022.9823833.

[2] A. Jha, R. Bulbule, N. Nagrale, T. Belambe, "Raspberry Pi-Powered Door Lock with Facial Recognition," 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), 2024, pp. 1-6. DOI: 10.1109/SCEECS61402.2024.10481920.

[3] A. D. Singh, B. S. Jangra, R. Singh, "Face Recognition Door Lock System Using Raspberry PI," 2022 International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 10, no. V, May 2022. Available: https://doi.org/10.22214/ijraset.2022.42663.

[4] F. N. Nwukor, "Face Recognition Door Lock System Using Raspberry Pi," 2022 Global Scientific Journal (GSJ), vol. 10, no. 8, pp. 1390-1393, Aug. 2022. Available: www.globalscientificjournal.com.

[5] D. G. Padhan, M. Divya, S. N. Varma, S. Manasa, V. C., and B. Pakkiraiah, "Home Security System Based On Facial Recognition," 2023 IEEE 3rd International Conference on Sustainable Energy and Future Electric Transportation (SEFET), 2023, pp. 979-8-3503-1997-2/23/$31.00 ©2023 IEEE, DOI: 10.1109/SEFET57834.2023.10244798.

Thank You!