Blue text on a white background

Description automatically generated

**PROJECT REPORT**

FaceSphere

**FACULTY of ENGINEERING and NATURAL SCIENCES  
MULTIDISCIPLINARY FACULTY ELECTIVE**

**ENS005 Artificial Intelligence Applications in Image Processing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student Name** | **Student Surname** | **Student Number** | **Department** |
| Emir | Yılmaz | 200722094 | Computer Engineer |
| Arzu Selin | Yaşar | 200722079 | Computer Engineer |
| Rawan Z. M. | Shat | 210408214 | Biomedical Engineer |
| İbrahim İsmail | Onay | 180723016 | Electrical Engineer |

**Course Instructor:** INDRİT MYDERRİZİ

**Summary**

FaceSphere is a web application for real-time recognition of video feeds through face detection with Python, Flask. The program checks the webcam stream and compares the detected faces with a premade database of users e.g. Upon successful recognition, the person is asked to look right and left to prevent spoofing issues while still within the same terminal When these three tests successfully prove who the user is, the system allows him or her entry and keeps track of the time this person entered in the DB. If an unauthorized person tries to enter, his or her picture is taken and stored as a case of wrong entry. The program also lets you add or remove users through their numbers in addition to showing recently entered persons.

**ORIGINALITY**

1. **Importance of the Topic:**

FaceSphere caters for the rising demand for secure and efficient entrance systems in different sectors, e.g., corporate offices, schools and secured premises. Security and reliability are improved, thanks to its incorporation of anti-spoofing measures.

1. **Originality of the Research Topic:**

The practical application of these technologies in a cohesive system highlights a novel approach in the use of real-time face recognition and anti-spoofing checks. This project equally applies the two technologies in an innovative manner.

**PURPOSE and GOALS**

The primary aim of this project is to develop a robust and secure face recognition system for access control that includes anti-spoofing measures. The specific goals include:

* Developing a web-based interface for real-time face recognition.
* Implementing anti-spoofing techniques by prompting users to turn their heads.
* Recording entry times and managing user data efficiently.
* Ensuring the system can handle unauthorized access attempts by logging and photographing such events.

**METHODOLOGY**

1. **System Design:**

The system is built using Python and Flask for the backend, with OpenCV and DeepFace libraries for face detection and recognition. The frontend interface is developed using HTML, CSS, and JavaScript.

1. **Data Collection and Training:**

User images are collected and stored in a database for verify by pre-trained face recognition model. The model uses these images to recognize faces accurately.

1. **Anti-Spoofing Measures:**

To prevent spoofing attacks, the system requires users to turn their heads left and right, verifying the consistency of the face data captured from different angles.

1. **Database Management:**

The system uses a database to store user images, entry logs, and manage user data. SQLite is used for simplicity and ease of integration with Flask by SqlAlchemy.

1. **Verification and Validation:**

Extensive testing is conducted to ensure the system accurately recognizes users and effectively prevents unauthorized access.

**PROJECT MANAGEMENT**

**Work – Time Schedule**

| **Work No** | **Name and Goals of the Work** | **Responsible Student** | **Time Range** | **Success Criteria and Contribution to the Success of the Project** |
| --- | --- | --- | --- | --- |
| 1 | System Design and Development | EMİR YILMAZ | 1-4 Weeks | Completion of the system architecture and initial implementation |
| 2 | Model Tests and FrontEnd aspects | İBRAHİM İSMAİL ONAY | 5-7 Weeks | Successful training of the face recognition model with a dataset of user images |
| 3 | Implementation of Anti-Spoofing Measures | EMIR YILMAZ  ARZU SELİN YAŞAR | 8-10 Weeks | Accurate detection and verification of head movements |
| 4 | Database Integration and Management | EMİR YILMAZ  RAWAN SHAT  İBRAHİM ONAY | 11-12 Weeks | Efficient data storage and retrieval |
| 5 | Testing and Validation | EMİR YILMAZ  ARZU SELİN YAŞAR | 13-14 Weeks | System performs well in recognizing faces and preventing unauthorized access attempts |

**Risk Management**

| **Work No** | **Major Risks** | **Risk Management (Plan B)** |
| --- | --- | --- |
| 1 | Inaccurate face recognition due to poor image quality | Implement image preprocessing techniques to enhance image quality |
| 2 | Failure in anti-spoofing measures | Enhance the algorithm by incorporating additional movement checks or 3D facial recognition techniques |
| 3 | Database management issues | Utilize more robust database solutions such as PostgreSQL or MongoDB |

**RESEARCH OPPORTUNITIES**

**RESEARCH OPPORTUNITIES TABLE**

| **Infrastructure/Equipment Type and Model in the Organization** | **Purpose of Use in the Project** |
| --- | --- |
| Computer Laboratory | Use of necessary computer equipment during the application phase |
| High-Resolution Webcam | Capturing high-quality images for accurate face recognition |

**IMPLICATIONS**

**EXPECTED IMPLICATIONS TABLE of the RESEARCH**

| **Implication Types** | **Expected Outputs Results Findings and Impacts** |
| --- | --- |
| Scientific/Academic | The project findings will be compiled into an article and published, contributing to the literature on face recognition technology |
| Economic/Commercial/Social | Potential for commercialization as a secure access control system for various institutions and organizations |
| Training Researchers | Undergraduate students will gain hands-on experience in AI and image processing, fostering research skills for future academic pursuits |

**FINDINGS**

The test was facilitated by the system’s ability to spot enrolled users on spot thus preventing unauthorized entry into the system. This success shows that the measures against spoofing were operational.

**CONCLUSION**

With anti-spoofing measures combined, live face recognition is made practical by The FaceSphere Project. Its performance in different test situations has proved that it could serve real life purposes. Coming up with methods to boost its capacity and incorporating more safety measures would be a dream of the future.

**APPENDICES**

**Sub-Teams**

**Computer / Electrical Engineering Team**

**Overview:**

Responsible for developing the backend system using Python and Flask and integrating the face recognition and anti-spoofing algorithms.

**Literature Survey:**

Review of current face recognition technologies and anti-spoofing measures.

**Constraints:**

Limited by the accuracy of face recognition in varying lighting conditions and the performance of the anti-spoofing algorithm.

**Requirements:**

Develop a robust and secure system that accurately recognizes faces and prevents unauthorized access.

**Methodology:**

Implemented using OpenCV, DeepFace library, and Flask, with SQLite for database management.

**System Integration:**

Integrated the face recognition and anti-spoofing components into a cohesive web-based application, ensuring seamless interaction between the frontend and backend systems.

**Computer / Biomedical Engineering Team**

**Overview:**

Responsible for the writing of the report and research on the project at hand.

**APPX-1: REFERENCES**

1. ﻿﻿﻿ **OpenCV for Face Detection and Recognition**

* **Title:** "Real-time Face Detection and Tracking Using OpenCV"
* **Authors:** C. R. Gour, S. Kumar
* **Publication:** International Journal of Computer Applications
* **Year:** 2010

1. **dlib for Facial Landmark Detection**

* **Title:** "One Millisecond Face Alignment with an Ensemble of Regression Trees"
* **Authors:** Vahid Kazemi, Josephine Sullivan
* **Publication:** IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
* **Year:** 2014

1. **Deep Learning for Face Recognition**

* **Title:** "DeepFace: Closing the Gap to Human-Level Performance in Face Verification"
* **Authors:** Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, Lior Wolf
* **Publication:** IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
* **Year:** 2014

1. **Flask for Web Development**

* **Title:** "Building Web Applications with Flask"
* **Authors:** Miguel Grinberg
* **Publication:** O'Reilly Media
* **Year:** 2018

1. **SQLAlchemy for Database Management**

* **Title:** "Mastering Flask Web Development"
* **Authors:** Daniel Gaspar
* **Publication:** Packt Publishing
* **Year:** 2018

1. **Face Verification with Deep Learning Models**

* **Title:** "FaceNet: A Unified Embedding for Face Recognition and Clustering"
* **Authors:** Florian Schroff, Dmitry Kalenichenko, James Philbin
* **Publication:** IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
* **Year:** 2015