TUNIS BUSINESS SCHOOL

CYBER SECURITY PROJECT

WEB BASED FACIAL AUTHENTICATION SYSTEM

SPRING 2024

TUNIS BUSINESS SCHOOL

Our Team

Islem Hamzaoui

BA/IT

Aya Essid

BA/IT

Yosr Boussarsar

BA/IT

Wadii Selmane

BA/IT

Eya Trabelsi

FIN/IT

Introduction

Facial authentication is taking the web by storm, offering a secure and convenient way to log in to applications and access online services.

This presentation sheds light on the core concepts of web-based facial authentication systems, exploring their unique characteristics and how they function within the web environment.

MAIN COMPONENTS

01

Web Interface

This user-friendly interface on a web browser allows users to interact with the facial authentication system. It typically includes instructions for the user to register and log in.

02

Client-Side Processing

In some systems, basic facial feature extraction or preprocessing might occur on the user's device before sending data to the server.

03

Secure Communication Channel

Encrypted communication protocols like HTTPS ensure the safe transmission of facial data between the user's device and the server.

04

Server-Side Processing

Facial Recognition engine powered by ML algorithms .
Feature Database to store authorized users' facial feature representations.
Matching and Verification algorithms.

05

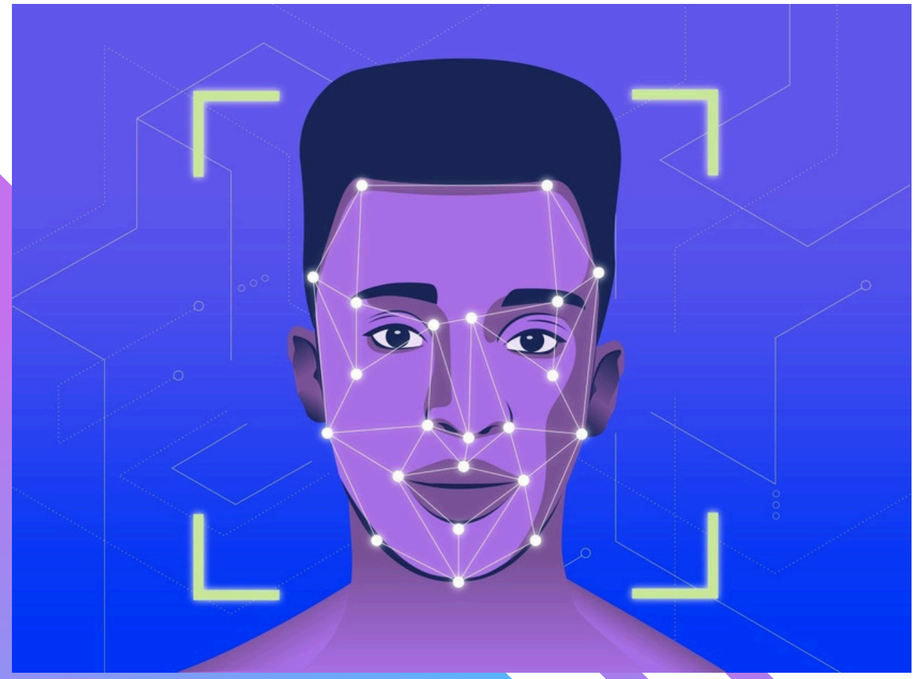
Authentication Decision

Based on the matching score, the system determines whether to grant access or prompt for alternative credentials.

FUNCTIONAL FLOW

- Webpage Access
- Web Interface Interaction
- Facial Image Capture:
- Data Transmission (Optional)
- Secure Communication and Data Transfer
- Server-Side Processing
- Authentication Response
- Access Granted/Denied

FACIAL RECOGNITION - KEY COMPONENT



The facial recognition process utilizes biometric technology to analyze and identify individuals based on facial features.
It starts with capturing facial images and extracting unique characteristics, which are then converted into mathematical templates for comparison with stored data.
Matching algorithms assess similarity to authenticate users, granting or denying access accordingly.
Continuous refinement mechanisms enhance accuracy over time.

CHARACTERISTICS

Detection

This stage involves finding a face within an image. Initially, the system extracts the facial area from the input image.

Matching

The generated faceprint is compared with prints stored in the system's database.
The algorithms assess similarity scores obtained from comparing features of the new face image with those of known faces.

Analysis (Face Mapping)

The system analyzes the facial features by marking specific landmarks on the face and their measurements are used to generate a unique code known as a 'faceprint' for each individual.

Recognition

Based on the similarity scores obtained during the matching stage, the system makes a decision regarding the identity of the individual.

FACIAL RECOGNITION - MAIN COMPONENTS

IMAGE ACQUISITION

FACE DETECTION

MATCHING AND RECOGNITION

FEATURE EXTRACTION

FEATURE COMPARISON

HAAR CASCADE CLASSIFIER

This machine-learning algorithm thrives in real-time environments. It dissects the image into Haar features, which are simple edge and line patterns.
By efficiently identifying these patterns within the image, the Haar cascade classifier can pinpoint the presence of a face.

PRINCIPAL COMPONENT ANALYSIS - PCA

PCA, also known as Eigenfaces, tackles the high dimensionality challenge. It analyzes the variations within a set of facial images and identifies the most significant patterns that distinguish one face from another PCA.
Then projects these high-dimensional images into a lower-dimensional subspace, capturing the essence of a face while reducing processing requirements. While PCA offers a solid foundation, future advancements might involve sparse representation techniques.

Matching Algorithms

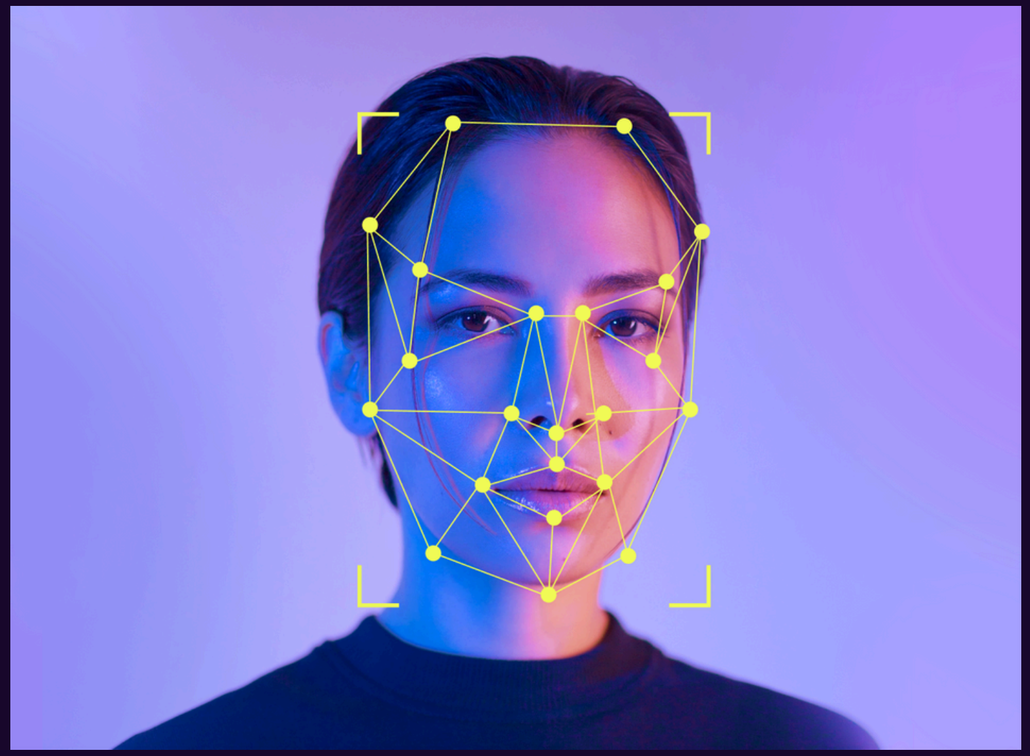
Matching algorithms compare the extracted features (Eigenfaces) of the unknown individual with the known faces stored in the database

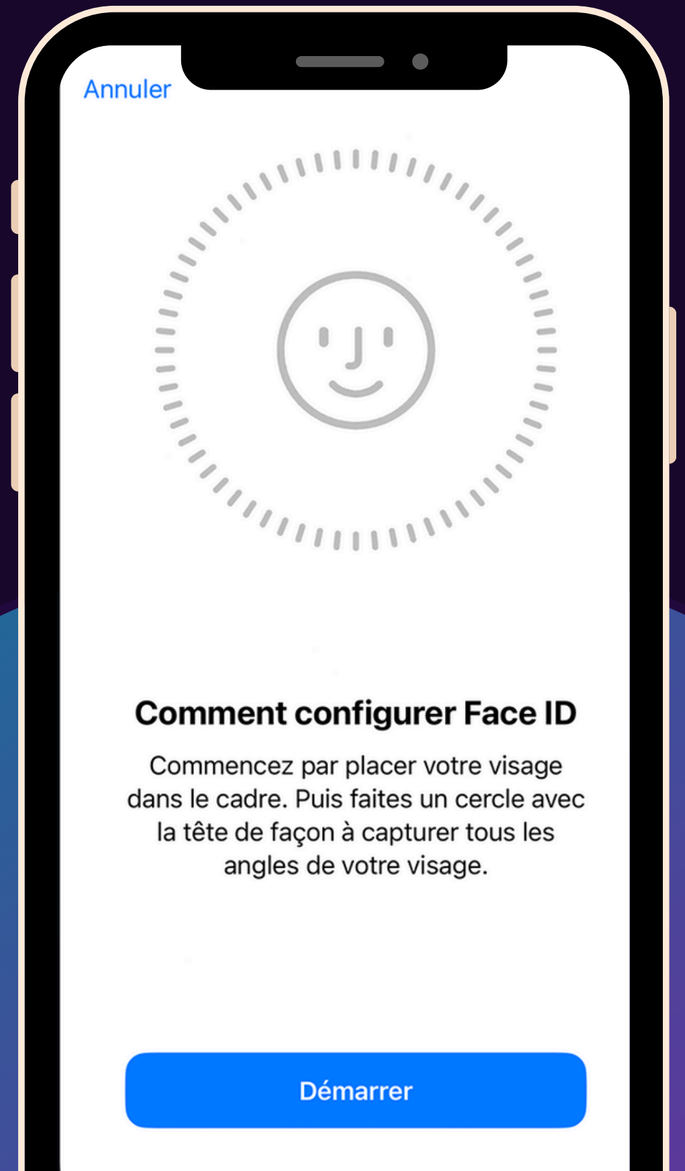
- **Algorithm 1: Eigenface (PCA) Matching**
- **Algorithm 2: Local Binary Patterns (LBP) Matching**

Deep Learning For Facial Recognition

Convolutional Neural Networks

CNNs eliminate the need for predefined features. Unlike traditional methods, they function as intelligent feature detectors, analyzing spatial relationships between pixels in face images. Specialized CNN layers capture the arrangement of features like eyes and nose, progressing from recognizing basic shapes to intricate facial details through a learning process.





Examples of facial recognition technologies

Cigna, a US-based healthcare insurer, allows customers in China to file health insurance claims which are signed using a photo, rather than a written signature, in a bid to cut down on instances of fraud.

British Airways enables facial recognition for passengers boarding flights from the US. Travellers' faces can be scanned by a camera to have their identity verified to board their plane without showing their passport or boarding pass.

Amazon previously promoted its cloud-based face recognition service named **Rekognition** to law enforcement agencies

Apple FaceID uses facial recognition to help users quickly unlock their phones, log in to apps, and make purchases.

EXISTING SOLUTIONS

Off-the-Shelf Solutions and APIs

Companies can opt for ready-made solutions like Microsoft Face API, Amazon Rekognition, or utilize existing facial recognition libraries such as DeepFace, FaceNet, InsightFace, among others

Custom Software Development

For specialized needs or industries, custom software development may be necessary to tailor facial recognition systems to specific requirement.

Facial Recognition Based on Machine Learning

Involves ML algorithms to recognize faces by extracting features from images and learning patterns from labeled data. These algorithms typically use techniques like PCA, LDA, or SVM

EXISTING SOLUTIONS

Facial Recognition Based on Deep Learning

Utilizes deep neural networks, such as CNNs, to automatically learn hierarchical representations of facial features from raw image data. These networks are trained on large datasets of labeled facial images..

Facial Recognition Without AI

Refers to traditional methods of face recognition that rely on handcrafted features and rule-based algorithms. These methods often use techniques like geometric feature extraction, template matching, or correlation-based matching.

FACIAL AUTHENTICATION USE CASES

- Healthcare
- Car Industry
- Unlocking phones
- Airports and border control
- Banking and Payment
- Tracking students
- Employees Attendance

ADVANTAGES

- Enhanced Security
- Reduced number of touchpoints
- Convenience
- Efficiency
- Personalization
- Accessibility

LIMITATIONS

- Accuracy Concerns and Bias
- Biometric Variability
- Imperfections
- Ethical Considerations
- Privacy Issues
- Ownership Challenges
- Legal and Regulatory Hurdles
- Resource Intensiveness
- Vulnerability to Attacks

A web-based facial authentication system offers a secure and convenient method for users to authenticate their identities using facial recognition technology. The upcoming section will outline the **design**, **components**, **working flows**, **roles**, **functions**, and **data exchanged** within such a system.

SYSTEM’S COMPONENTS

- CAPTURE DEVICE
- FACIAL RECOGNITION ENGINE
- DATABASE
- SECURE COMMUNICATION CHANNEL
- AUTHENTICATION SERVER
- USER INTERFACE

WORKING FLOWS

User Registration Flow

In this flow, a new user registers on the system by providing necessary details such as name, email, and creating a password.

Facial enrollment

The user is then prompted to take a photo of their face using their webcam (User's face image is captured from multiple angles for increased accuracy) or upload an existing photo.

Authentication Flow

In this flow, a user who wants to authenticate themselves logs in to the system using their email and a photo of their face or upload an existing one.

Administration Flow

In this flow, system administrators can manage user accounts, set up authentication policies, and monitor system activity. They can also configure the system to integrate with other applications or services.

USERS' ROLES

- **Enrolled User:** Individuals who have their facial data stored in the system for authentication purposes.
- **System Administrator:** Manages user enrollment, system configuration, and access control and ensures the security and performance of the system.
- **Developer:** A person who develops and maintains the system, integrates it with other applications or services and ensures its security and performance

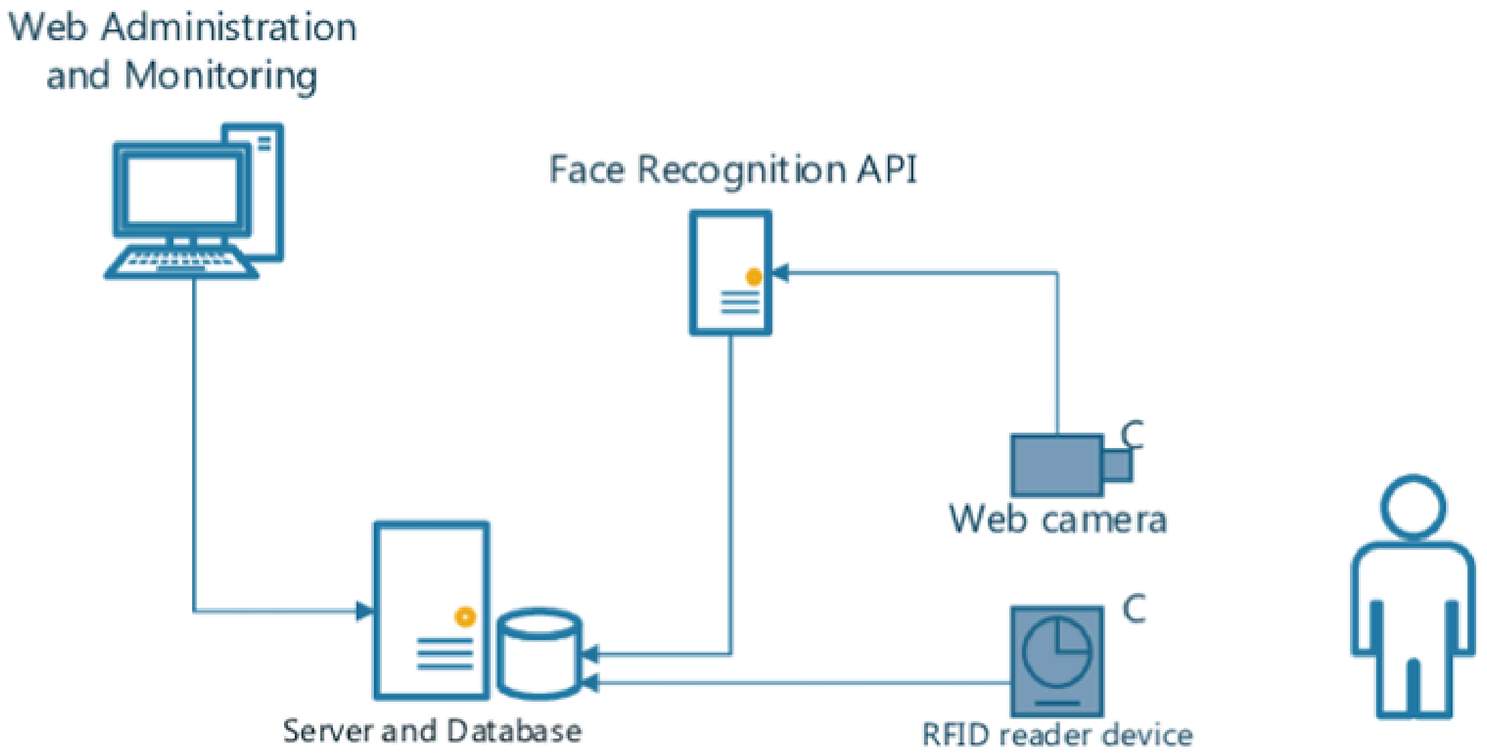
DATA EXCHANGE

- **Enrollment**: User's facial image and a unique identifier (ID number, username) are captured and stored in the database.
- **Authentication:** User's face is captured. The facial recognition engine extracts features and sends them to the server. The server retrieves the user's template from the database and compares it.
- **Success/Failure:** The server sends a success or failure message to the user interface based on the comparison result.

MAIN FUNCTIONS

- USER ENROLLMENT
- USER AUTHENTICATION
- FACIAL RECOGNITION
- PASSWORD RESET
- TWO-FACTOR AUTHENTICATION

AUTHENTICATION DIAGRAM



Building a web-based facial authentication system involves several stages, including **planning**, **development**, and **deployment**. This system typically requires various technologies and tools to perform facial recognition and authentication tasks. Here's an outline of the tools and development phases involved.

PLANNING PHASE

Requirement Analysis

Determine the objectives, target users, security requirements, and functional specifications

Architecture Design

Define the system architecture, including client-side and server-side components.

Technology Stack Selection

Choose appropriate technologies for the front-end, back-end, database, and machine learning

DEVELOPMENT PHASE

Front-End

- **CSS and HTML:** Used for designing the user interface .
- **JavaScript:** Used to handle user input and communicate with the server.

Back-End

- Server-Side Frameworks: **Python**
- Web Server: **Apache**
- Authentication: **JSON Web Tokens** for secure authentication and session management.
- API Integration: **PHP**

Database System

- **SQL** (MySQL)

ALGORITHMS

- Haar Cascade Classifier
- Facial Landmark Detection
- Support Vector Machine (SVM)
- PCA (Principal Component Analysis)

Python Libraries

OpenCV

PIL

NumPy

MySQL Connector

dlib

os

scikit-learn

TESTING PHASE

Unit Testing

Pytest (Python), for testing individual components.

Integration Testing

Test interactions between components to ensure they work as expected

User Testing

Real-world tests with users to ensure usability and security

DEPLOYMENT PHASE

Continuous Integration/Continuous Deployment (CI/CD)

GitHub Actions for automated testing and deployment.

Security Tools

Hash functions
SQL injection prevention methods
Tokenization

MAINTENANCE PHASE

Updates and Upgrades

Regular updates to ensure compatibility and security.

Bug Fixes

Address reported issues promptly.

User Feedback

Gather feedback to improve system usability and functionality.

Documentation

Maintain comprehensive documentation for developers and users.

CONCLUSION

Web-based facial authentication systems offer security, convenience, and efficiency, but face challenges like accuracy, privacy, and legal compliance.
Organizations must address these to ensure responsible use, including informed consent, explainable AI, and backup methods.
The system combines traditional techniques with facial recognition, enhancing user comfort and security.

GitHub link

https://github.com/YosrBoussarsar/web-based-facial-recognition-system-

THANK YOU