Computer Vision

Face Detection & Recognition

Task 5 - Team 11

Introduction

In the realm of computer vision, the task of face detection and recognition holds significant importance across various domains. This project aims to develop a robust face detection and recognition system utilizing Principal Component Analysis (PCA)/Eigen analysis. The system will be trained on the ORL (Olivetti Research Laboratory) dataset, a benchmark collection of facial images. By leveraging PCA for feature extraction, this system seeks to achieve efficient representation and classification of facial features. This report provides a comprehensive documentation of the project, from dataset loading and preprocessing to model training and performance evaluation.

Loading Dataset

The dataset utilized for this project is the ORL (Olivetti Research Laboratory) dataset, a well-established benchmark collection of facial images. The loading mechanism is implemented in the DatasetClass.py file.

Dataset Description

The ORL dataset consists of grayscale facial images captured under varying lighting conditions and facial expressions. Each image is associated with a specific individual, making it suitable for face detection and recognition tasks.

Dataset Loading Mechanism

The dataset loading process involves several key steps to organize the images into training and testing subsets.

Initialization

Upon initialization of the DatasetClass, essential variables and paths required for dataset loading are initialized. These include paths for training and testing images, as well as lists to store image paths, labels, and the number of images for training and testing.

Loading and Splitting

The dataset is iterated through, and images are split into training and testing subsets based on a specified parameter (training_num). For each person in the dataset, if the number of available images is sufficient for training (equal to or greater than training_num), images are added to the training subset; otherwise, they are added to the testing subset. This process ensures a balanced distribution of images for training and testing.

Labeling

Labels are assigned to each image based on the person it belongs to. These labels are crucial for training and testing the face recognition model. Additionally, the images_target list stores the target names associated with each image, facilitating result interpretation and visualization.

Conclusion

The dataset loading mechanism efficiently organizes the ORL dataset into training and testing subsets, laying the foundation for subsequent face detection and recognition tasks using PCA/Eigen analysis.

Face Detection & Recognition

The core of this project lies in the implementation of face detection and recognition using the ORL dataset and PCA/Eigen analysis. This section describes the classes and methods involved in the face detection and recognition process.

PCAClass

The PCAClass encapsulates the functionality for performing Principal Component Analysis (PCA) on the face images. It computes the principal components, reduces the

dimensionality of the data, and provides methods for recognizing faces based on their coordinates in the reduced feature space.
Key methods include:

- __init__(): Initializes the PCA object with image matrix, labels, target names, image dimensions, and quality percentage.
- reduce_dim(): Performs PCA to reduce the dimensionality of the image matrix.
- new_coord(): Projects a new image onto the PCA subspace to obtain its coordinates.
- recognize_face(): Identifies the person associated with a given set of coordinates in the PCA subspace.
- show_eigen_faces(): Displays the eigenfaces, which are the principal components representing the variability in the dataset.

Predictor

The Predictor class utilizes the PCA analysis from PCAClass for real-time face detection and recognition. It integrates with OpenCV for face detection using Haar cascades and overlays recognized face labels onto the detected faces.
Key methods include:

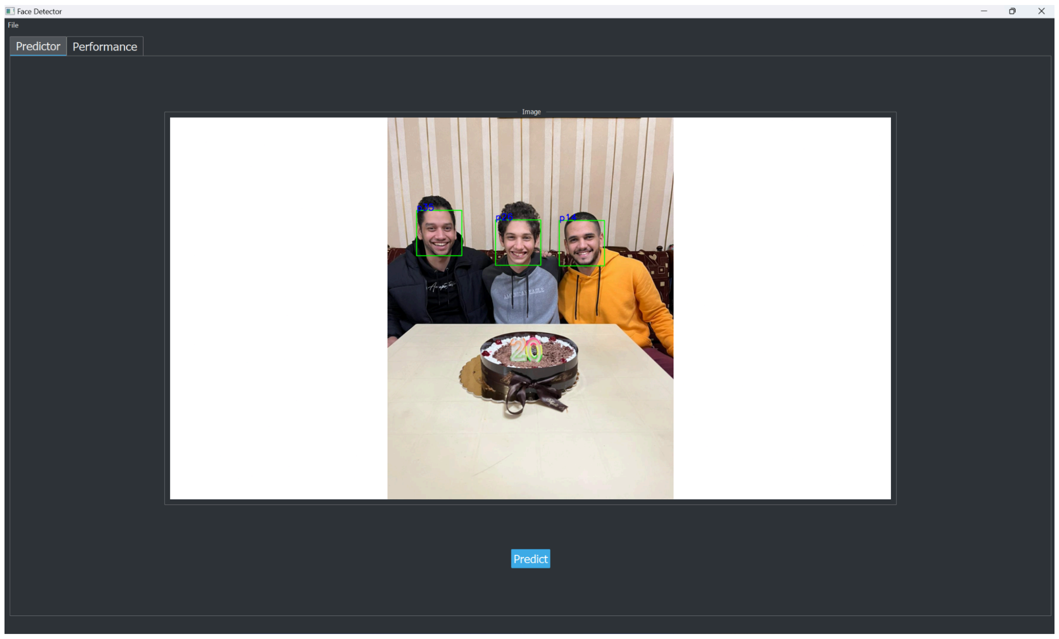
- __init__(): Initializes the Predictor object with dataset information and PCA analysis.
- predict(): Takes an image path as input, detects faces in the image using Haar cascades, extracts and scales the detected faces, computes their PCA coordinates, and predicts the identity of each face based on the PCA analysis.

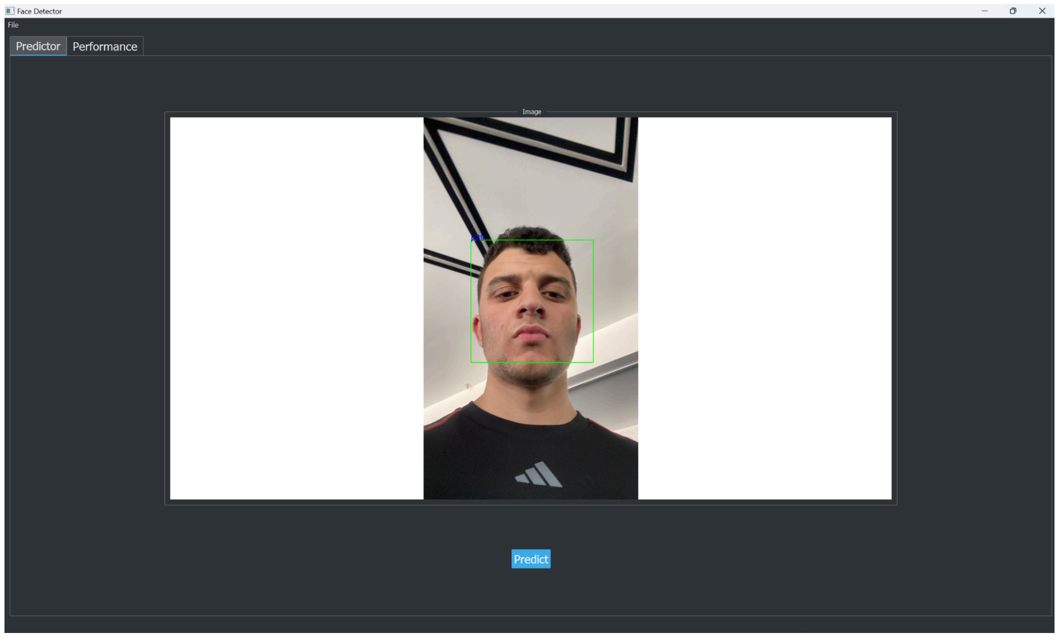
Face Detection and Recognition Workflow

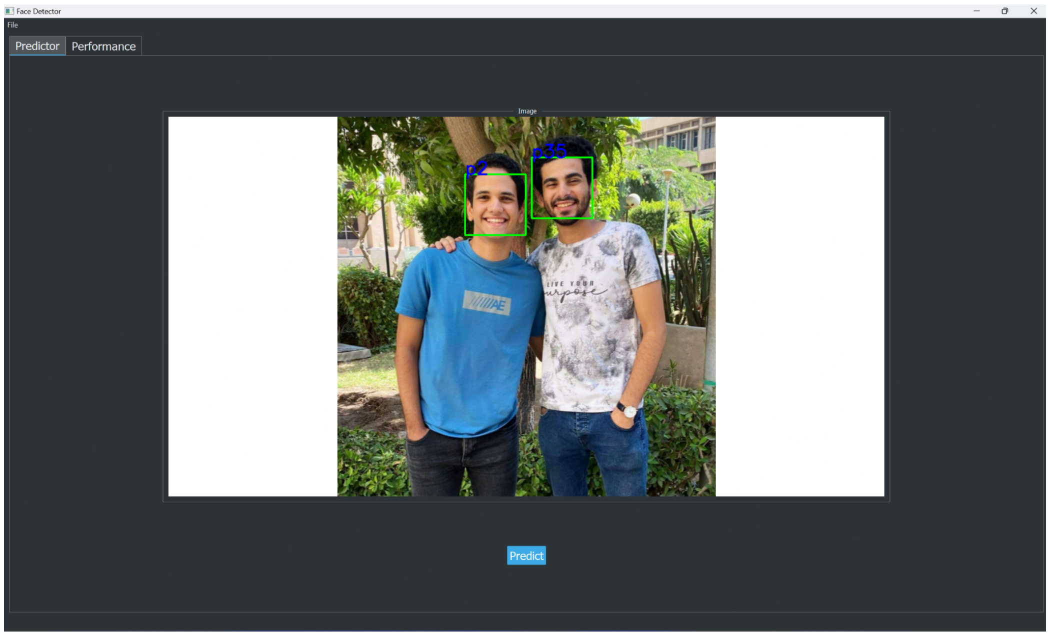
1. Face Detection: The Predictor class utilizes OpenCV's Haar cascades to detect faces in input images.
2. Feature Extraction: Detected faces are extracted, scaled, and converted to grayscale.
3. Dimensionality Reduction: PCA analysis from the PCAClass is applied to the extracted face images to reduce their dimensionality.
4. Face Recognition: Reduced-dimensional coordinates of the faces are used to predict the identity of each face.

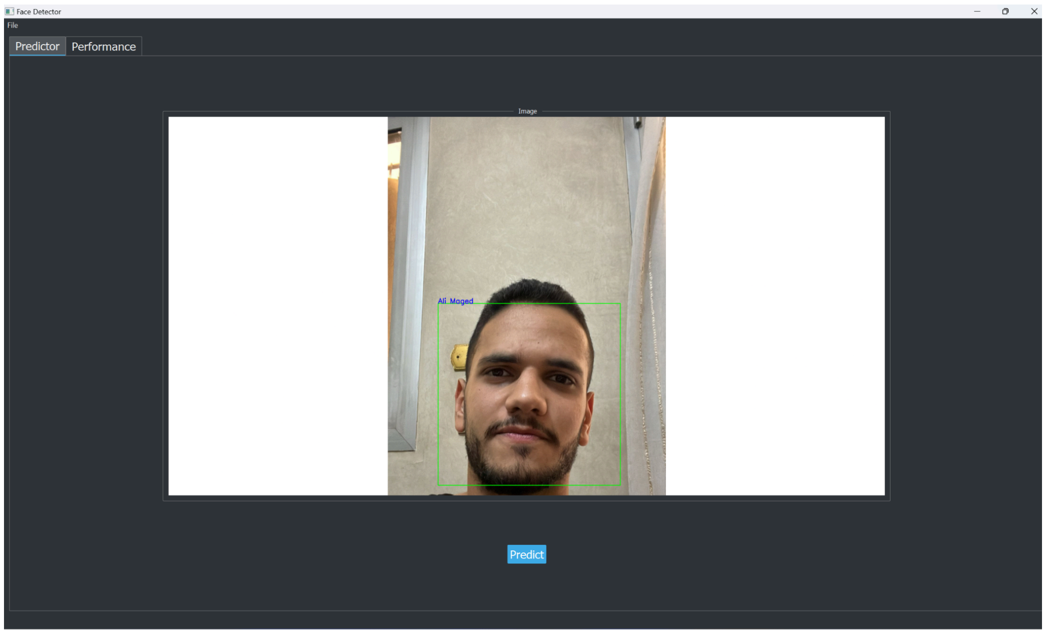
5. Display: Detected faces are annotated with predicted labels and displayed on the output image.

Here are some screenshots of the face detection and recognition:









<u>Performance Evaluation and ROC Curve</u>

Performance evaluation is a crucial aspect of any machine learning system. In this project, we assess the effectiveness of the face detection and recognition system using real-world performance metrics and visualize its performance using ROC curves.

Performance Evaluation

The performance evaluation is conducted on the testing subset of the ORL dataset. For each test image, the system predicts the identity of the face and compares it against the ground truth. The following metrics are used for evaluation:

- Correct Predictions: The number of correctly identified faces.
- Incorrect Predictions: The number of incorrectly identified faces.

The performance metrics provide insights into the system's accuracy and robustness in recognizing faces across different individuals.

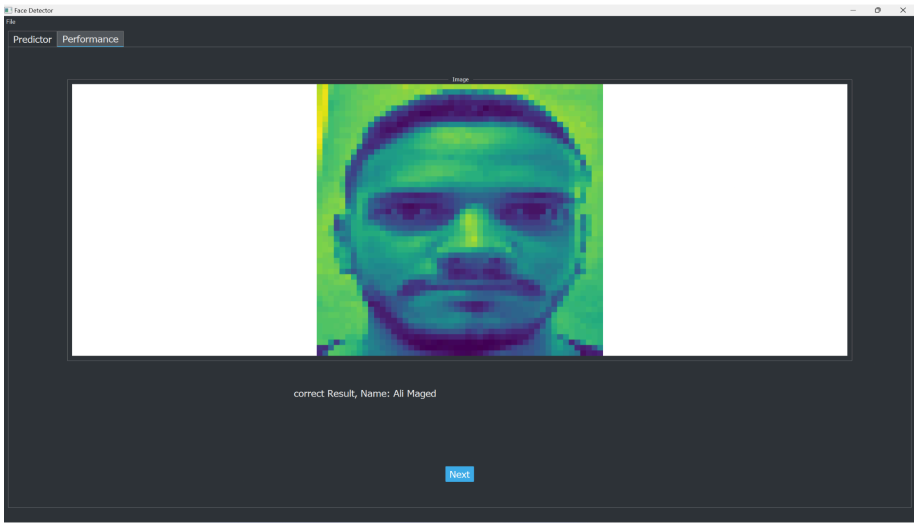
ROC Curve Generation

To further assess the system's performance, Receiver Operating Characteristic (ROC) curves are generated. The ROC curve visualizes the trade-off between the true positive rate and the false positive rate across different threshold values. Each class (individual) in the dataset is represented by a separate ROC curve, providing a comprehensive view of the system's discriminatory power for each class.

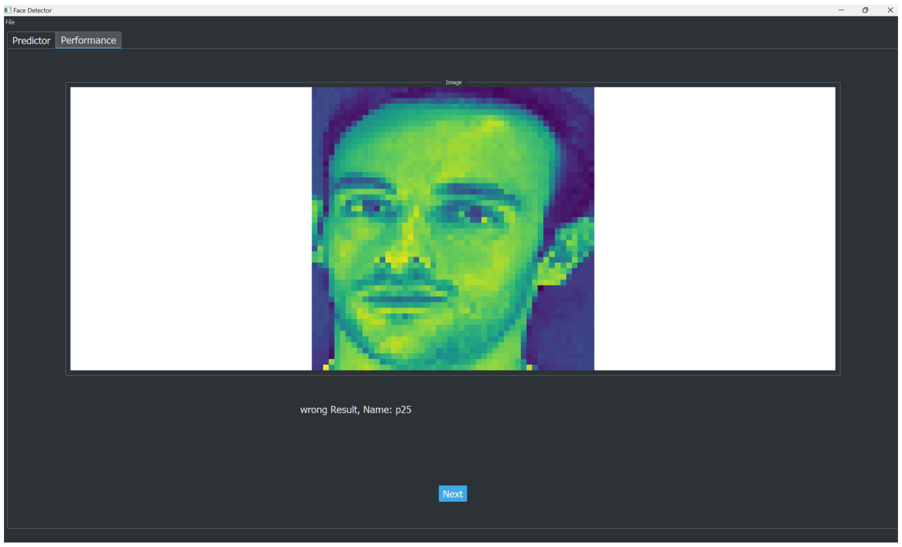
Methodology

1. Binarization: Ground truth labels and predicted scores are binarized for each class.
2. ROC Curve: ROC curves are computed for each class, showing the relationship between true positive rate and false positive rate.
3. Area Under Curve (AUC): The area under each ROC curve is calculated to quantify the system's discriminative performance.

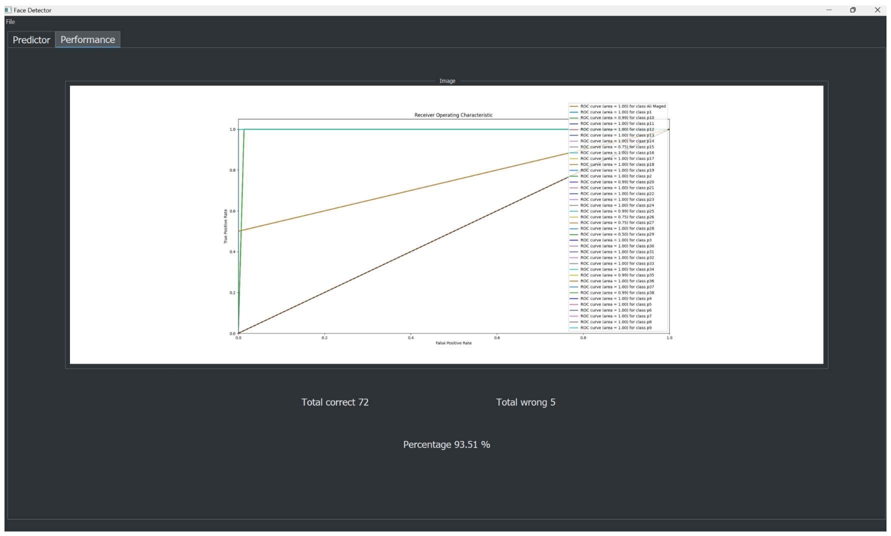
And here are some screenshots for the scores and the ROC Curve:



correct Result, Name: Ali Maged



wrong Result, Name: p25


Face Detector
File
Predictor
Performance
Image
Receiver Operating Characteristic
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 1.00) for class Ali Maged
ROC curve (area = 1.00) for class p1
ROC curve (area = 0.99) for class p10
ROC curve (area = 1.00) for class p11
ROC curve (area = 1.00) for class p12
ROC curve (area = 1.00) for class p13
ROC curve (area = 1.00) for class p14
ROC curve (area = 0.75) for class p15
ROC curve (area = 1.00) for class p16
ROC curve (area = 1.00) for class p17
ROC curve (area = 1.00) for class p18
ROC curve (area = 1.00) for class p19
ROC curve (area = 1.00) for class p2
ROC curve (area = 0.99) for class p20
ROC curve (area = 1.00) for class p21
ROC curve (area = 1.00) for class p22
ROC curve (area = 1.00) for class p23
ROC curve (area = 1.00) for class p24
ROC curve (area = 0.99) for class p25
ROC curve (area = 0.75) for class p26
ROC curve (area = 0.75) for class p27
ROC curve (area = 1.00) for class p28
ROC curve (area = 0.50) for class p29
ROC curve (area = 1.00) for class p3
ROC curve (area = 1.00) for class p30
ROC curve (area = 1.00) for class p31
ROC curve (area = 1.00) for class p32
ROC curve (area = 1.00) for class p33
ROC curve (area = 1.00) for class p34
ROC curve (area = 0.99) for class p35
ROC curve (area = 1.00) for class p36
ROC curve (area = 1.00) for class p37
ROC curve (area = 0.99) for class p38
ROC curve (area = 1.00) for class p4
ROC curve (area = 1.00) for class p5
ROC curve (area = 1.00) for class p6
ROC curve (area = 1.00) for class p7
ROC curve (area = 1.00) for class p8
ROC curve (area = 1.00) for class p9

Total correct 72

Total wrong 5

Percentage 93.51 %

Team Members:

1. Mohamed Sayed Mosilhe
2. Mina Adel
3. Mariam Magdy
4. Ali Maged
5. Abdallah Ahmed