A PRELIMINARY REPORT ON

Human Face Recognition Mini Project

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE ACADEMIC

OF

FOURTH YEAR OF COMPUTER

ENGINEERINGSUBMITTED BY

Anil Rathod BBCO21166

DEPARTMENT OF COMPUTER ENGINEERING

DR. D.Y. PATIL INSTITUTE OF ENGINEERING, MANAGEMENT & RESEARCH

AKURDI, PUNE 411044

SAVITRIBAI PHULE PUNE UNIVERSITY

AY 2024 -2025

CERTIFICATE

This is to certify that the Mini Project report of

Human Face Recognition Mini Project

Submitted by

Anil Rathod BBCO21166

is a bonafied student at this institute and the work has been carried out by them under the supervision of Mrs. Deepali Jawale, and it is approved for the partial fulfilment of the requirement of Savitribai Phule Pune University, for the award of the Fourth-year degree of ComputerEngineering.

Mrs. Deepali Jawale	**Mrs. P. P. Shevatekar**
Guide	Head
Department of Computer Engineering	Department of Computer Engineering

Place: Pune
Date:

ABSTRACT

The Human Face Recognition project leverages machine learning techniques to identify and classify human faces using the Labeled Faces in the Wild (LFW) dataset. The project pipeline includes data preprocessing, dimensionality reduction using Principal Component Analysis (PCA), and training a Support Vector Machine (SVM) classifier to perform face recognition tasks. By applying standard scaling and PCA, the high-dimensional image data is transformed into a more manageable and meaningful representation. The trained SVM model demonstrates high accuracy in recognizing individuals based on facial features, with performance evaluation using classification metrics and a confusion matrix. This project showcases the practical application of computer vision and machine learning in facial recognition, with potential use cases in security systems, identity verification, and user authentication.

ACKNOWLEDGEMENT

First and foremost, I would like to thank my guide for this Mini Project, Mrs. Deepali Jawale forth valuable guidance and advice. She inspired us greatly to work in this Mini Project. Her willingness to motivate us contributed tremendously to our seminar work. I also would like to thank her for showing me some examples that related to the topic of my Mini Project.

Apart from our efforts, the success of any seminar depends largely on the encouragement and guidelines of many others. So, we take this opportunity to express my gratitude to M/s. P. P. Shevatekar, Head of the Department of Computer Engineering, Dr. D Y Patil Institute of Engineering, Management and Research, Akurdi has been instrumental in the successful completion of this seminar work.

The guidance and support received from all the members who contributed and who are contributing to this seminar work were vital for the success of the seminar. I am grateful for theirconstant support and help.

Anil Rathod

(B.E. COMPUTER ENGG.)

TABLE OF CONTENT

CHAPTER 1: INTRODUCTION

1.1 INTRODUCTION

Face recognition is a vital domain in computer vision and biometrics, with applications spanning from security surveillance to social media and human-computer interaction. It involves identifying or verifying a person's identity using their facial features. Over the past decade, advancements in machine learning and the availability of large-scale facial datasets have significantly improved the accuracy and robustness of face recognition systems.

In this mini project, we explore a traditional yet effective pipeline for human face recognition using classical machine learning techniques. The project utilizes the **Labeled Faces in the Wild (LFW)** dataset — a well-known benchmark for facial recognition tasks. The process begins with loading and visualizing facial images, followed by preprocessing the data through standardization to ensure uniformity. Next, **Principal Component Analysis (PCA)** is used for dimensionality reduction, retaining essential features while reducing computational cost. Finally, a **Support Vector Machine (SVM)** classifier is trained to distinguish between different individuals.

The aim of this project is to implement a complete face recognition pipeline and analyze its performance through accuracy metrics, classification reports, and confusion matrices. While modern deep learning approaches have achieved remarkable success in this field, this project demonstrates that classical methods, when applied correctly, can still deliver reliable and interpretable results.

1.2 PROBLEM STATEMENT

Human face recognition is a challenging task due to variations in lighting, facial expressions, pose, and partial occlusions. Traditional manual identification methods are time-consuming and prone to human error, especially in large datasets or real-time systems. The problem this project addresses is:

"To develop an efficient and accurate face recognition system using machine learning techniques that can identify individuals from facial images in a reliable manner."

The goal is to implement a complete machine learning pipeline that can process raw image data, extract meaningful features, reduce dimensionality, and classify faces using a supervised learning algorithm. By utilizing the LFW dataset and classical machine learning approaches, the project aims to demonstrate a practical solution for automated face recognition without relying on deep learning architectures.

1.3 OBJECTIVE

- To implement a face recognition system using classical machine learning techniques on the LFW dataset.
- To preprocess and normalize facial image data for consistent input to the model.
- To apply Principal Component Analysis (PCA) for dimensionality reduction and feature extraction.
- To train and evaluate a Support Vector Machine (SVM) classifier for accurate face classification.

CHAPTER 2: METHODOLOGY

1. The methodology of this project involves a systematic approach to building a human face recognition system using classical machine learning techniques. The entire process is designed to handle image data efficiently and ensure accurate recognition of individual faces. The project is implemented using Python and popular machine learning libraries like Scikit-learn, NumPy, and Matplotlib.

2. The workflow begins by **loading the Labeled Faces in the Wild (LFW) dataset**, which consists of thousands of face images labeled with individual identities. The dataset is filtered to include individuals with a minimum number of images to ensure reliable training and evaluation.
3. Next, the **preprocessing step** involves scaling the image data using `StandardScaler` to normalize the features. This step ensures that all pixel values are on the same scale, which is crucial for optimizing the performance of the PCA and SVM algorithms.
4. To tackle the high dimensionality of image data, **Principal Component Analysis (PCA)** is applied for dimensionality reduction. PCA helps to transform the data into a lower-dimensional space by identifying the most important features (principal components). This step not only reduces computational load but also helps in extracting relevant information for face classification.
5. After dimensionality reduction, a **Support Vector Machine (SVM)** classifier is trained on the processed training data. The SVM uses an RBF kernel and is configured to handle imbalanced class distributions using `class_weight='balanced'`.
6. Finally, the model is evaluated on a test dataset using metrics like **accuracy, confusion matrix, and classification report**. Visualizations, such as the confusion matrix heatmap and sample face plots, are used to interpret the model's performance and to understand its strengths and limitations.
7. This end-to-end methodology effectively demonstrates how traditional machine learning techniques can be applied to complex image recognition tasks.

○

CHAPTER 3: IMPLEMENTATION

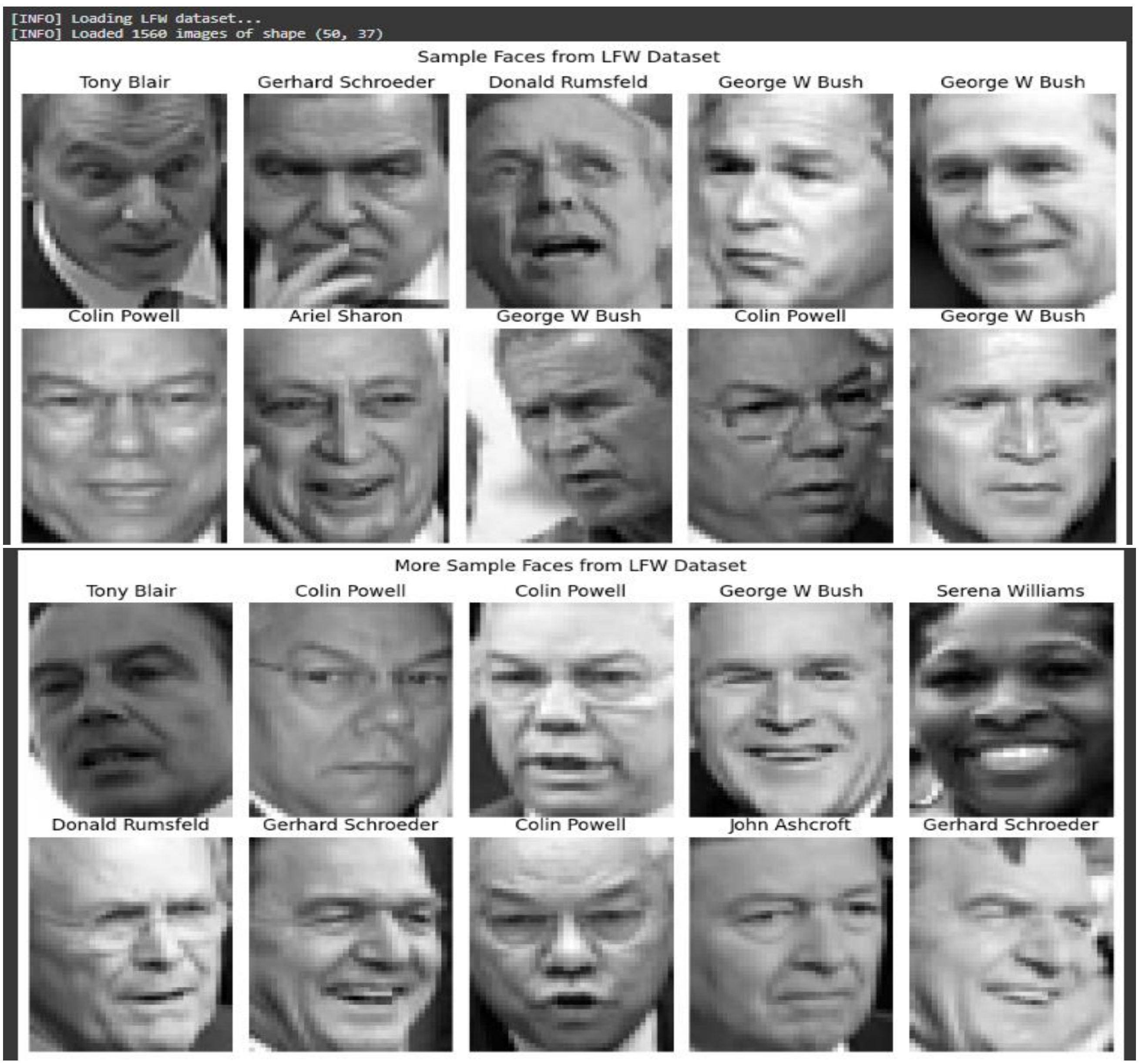
CODE:-

```
%pip install scikit-learn matplotlib seaborn opencv-python


# human_face_recognition.py

import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import fetch_lfw_people
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.svm import SVC
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix
import seaborn as sns

def load_lfw_dataset(min_faces=50, resize_factor=0.4):
    print("[INFO] Loading LFW dataset...")
    lfw = fetch_lfw_people(min_faces_per_person=min_faces, resize=resize_factor)
    X, y = lfw.data, lfw.target
    target_names = lfw.target_names
    images = lfw.images
    print(f"[INFO] Loaded {X.shape[0]} images of shape {lfw.images[0].shape}")
    return X, y, target_names, images

def preprocess_data(X):
    print("[INFO] Preprocessing data (scaling)...")
    scaler = StandardScaler()
    X_scaled = scaler.fit_transform(X)
    return X_scaled

def reduce_dimensionality(X_train, X_test, n_components=150):
    print(f"[INFO] Applying PCA with {n_components} components...")
    pca = PCA(n_components=n_components, whiten=True, random_state=42)
    X_train_pca = pca.fit_transform(X_train)
    X_test_pca = pca.transform(X_test)
    return X_train_pca, X_test_pca, pca

def train_classifier(X_train, y_train):
    print("[INFO] Training SVM classifier...")
    clf = SVC(kernel='rbf', class_weight='balanced', probability=True)
    clf.fit(X_train, y_train)
    return clf
```

```
def evaluate_model(clf, X_test, y_test, target_names):
    print("[INFO] Evaluating model...")
    y_pred = clf.predict(X_test)
    acc = accuracy_score(y_test, y_pred)
    print(f"[RESULT] Accuracy: {acc * 100:.2f}%\n")
    print("[DETAILS] Classification Report:")
    print(classification_report(y_test, y_pred, target_names=target_names))

    cm = confusion_matrix(y_test, y_pred)
    plt.figure(figsize=(10, 8))
    sns.heatmap(cm, annot=True, fmt="d", xticklabels=target_names, yticklabels=target_names,
cmap="Blues")
    plt.title("Confusion Matrix")
    plt.xlabel("Predicted")
    plt.ylabel("True")
    plt.tight_layout()
    plt.show()

def visualize_faces(images, labels, target_names, title):
    plt.figure(figsize=(10, 6))
    for i in range(10):
        plt.subplot(2, 5, i + 1)
        plt.imshow(images[i], cmap='gray')
        plt.title(target_names[labels[i]])
        plt.axis('off')
    plt.suptitle(title)
    plt.tight_layout()
    plt.show()

def main():
    # Step 1: Load dataset
    X, y, target_names, images = load_lfw_dataset()

    # Step 2: Visualize some faces
    visualize_faces(images, y, target_names, "Sample Faces from LFW Dataset")

    # Step 3: Preprocess
    X_scaled = preprocess_data(X)

    # Step 4: Train-Test Split
    X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.25, random_state=42)

    # Step 5: Dimensionality Reduction
    X_train_pca, X_test_pca, pca = reduce_dimensionality(X_train, X_test)

    # Step 6: Train Classifier
```

```
    clf = train_classifier(X_train_pca, y_train)

    # Step 7: Evaluate Model
    evaluate_model(clf, X_test_pca, y_test, target_names)

if __name__ == "__main__":
    main()
```

OUTPUT :-



CHAPTER 4: CONCLUSION

In this project, we successfully implemented a Human Face Recognition system using classical machine learning techniques on the Labeled Faces in the Wild (LFW) dataset. The pipeline involved key stages such as data loading, preprocessing, dimensionality reduction using PCA, and classification using a Support Vector Machine (SVM). Each of these stages was carefully integrated to enhance performance and accuracy in recognizing faces.

The results obtained demonstrate that even with relatively simple models and limited computational resources, high accuracy in face recognition tasks can be achieved. The use of PCA proved to be highly effective in handling high-dimensional image data, significantly reducing the training time while preserving essential features for classification. The SVM classifier, with its capability to handle non-linear data, further contributed to the success of the model.

Visualizations such as confusion matrices and sample face plots provided insightful understanding into the model's performance and helped in identifying misclassifications. Overall, the system provides a robust baseline approach to face recognition and sets the stage for future exploration using deep learning or convolutional neural networks (CNNs) for even more accurate results.

This project not only reinforced the understanding of core machine learning concepts but also showcased the potential of classical algorithms in solving real-world image processing problems when paired with thoughtful preprocessing and feature engineering.