



# Project Workflow Overview

Group7 Facial recognition

## 1. Project Objective

The aim of this project is to process the LFW dataset, extract features from images of faces, and use machine learning techniques to classify these faces. The approach includes face detection, feature extraction using HOG, dimensionality reduction using PCA, and classification using SVM models.

---

## 2. Data Processing Workflow

### 2.1 Dataset Preparation

- **Dataset:** LFW dataset organized in subfolders, each containing images of a unique person.
- **Output:**
  - Training features `X` and labels `y`.
  - Testing features `X_test` and labels `y_test`.
  - Array `Names` mapping label indices to person names.

### 2.2 Face Detection and Cropping

Faces are detected using OpenCV's Haar Cascade Classifier.

- **Process:**
  - For each image, detect all faces.
  - Select the face closest to the image center.
  - Crop and resize the detected face to `64x64` grayscale.
- **Special Handling:**
  - If no face is detected, the image is skipped.
  - Images from each person's folder are split into training and testing sets, with the first 10 images assigned to training and the rest to testing.

## 2.3 Output Data

The processed data includes:

- `X` : Training data of shape `(n_train, 64, 64)`.
  - `X_test` : Testing data of shape `(n_test, 64, 64)`.
  - `y` : Training labels.
  - `y_test` : Testing labels.
  - `Names` : Array of person names corresponding to label indices.
- 

## 3. Feature Extraction: HOG Features

Histogram of Oriented Gradients (HOG) is used for feature extraction.

- **Parameters:**
    - `pixels_per_cell=(8, 8)`
    - `cells_per_block=(2, 2)`
    - `orientations=9`
  - **Process:**
    - Each `64x64` image is transformed into a 1D feature vector.
    - The resulting feature matrix `X_hog` has a shape of `(n_samples, n_features)`.
- 

## 4. Dimensionality Reduction with PCA

Principal Component Analysis (PCA) is applied to the HOG features to reduce dimensionality while retaining 95% of the variance.

- **Benefits:**
  - Reduces computational cost.
  - Removes correlations between features.
- **Output:**
  - Transformed training data `X_train_pca` and testing data `X_test_pca`.

---

## 5. Model Training and Optimization

### 5.1 SVM Classifier

- A SVM classifier is trained on PCA-transformed training data.
- **Parameters:**
  - `kernel='linear'` : The kernel of the SVM
  - `class_weight` : weight of the class.

### 5.2 SVM Classifier with Grid Search

- A pipeline combining PCA and SVM with a linear kernel is built.
- Grid search is used to optimize the SVM hyperparameters:
  - `C` : `[0.1, 1, 10, 100]`
  - `gamma` : `[0.0001, 0.001, 0.01, 0.1]`
- 5-fold cross-validation evaluates performance.

---

## 6. Evaluation and Results

### 6.1 Metrics

Classification performance is assessed using:

- Accuracy
- Precision
- Recall
- F1-score

---

## 7. Summary

- The workflow processes the LFW dataset, extracting faces, applying feature extraction (HOG), reducing dimensionality (PCA), and training classification models SVM.
- SVM performance improves significantly with grid search optimization.

- Suggestions for improvement:
  - Use deep learning-based face detectors (e.g., MTCNN) for more robust face detection.
  - Replace HOG with CNN-based feature extraction for better classification results.