**2D FACE RECOGNIZATION SYSTEM**

**I. Model Files**

In the facial recognition and feature extraction system, two primary models are utilized: the **face detection model** (detectorModel) and the **face feature extraction model** (descriptorModel).

The face detection model (detectorModel) is based on the **Caffe** framework, consisting of two main files: a “.prototxt” file that defines the network architecture and a “.caffemodel” file that contains the pre-trained weights, often trained on large datasets like the **WIDER FACE Dataset**. This model typically employs architectures such as the **Single Shot Detector (SSD)** or **Faster R-CNN**, enabling efficient and fast face detection. It works by dividing the input image into a grid, predicting bounding boxes for detected faces along with confidence scores, and applying the **Non-Maximum Suppression (NMS)** algorithm to eliminate overlapping boxes. The input to the model is a preprocessed image resized to 300x300 pixels with pixel values adjusted by subtracting mean values ([104, 117, 123]) to normalize lighting conditions. The output is a list of bounding boxes and corresponding confidence scores representing the detected faces in the image.

The face feature extraction model (descriptorModel) maps each detected face to a fixed-length feature vector, acting as a unique "fingerprint" for the face. This model is based on **OpenFace** architecture. It takes cropped face images from the bounding boxes generated by detectorModel as input. Each face is resized to a fixed size (96x96) and normalized to ensure consistent output. The model generates a feature vector that can be used for tasks such as face comparison or classification, allowing the system to measure the similarity between different faces effectively.

A screenshot of a computer program

Description automatically generated

Hình 1. Model files

**II. Preprocessing**

In the preprocessing phase, the system prepares the input image data to ensure compatibility and consistency before performing facial detection and feature extraction. Images are loaded from the dataset, and their pixel values are normalized to fit the expected input range of the models. For face detection, the input image is resized and transformed into a blob using OpenCV's blobFromImage method. This blob standardizes the image dimensions (300x300 pixels) and applies mean subtraction based on predefined RGB values. This ensures that the input is scaled and centered appropriately, making the model more robust to variations in input size and lighting conditions.

A computer screen with green and white text

Description automatically generated

Hình 2. Preprocessing

**III. Face detection**

Face detection is handled using a pre-trained deep learning model stored in the file “**res10\_300x300\_ssd\_iter\_140000.caffemodel”**, which is based on the Single Shot Multibox Detector (SSD) framework. This model scans the image to identify regions likely to contain faces and outputs bounding boxes for each detected face. To ensure high accuracy, only faces with a confidence score greater than 0.5 are considered. Once detected, the bounding boxes are used to crop the faces from the original image. Each detected face is resized to a fixed resolution (e.g., 96x96 pixels) to ensure uniformity in downstream processing. The program supports detecting multiple faces per image, assigns each face an index, and prints it alongside the face. This enables handling images with multiple subjects effectively.

A screenshot of a computer screen

Description automatically generated

Hình 3. Face detection

**IV. Lighting adjustment**

To address variations in lighting conditions, the intensity of the pixel values is normalized. This process ensures the model is less affected by differences in brightness and contrast. The pixel values of the cropped face region (roi) are scaled by 1/255, reducing the pixel intensity range to [0, 1]. This normalization step helps focus on the structural characteristics of the face, making the model robust to lighting variations. Additionally, resizing to 96x96 pixels ensures consistency across all face regions before they are passed to the descriptor model for feature extraction.

A screenshot of a computer

Description automatically generated

Hình 4. Lighting adjustment

V. Algorithms

The model uses SSD (Single Shot Multibox Detector) algorithm. It is a fast and accurate object detection algorithm designed to detect multiple objects in an image in a single pass. SSD uses a common CNN model (such as VGG16) as a base network to extract features from the input image. In this case, the “.caffemodel” file is trained with a ResNet10 backbone, optimized for both performance and speed.