**Name**: Preethi Sekar **UTD Mail ID**: pxs163930@utdallas.edu

**FACE RECOGNITION IN A VIDEO**

**Statement**:

Face detection and recognition in a video using ‘Haar Cascade Classifiers’ and ‘Local Binary Pattern Histograms’ respectively. This project is implemented using Python 2.7.

**Introduction**:

Videos are images that are displayed in a sequence with a fast frequency such that the human eye is not able to perceive the change. Identifying a specific frame, which is the equivalent of an image, and applying image processing techniques will help identify the face in the video. This project focuses on specifically recognizing the human face in a video file/frame.

**Global Approach:**

The program takes on a global approach wherein an input image (frame taken from a video) is compared to a set of training images and decide whether the face appears in the video.

**Haar Cascade Classifiers:**

A machine learning based approach is used, wherein a cascade function is used to train numerous images. A plethora of positive and negative images are used to train the function. This classifier is used to identify the object (faces in our case) in the video. OpenCV(cv2), a library in Python2.7 provides .xml documents where the data is already pretrained. We make use of this to detect faces in the video frame.

**Local Binary Pattern Histograms:**

Once the face is detected, a descriptor called Local Binary Pattern is used to recognize the face. This is a machine learning technique which uses texture as the primary feature identifier, to recognize different faces. This utilizes the concept where a pixel is compared to its neighbouring pixels in order to recognize the object (as a face with regard to this project). After the neighbouring pixels are recognized, an 8-bit array is created which is later converted to decimal. This is done for every pixel value. And a histogram is constructed form this array. This is used for training the data samples.

**Face Recognition:**

The input image (taken from the video) is detected using Haar Cascade Classifiers and then the histograms are generated form the Local Binary Pattern Histograms approach. This is compared to many histograms formed from the set of frames taken from the dataset and a prediction is made to decide whether the face in the dataset appears in the video.

**Accomplished Goals:**

The project identifies faces in a video file. This translates to identifying if a person appears in a video or not. For example, if a specific actor is a part of a movie. A video/set of images is the dataset which is used in training. After which, given an input video, the program identifies if the face is present in that input video.

**Challenges:**

1. Input images (taken from the video) are converted to grayscale for our convenience. This implies that we are ignoring the colour information in the video.
2. Since the project only uses the features of the face overall the project is not so efficient for people with similar face structure.
3. Lighting of the room should be consistent since the images are converted to grayscale and this plays a significant role in recognition.
4. Other factors such as position, colour, light illumination, motion is ignored which if utilized would yield better results.
5. Clarity of the Video and the dataset poses to be a challenge in identifying the video.

**Future Scope:**

1. The video recognition in this program can be improvised and trained to work over the internet.
2. When Local Binary Pattern Histograms are compared with histogram of oriented gradients approach, which is a novel approach to identify the face, they would yield a better accuracy.

**References:**

1. <https://en.wikipedia.org/wiki/Local_binary_patterns>
2. <https://docs.opencv.org/3.3.0/d7/d8b/tutorial_py_face_detection.html>
3. <https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html>