**Face Emotion Detection using OpenCV and FisherFace Recognizer**

**ABSTRACT**

Face recognition is the process of identifying people through facial images, has numerous practical applications in the area of bio-metrics, information security, access control, law enforcement, smart cards and surveillance system.We aim to augment the utility of face recognition with emotion as well as gender detection for the full purview of our project. Emotion detection and computing aims to enable machines to recognize and synthesize human emotions. As we all know, a change of user’s emotion is one of the foundation of communication. Emotional states can motivate human’s actions, and can also supplement the meaning of communication. Thus,in this project we first went through OpenCV library which is aimed at real time computer vision and includes more than 2500 optimized algorithms,which includes a comprehensive set of both classic and state of the art computer vision and machine learning algorithms. OpenCV helped us to detect faces and various facial regions like eyes,nose,lips and jaw and dlib was an improvement over OpenCV. We have used dlib library for mapping exact facial characteristics to obtain a higher accuracy during training and testing. Here, Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software to solve real world problems. For gender and emotion detection we make use of FisherFace Recognizer included under OpenCV which we train using dataset from KDEF(Karolinska Directed Emotional Faces) and IMDB-WIKI.FisherFace uses Linear Discriminant Analysis (LDA) to determine the vector representation. It produces float value in the prediction. This also means that the result is better compared to Eigenface, that is considered first successful method of face recognition.

**CONTENTS**

Preface iv

Acknowledgements v

1 Introduction 1

2 Related Work 3

3 Proposed Work 4

3.1 OpenCV . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

3.2 Haar Classifier . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3.3 Dlib Library . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

3.4 FisherFace Recognizer . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

4 Experimental Setup and Results Analysis 12

4.1 Face detection using Haar Classifiers . . . . . . . . . . . . . . . . . . . 12

4.2 Face detection using Dlib Library . . . . . . . . . . . . . . . . . . . . . 15

4.2.1 Extracting parts of the face using dlib, OpenCV, and Python . . 15

4.3 Gender and Emotion detection using FisherFace Recognizer . . . . . . . 21

4.3.1 Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

4.3.2 Running the program . . . . . . . . . . . . . . . . . . . . . . . 21

4.3.3 Data Preprocessing . . . . . . . . . . . . . . . . . . . . . . . . 21

4.3.4 About Models . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

4.3.5 About FacePlay . . . . . . . . . . . . . . . . . . . . . . . . . . 24

References 30

**Chapter 1**

**Introduction**

Face recognition is the process of identifying people through facial images. It has numerous practical applications in the area of biometrics, information security, access control, law enforcement, smart cards and surveillance system. Face recognition system identifies a face by matching it with the facial database. It has gained great progress in the recent years due to improvement in design and learning of features and face recognition models. Face recognition plays a crucial role in applications such as security system, credit card verification, identifying criminals in airport, railway stations etc. Although many methods have been proposed to detect and recognize human face developing a computational model for a large data base is still a challenging task. That is why face recognition is considered as high level computer vision task in which techniques can be developed to achieve accurate results.

Gender recognition methods often rely on face recognition. There are many gender classification approaches and methods. The gender recognition might be one of the factors that fastens the person authentication because you limit the search for a person to 50 percent of the population.

We have also performed emotion detection in our project as human emotions can have great impact on the way of using software and in various other dimensions of our life. Therefore, robust and accurate recognition of human emotional states can play an important role in many software systems in different application fields. Most facial recognition systems function based on the different nodal points on a human face. The values measured against the variable associated with points of a persons face help in uniquely identifying or verifying the person. With this technique, applications can use data captured from faces and can accurately and quickly identify target individuals. We can perform fast, accurate face detection with OpenCV using a pre-trained deep learning face detector model shipped with the library. OpenCV ships out-of-the-box with pre-trained Haar cascades that can be used for face detection.

To build our face recognition system, we will first perform face detection, extract face embeddings from each face using deep learning, train a face recognition model on the embeddings, and then finally recognize faces in images with OpenCV.

A computer program that decides whether an image is a positive image or negative image is called a classifier. A classifier is trained on hundreds of thousands of face and non-face images to learn how to classify a new image correctly. OpenCV provides us with two pre-trained and ready to be used for face detection classifiers:

• Haar Classifier

• LBP Classifier

We have used Haar Classifier with OpenCV for face detection in this project. The Haar Classifier is a machine learning based approach, an algorithm created by Paul Viola and Michael Jones; which are trained from many many positive images (with faces) and negatives images (without faces).For improvement of Face detection we make use of Dlib Library.

Fisherface is one of the popular algorithms used in face recognition, and is widely believed to be superior to other techniques, such as eigenface because of the effort to maximize the separation between classes in the training process. And therefore we will use FisherFaces for emotion and gender detection.We train Fisherface Recognizer using images from KDEF(Karolinska Directed Emotional Faces) which contains images of 70 individuals each displaying 7 different emotions,each emotion being photographed(twice) from 5 different angles. Also images from IMDB-WIKI dataset have been taken which is largest publicly available dataset of face images with gender and age labels for training.It contains 4,60,723 face images from 20,284 celebrities from IMDB and 62,328 from wikipedia thus 5,23,051 in total.

**Chapter 2**

**Related Work**

Li Cuimei[1] has proposed solution in Eyes Detection and Mouth Detection with in a Human Face Candidate.A new human face detection algorithm is proposed to implement a stronger whole detection system by appending three weak classifiers, i.e., a classifier based on skin tone histogram matching, a classifier based on eyes detection, and third, a classifier based on mouth detection.

Chung-Hua Chu, Yu-Kai Feng [2] proposed a method that detects the movements of eyeball and the number of eye blinking to improve face recognition for screen unlock on the mobile devices. In some cases, people can use the photos and face masks to hack mobile security systems, so they proposed an eye blinking detection, which finds eyes through the proportion of human face.

Bruce Poon, Hong Yan [3] proposed a PCA Based Human Face Recognition with Improved Method for Distorted Images due to Facial Makeup. Facial makeup may change the appearance of a face which can degrade the accuracy of an automatic face recognition system. Experiment results show that by applying the Gradientfaces technique at the preprocessing stage which computes the orientation of the image gradients in each pixel of the face images the recognition rates can be improved for a mixture of facial images with makeup and nonmakeup.

H. Salih and L. Kulkarni [4] proposed to present the methodologies in terms of feature extraction and classification used in facial expression and/or emotion recognition methods with their comparative study. The comparative study is done based on accuracy, implementation tool, advantages and disadvantages.

**Chapter 3**

**Proposed Work**

Our entire project is divided into the following parts :

1. Visualization of facial landmarks and implementing facial recognition on the frames obtained by the web-cam or on sample images. We have two approaches for it :

• OpenCV library

• Haar Classifier

• Dlib Library

2. Emotion Detection to myriad of emotions ranging from happy to sad, disgusted, surprised, afraid, angry and neutral. We have used the following technique to obtain a high degree of accuracy : • FisherFace, OpenCV and Dlib.

**3.1 OpenCV**

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. There are bindings in Python, Java and MATLAB/OCTAVE. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. The library is used extensively in companies, research groups and by governmental bodies. OpenCV was designed for computational efficiency with a strong focus on real-time applications. The library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous computing platform. To be able to recognize emotions on images we will use OpenCV. OpenCV has a few face recognizer classes that we can also use for emotion recognition. They use different techniques, of which we will mostly use the Fisher Face one.

**3.2 Haar Classifier**

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. Here we will work with face detection. Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then we need to extract features from it. For this, Haar features shown in the below image are used. They are just like our convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle.

**3.3 Dlib Library**

Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems.Its design is heavily influenced by ideas from design by contract and component-based software engineering. Thus it is, first and foremost, a set of independent software components. It is open-source software released under a Boost Software License.Dlib contains software components for dealing with networking, threads, graphical user interfaces, data structures, linear algebra, machine learning, image processing, data mining, XML and text parsing, numerical optimization, Bayesian networks, and many other tasks. The facial landmark detector implemented inside dlib produces 68 (x, y)-coordinates that map to specific facial structures. These 68 point mappings were obtained by training a shape predictor on the labeled iBUG 300-W dataset.

**3.4 FisherFace Recognizer**

A key problem in computer vision, pattern recognition and machine learning is to define an appropriate data representation for the task at hand. One way to represent the input data is by finding a subspace which represents most of the data variance. This can be obtained with the use of Principal Components Analysis (PCA). When applied to face images, PCA yields a set of eigenfaces. These eigenfaces are the eigenvectors associated to the largest eigenvalues of the covariance matrix of the training data. The eigenvectors thus found correspond to the least-squares (LS) solution. This is indeed a powerful way to represent the data because it ensures the data variance is maintained while eliminating unnecessary existing correlations among the original features (dimensions) in the sample vectors. When the goal is classification rather than representation, the LS solution may not yield the most desirable results. In such cases, one wishes to find a subspace that maps the sample vectors of the same class in a single spot of the feature representation and those of different classes as far apart from each other as possible. The techniques derived to achieve this goal are known as Discriminant analysis (DA). The most known DA is Linear Discriminant Analysis (LDA), which can be derived from an idea suggested by R.A. Fisher in 1936. When LDA is used to find the subspace representation of a set of face images, the resulting basis vectors defining that space are known as Fisherfaces. Fisherface algorithm considers the ratio between the variation of one person and that of another person. That is to say, it maximizes the determinant of between-class scatter matrix simultaneously, minimizing the determinant of within-class scatter matrix. Fisherface procedure is as the following. Let there be total N images and total c persons. Suppose the number of images from one person is K. From PCA we can get N1 eigenfaces. To minimize the determinant of within-class scatter matrix and maximize that of between-class scatter matrix, we constitute the Sw -1 sb matrix and get Fisherfaces.

**Chapter 4**

**Experimental Setup and Results Analysis**

We have used Haar Classifier provided in OpenCV for face detection and then Dlib which is improvement over OpenCV for face detection.

We have also used Fisher Face Recognizer provided in OpenCV for emotion detection. The latest version of OpenCV we are using is 3.2.0.

**4.1 Face detection using Haar Classifiers**

The following dependencies are required for detection using haar classifiers:

1.OpenCV should be installed.

2.Python should be installed.

3.Matplotlib should be installed.

Now, how do we detect a face from an image using the CascadeClassifier we just loaded? Well, again OpenCV’s CascadedClassifier has made it simple for us as it comes with the function detectMultiScale, which detects exactly that. Next are some details of its options/arguments:

• **detectMultiScale(image, scaleFactor, minNeighbors):** This is a general function to detect objects, in this case, it will detect faces since we called in the face cascade. If it finds a face, it returns a list of positions of said face in the form Rect(x,y,w,h)., if not, then returns None.

• **Image:** The first input is the grayscale image. So make sure the image is in grayscale.

• **ScaleFactor**: This function compensates a false perception in size that occurs when one face appears to be bigger than the other simply because it is closer to the camera.

• **minNeighbors:**This is a detection algorithm that uses a moving window to detect objects, it does so by defining how many objects are found near the current one before it can declare the face found.

**4.2 Face detection using Dlib Library**

**4.2.1 Extracting parts of the face using dlib, OpenCV, and Python Requirements**

1. dlib library

2. imutils to the latest version,

ensuring we have access to the face utils submodule: pip install –upgrade imutils

**Methodology:**

We perform the following steps:

• Import our required Python packages.

• Parse our command line arguments.

• Instantiate dlibs HOG-based face detector and load the facial landmark predictor

• Load and pre-process our input image.

• Detect faces in our input image.

**4.3 Gender and Emotion detection using FisherFace Recognizer**

**4.3.1 Requirements**

1.Python3

2.OpenCV

**4.3.2 Running the program**

Run the file facifier.py with the command python facifier.py

The program would ask you to use your webcam. Type y for yes and n for no.

**Webcam Mode**: If you choose y, the program would continuously process the images that is recorded through your webcam.

**Static Image Mode**: Otherwise, if you chose n, the program would ask for an image file to be processed. Put the image you want to process, preferably in jpg format, to the folder data/sample and simply type the file name to process it.

**4.3.3 Data Preprocessing**

Fisherface recognizer requires every training data to have the same pixel count. This raises a problem because the dataset from KDEF and IMDB does not have uniform size and thus produces error during training. To address this problem, emotion data prep.py and gender data prep.py are created. Both of them use face detection algorithm from face detection.py to detect faces in photos. Then, the picture would be normalized to uniform size (350px x 350px) and saved in grayscale to speed up the training process.

**4.3.4 About Models**

The included models are essential for the program to detect faces, emotions.

**HaarCascade:** These models are provided by OpenCV and allows the program to detect human faces. After some manual and automated testings, I decided to use the first alternate version.

If for some reason you want to change the way this program detect human faces, open face detection.py, search the following line:

faceCascade = cv2.CascadeClassifier(’models/haarcascade frontalface alt.xml’) and change the model path to the desired one.

**Emotion Classifier:** These models are created with train emotion classifier.py. Each model is trained with dataset from KDEF. There are 2 versions: normal and decent. The normal version is trained with all the data from KDEF, while the decent version is trained with modified data from KDEF.

Modified here means deleting obviously misleading emotions. For example, there was a picture labelled sad that shows the person smiling while having tears around the eyes. It is very unusual for people to smile while crying, but this one person does it. To achieve better result, the said picture is removed from dataset. Another example, a person shows no real emotion in a picture labelled angry. That particular picture is then re-labelled as neutral.

**4.3.5 About FacePlay**

Emotions under FacePlay FacePlay can classify 7 basic emotions: afraid, angry, disgusted, happy, neutral, sad, and surprised. Meaning of colored boxes around the faces The box highlights any detected face whether from the webcam or in a static image. A blue box indicates that Facifier classified that person as a male and a red box indicates that Facifier classified that person as a female.

References

[1] Li Cuimei,Qi Zhiliang,Jia Nan,Wu Jianhua , ”Human face detection algorithm via Haar cascade classifier combined with three additional classifiers” 2017 13th IEEE International Conference on Electronic Measurement Instruments (ICEMI).

[2] Chung-Hua Chu and Yu-Kai Feng,Member of IEEE, ”Study of Eye Blinking to Improve Face Recognition for Screen Unlock on Mobile Devices ,” J Electr Eng Technol.2017.

[3] Bruce Poon, M. Ashraful Amin, and Hong Yan, ”PCA Based Human Face Recognition with Improved Method for Distorted Images due to Facial Makeup,” International MultiConference of Engineers and Computer Scientists 2017 Vol I, IMECS 2017, March 15 - 17, 2017, Hong Kong .

[4] H. Salih and L. Kulkarni, ”Study of video based facial expression and emotions recognition methods,” 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, 2017, pp. 692-696.doi: 10.1109/ISMAC.2017.8058267.