SUKKUR IBA UNIVERSITY

MACHINE LEARNING

FINAL PROJECT REPORT

PROJECT TITLE: EMOJIFY

SUBMITTED BY:

AMJAD ALI (023-19-0131)
PREM SAGAR (023-19-0039)
HAMMAD AHMED (023-19-0018)

SEMESTER: BSCS-VII (A)

SUBMITTED TO: DR SHER MUHAMMAD DAUDPOTA

TABLE OF CONTENTS

1. INTRODUCTION

In today's world, everyone is using emojis with the textual messages while communicating someone. Emojis are the type of language which completely define the emotion of the individual. Emojis are used to indicate non-verbal cues. These cues have become essential part of product review, chatting, brand emotion etc. Today's generation prefer to express the emotions through emoticons not by typing a long text while communicating with each other. As, it is easy to detect human emotion from image. In this project we will classify human facial expressions to filter and map corresponding most suitable. Therefore, we thought of making customized emojis. Emojify is a software which deals with the creation of Emojis. So, we are creating a convolutional neural network which will be used to convert the facial expressions. We will be using Fer2013 dataset for this project.

2. PROBLEM STATEMENT

Humans are using several emojis to show moods or feelings in the online mode of communication which is not basically face to face. Emojis act as nonverbal cues for humans. Without them, online chatting and other areas seem emotionless which is not favorable for humans because humans are Social Animals. They have become a crucial part of emotion recognition, online chatting, brand emotion, product review, and a lot more. This deep learning project will allow us to see things more colorful in chatting world, we can easily recognize the emotions as well as we can obtain the personalized emoji for the individual. In this way, it will be very easy to detect the emotions of person.

3. DATASET DISCUSSION

The dataset we are going to use in our project named as **FER2013 Dataset (facial expression recognition)**. The following dataset consists of grayscale face photos with a resolution of 48 by 48 pixels. The images are equally spaced apart and are centered. The training set consists of **28,709 examples** and the public test set consists of **3,589 examples**.

This dataset includes seven facial expressions of the following emotions:

1. 0: angry
2. 1: disgust
3. 2: feat
4. 3: happy
5. 4: sad
6. 5: surprise
7. 6: natural

NOTE: The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral).

4. METHODOLOGY

This section describes the methodology that has been followed to develop the system. The system has been developed using Python language. First, we have tried CNN model and its architecture and pretrained model from python framework named "deep face" and then we selected the outperforming model.

4.1. DATA PRE-PROCESSING

We have performed various type of data pre-processing on dataset FER2013 like data cleaning, transformation, and data augmentation. For example, we decoded images into RGB grids of pixels and then transformed the images into to grayscale. After that, we converted the images into floating-point tensors. Then finally, we rescaled the pixel values from 0 to 255 into the [0, 1] etc.

4.2. PRE-TRAINED MODEL

In this step, we created a Convolutional Neural Network available in Keras library. As we have data in image format, so we have chosen CNN as it works best for grid-like data. This model will be trained on FER2013 dataset to recognize the emotions from facial expression in images. The model was evaluated using test data and we performed hyperparameter tuning until model got suboptimal validation score. CNN model has 60% validation accuracy on FER13 dataset.

4.3. NON-PRE-TRAINED MODEL

We tried emotion recognition from pretrained model available in the python framework named as "deepface". Deepface contains pretrained for emotion recognition, the Facebook DeepFace algorithm was trained on a labeled dataset of four million faces belonging to over 4'000 individuals, which was the largest facial dataset at the time of release. The approach is based on a deep neural network with nine layers. The Facebook model achieves an accuracy of 97.35% (+/- 0.25%) on the LFW dataset benchmark.

4.4. HAAR CASCADE CLASSIFIER

After selecting the model, haarcascade_frontalface_default.xml available in OpenCV's library was loaded. It is an Object Detection Algorithm, which identify faces in an image or a video. We used this model to get the bounding boxes of the identified faces in the webcam. Then, obtained boxes were fed to the model for emotion classification.

4.5. MAPPING TO EMOJI

After getting output from the model for each input frame, we mapped the facial expression coming from the frame to its corresponding emoji.

4.6. WEB APPLICATION AND DEPLOYMENT

In this step, we developed the web application using flask framework. The web application is used to provide an interface to the user to take video using webcam and show the corresponding emoticon for each facial expression coming from the frame in the video. After that we deployed our application on a free web hosting service.

5. MAJOR OUTCOMES

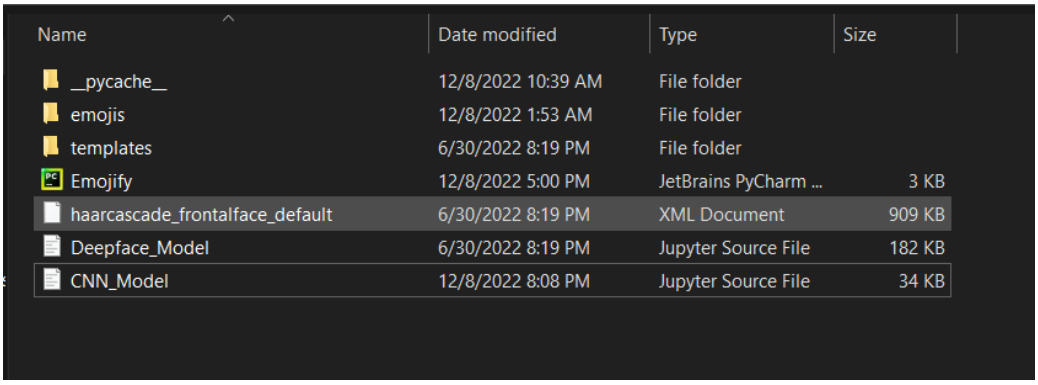
The main goal of this project is to learn about human facial expressions and then show emoji like facial expression. After training on dataset model will be able to detect emotion from facial expression and will show emoji according to that emotion.

6. CONCLUSION:

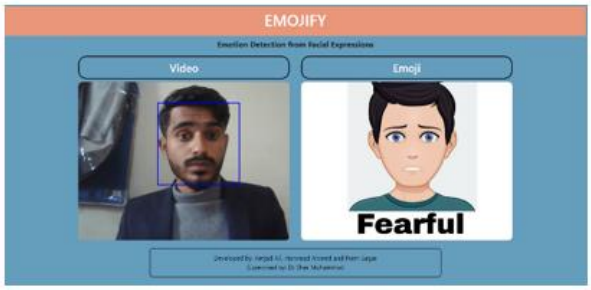
People in today's generation tend to use non-verbal cues like emoticons to communicate, so we thought why not bring out our emojis with advances in computer vision and deep learning. It is now possible to recognize human emotions from images. In this deep-learning project, we will classify human facial expressions to filter and map corresponding emojis.

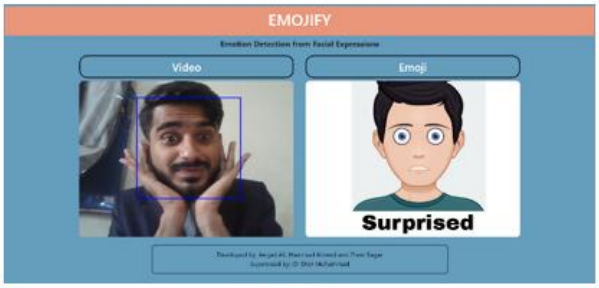
7. PROJECT IMPLEMENTATION

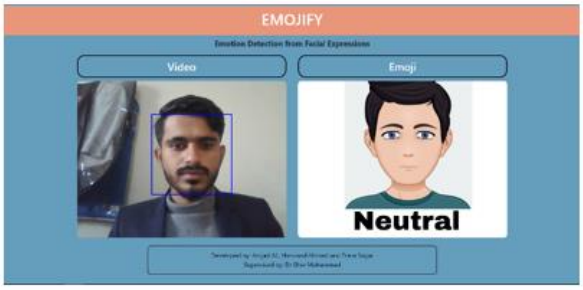
7.1. PROJECT STRUCTURE

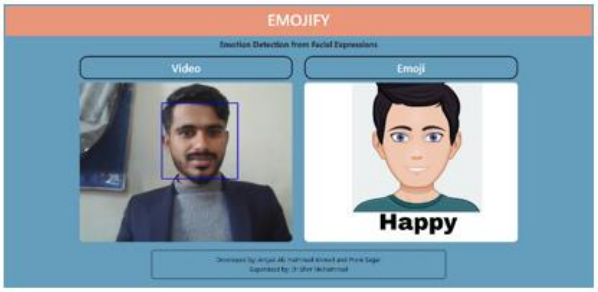


7.2. PROEJCT RESULT









7.3. CODE

https://github.com/amjadali070/Emojify_ML_Project

8. REFERENCES

https://www.kaggle.com/datasets/msambare/fer2013

https://data-flair.training/blogs/create-emoji-with-deep-learning/

https://www.irjet.net/archives/V8/i11/IRJET-V8I11164.pdf

https://viso.ai/computer-vision/deepface/