Emotion and Gender Recognition from Facial Image Data using Deep Learning

Samarth Nigam(19307R002)
Sachin Doifode (193079033)
Varsha S (193079005)

***Abstract*—Facial expression and gender recognition has been an active research area over the past few decades. Its applications in todays' world go from as simple as social media to criminal activities observation. This report discusses some methodologies undertaken for facial emotion and gender recognition. The work discussed in this report is inspired from several works proposing an end-to-end framework for facial expression and gender recognition, using deep learning models.The report explores around the classification of emotion and gender with the utilization of Deep Learning Models. This exploration includes both self made and pre-made model approaches.**

I. Introduction

Emotions are an inevitable portion of any interpersonal communication. They can be expressed in many different forms which may or may not be observed with the naked eye. Therefore, with the right tools, any indications preceding or following them can be subject to detection and recognition. There has been an increase in the need to detect a person's emotions in the past few years. There has been interest in human emotion recognition in various fields including, but not limited to, human-computer interface, animation, medicine, and security [5].

Emotion recognition can be performed using different features, such as face, speech , EEG, and even text. Among these features, facial expressions are one of the most popular due to a number of reasons; they are visible, they contain many useful features for emotion recognition, and it is easier to collect a large data set of faces (than other means for human recognition).

Automatic gender classification is a fundamental task in computer vision, which has recently attracted immense attention. It plays an essential role in a wide range of real-world applications such as targeted advertisement, forensic science, visual surveillance, content-based searching, human-computer interaction systems, etc. It concentrates on the efforts of perceiving human visual processing and recognizing relevant features that can be used to distinguish between female and male individuals. However, gender classification is still an arduous task due to various changes in visual angles, face expressions, pose, age, background, and face image appearance [4].

2 major approaches have been taken for this project. The first is using a VGG architecture and the second is using a Deep Convolutional Neural Network made from scratch.
The Deep Conventional Neural Network(D-CNN) technique comprises the phases of accepting the image as input and then transforming input images for further processing, dimension reduction, feature extraction, feature procurement, and classification, in this sequence. A convincing gain of the D-CNN over another traditional method in feature recognition technique is its capability to simultaneously perform following tasks like features extraction, reducing data dimension, and classification in the particular organized network structure. It can speed up recognition process and provide the result with high accuracy and minimum cost.

VGG16 is a convolutional neural network model achieves 92.7 per cent top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. Its architecture and weights can directly be used as a pretrained model for feature extraction. In this work, certain detection layers are placed at the output of VGG16 architecture for emotion and gender recognition.

The results for the following approaches have been evaluated on the basis of accuracy of detection, cross entropy loss and confusion matrices. The models have also been tested over new images and give sufficiently good results. Further, a simple GUI has been made to test the models for new live data in image as well as video mode.

II. Datasets

This section consists the basic description of the data sets used for this work. 2 different data sets are used for the emotion detection and gender detection.

A. Dataset for Emotion Recognition

The dataset used for emotion recognition is **FER2013**. The Facial Expression Recognition 2013 (FER2013) database was first introduced in the ICML 2013 Challenges in Representation Learning. This dataset contains 35,887 images of 48x48 resolution, most of which are taken in wild settings. Originally the training set contained 28,709 images,and validation and test each included 3,589 images. This database was created using the Google image search API and faces are automatically registered. Faces are labeled as any of the six cardinal expressions as well as neutral. Compared to the other datasets, FER has more variation in the images, including face occlusion (mostly with hand), partial faces, low-contrast images, and eyeglasses.

The dataset is available in form of a csv file with 3 columns while the training examples were stored in rows. The columns are namely, Emotion, Pixels, Dataset. The Emotion

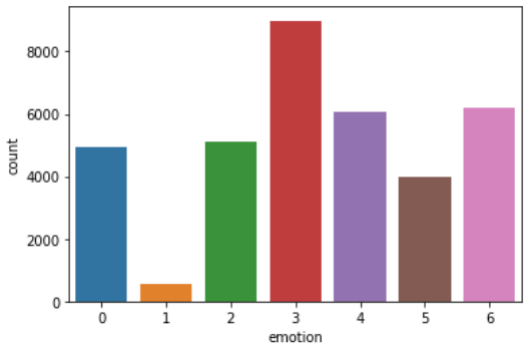


Fig. 1. Dataset sample space. Angry : 0, Disgust :1, Fear :2, Happy : 3 ,Sad : 4 , Surprise : 5, Neutral : 6

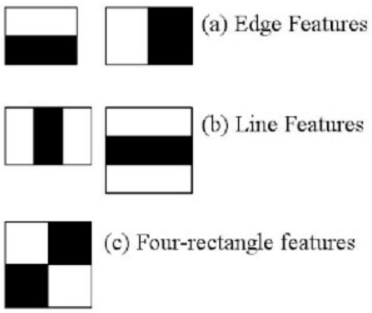


Fig. 2. Haar features

columns contains the emotion tag for corresponding image while the Dataset column assigns the image into Training, Test or Validation data. Some self compiled images were used for further testing the emotion recognition model. For testing further on new images, a GUI has been built which can work on live images and save them on the device memory. For the particular Emotion Recognition problem, the GUI also contains live video recognition feature.

B. Dataset for Gender Recognition

The dataset used for gender recognition is **Dataset for gender recognition from Kaggle** [7] . The data set is of cropped images of male and female . It is split into training and validation directory. Training contains 23,000 images of each class and validation directory contains 5,500 images of each class. The input images are of dimension 150*150 and are colored. There is no different test set in this particular dataset.

Some self compiled images were used for further testing the gender recognition model. For testing further on new images, a GUI has been built which can work on live images and save them on the device memory.

III. Analysis Pipeline

A. Haar Cascade Face Detection

A facial identification system is a technology capable of identifying a face of a person from a digital image or a video frame from a video source. Haar Cascade classifier is based on the Haar Wavelet technique to analyze pixels in the image into squares by function. This uses "integral image" concepts to compute the "features" detected. Haar Cascades uses the Ada-boost learning algorithm which selects a small number of important features from a large set to give an efficient result of classifiers then use cascading techniques to detect the face in an image.

Object Detection using Haar feature-based cascade classifiers is an effective object detection method [1]. 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 the sum of pixels under white rectangle from the sum of pixels under black rectangle.

For this work, we have used pretrained Haar Cascade Face Detectors during test time. They were not required in train time as both the emotion and gender datasets contained only facial images. In test time, they are used to just sample out the faces from the whole new image.

B. Model 1: VGG-Network

- VGG is an acronym for the Visual Geometric Group from Oxford University and VGG-16 is a network with 16 layers proposed by the Visual Geometric Group. These 16 layers contain the trainable parameters and there are other layers also like the Max pool layer but those do not contain any trainable parameters. This architecture was the 1st runner up of the Visual Recognition Challenge of 2014 i.e. ILSVRC-2014 [2].
- We modified the last layer for 7 class classification problem of emotion detection and 2 class classification problem of gender detection.
- SGD was used as the optimiser.
- loss function - Cross entropy is used as a loss function. Categorical loss function is used for multiclass classification in emotion detection. Binary cross entropy loss function is used for two class classification in gender recognition.

C. Model 2: Deep - Convolutional Neural Network

- We trained a D-CCN from the scratch with 6 convolutional layers. To generalize the network, dropouts are used in regular intervals. To prevent overfitting and provide regularization, batch normalization is also used. The model works on the input image and creates feature maps at each layer and uses highly specific features to solve multi class classification problems. It also has some residual skip connections from starting layers to the end layers. 'ELU' is used as the activation because it avoids the dying relu problem but also performs well

as compared to LeakyRelu at least in this case. The 'he_normal' kernel initializer is used as it suits ELU. Batch Normalization is also used for better results [6].

- Callback - A callback is an object that can perform actions at various stages of training (e.g. at the start or end of an epoch, before or after a single batch, etc). Two Callbacks used are
  1) Early stopping - Monitors the performance of the model for every epoch on a held-out validation set during the training, and terminates the training conditional on the validation performance. This prevents overfitting.
  2) ReduceLROnPlateau - Models often benefit from reducing the learning rate by a factor of 2-10 once learning stagnates. This callback monitors a quantity and if no improvement is seen for a 'patience' number of epochs, the learning rate is reduced.
- Optimiser - Adam optimizer is used. It can be looked at as a combination of RMSprop and Stochastic Gradient Descent with momentum. It uses the squared gradients to scale the learning rate like RMSprop and it takes advantage of momentum by using moving average of the gradient instead of gradient itself like SGD with momentum [3].
- Loss function - Cross entropy is used as a loss function. Categorical loss function is used for multi-class classification in emotion detection. Binary cross entropy loss function is used for two class classification in gender recognition.
- Plots - A violin plot is a method of plotting numeric data. It is similar to a box plot, with the addition of a rotated kernel density plot on each side. Violin plots are similar to box plots, except that they also show the probability density of the data at different values, usually smoothed by a kernel density estimator.

D. Steps followed for Emotion Detection

- The dataset was in csv format. We had to extract pixels of each image and reshape.
- We used 48*48 dimension gray-scale images of FER dataset to train the model. We dropped the disgust data as its sample space was very less. So the model was trained for a 6 emotion detection problem rather than 7.
- ImageDataGenerator is used to augment the data by applying any random transformations on each training image as it is passed to the model. This will make the model robust and also save up on the overhead memory.
- We used Adam optimiser with learning rate 0.001, a batch size of 32.
- Elu activation function with kernel intitalizer he_normal was used in the network.
- Early Stopping had a patience value of 11 while ReduceLROnPlateau had a patience value of 7.
- A Second model i.e., VGG network was trained both from scratch and with pre-trained weights. But it did not give good results.

E. Steps followed for Gender Detection

- We used 150*150 dimension images to train the model.
- ImageDataGenerator is used to augment the data by applying any random transformations on each training image as it is passed to the model. This will make the model robust and also save up on the overhead memory.
- We used Adam optimiser with learning rate 0.001, a batch size of 32.
- Elu activation function with kernel intitalizer he_normal was used in the network.
- Early Stopping had a patience value of 11 while ReduceLROnPlateau had a patience value of 7.
- A Second model i.e., VGG network was trained both from scratch and with pre-trained weights. But it did not give good results.

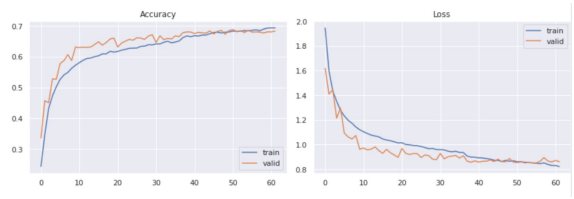
IV. Results

A. Emotion Detection

For testing purpose, we used Haar Cascade face detection to find the faces in the test image and then applied the model to predict the emotion of the face.

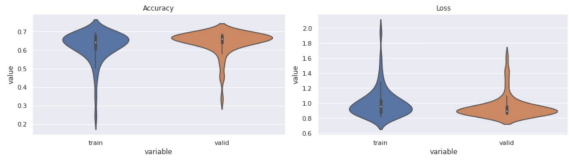
1) D-CNN

a) The following are the accuracy and loss plots



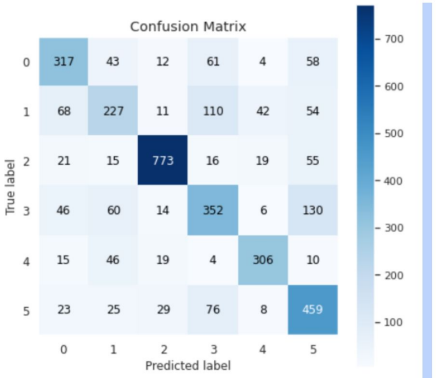
The epoch's history shows that accuracy gradually increases and achieves +68% accuracy on both training and validation set, but at the end the model starts overfitting training data. This issue was handled using early stopping where the training was stopped as soon as a dip in validation accuracy began to appear.

b) The following are violin plots of loss and accuracy,



We can observe that the validation accuracy distribution is more around the 0.65 and validation loss distribution is more around the 0.85.

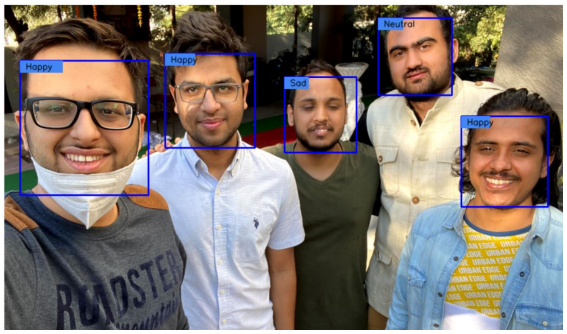
c) The following is the confusion matrix obtained while training with this model,

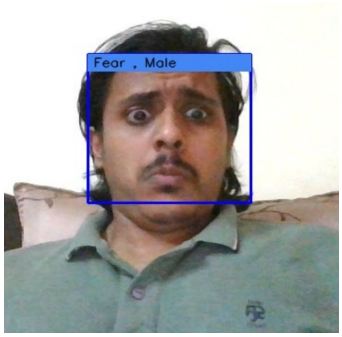


The confusion matrix clearly shows that our model is doing the best job on the classes 'happy' and 'neutral' but it's performance is low on the class 'fear'. It performs sufficiently well on the left classes. One of the reasons for low performance could be the fact that these two classes have less data.

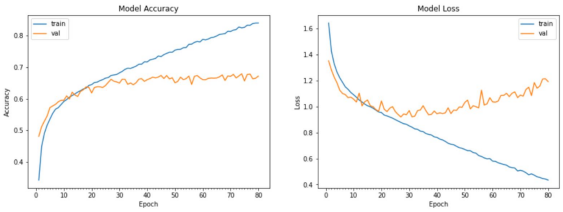
d) Test Results Examples

D-CNN architecture gave an accuracy of around 75 per cent on the test set of FER2013. This is evident from the very good emotion detection as shown in below images. During testing on new images, there were very less wrong predictions. Thus this model has been used for final evaluations and on GUI too.





The following is the accuracy and loss plot.



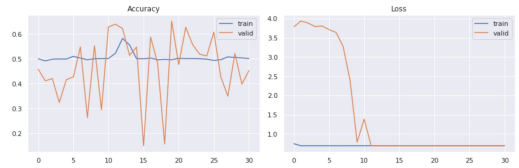
As it is evident from the above plots, this model doesn't train well using VGG Net. During testing it gave an accuracy of mere 51 per cent which is not at all good.

B. Gender Detection

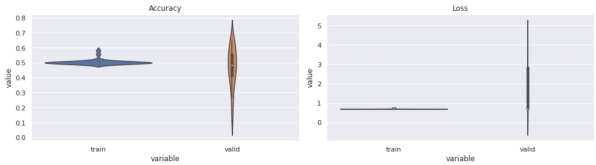
For testing purpose, we used Haar Cascade face detection to find the faces in the test image and then applied the model to predict the gender of the face.

1) D-CNN

a) The following are the accuracy and loss plots



b) The following are violin plots of loss and accuracy,



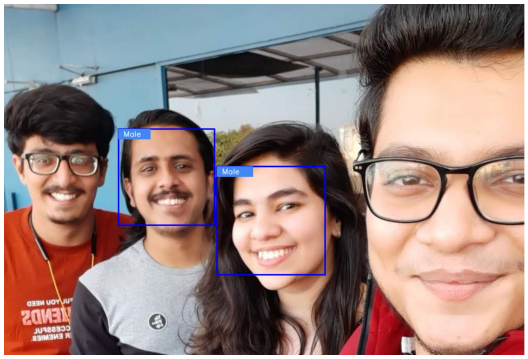
We can observe that the validation accuracy distribution is more around the 0.55.

c) Test Results Examples

The gender recognition doesn't work too good for this model. It is evident from accuracy and loss plots too as validation accuracy is not constant. Still some better detections and some bad detections are shown below.

i) Good test example

ii) Bad test example



V. GUI

A. Basic Build

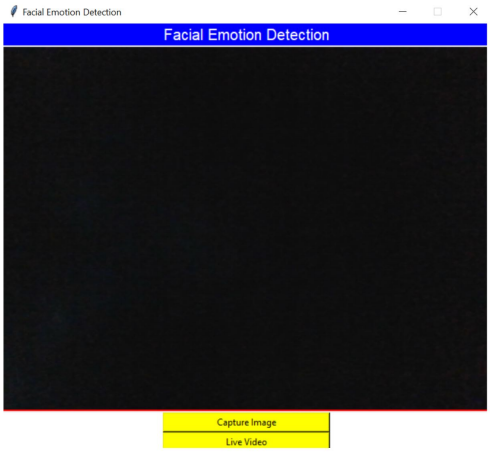
Once the model training and testing is finished, we move on to test the models functioning on new data. For this purpose, a simple GUI is implemented which works on live data. This GUI is made on python and certain packages are required to run it. The basic build of this GUI is discussed below-

The main window of GUI consists of basically 2 interactive sections.

The first section is essentially a display window which is used to display the live real time image captured by the webcam of the machine. This is also the window where further functionalities of the software are implemented which include real time detection of emotions from live feed and a snapshot feature to take the picture of current emotion and current gender. This section covers the most part of the main window.

The second section consists of 2 buttons which when clicked perform 2 basic tasks.

- Button1 : The upper button is named 'Capture Image'. On clicking this button the GUI takes the snapshot of the current picture and predicts the emotion and gender of the face present in it. This processed image is then stored in the 'Images' folder available in the folder. The obtained emotions and genders are displayed in the white space to the right of the buttons.
- Button2 : The lower button is named 'Live Video'. On clicking this button, an iterative function is called which basically predicts frame wise emotion of the live feed from webcam. The results in this video mode are displayed in a facial bounding box over the live image in the display section itself.



Both the above button functionalities work for both single and multiple face recognition. In either of the modes, it predicts the emotions with very good accuracy. The gender model is currently under training so the model included has less accuracy.

B. Programming Overview

The GUI is named 'emotion_detector.py'. It is made using the 'tkinter' module of python which is a popular module to make docker applications like a GUI. Following basic coding steps have been applied in the gui for detection purposes-

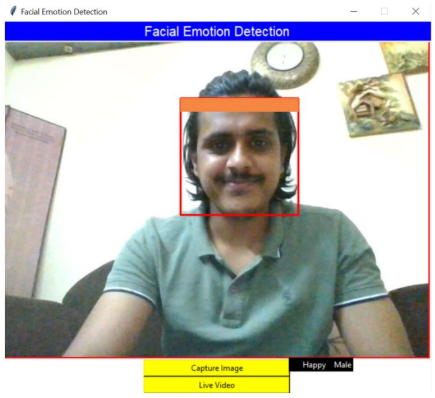
- For display, the programming has been done in the 'tkinter' module of python. It is used to create the main window with all its sections along with the buttons.
- Rest programming is in python and uses concepts of object oriented programming. This part essentially includes the setting up of all functions to capture images and to process them.

C. Using the GUI

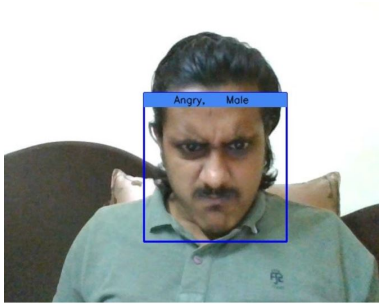
1) On initial opening of the GUI, the display section consists of the live image being captured by the webcam. On this live image, frame wise pre trained haar cascade face detectors are applied and thus 'orange' boxes can be seen around all faces in the display screen. They don't have any detection values over them.
2) On pressing the Capture button, the software basically picks that particular frame from the live feed, applies haar cascading to detect faces and then resizes these facial images. These resized images are then passed into the 'predict' function of the preloaded model weights for emotion and gender detection. The predictions are then printed on the white space to the right of the buttons. The Image captured is stored in the 'Images' folder of the software directory along with the detection results.
3) On pressing the Live Video button, the software iteratively picks up each frame of the live video, applies face detection followed by emotion detection. For good latency, an fps of 30 is considered. It is worth mentioning that the processing time of the model per frame should be less than the frame time for good latency. Currently, without GPU there is a factor of latency too in this version of the GUI.
4) If the Live Video button is pressed once, the GUI can't go back to the initial mode i.e. on pressing the Capture button after the Live Video button will not make Live Video mode quit. It would continue and just a frame would be saved.

D. Snapshots from GUI

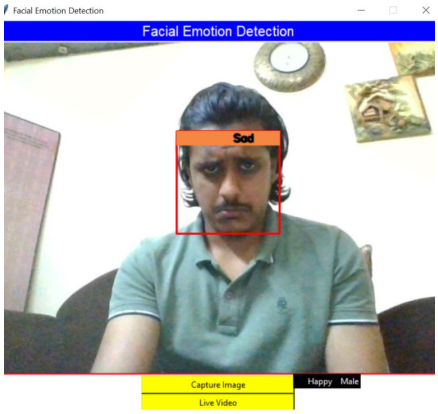
1) Initial Window



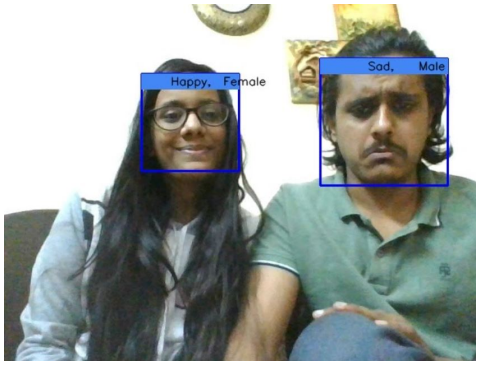
2) Captured Image



3) Video Mode



4) Multiple Faces



E. Video Emotion Detector

In addition to the GUI python file, a video emotion detector file has also been included in the software folder. It contains a code to take a video file as input and output a video file where all facial emotions in the input file are detected. A sample output file is included in the software folder too. To look at the functioning of this file, the readme file present in the software folder needs to be referred. This python file may take more or less time to execute based on the processing power of the device. If the device contains fast GPU the video processing would be fast else it may take some time.

VI. Discussion and Conclusion

This section discusses the observations and the inferences we made during the course of the project.

- During the literature survey, we found that VGGNet is a standard network used for image classification problem. While using it for emotion recognition, we observed that it did not give good accuracy(test accuracy 51 %).
- During the initial train runs, we used all the labels for emotion recognition which degraded the performance on the whole. When we selectively removed a few labels, we got better performance. The reason maybe due to the fact that the removed labels had very less training data.
- D-CNNs performed good on emotion data(test accuracy 74 %). Reasons can be successive usage of batchnorms and dropouts to prevent overfitting, presence of skip connections which are found to help in a multiclass classification problem.
- Gender recognition has scope of improvement while being trained on D-CNN(test accuracy 60 %). This low accuracy can be attributed to the fact that the cues to differentiate between the male and female are less than the cues to classify emotions. Hence the same model gave low accuracy for gender recognition compared to emotion recognition.
- In Emotion recognition problem the following were observed.
  1) Most images which had faces with teeth seen were labelled 'happy'.
  2) Images which had faces with wide open eyes and shrinked mouth seen were labelled 'surprised'.
  3) Images which had wrinkles or lines on the forehead, 'sad' emotion was predicted.
  4) Images which had deflated cheeks, narrowed eyes and lowered brow, 'angry' emotion was predicted.
  5) Images with faces which have slightly raised eyebrows, a taut brow were predicted 'fear'.
- In Gender recognition problem the length of hair and presence of facial hair were mainly the features which the model perceived.
- When the 'neighbour' parameters of Haar cascade function were tweaked, we observed that the more the value of it, less the faces detected in the image. If we decrease the 'scale factor', more number of faces detection in the image.

References

[1] Li Cuimei, Qi Zhiliang, Jia Nan, Wu Jianhua *Human face detection algorithm via Haar cascade classifier combined with three additional classifiers*.
https://ieeexplore.ieee.org/document/8265863

[2] Karen Simonyan, Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale image Recognition*
https://arxiv.org/pdf/1409.1556.pdf

[3] Diederik P. Kingma, Jimmy Ba *Adam: A Method for Stochastic Optimization*
https://arxiv.org/abs/1412.6980

[4] Amit Dhomne, Ranjit Kumar, Vijay Bhan *Gender Recognition Through Face Using Deep Learning*
https://www.sciencedirect.com/science/article/pii/S1877050918307853

[5] Shervin Minaee, Amirali Abdolrashidi *Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network*
https://arxiv.org/pdf/1902.01019.pdf

[6] Kaggle Code Reference
https://www.kaggle.com/gauravsharma99/facial-emotion-recognition

[7] Kaggle Dataset Reference
https://www.kaggle.com/cashutosh/gender-classification-dataset