Real-Time Facial Emotion Recognition Using Deep Learning

Abhishek Parve[1]
IT Engineer
Nerul, Navi Mumbai

Ravina Parve[2]
Web Developer
Nerul, Navi Mumbai

Abstract — As humans we have an ability to identify people's emotions based on their expressions. But achieving the same task with a computer algorithm is quite challenging. With the recent advancement in computer vision and machine learning, it is possible to detect emotions from images. This technology focuses on detecting emotions based on real-time video. The dataset containing 20,000 plus images of human expression is used for training the deep learning model for detecting the emotion. Haar Cascade frontal face classifier is used to detect the face from the video. Convolutional Neural Network (CNN) is used to predict the real-time expression from the live video stream through webcam.

Index Terms —Deep Learning, Convolutional Neural Network (CNN), Face Detection, Haar Cascade, Emotion Detection.

1. Introduction

Facial expression is the common signal for all humans to convey the mood. Expression tells us a lot about a person's current state of mind. The idea is to display the emotion of a person through live video stream. The dataset containing 20,000 plus images of human faces which are labelled on the scale of 0 to 6, where 0, 1, 2, 3 labels indicate angry, disgust, fear and happy emotion whereas 4, 5, 6 as sad, surprise and neutral emotions respectively.

This huge training dataset is provided as an input to our Convolutional Neural Network model where the dataset is trained based on the labels. On the other hand, Viola-Jones Face Detection Technique, popularly known as Haar Cascades is used as a classifier for face detection. Though there are many Haar cascade classifier for eyes, mouth, nose detection. But for this technology we have only made use of Haar cascade frontal face classifier.

We are using real-time video stream from webcam to display our predicted emotions on subject's face by displaying the square on the face and stating the predicted emotion on the screen.

2. Related Work

Previously there were many approaches which were used, one of the models proposed by "Standford University" professors called "Facial Emotion Recognition Real-Time" was able display the emoji on the screen based on the emotion expressed by the subject through real time video. They made use of custom trained VGG S network with a face-detector provided by OpenCV. The limitations of this model were any shadow on a subject's face would cause an incorrect classification of 'angry'.

3. METHODOLOGY

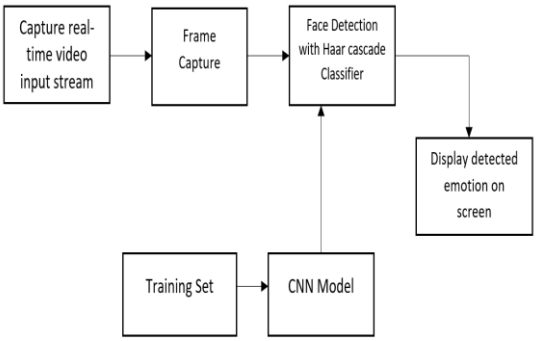

Fig 1.0 Block Diagram of RTFER

Initially the real time video input is captured through webcam and the face is detected from the video stream. The detected face is shown by a rectangle of green color around the face. Viola-Jones frontal face Haar Cascade classifier is used to detect the face. Once the face is detected, an emotion is displayed on the screen near the rectangle. We have used Convolutional Neural Network model for prediction of emotions and have achieved around 85% prediction accuracy for our model.

This model is trained on Kaggle's dataset containing 20,000 plus face images which are labelled on the scale of 0 to 6 based on the emotion. Kaggle is an online community for data scientists which allows users to find and publish data sets, explore and build models in a web-based data-science environment.

Major components of our technology are described below in details:

A. Haar Cascade Classifier

Haar Cascade is an Object Detection Algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper. We have used python's machine learning library known as OpenCV, which has various pre-trained classifiers. There are many Haar cascade classifier for eyes, mouth and nose detection, but for this technology we have only made use of Haar cascade frontal face classifier.

This classifier is trained on lot of positive and negative images. This process is known as feature extraction. For this classifier we have used "*Haarcascade_frontalface_default.xml*" file as a training data. To detect the face in the video we have used

detectMultiscale module from OpenCV. This creates a green colour rectangle with coordinates (x,y,w,h) around the face detected in the image.

Following were the important parameters which were considered.

a) scaleFactor

The value indicates how much the image size is reduced at each image scale. A lower value uses a smaller step for downscaling. It has a value of (x. y) where x and y are arbitrary values that we can set.

b) minNeighbors

This parameter specifies how many "neighbors" each candidate rectangle should have. A higher value results in less detections but it detects higher quality in an image. We can use a value of Y that signifies a finite number.

c) minSize

The minimum object size. By default, it is (30,30). The smaller the face in the image, it is best to adjust the minSize value lower.

B. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) are similar to typical neural network. It takes input as images. We have used Kaggle's dataset of 20,000 plus images as training data for our model. This model allows to incorporate properties that make the training process much more efficient and vastly reduces the number of parameters in the network.

Layers used in the model for training.

a) Max Pooling

The pooling layer's main objective is to reduce the spatial dimensions of the data propagating through the network. Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter.

b) Fully Connected Layer

In this we flatten the output of the last convolutional layer and connect every node of the current layer with every other node of the next layer. This layer basically takes input the output of the preceding layer, whether it is a convolutional layer, ReLU or pooling layer and outputs an n dimensional vector. Where 'n' is the number of classes pertaining to the problem.

4. Conclusion

In our proposed model we were able to successfully detect the person's emotion through webcam and display it on the screen. We were able to achieve prediction accuracy of around 85%.

Detecting emotions with technology is quite a challenging task, yet one where machine learning algorithms have shown great promise. By using Facial Emotion Recognition, businesses can process images, and videos in real-time from video feeds of a user interacting with the product, and the video can then be analysed manually to observer the users' reactions and emotions. Using facial emotion detection, smart cars can alert the driver when he/she is feeling drowsy. This technology would be very helpful for human interaction that are online such as interviews, meetings and examinations.

This model can be used with other innovations with more complex algorithms thus has more scope to grow and improve.

5. References

1. S. Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal, "Recurrent neural networks for emotion recognition in video," ICMI, pp. 467–474, 2015.

2. Z. Yu and C. Zhang, "Image based static facial expression recognition with multiple deep network learning," in Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, ICMI '15, (New York, NY, USA), pp. 435–442, ACM, 2015.

3. B. Kim, J. Roh, S. Dong, and S. Lee, "Hierarchical committee of deep convolutional neural networks for robust facial expression recognition," Journal on Multimodal User Interfaces, pp. 1–17, 2016.

4. G. Levi and T. Hassner, "Emotion recognition in the wild via convolutional neural networks and mapped binary patterns," in Proc. ACM International Conference on Multimodal Interaction (ICMI), November 2015

5. P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on, pp. 94–101, June 2010.

6. T. Liu, Z. Chen, H. Liu, Z. Zhang, and Y. Chen, "Multi-modal hand gesture designing in multi-screen touchable teaching system for human-computer interaction," in Proceedings of the Second International Conference on Advances in Image Processing, pp. 100–109, Chengdu China, June 2018.