FACIAL EXPRESSION RECOGNITION

Final Project Report

Submitted to Prof. Natarajan P

Course: Image Processing

Course code: CSE4019

Slot: B1+B2+TB1+TB2

Team Members:

Jyothi K C – 19BDS0144

Janhavi KHARE- 19BCE2412

ABSTRACT:

Behaviours, actions, poses, facial expressions and speech; these are considered as channels that convey human emotions. Extensive research has been carried out to explore the relationships between these channels and emotions. This project proposes a system which automatically recognizes the emotion represented on a face.

In 1969, Ekman and Friesen recognized a universality among emotions in different groups of people despite the cultural differences & classified six emotional expressions to be universal: happiness, sadness, disgust, surprise and fear.

Recent improvements have extended the applicability of facial emotion recognition to areas like chat room avatars and video conferencing avatars. The ability to recognize emotions can be valuable in face recognition applications as well. Suspect detection systems and intelligence improvement systems meant for children with brain development disorders are some other beneficiaries.

INTRODUCTION:

Facial expressions are not only a natural form of displaying emotions but also as key non-verbal communication technique. Efficient methods to recognize facial expressions can lead to striking improvements in the area of human computer interaction. Artificial Intelligence relies on facial emotion recognition to gain intelligence to model human emotions convincingly in robots.

Thus, a neural network based solution combined with image processing is used in classifying the universal emotions: Happiness, Sadness, Anger, Disgust, Surprise and Fear. Colored frontal face images are given as input to the prototype system.

After the face is detected, image processing-based feature point extraction method is used to extract a set of selected feature points. Finally, a set of values obtained after processing those extracted feature points are given as input to the neural network to recognize the emotion contained.

PROPOSED METHODOLOGY:

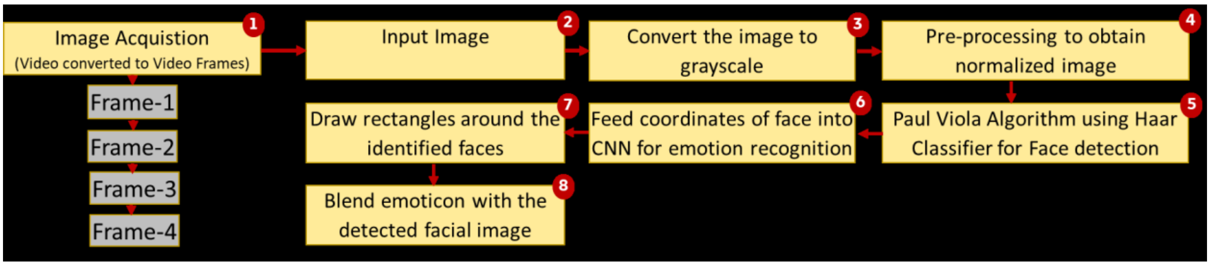

Figure 1. Flowchart of the proposed method

The proposed methodology involves image acquisition, pre-processing, face detection, segmentation, morphological processing and area calculation. Below is the algorithm of the proposed model.

Algorithm:

1. Read the input video frame image & convert the image into grayscale image

2. Pre-processing : Remove unwanted noise and blur and enhance the input image with filters

3. Apply Viola-Jones algorithm to detect the face region

4. Use bounding box method and crop the face region

5. Skin color segmentation & threshold value is used to extract non skin regions

6. Apply morphological operations to extract continuous boundaries of non-skin region namely eyes and mouth

7. Mask the boundary from the original image & Extract the mouth region

8. Area is calculated from extracted mouth region

9. Feed data into CNN and recognize facial emotions based on the value of area

10. Blend the emoticon for the recognised emotion with the detected facial region.

Advantages:

1. Extremely fast feature computation due to cascade of classifiers.

2. Does not scale the image but scales the features of the image.

3. Efficient feature selection Scale and location invariant detector gives accurate results.

Disadvantages:

1. Effective only on frontal images of faces not side pose.

2. Sensitive to lighting conditions.

3. Training is slow.

BACKGROUND

CONVOLUTIONAL NEURAL NETWORKS(CNN)

Convolution neural network (CNN) is widely used in image detection and recognition because it can recognize features regardless of the appeared position. In a neural network, each node in the previous layer gives effects to all nodes in the next layer. However, in CNN, only several nodes in the current layer give effects to the nodes in the next layer. So, CNNs can use local correlation. It means that CNN learns features from the images. CNN, it has two special layers such as a convolution layer and a pooling layer. In the convolution layer, features are extracted by convoluting filter to inputs. In the pooling layer, an input is down sampled to decrease the effect of small position shifting. CNN is consisted of some sets of these two types of special layers and normal NN. By using the pooling layer, the deep learning model is robust to the minor changes in the images. A critical feature of CNNs is their ability to achieve ‘spatial invariance,’ which implies that they can learn to recognize and extract image features anywhere in the image. There is no need for manual extraction as CNNs learn features by themselves from the image/data and perform extraction directly from images. This makes CNNs a potent tool within Deep Learning for getting accurate results.

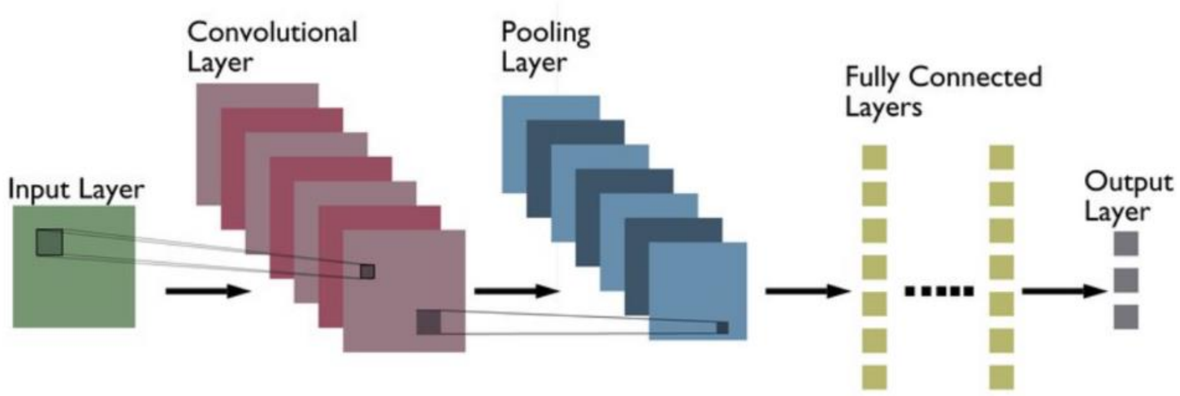

Figure 2. Architecture of CNN model

The following are definitions of different layers shown in the above architecture:

Convolutional layer: Convolutional layers are made up of a set of filters (also called kernels) that are applied to an input image. The output of the convolutional layer is a feature map, which is a representation of the input image with the filters applied. Convolutional layers can be stacked to create more complex models, which can learn more intricate features from images.

Pooling layer: Pooling layers are a type of convolutional layer used in deep learning. Pooling layers reduce the spatial size of the input, making it easier to process and requiring less memory. Pooling also helps to reduce the number of parameters and makes training faster. There are two main types of pooling: max pooling and average pooling. Max pooling takes the maximum value from each feature map, while average pooling takes the average value. Pooling layers are typically used after convolutional layers in order to reduce the size of the input before it is fed into a fully connected layer.

Fully connected layer: Fully-connected layers are one of the most basic types of layers in a convolutional neural network (CNN). As the name suggests, each neuron in a fully-connected layer is Fully connected- to every other neuron in the previous layer. Fully connected layers are typically used towards the end of a CNN- when the goal is to take the features learned by the previous layers and use them to make predictions. For example, if we were using a CNN to classify images of animals, the final Fully connected layer might take the features learned by the previous layers and use them to classify an image as containing a dog, cat, bird, etc.

Working of CNN :

1. First, the use of small CNNs alleviates us from slow performances in hardware-constrained systems such robot platforms.

2. And second, the reduction of parameters provides a better generalization under an Occam's razor framework.

3. Our first model relies on the idea of eliminating completely the fully connected layers.

4. The second design joins the cancellation of the completely associated layer and the consideration of the consolidated profundity shrewd distinguishable convolutions are remaining modules.

5. Both architectures were trained with the ADAM optimizer

6. We use Batch Normalization for better accuracy

7. Convolutional 2D network has been used to train the models

8. Pre trained models are available to test the testing sets.

9. We take the real-time video from the patients

10. Put it under processing

11. Output is displayed.

Literature Survey

Year of Publication	Author	Title	Advantages	Disadvantages
2015	R. Kaur and E. Himanshi	Face recognition using Principal Component Analysis	1. Data compression is accomplished by the low dimensional subspace representation. 2. It reduces the total entropy of data. 3. No knowledge of geometry and transparency of faces is required.	1. Sensitive to scale. 2. Recognition rate decreases with varying pose and illumination. 3. Considerable computational effort is required for the generation of Eigen values of the covariance matrix.
1991	M. Turk, and A. Pentland,	Eigenfaces for Recognition	1. Efficient and easy to implement . 2. Raw data can directly be used without any processing. 3. No knowledge of reflection and geometry of faces is required.	1. Varying light intensity, Scale and Orientation of an image affect in decreasing the accuracy. 2. Finding the eigenvectors and eigenvalues are time consuming. 3. Tested face images in experiments are taken in uniform

				background which is not suitable practically.
2012	Shah, Parin M	Face Detection from Images Using Support Vector Machine	1. Provide good accuracy. 2. Power of flexibility is high, the risk of over-fitting is less in SVM, scales relatively well to high dimensional data. 3. Works well with even unstructured and semi structured data like text, Images and trees	1. Complex, slow and takes a lot of memory. 2. Unsuitable for larger trainings sets due to lower execution speed, difficulty in training. 3. Low performance when the data set has more noise i.e. target classes are overlapping and in cases where the number of features for each data point exceeds the number of training data.
2007	S. A. Nazeer, N. Omar and M. Khalid	Face Recognition System using Artificial Neural Networks Approach	1. Good at capturing fairly complex patterns while keeping good generalization capabilities. 2. While it is slow to train, it is fast to use and its execution speed is independent of the size of the data it was trained on.	1. High processing time required for large neural networks. 2. The neural network needs training to operate and Detection rate Depend on facial features. 3. The modification of ANN is very complex.

			3. Very well suited for more complex real-world problems.	
2009	F. Z. Chelali, A. Djeradi and R. Djerad	Linear discriminant analysis for face recognition,	1. Solves illumination problem to maximize the ratio of between class scatter to within class scatter. 2. LDA achieves object reconstruction. 3. Projects features in higher dimension space into a lower dimension space.	1. Fails when all scatter matrices are singular. 2. Faces Small sample size problem when the no. of training samples are less than the dimension of feature spac 3. Can only classify faces“known” to the database.
2002	M. S. Bartlett, J. R. Movellan and T. J. Sejnowski	Face recognition by independent component analysis	1. Good performance in pattern recognition, noise and data reduction. 2. Powerful data representation as its goal is to provide an image decomposition and representation.th at is independent rather than uncorrelated	1. Cannot rank the order of dominant component. 2. At different point of times there is little variation in the answer (dependent components extra --cted through an iterative optimiz- ation procedure).

2005	Caifeng Shan, Shaogang Gong and P. W. McOwan	Robust facial expression recognition using local binary patterns	1. Recognition rate is unaffected by localization errors. 2. Computational simplicity. 3. Robust to monotonic gray-scale change.	1. Lower recognition rate than other similar methods. 2. Binary data produced by it is sensitive to noise. 3. Unsuitable for large scale databases due to long histograms produced which decrease speed.
2002	Thang V. Pham, Marcel Worring, Arnold W.M. Smeulders	Face detection by aggregated Bayesian network classifiers	1. Doesn't require much training data. 2. Can Handle both discrete and continuous data. 3. Highly scalable and fast, insensitive to irrelevant data.	1. Accuracy less than other complex algorithms since its impractical to have all mutually independent attributes. 2. Assigns a 0 probability value(called zero frequency). to a categorical variable if its category is not observed in the training set.
2013	D. Peleshko and K. Soroka	Research of usage of Haar-like features and AdaBoost algorithm in Viola-Jones method of object detection	4. Extremely fast feature computation due to cascade of classifiers. 5. Does not scale the image but scales the features of the image.	4. Effective only on frontal images of faces not side pose. 5. Sensitive to lighting conditions. 6. Training is slow.

			6. Efficient feature selection Scale and location invariant detector gives accurate results.	

CODE:

```
from keras.models import model_from_json
import numpy as np
import cv2
class FacialExpressionModel(object):
EMOTIONS_LIST = ["ANGRY", "DISGUST", "FEAR", "HAPPY", "SAD", "SURPRISE",
"NEUTRAL"];
def __init__(self, model_json_file, model_weights_file):
with open(model_json_file, "r") as json_file:
loaded_model_json = json_file.read()
self.loaded_model = model_from_json(loaded_model_json)
self.loaded_model.load_weights(model_weights_file)
print("Model loaded from disk")
self.loaded_model.summary()
def predict_emotion(self, img):
self.preds = self.loaded_model.predict(img)
self.preds[4:6] += 0.1
self.preds[1:3] += 0.2
lbl = np.argmax(self.preds)
return FacialExpressionModel.EMOTIONS_LIST[lbl], lbl
rgb = cv2.VideoCapture(0)
facec = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
font = cv2.FONT_HERSHEY_SIMPLEX
emo_happy = cv2.imread('happy.png',1)
emo_sad = cv2.imread('sad.png',1)
emo_fear = cv2.imread('fear.png',1)
emo_disgust = cv2.imread('disgust.png',1)
emo_surprise = cv2.imread('surprise.png',1)
emo_angry = cv2.imread('angry.png',1)
emo_neutral = cv2.imread('neutral.png',1)
emoji =
[emo_angry,emo_disgust,emo_fear,emo_happy,emo_sad,emo_surprise,emo_neutral]
def __get_data__():
_, fr = rgb.read()
gray = cv2.cvtColor(fr, cv2.COLOR_BGR2GRAY)
gray = cv2.equalizeHist(gray)
faces = facec.detectMultiScale(gray, 1.25, 5)
return faces, fr, gray
def start_app(cnn):
skip_frame = 10
data = []
flag = False
```

```
emo=None
while True:
faces, fr, gray_fr = __get_data__()
for (x, y, w, h) in faces:
fc = gray_fr[y:y+h, x:x+w]
fc = cv2.normalize(fc,None,0,255,cv2.NORM_MINMAX)
roi = cv2.resize(fc, (48, 48))
pred, lbl = cnn.predict_emotion(roi[np.newaxis, :, :, np.newaxis])
cv2.putText(fr, pred, (x, y), font, 1, (255, 255, 0), 2)
cv2.rectangle(fr,(x,y),(x+w,y+h),(255,0,0),2)
x1 = x + w//2
y1 = y + h//2
emo = emoji[lbl]
emo = cv2.resize(emo,(h,w))
fr[y:y+h,x:x+w] = cv2.addWeighted(fr[y:y+h,x:x+w],0.5,emo,0.5,0)
if cv2.waitKey(1) == 27:
cv2.destroyAllWindows()
break
cv2.imshow('Facial Expression Recognition', fr)
cv2.destroyAllWindows()
if __name__ == '__main__':
model = FacialExpressionModel("model1.json", "chkPt1.hdf5")
cap = cv2.VideoCapture('startV.mp4')
while True:
ret, frame = cap.read()
if ret:
frame = cv2.resize(frame, (1366, 800))
cv2.imshow('Facial Expression Recognition',frame)
cv2.waitKey(1)
else:
break
cap.release()
start_app(model)
```

RESULTS AND DISCUSSION

We obtained an accuracy score of 68.68% after 16 epochs using our proposed system design. While, using the pre-trained model, the accuracy rate was observed to be 88.68%.

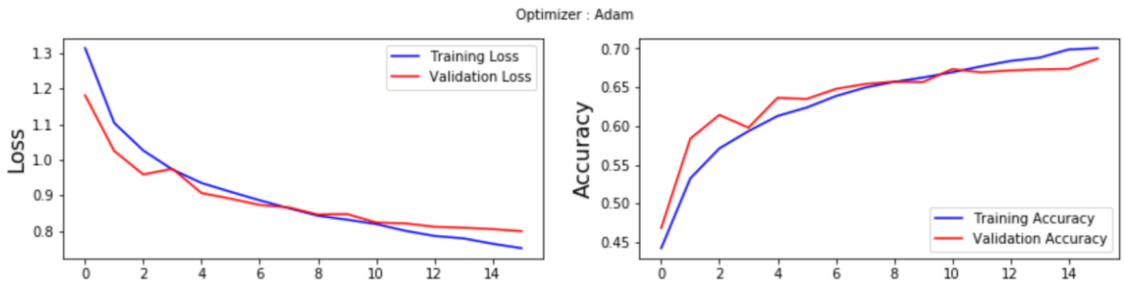

Figure 3. Plotted Graph showing Loss and Accuracy

Test cases :

Multiple live video feeds of faces fed into the system are taken as test cases here. While the overall system has an accuracy of 68.68 %, we observed that our system achieved almost perfect accuracy for happy and neutral faces (this is because of higher training for these images as more of them are present in the training set).

CONCLUSION AND FUTURE WORK

We have tried a general building plan for making ongoing CNNs. Our proposed structures have been deliberately worked keeping in mind the end goal to decrease the measure of parameters. We started by taking out the completely associated layers and by decreasing the measure of parameters in the rest of the convolutional layers by means of profundity astute distinguishable convolutions.

We have demonstrated that our proposed models can be stacked for multi-class classifications while keeping up ongoing derivations. Specifically, we have built up a dream framework that performs confront recognition, sex classification what's more, feeling classification in a solitary incorporated module. We have accomplished human-level execution in our classification's errands utilizing a solitary CNN that use current engineering develops.

In this project report, we saw how convolution neural network can help us with identifying the facial emotion one possesses.

In summary:

- A ConvNet architecture is in the simplest case a list of Layers that transform the image
  volume into an output volume (e.g. holding the class scores).
- There are a few distinct types of Layers (e.g. CONV/FC/RELU/POOL are by far the most popular).
- Each Layer accepts an input 3D volume and transforms it to an output 3D volume through a differentiable function.
- Each Layer may or may not have parameters (e.g. CONV/FC do, RELU/POOL don't)
- Each Layer may or may not have additional hyper parameters (e.g. CONV/FC/POOL do, RELU doesn't).

REFERENCES

1. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol 1, pp I-I. IEEE

2. V. Wati, K. Kusrini and H. A. Fatta, "Real Time Face Expression Classification Using Convolutional Neural Network Algorithm," 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 2019, pp. 497-501, doi: 10.1109/ICOIACT46704.2019.8938521.

3. El Maghraby A, Abdalla M, Enany O, El Nahas MY (2013) Hybrid face detection system using a combination of Viola-Jones method and skin detection. Int J Comput Appl 71(6):15–22

4. Real-time Convolutional Neural Networks for Emotion and Gender Classification Octavio Arriaga, Paul G. Pl oger, Matias Valdenegro, Hochschule onn- Rhein-Sieg, Sankt Augustin Germany, Hochschule onn-Rhein-Sieg, Sankt Augustin Germany, Heriot-Watt University. Edinburgh, UK. Francois Chollet. Xception: Deep learning with depth wise separable convolutions. CoRR, abs/1610.02357, 2016.

5. Zhu Q, Avidan S, Yeh M-C, et al. Fast human detection using a cascade of histograms of oriented gradients. In: IEEE computer society conference on computer vision and pattern recognition, New York, NY, 17–22 June 2006, pp.1491–1498.

6. Paisitkriangkrai S, Shen C and Zhang J. Face detection with effective feature extraction. In: 10th Asian conference on computer vision, Queenstown, New Zealand, 8–12 November 2011, pp.460–4

7. P. I. Wilson and J. Fernandez, Facial feature detection using Haar classifiers, Journal of Computing Science in Colleges, 21(4) April (2006) 127–133.

8. Emotion recognition using facial expressions Pawel Tarnowski, Marcin Kolodziej, Andrzej Majkowski, Remigiusz J. Rak.

9. A Survey on Human Emotion Recognition Approaches, Databases and Applications C.Vinola, K. Vimaladevi

10. Emotion Recognition from Facial Expressions using Multilevel HMM Ira Cohen, Ashutosh Garg, Thomas S. Huang, Beckman Institute for Advanced Science and Technology.

11. Bartlett MS, Littleworth G, Fasel I, Movellan R (2003) Real-time face detection and facial expression recognition: development and application to human-computer interaction. In: Proceedings CVPR workshop on computer vision and pattern recognition for human-computer interaction, vol 5

12. Chibelushi CC, Bourel F (2003) Facial expression recognition: a brief tutorial overview. CVonline: On-Line Compendium of Computer Vision 9

13. Muhtahir, O. Oloyede, Gerhard P. Hancke and Herman C. Myburgh Improving Face Recognition Systems Using a New ImageEnhancement Technique, Hybrid Features and the Convolutional Neural Network, IEEE Access, December, 2018.

14. Tanmoy Paul,Unmmmul Afia Shammi, Mosabbear Uddin A F Med,Ra Shehur Rahmn,Syoji Atiqur Rahman AhADA Study on Face Detection Using Viola Jones Algorithm for Various Backgrounds Angles and Distances.

15. Viola, P., Jones, M.J. Robust Real-Time Face Detection. International Journal of Computer Vision 57, 137–154 (2004).