

Integration of student gestures and facial emotion recognition to virtual meeting platforms along with the statistics of their concentration level during the class.

AUTHORS:

Kolli Sukhdev AP19110010276

Nallapaneni Sai Suprabhanu AP19110010232

Gopu Sai Alekhya AP19110010235

Valeti Manish AP19110010204

Mentor: Dr. Ravi Kant Kumar

Problem Statement


- during online lectures that not all the attendees are entirely active.
- speaker cannot understand the mood of the class.
- proctoring of exams, interviews in virtual mode required high bandwidth, RAM.



Facial recognition has become one of the most researched and sought-out topics in recent years. There is a huge amount of research work that is being done around the world in the same area. It is used in many applications such as camera surveillance.

Emotion detection is a subset of facial recognition which focuses mainly on the emotions that a human face is expressing.





Facial recognition has become one of the most researched and sought-out topics in recent years. There is a huge amount of research work that is being done around the world in the same area. It is used in many applications such as camera surveillance.

Emotion detection is a subset of facial recognition which focuses mainly on the emotions that a human face is expressing.



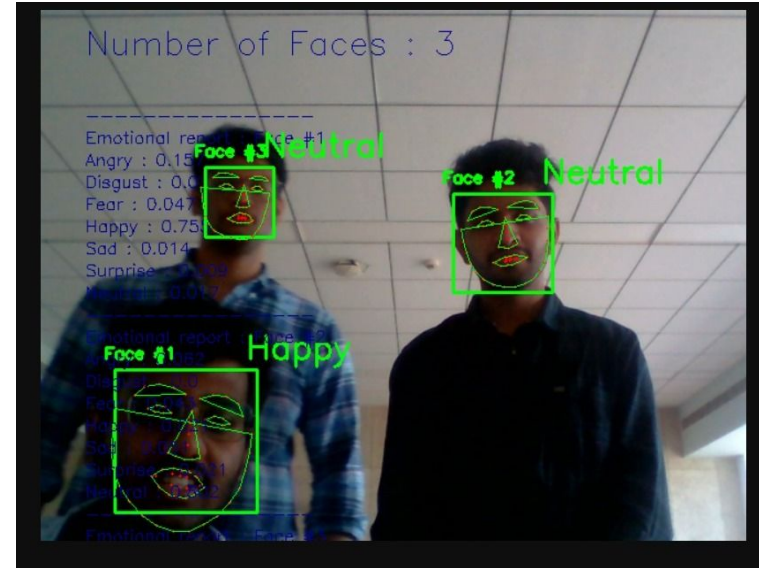
Aim of the project

The main aim of this project is to integrate the emotion detection methods into virtual meeting platforms like zoom and google meet where we can get the mood of the class. The model we developed helps in getting the overall statistics of the mood and emotion that the participants or students attending the meet are in.



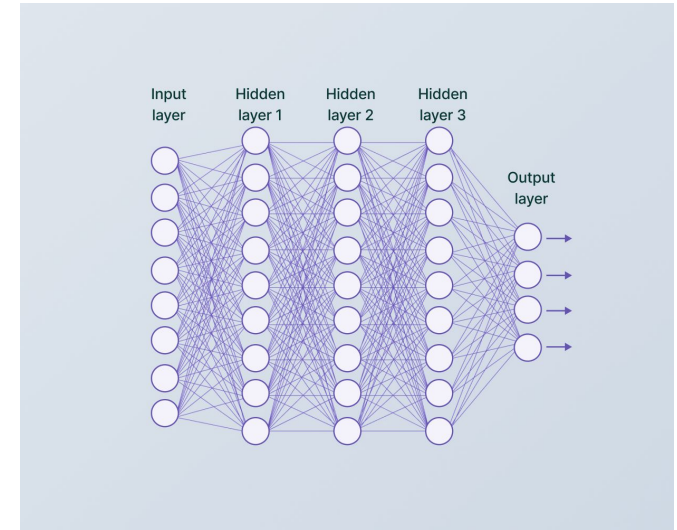
What's Different

The main important difference between our model and various other models is, that it can also work well under low light conditions and with a low quality webcam. The reason behind choosing the FER2013 dataset is it has more number of gray scale low quality images when compared to the other widely used datasets like AffectNet and Ascertain.

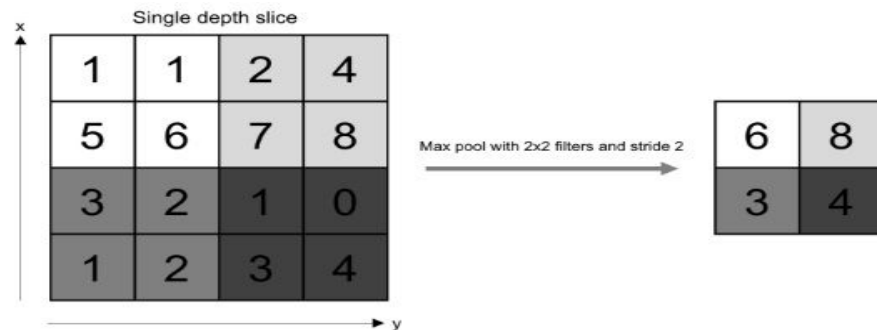


Approach

- The new era requires new standards and the Convolutional Neural Networks (CNNs) are the unique types of neural networks for processing the data with a grid-like topology.
- In traditional SVM approaches, selecting the filters like Gabor filters and the architecture of the filters was a large part of the job in order to extract as much information from the image as feasible.
- With the rise of deep learning and machine learning and with increased computing power, this work can now be automated.




- At each convolution step, for each input, we apply an activation function (typically ReLU). So far, we have only added dimensionality to our initial image input.
- To reduce the dimension, we then apply a pooling step.
- Pooling involves a down sampling of features so that we need to learn less parameters when training. The most common form of pooling is max-pooling



We have now covered all the ingredients of a convolution neural network :

- Convolution layer
- Activation
- Pooling layer
- Fully connected layer, similar to a dense neural network

The order of the layers will be: **MaxPool(ReLU(Conv(X)))**



```
def createModel2():
```

```
    model = Sequential()
```

```
    model.add(Conv2D(32, (3, 3), padding='same', activation='relu',  
                    input_shape=input_shape))
```

```
    model.add(MaxPooling2D(pool_size=(2, 2)))
```

```
    model.add(BatchNormalization())
```

```
    model.add(Conv2D(32, (3, 3), activation='relu'))
```

```
    model.add(MaxPooling2D(pool_size=(2, 2)))
```

```
    model.add(BatchNormalization())
```

```
    model.add(Conv2D(32, (3, 3), padding='same', activation='relu'))
```

```
    model.add(MaxPooling2D(pool_size=(2, 2)))
```


```
    model.add(Conv2D(32, (3, 3), padding='same', activation='relu'))
```

```
    model.add(Flatten())
```

```
    model.add(Dense(512, activation='relu'))
```

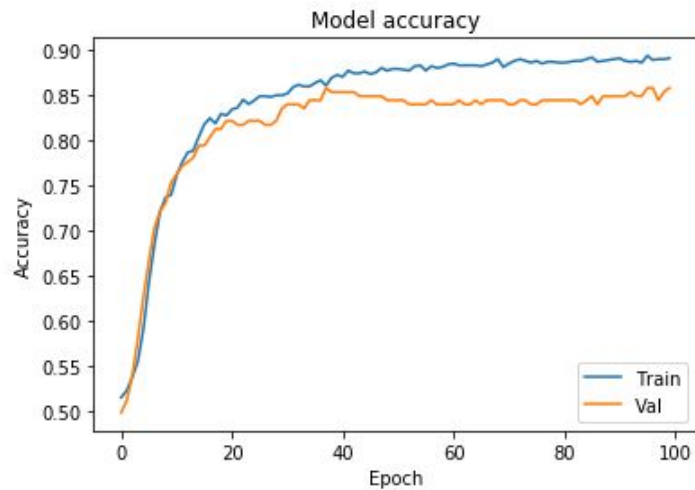
```
    model.add(Dense(nClasses, activation='softmax'))
```

```
    return model
```

- 
- This simple architecture produces over 440'000 parameters to estimate.
 - The computation time is around 8 hours on local machine. In order to prevent overfitting, we also apply Keras built-in data generation module.
 - The optimizer we chose is RMSprop, an optimizer that divides the learning rate by an exponentially decaying average of squared gradients.
 - The loss function we use is the categorical cross-entropy since we face a classification problem. Finally, the metric we use is accuracy.

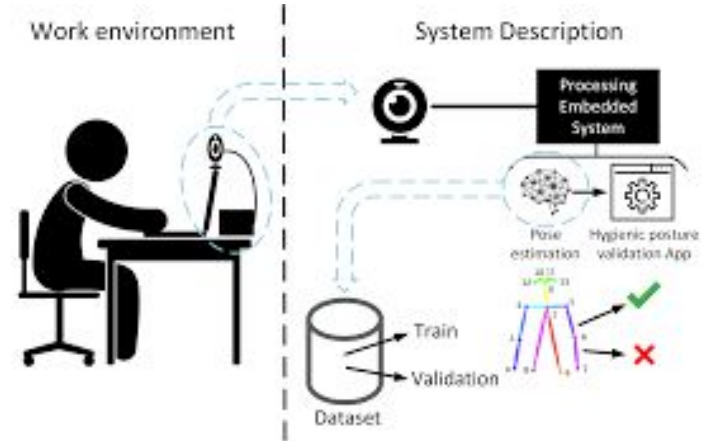
Layer (type)	Output Shape	Param #
conv2d_2 (Conv2D)	(None, 48, 48, 32)	320
max_pooling2d_2 (MaxPooling2D)	(None, 24, 24, 32)	0
batch_normalization_1 (Batch Normalization)	(None, 24, 24, 32)	128
conv2d_3 (Conv2D)	(None, 22, 22, 32)	9248
max_pooling2d_3 (MaxPooling2D)	(None, 11, 11, 32)	0
batch_normalization_2 (Batch Normalization)	(None, 11, 11, 32)	128
conv2d_4 (Conv2D)	(None, 11, 11, 32)	9248
max_pooling2d_4 (MaxPooling2D)	(None, 5, 5, 32)	0
conv2d_5 (Conv2D)	(None, 5, 5, 32)	9248
flatten_1 (Flatten)	(None, 800)	0
dense_1 (Dense)	(None, 512)	410112
dense_2 (Dense)	(None, 7)	3591
Total params: 442,023		
Trainable params: 441,895		
Non-trainable params: 128		


Results and Accuracy



Future Scope

1. Concentration and understanding level of students can't be judged just through the emotions. In addition to the emotion, posture is also an important feature to estimate the aforementioned attributes. Detecting the posture by training the model using the conventional techniques like in case of emotions wouldn't be much effective, rather we need to come up with a different approach.





2. After Integrating the complete model to a virtual meeting application, we need to focus on memory management and data consumption of application. Rather than storing the whole video data all over the period, we can simply store a small fragment of video data only when there is certain disturbance in the posture of the student. To do this we can generate a buffer of desired time frame and drop the buffer if that's clean without any disturbance.

3. Data consumption also is one of the huge obstacles that hold back the addition of more complex features to virtual meeting applications. To overcome this video data of the participants won't be continuously transferred to the host, rather disturbance in the posture of a participant would bring a pop up on the host's screen, this will allow the host to choose if the participant has to be manually observed or not.



References

The Facial Emotion Recognition Challenge from Kaggle,

<https://www.kaggle.com/c/challenges-in-representation-learning-facial-expressionrecognition-challenge/data>

End-to-End Multimodal Emotion Recognition using Deep Neural Networks,

<https://arxiv.org/pdf/1704.08619.pdf>

Facial Expression Recognition using Convolutional Neural Networks: State of the Art,

<https://arxiv.org/pdf/1612.02903.pdf>



THANK YOU