Face Emotion Detection

Group 3:   **Mentor: Prof. Mahadeva Prassanna**
Joshitha Gandra
Vaishnavi Patil

Objective: Classify the emotion on a person's face into one of seven categories, using deep convolutional neural networks.

Dataset

FER-2013 (published at the International Conference on Machine Learning (ICML)). This dataset consists of 35887 grayscale, 48x48 sized face images with seven emotions - **angry, disgusted, fearful, happy, neutral, sad, and surprised.**

Planned Approach

- **Haar cascade method** -to detect faces.
- Algorithms:
    - **ML: KNN, SVM**
    - **DL: CNN.**
- The Detected emotion will be displayed

Tools and Libraries: OpenCV, Tensorflow

TIMELINE

PHASE 1: Face Detection + Data Preprocessing (Data Splitting, standardization) + Train test splitting and data preparation, ML

PHASE 2: Apply DL algorithms and compare results.

PHASE 3: Improve the performance by using transfer learning techniques(like VGG).

References
Research Paper: https://arxiv.org/pdf/1307.0414.pdf
Dataset: https://www.kaggle.com/deadskull7/fer2013

Github: https://github.com/atulapra/Emotion-detection
Blogs/articles: TowardsDataScience   https://algorithmia.com/blog/introduction-to-emotion-recognition
https://github.com/rishabhjainps/Facial-Expression-Recognition

Minutes of the meeting -1:

- Dataset

- Discussed approach and algorithms

- Timeline

Minutes of the meeting -2:

- Implemented ML algorithms

  - SVM : 46.05% accuracy

  - KNN : 40.5% accuracy

- Implemented CNN

  - Training accuracy : 75.45%, Validation accuracy : 61.33%

NEXT Evaluation:

- Improve accuracy/performance of the deep learning model.

Evaluation :2

Since the FER2013 dataset is quite small and unbalanced, we have used transfer learning to boost the accuracy of our model.

Transfer Learning: In transfer learning, the knowledge of an already trained machine learning model is applied to a different but related problem. Main advantages are saving training time, better performance of neural networks (in most cases), and not needing a lot of data.

Preprocessing required for transfer learning models:

These new networks expected RGB images of no smaller than 197x197, So we resized and recolored the 48x48 grayscale images in FER2013 during training time

1) ResNet 50 Model: [Colab](#)

ResNet50 is a variant of <u>ResNet model</u> which has 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer. It has 3.8 x 10^9 Floating points operations.

Fundamental Concept of ResNet: (Residual Network)

In general, in a deep convolutional neural network, several layers are stacked and are trained to the task at hand. The network learns several low/mid/high level features at the end of its layers. In residual learning, instead of trying to learn some features, we try to learn some residual. Residual can be simply understood as subtraction of features learned from input of that layer. ResNet does this using shortcut connections (directly connecting input of the nth layer to some (n+x)th layer. It has proved that training this form of networks is easier than training simple deep convolutional neural networks and also the problem of degrading accuracy is resolved.

ResNet50 is a deep residual network with 50 layers. It is defined in Keras with 175 layers
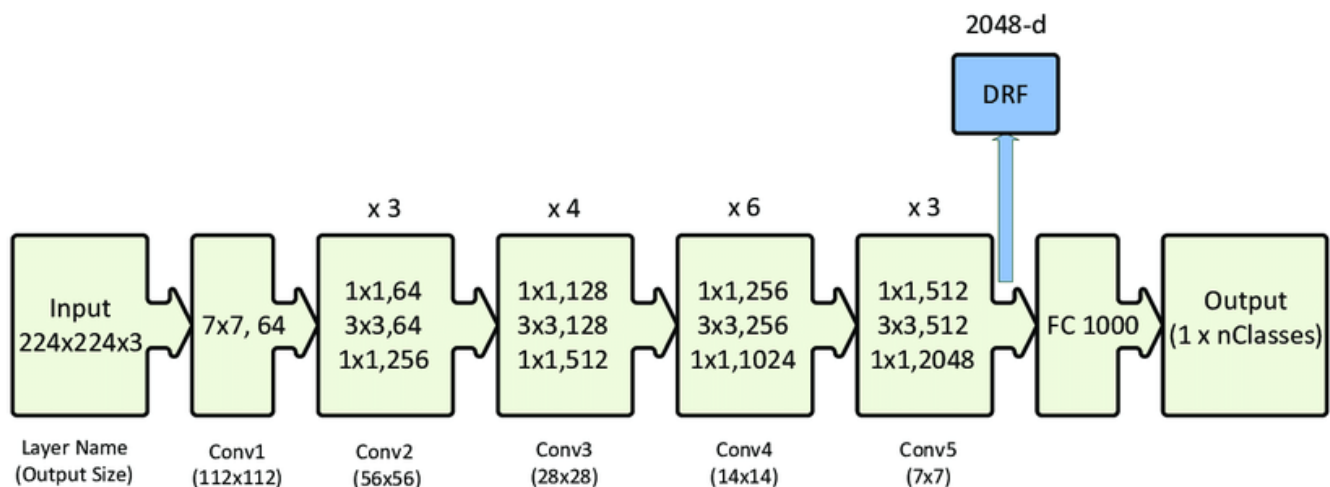

Our tuning:

1. Replaced the original output layer with two FC layers of sizes 4,096 and 1,024 respectively and a softmax output layer of 7 emotion classes.
2. We froze the first 170 layers in ResNet, and kept the rest of the network trainable.
3. Optimizer: Adam
4. Learning rate = 0.01
5. Batch size = 32

Accuracy: 85.71%

2) VGG -16 : [Colab](#)

1. Much shallower than ResNet50 : with only 16 layers, but VGG16 is more complex and has many more parameters.
2. VGG 16 has convolution layers of 3x3 filter with a stride 1 and always used same padding and maxpool layer of 2x2 filter of stride 2. It follows this arrangement of convolution and max pool layers consistently throughout the whole architecture.
3. In the end it has 2 FC(fully connected layers) followed by a softmax for output.
4. The 16 in VGG16 refers to it has 16 layers that have weights.
5. This network is a pretty large network and it has about 138 million (approx) parameters.
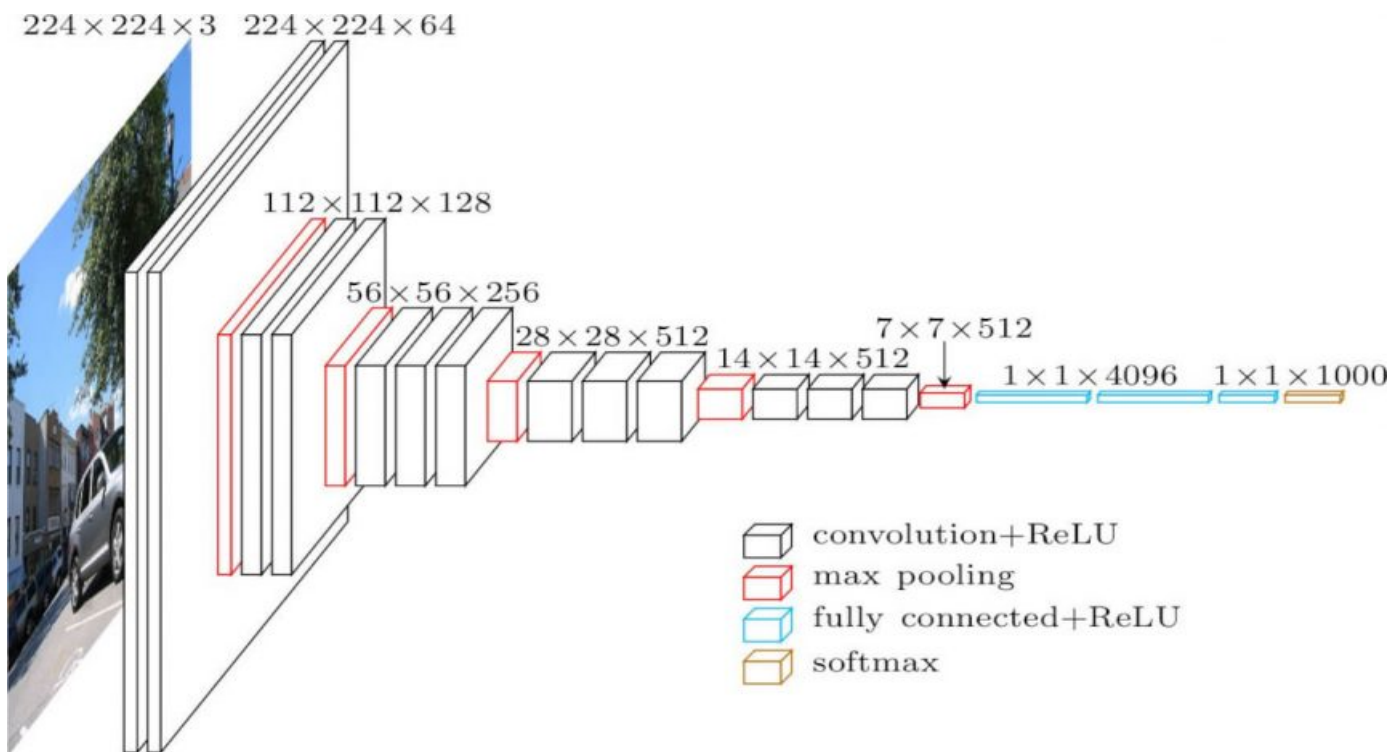
Tuning:

1. We kept all pre-trained layers frozen and added two FC layers of size 4096 and 1024 respectively with 50% dropout.
2. Optimizer: Adam

Accuracy: 85.79%

Overall Comparison:

Algorithm	Depth (Layers)	Parameters	Accuracy

SVM	–	1. Linear Kernel 2. RBf Kernel	37.4% 46.05%
KNN	–	1. K=1 2. K=2 3. K=3(def)	1. 40.5% 2. 26.8% 3. 34.92%
CNN	5	2.4(m)	61.33%
ResNet 50	50	25(m)	85.71%
VGG-16	16	138(m)	85.79%

Conclusion:

Transfer Learning Models increased the accuracy of the model compared to traditional CNN, which gave better results than ML algorithms like SVM, KNN.