# Emotion Recognition From Facial Expressions

Nitin Chakravarthy Gummidela, Manasa Donepudi and Shruthi Reddy Rodda

*Abstract*— **Expressions are one of the most important components of communication, Non-Verbal cues. Humans have been naturally trained to recognize facial expressions as a result of many social interactions. So it might seem a trivial task for a human to recognize expressions. But machines have a difficult time with the task. Facial emotion extraction can be used by retailers to evaluate customer interest or a hospital can monitor a patient's emotional state. Facial expression based emotion recognition is one of the topics in the field of computer vision that is receiving a lot of attention since the early nineties. In this paper, support vector machines and convolutional neural networks were used as a solution to this problem and the results were compared. Additional features such as 'Histogram of Gradients (HOG)' and 'Facial landmark' features were also explored as inputs. Also the behaviour of the neural networks were analyzed by diving into the network and interpreting the layer activations.**

## I. INTRODUCTION

Facial emotion recognition is the task of identifying emotion based on facial expression. Human facial expressions although complex in nature can be classified into 7 basic classes Happy, sad, Anger, Fear, Disgust, Surprise and Neutral. Certain set of muscles are activated for a certain expression. For example, cheek muscles while smiling and eyebrow muscles while angry. Humans are able to recognize these patterns in the muscle groups as expressions, as a result of infinitely many social interactions. In fact, children, who are only 36 hours old, can interpret some very basic emotions from faces[14]. But this trivial task is challenging for a computer. The task of facial emotion recognition is a challenging one as it has to be generalize so that it can be applied to unseen images or those that are captured in a wild setting. This paper tackles the facial emotion recognition problem using multiple machine learning algorithms, explains and compares the performance of each algorithm. Support Vector Machines, were used as a baseline for the comparative study. Various Convolutional Neural networks (CNNS) were implemented with images as inputs. Ablation study was performed with different hyper parameters to find out the best configuration for the best model. Also a CNN with SVM[9] on top was also explored. It was noticed that using SVM activation instead of a generic softmax activation resulted in a consistent increase over all the networks used. Further features such as HOG and Facial Landmarks were also used as inputs to the models. Extensive analysis was performed on the convolutional neural networks to explain their behaviour using GradCAM[15] and using the inter layer activations. To train our facial emotion classifers, we use the FER2013 dataset from the kaggle competition [8]

## II. RELATED WORK

Classical approach for facial expression recognition are based on Facial Action coding System (FACS)[12] which involves identifying facial muscles causing change in expression. Cootes et al[13] proposed a model that uses facial landmarks and perform PCA to derive Action Units(AUs) and classify use a single layered network to classify facial expressions. Early researchers tried to extract the best high-dimensional feature representation of faces[1], and find most important set of features with dimension reduction techniques such as PCA and sparse learning. Matthew Day exploits faical landmarks for emotion recognition[11]. [10] uses histogram of oriented gradients as inputs and trains an SVM. With later development of convolution neural networks, LeNet architecture[6] paper provides the message that better pattern recognition systems can be built by relying more on automatic learning and less on hand-designed heuristics.

In [7], a deep CNN with five convolution layers was developed for ImageNet[17] classification. ImageNet is 1.2 million sample data-set and has 1000 different classes. AlexNet[7] is known to perform well on the ImageNet data-set. The highest reported accuracy from leaderboard of kaggle competeion for this dataset is 71% [8]. Yichuan[9] has submitted a winning solution by a submission consisting of using a simple Convolutional Neural Network with linear one-vs-all SVM at the top of it instead of a softmax function. Simonyan et.al proposed a very deep convolutional neural network architecture (VGG16) for large scale image classifiation[5]. Sang et.al proposed a state of the art CNN with SMV on top for the Kaggle FER2013 data-set[14]. Arriaga et. al built a real time gender and emotion classifier based on the FER2013 data-set[2]

## III. MODEL AND APPROACHES

### A. *SVM with PCA classfier*

As a baseline for the comparative study we use SVMs using PCA reduced images and inputs. We perform a nested 5 fold cross validation where the outer loop is used to tune the number of principle components chosen and the inner loop is used to tune the hyper parameters of the SVM classifier for the best accuracy. The following are the parameters that were use in the tuning.

- number of principle components - [25, 50, 100, 200, 500, 1000]
- kernel - [linear function, Radial Basis function]
- C - [1e3, 5e3, 1e4, 5e4, 1e5]

- gamma - [0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1]

### *B. SVM with HOG and Facial Landmark features:*

Further we explored into SVM classifiers trained on higher level features such as 'Histogram of Oriented Gradients (HOG)' and Facial Landmarks. Facial Landmark features are used to localize and represent salient features such as jawline, eyes, cheek bones. Histogram of oriented gradients technique counts the occurrences of gradient orientation in localized portions of an image. 'dlib'[18] a in python library was used to extract these features from the images. We train an SVM classifier on these features using the hyperparameters tuned in the above section.

### *C. Convolutional Neural Networks*

We draw inspiration from three different CNN architectures, LeNet[6], AlexNet[7] and VGG16[5]. These are known CNNs that perform well on a image classification problems. We adapt these CNN architectures to our problem and use these to train on the Kaggle FER2013[8] dataset. The following are the configurations of the networks used in the study.

*1) Shallow CNN:* The Shallow CNN networks we used was based on the LeNet architecture (fig:1) which is a 2-layer convolutional network. The network architecture for our 2-Layer CNN model is:
(a) A 5x5 conv layer with 32 output channels, followed by a max pooling of size 2x2.
(b) A 5x5 conv layer with 64 output channels, followed by a max pooling of size 2x2.
(c) fully-connected layer with 2048 output neurons with dropout 0.3.
(d) fully-connected layer with 128 output neurons with dropout 0.3.
(e) fully-connected layer with 7 output neurons and softmax activation.
All convolution and fully connected layers use ReLU activation function.

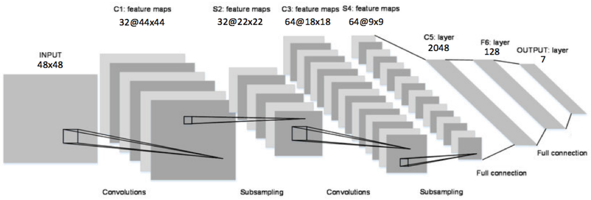


Fig. 1: LeNet Architecture

*2) Deep CNN models:* Further, we have also experimented with two deep CNN models, with architectures similar to AlexNet[7] and VGG16[5]. The following are the network architectures.

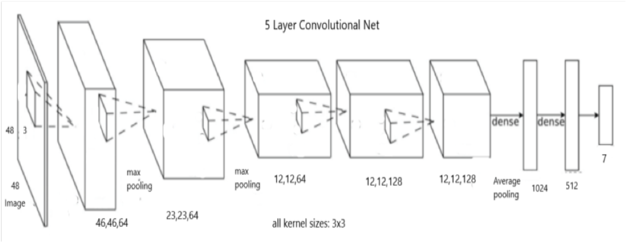


Fig. 2: AlexNet Architecture

*3) 5-Layer CNN Model:* The network architecture for our 5-Layer CNN model is shown in Fig. 2
(a) A 3x3 conv layer with stride 1 and 64 output channels, followed by a max pooling of size 3x3 with stride 2.
(b) A 3x3 conv layer with stride 1 and 64 output channels, followed by a max pooling of size 3x3 with stride 2.
(c) A 3x3 conv layer with stride 1 and 64 output channels.
(d) A 3x3 conv layer with stride 1 and 128 output channels.
(e) A 3x3 conv layer with stride 1 and 128 output channels, is followed by average pooling.
(f) fully-connected layer with 1024 output neurons with dropout 0.4.
(g) fully-connected layer with 512 output neurons with dropout 0.4.
(h) fully-connected layer with 7 output neurons and softmax activation.
All convolution and fully connected layers use ReLU activation function.

*4) 12-Layer CNN Model::* The network architecture for our 12-Layer CNN model[14] is as follows:
(a) 2 3x3 conv layer with stride 1 and 32 output channels, followed by a max pooling of size 3x3 with stride 2.
(b) 2 3x3 conv layer with stride 1 and 64 output channels, followed by a max pooling of size 3x3 with stride 2.
(c) 2 3x3 conv layer with stride 1 and 128 output channels.
(d) 3 3x3 conv layer with stride 1 and 256 output channels.
(f) fully-connected layer with 256 output neurons with dropout 0.4.
(g) fully-connected layer with 256 output neurons with dropout 0.4.
(h) fully-connected layer with 7 output neurons and softmax activation.
All convolution and fully connected layers use ReLU activation function.

### *D. SVM over global features extracted from CNNs:*

Once we train the CNNs, tune the hyper parameters and obtain the best model, We train a SVM classifier over the activations of the last layer from each CNN. This is done since the activation of the last layer of a CNN are representative of the global features of an image.

## IV. DATA SET

The data-set we used for this project was KAGGLE Facial Expression Recognition2013 challenge dataset[8].

This data-set consists of pre-cropped 48x48 gray scale images. The data set is divided into three sections.

- Training samples - 28709
- Public Test samples - 3,589
- Private Test samples - 3589

All the images are associated with one of the seven emotion class labels - Anger, Disgust, Fear, Happiness, Sadness, Surprise and Neutral.The public test set was made public during the competition to serve as a validation set for model development. The final test set, was used to determine the winner of the competition.

### A. *Data-set Analysis:*

FER2013[8] is a very challenging data-set for the emotion recognition problem. This is due to fact that the data-set consists of many wrongly labeled samples and also images that are not faces as illustrated by Fig. 4. Thus the model has to generalize and predict correctly even in the case of a wrongly classified input. This is a hard problem to handle. Further the data-set is imbalanced. The training data distribution shows that the number of samples corresponding to the emotion 'Disgust' are very less compared to the number of samples corresponding to the emotion 'Happy'. Where as other emotions have relatively a similar number of samples.

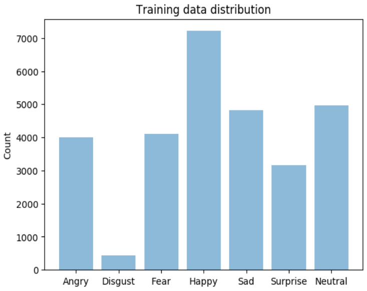


Fig. 3: Training data distribution

fig:3 shows the distribution of data in the training and testing sets that were used in several models in this project.

## V. RESULTS

The following are the results from the studies performed.

### A. *SVM methods:*

The following are the hyper parameters that were obtained by performing the crossvalidation technique for SVM using principal components as our features.

- number of principle components 200
- kernel - Radial Basis function
- C - 1e3
- gamma - 0.0001

Fig: 5 shows the confusion matrix for the classifier and Fig: 6 shows the classification reports. From both the confusion matrix and the precision recall table, we can observe that the ”disgust” class is completely wrongly classified by SVM. This is because, the number of training data points of ”disgust” class are very low compared to other classes and the SVM model is biased towards majority class.

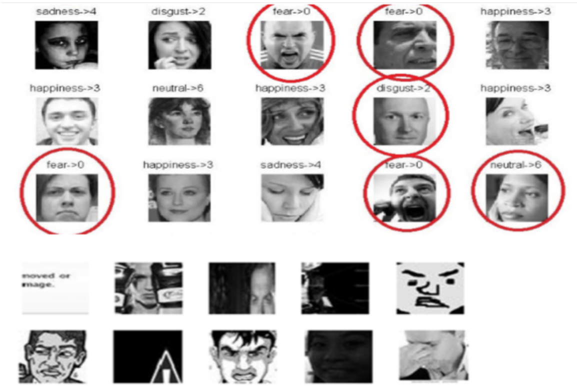


Fig. 4: Wrongly labeled images and images with no faces in the data-set[19]

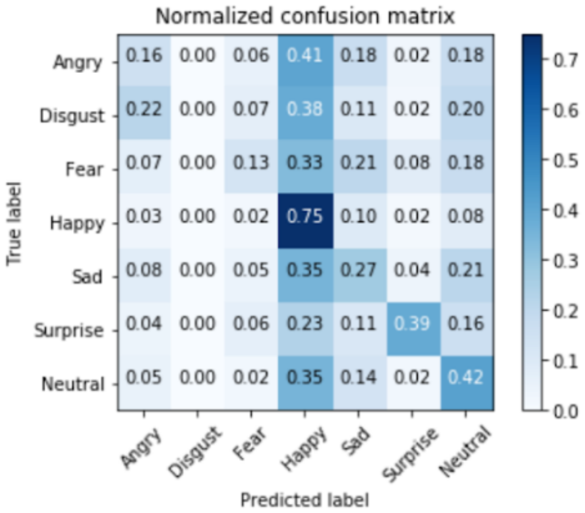


Fig. 5: SVM on PCA confusion matrix

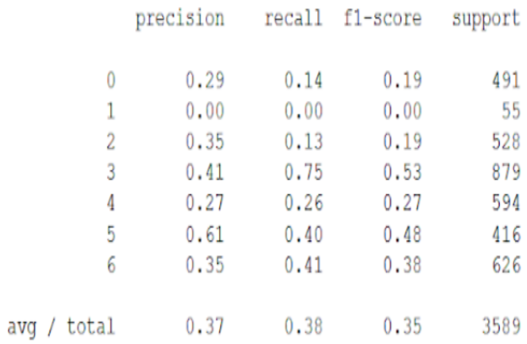
Further we use SVM on HOG and Facial landmark features with the same hyper parameters. The accuracy for both the models are reported in tableI. From the accuracy table, we can see that SVM on HOG and Landmark Facial features have performed much better.

| Model | Accuracy |
|---|---|
| SVM on PCV | 39.98 |
| SVM on HOG and Face Landmarks | 48.2 |

TABLE I: SVM models accuracy

### B. *CNN models:*

Ablation studies were performed on number of hyper-parameters to reach to a model with the highest classification



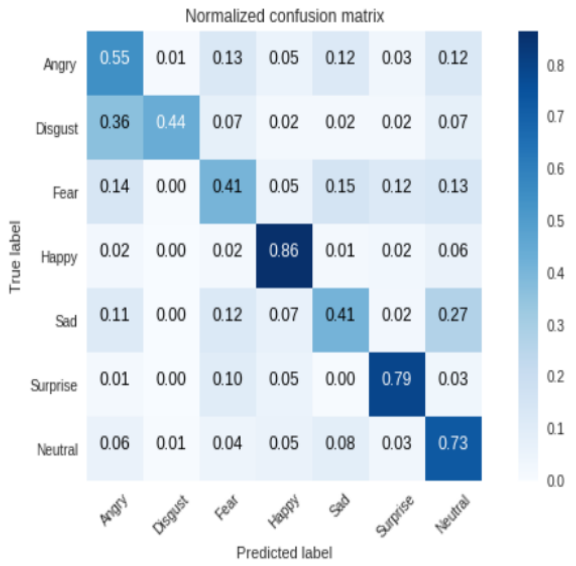
```
             precision    recall  f1-score   support

          0       0.29      0.14      0.19       491
          1       0.00      0.00      0.00        55
          2       0.35      0.13      0.19       528
          3       0.41      0.75      0.53       879
          4       0.27      0.26      0.27       594
          5       0.61      0.40      0.48       416
          6       0.35      0.41      0.38       626

avg / total       0.37      0.38      0.35      3589
```

Fig. 6: SVM on PCA classification metrics

metrics. The model that gave the best performance was the 12 layered network and the hyper parameters are as mentioned in table: II. The confusion matrix of the best model is as shown in fig 7 and fig 8 shows us the ROC curves corresponding to all the prediction accuracies of each class of the best model.

| Hyper parameter | Value |
|---|---|
| Batch Size | 300 |
| kernel size | 3 |
| optimizer | AdaDelta |
| Dropout | 0.5 |

TABLE II: Best model hyper parameters



| | Angry | Disgust | Fear | Happy | Sad | Surprise | Neutral |
|---|---|---|---|---|---|---|---|
| Angry | 0.55 | 0.01 | 0.13 | 0.05 | 0.12 | 0.03 | 0.12 |
| Disgust | 0.36 | 0.44 | 0.07 | 0.02 | 0.02 | 0.02 | 0.07 |
| Fear | 0.14 | 0.00 | 0.41 | 0.05 | 0.15 | 0.12 | 0.13 |
| Happy | 0.02 | 0.00 | 0.02 | 0.86 | 0.01 | 0.02 | 0.06 |
| Sad | 0.11 | 0.00 | 0.12 | 0.07 | 0.41 | 0.02 | 0.27 |
| Surprise | 0.01 | 0.00 | 0.10 | 0.05 | 0.00 | 0.79 | 0.03 |
| Neutral | 0.06 | 0.01 | 0.04 | 0.05 | 0.08 | 0.03 | 0.73 |

Fig. 7: confusion matrix of the best configuration of the 12 layered network

From the confusion matrix, we can see that most of the data points of the classes ”happy” and ”surprise” are correctly classified.We can see the same being reflected in the ROC curve. We can also see that, almost half of the ”disgust” class data points are classified correctly using CNN’s which is not the case in SVM. Quite a number of ”disgust” images are classified as ”angry” due to high similarities in their features. Also, ”sad” images are classified ”neutral” because of similar features. More intuition behind this, can be seen in Visualization part.

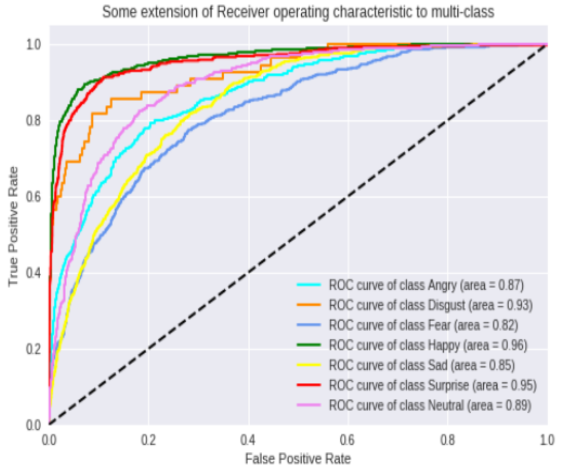


Fig. 8: ROC curves of the best configuration of the 12-layered network

### C. *Softmax vs Multiclass SVM activation:*

Most deep learning methods for classification using fully connected layers and convolutional layers use softmax activation function for prediction and minimize cross-entropy loss, with few exceptions. Here, softmax layer was replaced with a linear support vector machine and implemented the Convolutional Neural Networks described with linear one-vs-all SVM at the top. Learning minimizes a margin-based loss instead of the cross-entropy loss. Tang et.al[9], has shown that for some deep architectures,a linear SVM top layer instead of a softmax is beneficial, and our results are also inline with this observation. This addition added to roughly 1% increment in test accuracy over all the CNN models. Table:III shows the increase in the accuracy for different networks

| Model | Softmax | Multiclass SVM |
|---|---|---|
| 2-layer-net | 0.562 | 0.569 |
| 5-layer-net | 0.611 | 0.618 |
| 12-layer-net | 0.639 | 0.645 |

TABLE III: Softmax vs Mulitclass SVM activation

The Table IV shows the precision, recall, f1-scores and accuracy of the best case model in each configuration.The 12-layered model topped by SVM has performed best and SVM with PCA has given us the least scores.

## VI. Analysis

### A. *Training Process:*

When training models that can over-fit, we can see that the training loss decreases continuously over time, while the validation loss has a U-shaped curve. Increase in validation

| Model | Avg Precision | Avg. Recall | avg. F1 scores | Accuracy (%) |
|---|---|---|---|---|
| SVM (PCA) | 0.37 | 0.38 | 0.35 | 39.98 |
| SVM (HOG and Face landmarks) | 1.0 | 0.48 | 0.65 | 48.2 |
| 2-layered net | 0.55 | 0.56 | 0.55 | 56.2 |
| 5-layered net | 0.62 | 0.62 | 0.62 | 61.1 |
| 12-layered net | 0.63 | 0.64 | 0.63 | 63.9 |
| 12-layered net + Mul. SVM loss | 0.64 | 0.64 | 0.64 | 64.5 |

TABLE IV: Classification metrics for best configuration of all the models

loss implies that the model is over-fitting. So, the training process was stopped when the validation loss was the least to get a model that fit better to the data distribution. The fig9 and fig10 shows the train/validation accuracy and loss curves correspondingly

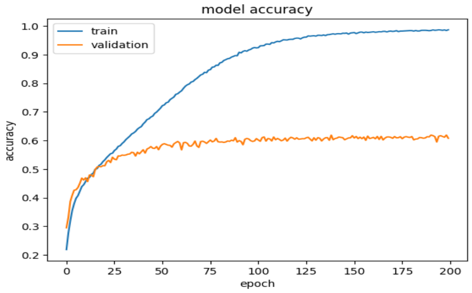


Fig. 9: 5-Layer ConvNet: Accuracy vs epochs

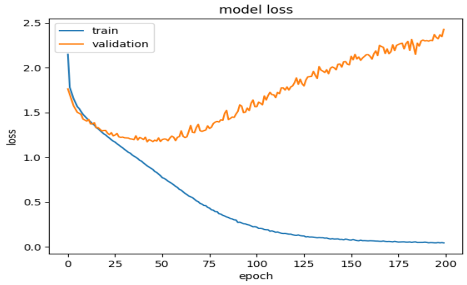
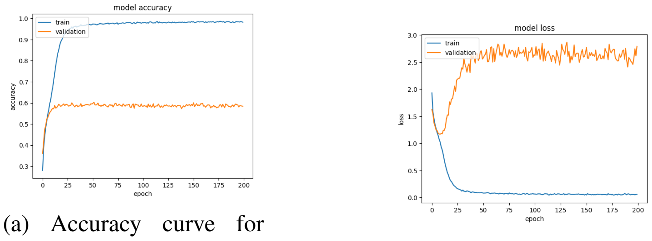


Fig. 10: 5-Layer ConvNet Training Curves: Loss vs epochs

*B. Ablation study:*

An ablation study was performed for the 5 layered network over different hyper parameters like optimization functions, dropout rates, different number of convolution layers, different batch sizes etc.

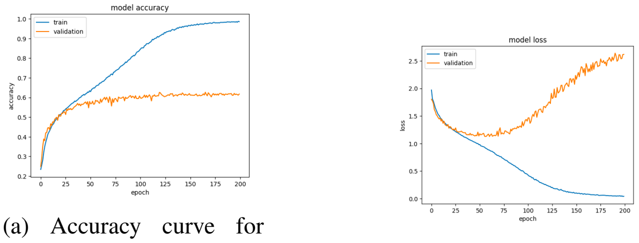
*1) Optimizers::* Optimization functions were varied and the accuracy and loss vs time plots are shown in the Fig 11. The traditional Batch Gradient Descent will calculate the gradient of the whole Data set but will perform only one update. Stochastic Gradient Descent(SGD) updates the parameters for each training example. It performs one update at a time which cause fluctuations in loss due to high variance in parameter updates. These frequent updates help to discover possible better local minima. Due to the high variant oscillations SGD takes longer durations to converge. Using momentum accelerates SGD by navigating along the relevant direction and softens the oscillations in irrelevant directions. SGD has given us the best performance for a 5-Layer ConvNet with an accuracy of 0.63 when momentum was set to 0.9. Adam converges very fast and the model learns faster. fig 12 and fig 13 shows the training curves for different optimization functions, for a 5-Layer ConvNet model. The model using Adam optimizer has converged faster than other models and the validation loss is minimum around 25 epochs. Other optimizers such as RMS-PROP and Adadelta were also experimented with. Adadelta, a variant of AdaGrad, resulted in an accuracy of 0.613. The learning rate plays a very important role in optimization. We have choosen optimal learning rates in all these cases. table: V shows optimizer used vs. changes in accuracy.



(a) Accuracy curve for Adam

(b) Loss curve for Adam

Fig. 11: 5-Layer ConvNet Training Curves



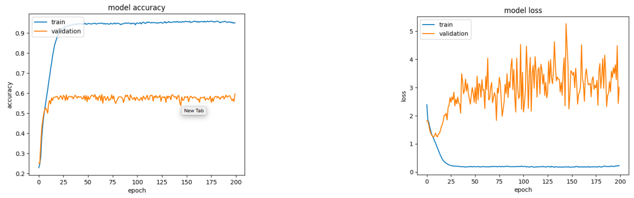
(a) Accuracy curve for SGD

(b) Loss curve for SGD

Fig. 12: 5-Layer ConvNet Training Curves

| Optimizer | Accuracy |
|---|---|
| AdaDelta | 0.611 |
| SGD | 0.630 |
| Adam | 0.588 |
| RMS prop | 0.584 |

TABLE V: Optimizer vs Accuracy



(a) Accuracy curve for RMS-PROP

(b) Loss curve for RMS-PROP

Fig. 13: 5-Layer ConvNet Training Curves

*2) Batch Size::* table: VI shows batch size vs accuracy. In can be seen that a batch size of 300 yielded in the best accuracy. Further increasing the batch resulted in a reduction of the accuracy.

| Batch size | Accuracy |
|---|---|
| 32 | 0.607 |
| 100 | 0.6032 |
| 300 | 0.6132 |
| 500 | 0.5764 |

TABLE VI: Batch size vs Accuracy

*3) Dropout:* The term dropout refers to dropping out units (both hidden and visible) in a neural network. Dropout helps reducing the interdependent learning amongst the neurons. Dropout is regularization method in neural networks. 5-layer ConvNets was trained with and without dropouts. The network with dropout performed better than that without dropout. The network without dropouts gave us an accuracy of 0.55. Further multiple dropout values were used and accuracies were recorded as given in the table: VII. A trend was noticed that as the dropout increases the accuracy inceases upto 0.5 and then starts decreasing. This is due to bias variance tradeoff principal, since low dropout implies complicated model and could result in higher bias and high drop out could result in a model with high variance.

| Dropout | Accuracy |
|---|---|
| 0.1 | 0.566 |
| 0.25 | 0.583 |
| 0.4 | 0.601 |
| 0.5 | 0.63 |

TABLE VII: Dropout vs Accuracy

*4) CNN - Number of Filters:* table: VIII shows number of kernels vs accuracy. It was seen that a convolution neural net with increasing number of kernels perform better for a classification problem. This is due to the fact that as the layer number increases in the network the kernel receptive field should be maximum on the input image so that it would take global features in the image into consideration.

| layer1 | layer2 | layer3 | layer4 | layer 5 | Accuracy |
|---|---|---|---|---|---|
| 64 | 64 | 64 | 64 | 64 | 0.586 |
| 128 | 128 | 128 | 64 | 64 | 0.612 |
| 64 | 64 | 64 | 128 | 128 | 0.608 |

TABLE VIII: Number of kernels per layer vs accuracy

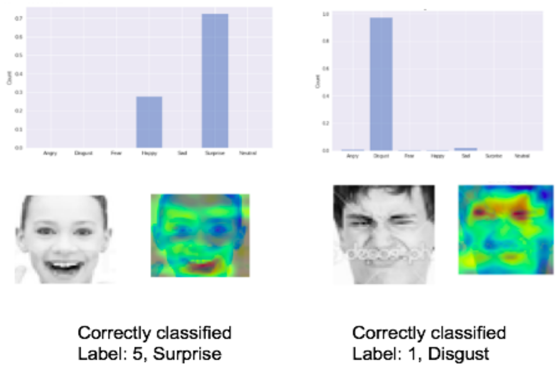
*5) Number of Convolutional Layers::* It was noticed that as number of convolutional layers increases, the accuracy of the model increases. But the improvement from 5 Layer Net to 12 Layer Net was less compared to that of 2 Layer Net to 5 Layer Net. With increase in number of layers greater than 12, the accuracy might increase very less, till a certain point.

*6) Number of Fully Connected Layers::* The convolutional layer is flatten and is followed by a fully connected layer.A fully connected layer, looks at what high level features most strongly correlate to a particular class and has particular weights so that when the products between the weights and the previous layer are computed, the correct probabilities for the different classes are obtained. We have experimented with several fully connected layers followed by flattening and found that two fully connected layers worked best.

## VII. Visualization Analysis

Several approaches have been developed for visualizing and understanding the CNN's. We applied visualization techniques to gain more insight of how our convolution layers work and the specific features of the facial images that attributed to classification.

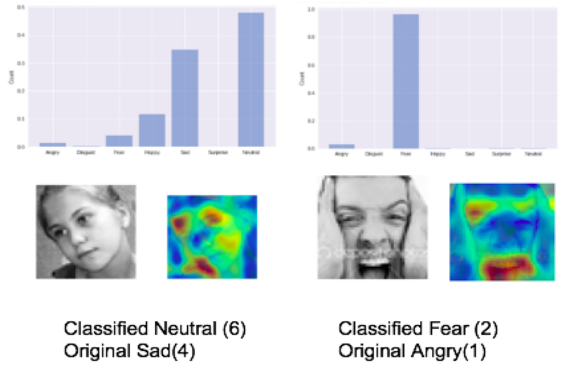
The most straight-forward technique is to visualize the convolution layer activations. This allow us to understand, the input patterns that activate a particular filter. We have shown these activations for one particular image on our best model in the Fig. 16. So from the Fig. 16, we can see that, initially the activations are dense, and with each layer the activations become more sparse and localized.The low level filters work as edge detectors and as we go deeper they tend to capture high level features.



Correctly classified
Label: 5, Surprise

Correctly classified
Label: 1, Disgust

Fig. 14: Correctly classified Images



Classified Neutral (6)
Original Sad(4)

Classified Fear (2)
Original Angry(1)

Fig. 15: Wrongly classified Images

The other visualization technique we have implemented is GRAD-CAM++, an improved gradient based class activation map. In the figures below, we have shown the test images, their corresponding softmax activations as a bar graph and their associated GRAD-CAM++ image for the 5-layer convolution net model. In the Fig. 14 , the smiling girl has highest softmax probability for Surprise and indeed the original label is Surprise. Here, the major features that contributed to the classification is mouth. The second image in Fig. 14 is correctly classified as disgust, the major features are the eyebrows,eyes and mouth. The images in Fig. 15 are wrongly classified and from their softmax values, we can see that the second highest softmax value is the original-correct label.

## VIII. CONCLUSIONS

In this project, a simple straight-forward computer vision classification problem, facial emotion classification was addressed. The dataset used is diverse and has images with varying illumination conditions, faces with glasses and beards. The best accuracy that we have obtained is 64.5% using a 12 layered model with SVM on top. Of all the 7 emotions, happy and surprised were classified correctly consistently since majority of the data were these two emotions. It was noticed that SVM performs better on global features rather than directly using dimensionality reduced images. It was also noticed that SVMs couldn't classify classes with low number of samples in the training data where as the CNN models were able to account for them. Further for CNNs it was noticed that models with higher complexity such as higher depth and higher number of filters were able to classify better till a certain point. Similar trend applied for dropouts also. It was also noticed that the rate at which accuracy increases with the number of layers follow an exponentially decreasing trend.

## IX. WORK DIVISION:

Each team member contributed substantially to the project.
SVM with PCA, SVM with HOG and facial Landmarks: Nitin
2-Layer Net:Shruthi
5-Layer Net:Manasa
12-Layer Net:Nitin
Hyper-parameter Tuning: Shruthi, Manasa
SVM on CNN instead of softmax: Nitin, Shruthi
ROC curves/confusion matrices: Manasa,Nitin
Visualizations, Grad-cam, Grad-cam++: Shruthi, Manasa

## X. FUTURE WORK:

- Exploring bigger nets such as GoogLeNet, ResNet.
- Using models with pre-trained weights on a different data-set and train the model on FER2013 data-set.
- Use the model trained on FER2013 data-set to perform emotion classification task on samples from other data-set to explore transfer learning.
- Models with residual connections are generally known to perform well in computer vision and computer graphics problems. So, addition of residual connections to the models we use and see if any improvements can be noticed.
- Use the best model to build a real time face recognition and emotion classification application.

## ACKNOWLEDGMENT

We thank Prof.Jack Mortazavi of CSCE633 for equipping us with a practical understanding of various Machine Learning algorithms.

## REFERENCES

[1] Zhang, Zhengyou. "Feature-based facial expression recognition: Sensitivity analysis and experiments with a multilayer perceptron." International journal of pattern recognition and Artificial Intelligence 13, no. 06 (1999): 893-911.

[2] Arriaga, Octavio, Matias Valdenegro-Toro, and Paul Plger. "Real-time Convolutional Neural Networks for Emotion and Gender Classification." arXiv preprint arXiv:1710.07557 (2017).

[3] Goodfellow, Ian J., Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski et al. "Challenges in representation learning: A report on three machine learning contests." In International Conference on Neural Information Processing, pp. 117-124. Springer, Berlin, Heidelberg, 2013.

[4] Mollahosseini, Ali, David Chan, and Mohammad H. Mahoor. "Going deeper in facial expression recognition using deep neural networks." In Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on, pp. 1-10. IEEE, 2016.

[5] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[6] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

[7] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." In Advances in neural information processing systems, pp. 1097-1105. 2012.

[8] https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge

[9] Tang, Yichuan. "Deep learning using linear support vector machines." arXiv preprint arXiv:1306.0239 (2013).

[10] Bristow, Hilton, and Simon Lucey. "Why do linear SVMs trained on HOG features perform so well?." arXiv preprint arXiv:1406.2419 (2014).

[11] Day, Matthew. "Exploiting facial landmarks for emotion recognition in the wild." arXiv preprint arXiv:1603.09129 (2016).

[12] P. Ekman and E. L. Rosenberg. What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System(FACS). Oxford University Press, USA, 1997.

[13] T. F. Cootes, C. J. Taylor, et al. Statistical models of appearance for computer vision, 2004.

[14] Sang, Dinh Viet, and Nguyen Van Dat. "Facial expression recognition using deep convolutional neural networks." Knowledge and Systems Engineering (KSE), 2017 9th International Conference on. IEEE, 2017.

[15] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 618-626.

[16] https://adeshpande3.github.io/A-Beginner's-Guide-To-Understanding-Convolutional-Neural-Networks/

[17] http://image-net.org/challenges/LSVRC/2016/index

[18] https://github.com/davisking/dlib

[19] https://github.com/amineHorseman/facial-expression-recognition-using-cnn

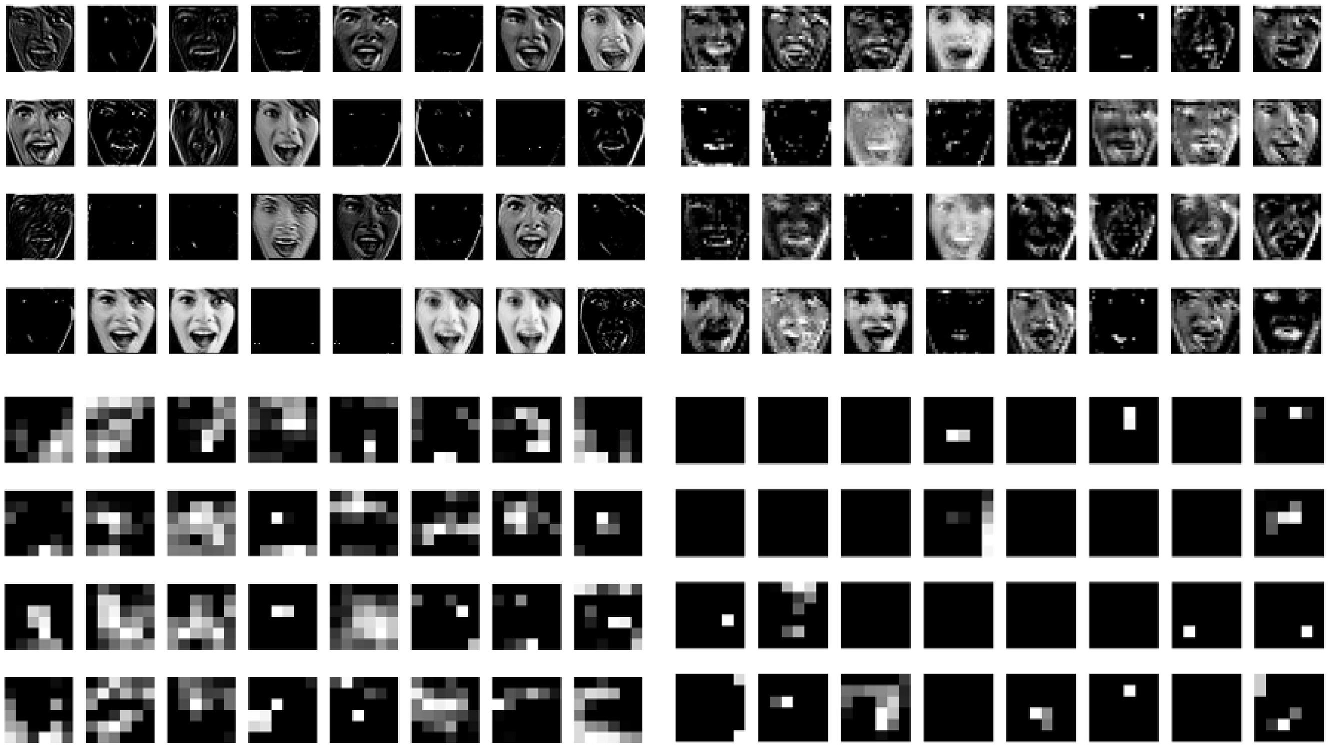


Fig. 16: Layer activations visualization, top left: Layer 1, top right: Layer 3, bottom left: Layer 8, bottom right: Layer 9