ECE 176 Final Project

Sialoi Ta'a staa@ucsd.edu
Allison Moya amoya@ucsd.edu
Department of Electrical and Computer Engineering, University of California, San Diego
San Diego, CA 92093 USA

Abstract

The proposed project focuses on Facial Expression Recognition with Mood Prediction, aiming to utilize computer vision and deep learning techniques for emotion analysis in facial expressions. The project aims to develop a system that is capable of accurately recognizing facial expressions and predicting the corresponding mood states. This aims not only to attain facial expression recognition but also to explore the topic of emotional comprehension. It can apply to applications involving human-computer interaction, virtual reality, mental health monitoring and more. We look to solve this problem with the MobileNet and Resnet architecture and see which one can accurately capture the visual patterns hidden inside our dataset.

1. Introduction

The problem we are looking to solve is if facial recognition can effectively perform mood analysis. The motivation behind this project is finding out if there are some underlying representations that can be detected in a facial expression by a neural network that we can't observe just with the naked eye and our own intuition. It's often been said that you can find anyone smiling, but you'll never truly know what underneath the smile. Whether it be anger, sadness, anxiety or even nausea, many can hide it through a smile and fool anyone. This is also linked to mental health issues as it has been common for those who have depression to hide it with a smile and go unnoticed or unaddressed.

2. Method

We will be using MobileNet V1's architecture and ResNet's architecture as listed in the 2017 articles respectively, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications" [1] and "Aggregated Residual Transformations for Deep Neural Networks" [2] without skip connections. The decision to use two different networks is because we wanted to see which one would carefully find the pattern. In our Experiment section, we will display the results of each model after hyperparameter tuning and show what the loss curves are like.

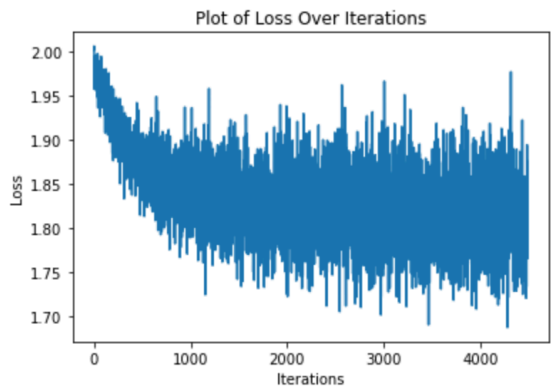
3. Dataset

We will be using the FER-2013 [3] (Facial Expression Recognition 2013). This is a well known and documented data set that has labeled over 35k images with a corresponding mood: Anger, Disgust, Fear, Happy, Neutral, Sad and Disgust. It consists of only image files and directories named after the corresponding mood separated by a test and train folder. The dataset consists of grayscale images, each of size 48x48 pixels.

4. Experiment

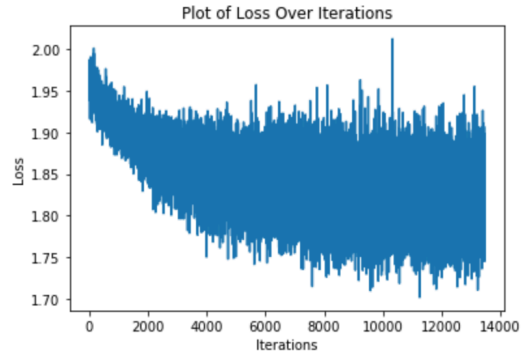
The first architecture we used was MobileNet's architecture. In the MobileNet_Experiment Jupyter notebook,[4] we document our experiment with the MobileNet architecture on the FER-2013 dataset. The process for running this experiment was the same as the first experiment however we found that stochastic gradient descent (SGD) was a much better optimizer than

Adam was in this case. Here are the graphs that were created. First is the graph for 10 epochs:




```
Checking accuracy on test set
Got 1777 / 7178 correct (24.76)
```

Second is the graph for 30 epochs:




```
Checking accuracy on test set
Got 1774 / 7178 correct (24.71)
```

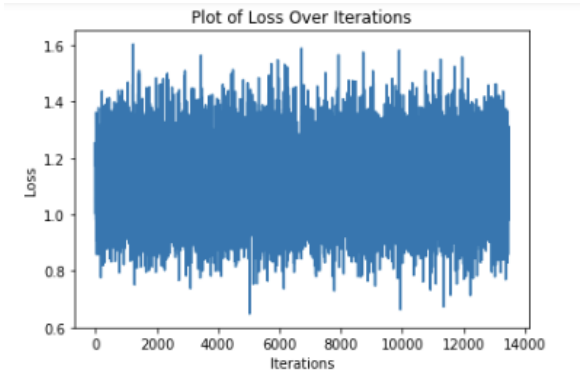
The first graph was run with a learning rate of 0.00001 and a momentum of 0.9. The second graph's learning rate was 0.000001 and the momentum was 0.95. The reason why the hyperparameters are different between graphs is because the learning rate reduction and momentum increase was an attempt in trying to decrease the oscillatory behavior of the first graph. What is similar in both of these graphs is that they look like they converge to around 24.7% accuracy and plateau there. It could be possible that the characteristic of MobileNets, which would be the depth convolutions, wasn't a good solution to the problem presented at the moment. While hyperparameter tuning didn't provide us with great results in this experiment, we can confirm that the MobileNet's architecture is not suited for this problem and should move onto another network.

We then moved onto the Resnet architecture for our next experiment. In the ResNet_Experiment jupyter notebook,[4] an Adam optimizer with a learning rate of 0.01 was employed for training. The model architecture, named CustomNet, comprises convolutional layers followed by normalization, activation functions, and dropout layers to mitigate overfitting. Designed for facial recognition tasks, the network outputs scores for different facial expressions. During the training process, the model undergoes multiple epochs, first with ten and second with thirty, using training helper functions we define, handling the training loop. Dropout layers are incorporated to prevent overfitting by randomly deactivating a portion of input units during training. This helps with focusing on certain patterns during the training duration and helping the model learn them one by one instead of all at once. Learning all of them all at once could cause overfitting and make the model have poor generalization when using images outside of the training set. To present the experiment results, accuracy and loss curves are employed. Visualization of the metrics shows the model's performance over time.

Accuracy Results after 10 epochs:


```
Checking accuracy on test set
Got 4048 / 7178 correct (56.39)
```

Accuracy Results after 30 epochs:




```
Checking accuracy on test set
Got 4040 / 7178 correct (56.28)
```

The accuracy results on the test set indicate that the model achieved 56.39% accuracy after 10 epochs and 56.28% accuracy after 30 epochs of training. While the accuracy slightly decreased after 30 epochs, it remained relatively stable, indicating that the model's performance remained relatively consistent over a longer training period. This consistent range of values from oscillating over time means that the loss plateaued and couldn't reach a higher accuracy, reaching maximum performance from this model. This model reaching peak performance within 10 epochs shows how great the model captured the class discernable patterns within the dataset in a short amount of time. While this version network of ResNet didn't have skip connections, the accuracy of this model foreshadows how ResNet would perform on this dataset and how skip connections could only increase the accuracy from there.

4. Limitations

Firstly the architecture of the neural network may not be optimal for capturing the intricate features present in facial expressions. Despite incorporating convolutional layers, batch normalization, group normalization, and dropout, the model's capacity to learn complex spatial patterns might be limited. More advanced architectures such as recurrent neural networks could potentially yield better performance by focusing on salient facial regions.

Furthermore, the training process and hyperparameter optimization are essential factors influencing the model's convergence and performance. Although the Adam optimizer was utilized with a fixed learning rate, exploring alternative optimization algorithms and adaptive learning rate strategies could potentially accelerate convergence and improve generalization. Additionally, hyperparameter tuning, including batch size, dropout rates, and regularization techniques, could further optimize the model's performance.

A good way to increase the model's accuracy is by either adding skip connections, skip connection block groups, or creating a more complex model to find more possible complex patterns that could exist in the dataset. By incorporating skip connections or skip connection block groups, the model can facilitate the flow of gradients and alleviate the vanishing gradient problem, enabling the network to learn more patterns more effectively. Creating a more complex model with additional layers or increasing the depth of existing layers could enhance the model's capacity to capture intricate features in facial expressions but this is a fine line as adding more layers would increase the likelihood of overfitting the model to our training dataset.

5. Conclusion

In this study, we addressed the problem of Facial Expression Recognition with Mood Prediction using computer vision and deep learning techniques. By accurately recognizing facial expressions and predicting corresponding mood states, our aim was to contribute to emotional comprehension, which holds implications for various applications, including human-computer interaction, virtual reality, and mental health monitoring.

To approach this problem, we employed the architectures of MobileNet v1 and ResNet, using their strengths in capturing intricate features from facial images. Exploring the FER-2013 dataset, which consists of labeled images across seven mood categories, we trained our models using hyperparameter tuning and optimization techniques.

Our experiments revealed interesting insights into the performance of the models. While the accuracy achieved by both models remained relatively stable, indicating consistent performance over time, it also plateaued at a certain threshold. This suggests that the models reached their maximum potential with the given architecture and dataset, signaling a limitation in capturing more nuanced facial expressions.

Although our current approach did not fully capture the solution, experiment results indicate that our next step

should involve exploring ResNet and ResNeXt structures with skip connections. By incorporating skip connections, these models can facilitate the flow of gradients and alleviate the vanishing problem, potentially enhancing the network's ability to learn intricate features present in facial expressions.

While our approach shows promise in addressing facial expression recognition with mood prediction, there are still opportunities for refinement and enhancement. By iteratively improving the model architecture and optimization strategies, and exploring more advanced architectures with skip connections, we aim to contribute towards a more robust and accurate system for emotional analysis in facial expressions.

5. References

[1] Howard, A. G., Zhu, M., Chen, B., Wang, W., Weyand, T., Andreetto, M., Adam, H., & Kalenichenko, D. (2017, April 17). Arxiv. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. https://arxiv.org/pdf/1704.04861.pdf

[2] Xie, S., Girshick, R., Dollar, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2017.634

[3] FER-2013 Dataset Link: https://www.kaggle.com/datasets/msambare/fer2013?resource=download

[4] GitHub Repo

[5] Professor Wang, Xiaolong