Physical Layer Data Augmentation Techniques for Face Recognition and Face Emotion Classification Data

Ashish Vinodkumar
MIDS
Duke University
ashish.vinodkumar@duke.edu

Pranav Manjunath
MIDS
Duke University
pranav.manjunath@duke.edu

Abstract

In this paper, we explore various physical layer augmentation techniques such as top-half mask, bottom-half mask, one-fourth mask, along with an image data augmentation technique, sharpness kernel, on the Face Recognition and Emotion Classification dataset using a VGG-16 model architecture. We learn that physical layer augmentation and data augmentation, in general, boosts model performance and reduces overfitting compared to the baseline. Specifically, masking one-fourth of the pixels in the physical layer, results in an accuracy of 69.76% similar to that of the baseline (no-mask) model of 72.31%. This shows us that 1/4th of the pixels can be discarded in a Face Emotion Classification experiment while still maintaining near baseline accuracy and reducing the model training time by 30% per epoch.

1 Introduction

Face emotion classification consists of a two step process, face detection and face emotion classification. With an input image, the model should first detect if a face is present by constructing a bounding box on the face. Once the face is detected, the face then goes through an emotion classification. In the area of deep learning, data pre-processing is an important stage that ensures readiness of data for model training. Particularly in face detection and emotion classification applications, one of the important pre-processing techniques that is commonly applied is called data augmentation. Data augmentation is particularly helpful when the amount of images available is scarce and hard to be collected. In this particular project, we are interested in observing the effects of physical layer augmentation. Specifically, assess accuracy metrics of models trained on images that are masked at the "camera lens" level, either in the top-half, bottom-half, or three-fourths lens pixel level. Performing such physical layer data augmentation will directly limit the numbers of pixels incident on the camera lens. This paper looks to identify if such physical layer augmentation combined with general "post-image capture" data augmentation techniques such as sharpness kernel, can yield comparable or better results compared to the baseline non-augmentation model.

2 Related Work

For this project, the team aims to follow an earlier work on assessing the effects of physical layer augmentation techniques, applied on a face detection and emotion classification dataset, by testing model performance metrics such as accuracy, precision, recall, and f1 score[1]. Specifically, the team focuses on three physical layer augmentation techniques: top-half mask, bottom-half mask, one-third mask, along with one data augmentation technique, sharpness kernel, to assess model performance metrics. The team shows that some augmentation techniques are able to extract face detection and emotion classification image statistics more effectively than others, which will lead to better predictive performance in the corresponding models.

We learnt from [6], that enhancing facial features can help in face emotional classification. Hence we implemented the sharpness kernel as a data augmentation technique to improve model performance

3 Method

3.1 Data Pre-Processing

We obtained our dataset from Kaggle. We had 2 separate datasets, one for Face Detection[2] and the other for Face Emotion Classification[3]. For the Face Detection dataset pre-processing, we parsed through the individual images in their respective folders, and created a root level folder with all images inside of it. For the Face Emotion Classification dataset pre-processing, we parsed through the folders containing images for the 3 emotions of interest: happy, neutral, and sad. We then created a table schema consisting of the filepath of the image and its corresponding emotion label. The dataset we used has a variety of people in terms of gender, race, and color, making sure that the model can be generalized to a larger population.

3.2 Model Architecture and Hyperparameters

For the Face Detection model, we performed transfer learning, by using an existing pre-trained model: CV2 Cascade Classifier[4]. Using the CV2 Cascade classifier allowed us to detect the face coordinates allowing us to create a 64x64 image crop around the face.

For the Face Emotion Classification model, we followed the study by Porcu et al. (2020), and used VGG-16 as our model architecture. This VGG-16 model architecture allowed us to create the baseline model and all 3 physical layer augmentations along with 1 data augmentation experiment as outlined in the paper. For all the 4 models, we set the learning rate to 1e-3. We further experimented with a dropout parameter of 0.05, and a L2 regularization of 1e-4. To account for model overfitting, we added dropout layers and L2 regularization to the model and tried a variety of options. However, we noticed that the model without dropout and L2 regularization performs better on the validation dataset and hence we decided to not include dropout and regularization in our model. Please see the Github repo for more in depth analysis.

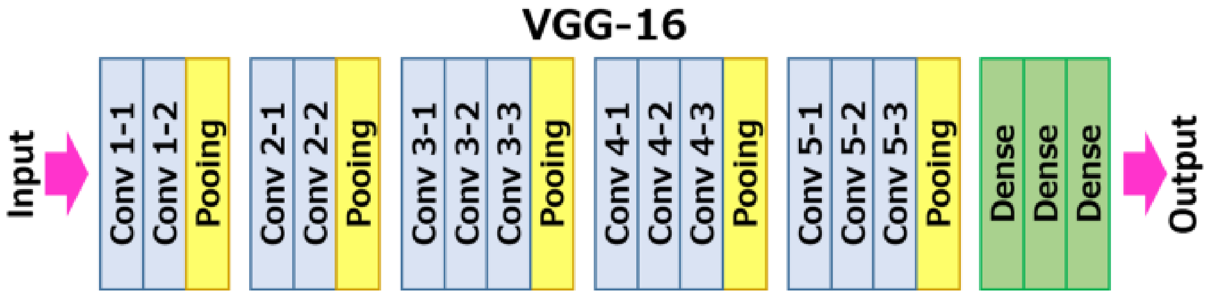


Figure 1: VGG-16 Model Architecture

3.3 Physical Layer Augmentation

To compare the effect of different physical and data augmentation techniques on the model's predictive performance, we used a baseline model that was trained on the entire 64x64 image, to then compare against the physical layer augmented models. We explored at least 3 distinct values for each physical layer augmentation and data augmentation (sharpness kernel), to ensure the best experimental result from each technique is used in the model comparison.

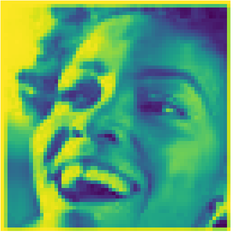


Figure 1: Original (64x64) Image

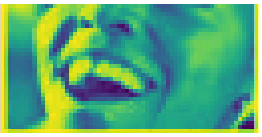


Figure 2: Top-Half Mask (32x64) Image

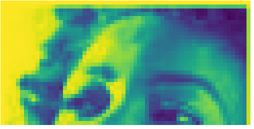


Figure 3: Bottom-Half Mask (32x64) Image

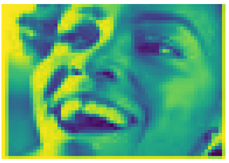


Figure 4: One-Fourth Mask (48x64) Image

3.3.1 Top-Half Mask

Top-Half Mask is a physical layer augmentation technique that involves reducing the number of pixels incident on the camera lens, i.e we block all pixels incident on the camera lens in the top-half of the image. Given that our original image is of size 64x64, removing the top-half pixels results in an image of 32x64. Figure.1 above represents the original non-masked 64x64 image, and Figure.2 represents the same image with a top-half mask applied to it.

3.3.2 Bottom-Half Mask

Bottom-Half Mask is a physical layer augmentation technique that involves reducing the number of pixels incident on the camera lens, i.e we block all pixels incident on the camera lens in the bottom-half of the image. Given that our original image is of size 64x64, removing the bottom-half pixels results in an image of 32x64. Figure.1 above represents the original 64x64 image, and Figure.3 represents the same image with a bottom-half mask applied to it.

3.3.3 One-Fourth Mask

One-Fourth Mask is a physical layer augmentation technique that involves reducing the number of pixels incident on the camera lens, i.e we block all pixels incident on the camera lens except for the middle three-fourth of the image. Given that our original image is of size 64x64, removing one-fourth pixels results in an image of 48x64. Figure.1 above represents the original 64x64 image, and Figure.4 represents the same image with a one-fourth mask applied to it.

3.4 Training and Evaluation

In terms of epoch number for model training, each model was trained on 25 epochs with each batch size equaling 64 images. We had a total of 8989 happy, 6198 neutral, and 6077 sad images. These images were split into a 80-20 training testing split. During each epoch, the random shuffle for the training loader function was set to True to ensure the augmented and original images were blended in each batch of images. After each training epoch, each model was tested with the batch size of 64 images. Once all 25 epochs of testing is finished, the highest testing accuracy is saved for comparison purposes. For the hyperparameter used in each data augmentation technique, at least 3 different values were explored, and the one with highest testing accuracy was chosen as the final hyperparameter for that specific augmentation technique. Below are the final hyperparameters chosen for each augmentation technique:

Learning Rate: 1e-3
Dropout Parameter: 0.05
L2 Regularization: 1e-4

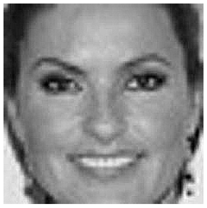
We further used a data augmentation technique, sharpness kernel, to enhance the image, as described by Vepuri, Ksheeraj Sai (2021)[6]. This allows for better edge identification and model prediction with emotion classification.

Sharpness Kernel:

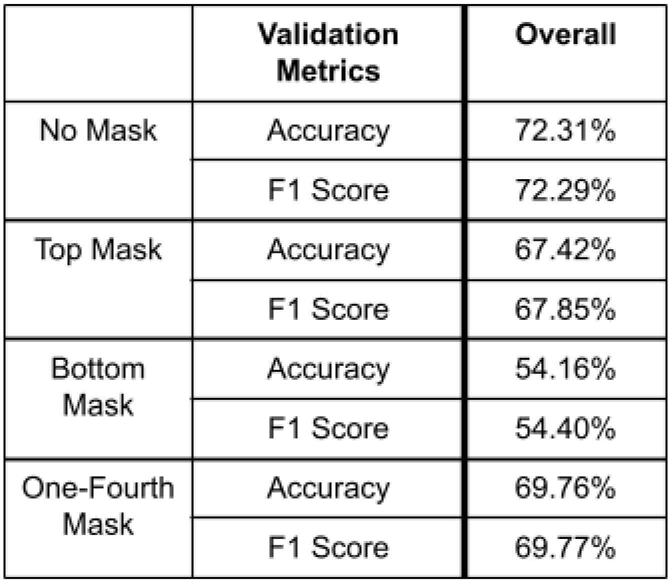
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Original Image

Sharpened Image



4 Experiment Results.



	Validation Metrics	Overall
No Mask	Accuracy	72.31%
	F1 Score	72.29%
Top Mask	Accuracy	67.42%
	F1 Score	67.85%
Bottom Mask	Accuracy	54.16%
	F1 Score	54.40%
One-Fourth Mask	Accuracy	69.76%
	F1 Score	69.77%

Table-1: Model Validation Performance on various masks

Table-1 describes the weighted average and F1 score for the baseline (no-mask) and 3 different masks. As expected, the baseline model (no-mask), has the highest performance metrics when predicted facial emotions, as the entire 64x64 image is available for model training and evaluation. Interestingly, the One-Fourth Mask model has a nearly close performance metric to the baseline (no-mask) model. This intuitively makes sense as the one-fourth mask ideally looks at the face from Eyebrow to Chin and most facial emotions can be represented in this region. We notice that images with a bottom mask (Hair to Nose) performs the worst when compared to all the other masks, indicating that it is difficult to identify facial emotions from the top half of the face. When comparing both halves of the face, through model performances we can conclude that the bottom half of the face (Top Mask) does a much better job when predicting facial emotions. However, it is important to note that these images are of 64 x 64 size and hence the resolution is not as clear.

Precision	Happy	Sad	Netural
No Mask	86%	61%	65%
Top Mask	85%	55%	58%
Bottom Mask	64%	54%	42%
One-Fourth	86%	56%	65%

Table-2: Precision for each Emotion

Recall	Happy	Sad	Netural
No Mask	84%	71%	56%
Top Mask	77%	60%	60%
Bottom Mask	61%	50%	48%
One-Fourth	81%	74%	50%

Table-3: Recall for each Emotion

The emotion Happy can be best classified with the highest precision of 86% and recall of 84% as shown in Table-2 and-3, with the baseline (no-mask) model. Compared to Happy, Sad and Neutral are more harder to predict - this can be concluded based on the varying levels of precision and recall amongst the various masks. It is observed that the Bottom-Mask performs the worst when predicting Neutral. We can infer from this that the mouth is important when predicting the neutral emotion. In terms of Precision, the baseline (no mask) model performs the best in predicting all three emotions. The three fourths mask does a very comparable job as well - however, the precision for the emotion is 5% less than that of baseline no-mask model (56% of images that the model predicts as Sad are actually sad). In terms of Recall, the baseline (no mask) model performs better for Happy, the one-fourth mask model performs better for sad and the top mask model performs better for neutral.

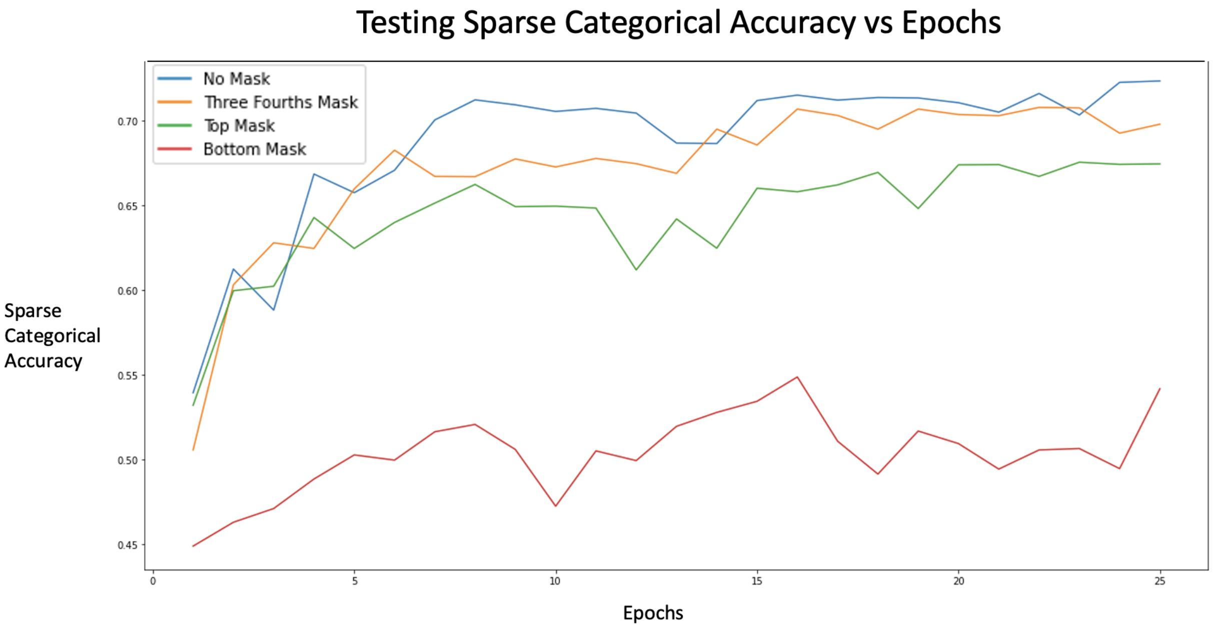
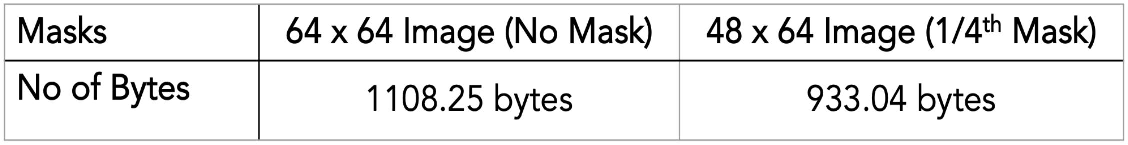


Figure 5: Testing Sparse Categorical Accuracy per Epoch

Figure 5 shows the sparse categorical accuracy for each of the four different masks by epoch. Each mask has been trained over 25 epochs. We can see that while the baseline (no mask) model performs the best, the one fourth mask performs very close to the baseline (no-mask) model. This indicates that we could just use 3/4th of the face to nearly identify the intended facial emotion of the person. It can be seen that the bottom mask model through the 25 epochs performs the worst with the highest sparse categorical accuracy of 54%. From this analysis, we understand that when compared to the top half of the face, the bottom half of the face is more important in classifying facial emotions into Happy, Sad and Neutral.

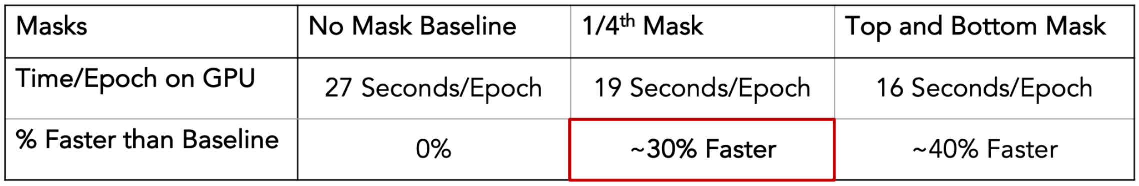
The importance of removing 1/4th of the face image is that we can make model training faster and save more in memory. Table 4 shows that on average black and white 48 x 64 images (1/4th mask), take up 16% less memory when compared to the average black and white full 64 x 64 images.



Masks	64 x 64 Image (No Mask)	48 x 64 Image (1/4th Mask)
No of Bytes	1108.25 bytes	933.04 bytes

Table 4: Average Size in Bytes of Entire Black and White Image and 1/4th Masked Image

Table 5 depicts the time the model takes to train with the various masked images (Model was trained on Google Colab Pro with GPU and High Ram Utilization). We can see that when ¼ of the pixels are removed, the training time is 30% faster than when the entire image is utilized, per epoch. Similarly we notice that the top and bottom mask perform 40% faster than the baseline (no-mask) image. However, their model performance in terms of classification metrics is not comparable to the baseline (no-mask) model.



Masks	No Mask Baseline	1/4th Mask	Top and Bottom Mask
Time/Epoch on GPU	27 Seconds/Epoch	19 Seconds/Epoch	16 Seconds/Epoch
% Faster than Baseline	0%	~30% Faster	~40% Faster

Table 5: Time per Epoch for each Mask

5 Conclusion / Discussion

Our project aimed to understand the importance of physical masks in predicting facial emotion (Happy, Sad, and Neutral) through images. We used a two step approach where the image goes through a facial detection model (openCV Cascade Classifier) and then to custom VGG-16 models to identify and predict facial emotions. We observed that the one fourth mask model performs very similarly to the baseline (no mask) model and that each mask model performs the best when predicting the emotion Happy. Removing 1/4th of the image results in 30% faster model training and 16% less memory. In other words, the model performance indicates that the forehead is not extremely important while the bottom half of the face is important in predicting these facial emotions.

We have also devised a front-end Flask App where users can input their images into the app and the models would predict the face emotion of the inputted image (See more in the Appendix). As part of our model classification result, we took a 2 pronged approach: best-mask-model classification and majority ensemble voting classification. With the best-mask-model classification, we used the one-fourth mask model as it had near baseline accuracy and was the clear winner compared to other masked models, as discussed above. With the majority ensemble voting classification, based on the majority classification from each of the masked models and the baseline model, the label/class with the highest vote was chosen as the final classification output.

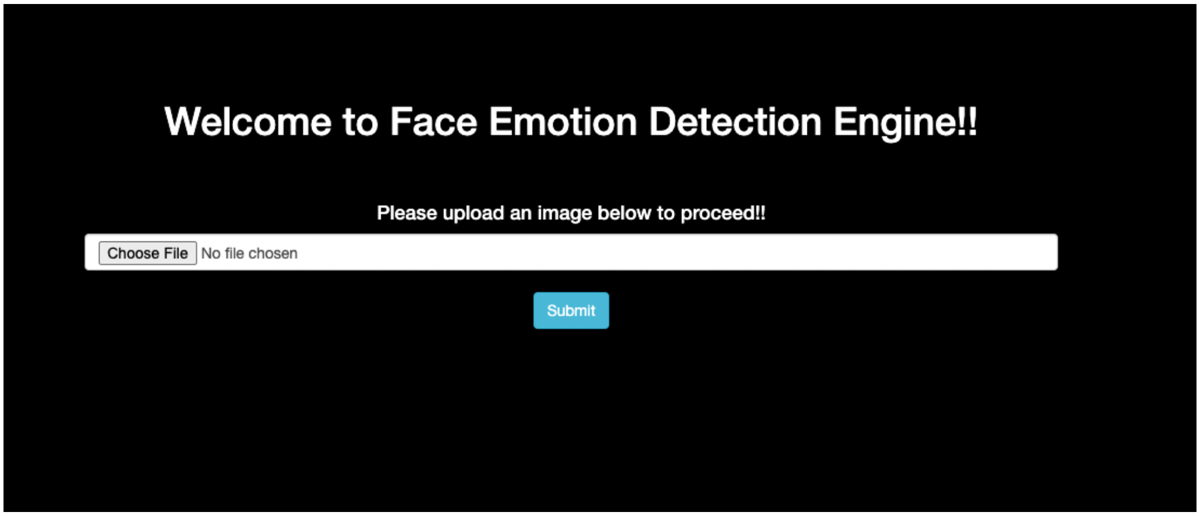
6 Reference

1. Pei, Zhao, et al. "(PDF) Face Recognition via Deep Learning Using Data Augmentation Based on Orthogonal Experiments." *ResearchGate*, https://www.researchgate.net/publication/336061375_Face_Recognition_via_Deep_Learning_Using_Data_Augmentation_Based_on_Orthogonal_Experiments.
2. Li, Jessica. "Labelled Faces in the Wild (LFW) Dataset." *Kaggle*, 17 May 2018, https://www.kaggle.com/jessicali9530/lfw-dataset.
3. Oheix, Jonathan. "Face Expression Recognition Dataset." *Kaggle*, 3 Jan. 2019, https://www.kaggle.com/jonathanoheix/face-expression-recognition-dataset/discussion/152419.
4. "Cascade Classifier." *OpenCV*, https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html.
5. Porcu, Simone, et al. *Evaluation of Data Augmentation Techniques for Facial ...* https://www.researchgate.net/publication/345940392_Evaluation_of_Data_Augmentation_Techniques_for_Facial_Expression_Recognition_Systems.
6. Vepuri, Ksheeraj Sai, "Improving Facial Emotion Recognition with Image processing and Deep Learning" (2021). Master's Projects. 1030.
7. Wu, Shengbin. "Expression Recognition Method Using Improved VGG16 Network Model in Robot Interaction." *Journal of Robotics*, Hindawi, 20 Dec. 2021, https://www.hindawi.com/journals/jr/2021/9326695/.

7 Appendix

7.1 Web Application

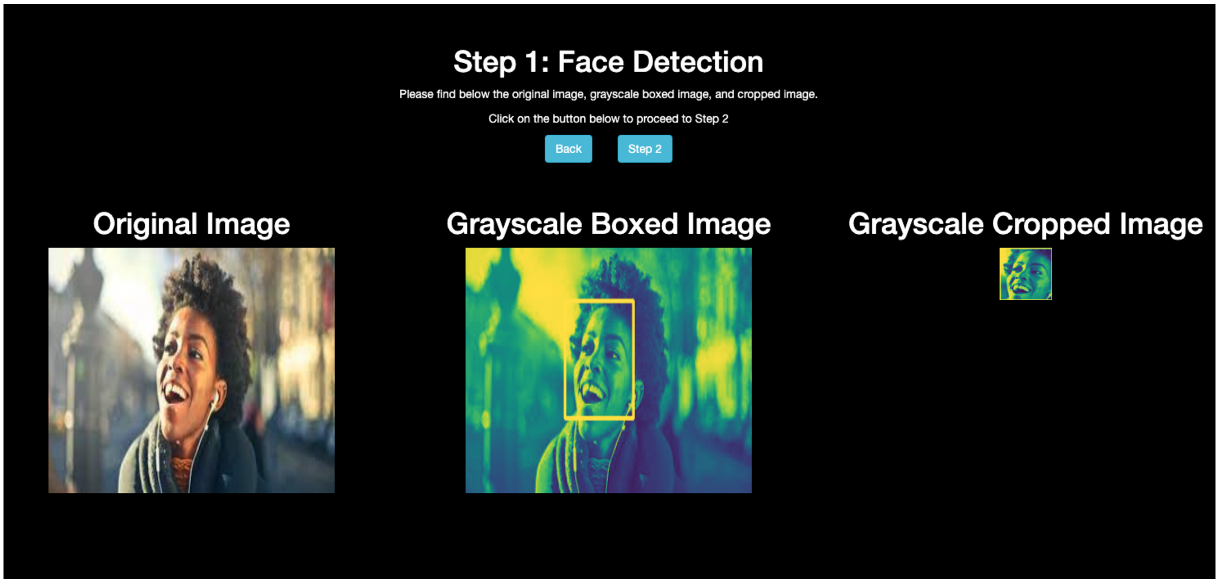
We utilized Flask, a micro web framework written in Python, to develop a web application for our models. Flask allowed us to effortlessly create a Front-End User Interface to present our model findings in an intuitive manner. We used a combination of HTML, CSS, JavaScript, and Flask to orchestrate the web application. Specifically, we first created a home screen that would allow users to upload a face image of their choice (Figure 6 below). We then created a "Step 1" UI page that showcased how the CV2 Face Detection model was being utilized to identify the face in the image, and crop the image pertaining only to the face in a 64x64 frame (Figure 7 below). Finally, we created a "Step 2" UI page that showcased our custom baseline VGG-16 model and the respective masked (top-half, bottom-half, and one-fourth masked) models, and its classification performance against the baseline (Figure 8 below).



Welcome to Face Emotion Detection Engine!!

Please upload an image below to proceed!!

Figure 6: Home Page UI: Users can upload a face image of their choice.

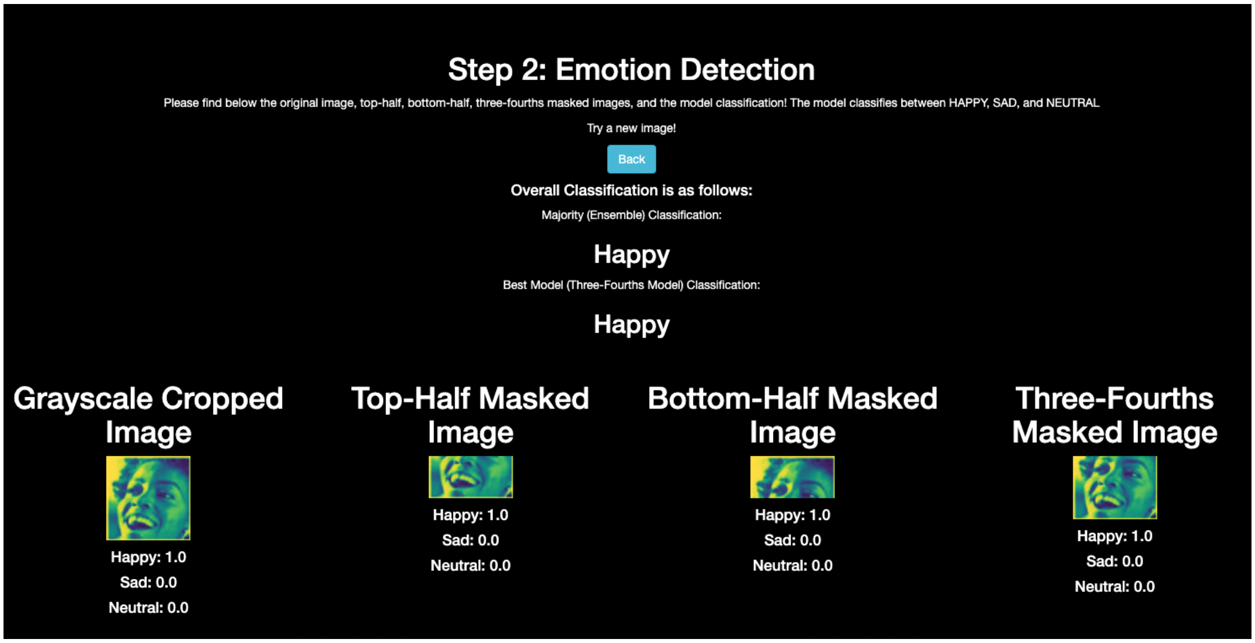


Step 1: Face Detection

Please find below the original image, grayscale boxed image, and cropped image.

Click on the button below to proceed to Step 2

Figure 7: Step 1 Page UI: Users can see CV2 Cascade Classifier for Face Detection.



Step 2: Emotion Detection

Please find below the original image, top-half, bottom-half, three-fourths masked images, and the model classification! The model classifies between HAPPY, SAD, and NEUTRAL

Try a new image!

Overall Classification is as follows:

Majority (Ensemble) Classification:

Happy

Best Model (Three-Fourths Model) Classification:

Happy

Grayscale Cropped Image

Happy: 1.0
Sad: 0.0
Neutral: 0.0

Top-Half Masked Image

Happy: 1.0
Sad: 0.0
Neutral: 0.0

Bottom-Half Masked Image

Happy: 1.0
Sad: 0.0
Neutral: 0.0

Three-Fourths Masked Image

Happy: 1.0
Sad: 0.0
Neutral: 0.0

Figure 8: Step 2 Page UI: Users can see Masked models and their classification predictions.