

Using Style Transfer and OpenCV color segmentation on CNNs to Accentuate and Identify Possible Facial Characteristics due to Illness.

- **Motivation**

Nobody likes to get sick. However, there could be specific facial cues that can be signals of illness. This paper will focus on analyzing facial features of healthy and ill people, accentuating the possible differences with two approaches, style transfer and color segmentation with the OpenCV package. This can be useful for the detection of possible future illnesses in patients, where specific facial patterns can be examined in order to make an accurate prediction of the disease that may be present.

- **Data and Cleaning Process**

I will be using the RUG dataset found at https://github.com/J1C4F8/deep_learning_acute_illness/tree/main/data/parsed. This dataset is divided into 'healthy' and 'sick' images of people. It contains close-ups of the faces where the differences of ill vs healthy are highly noticeable. The cleaning will mainly consist of positioning the adequate image size dimensions, so later it can be inputted for training in a Convolutional Neural Network (CNN) architecture. All images must be the same dimensions with batch, color channel, width, and height matching. The datasets are already nicely cropped and background friendly, facilitating the cleaning and division of train and validation sets.

- **Technical Approach**

Using style transfer and color image segmentation with the OpenCV library for image processing, I will highlight the drastic facial differences between healthy and sick. This will give a weighted value to specific parts of the face. Further, these images will be passed into a CNN with a weighted mask of facial features, which will be trained and tested on a larger batch of images. This is essentially a classification problem, where the labels are 'sick' and 'healthy', comparing how the resulting model classifies new batches of images based on the two approaches for accentuating the illness features. A more robust classification may pretend to create more labels to identify various types of illnesses, however, this is just a suggestion for a more involved classification problem. This will help with the detection of specific diseases. Note that this dataset only pertains to lipopolysaccharide (LPS) infused patients.

- *Style Transfer with VGG19*

After some literature review, the most optimal pretrained CNN for neural style transfer is VGG-19. The architecture of this network performs best at tasks like feature detection, facilitating the style transfer. Other convoluted networks such as ResNet or DesNet, appear not

to display as smooth results as VGG structures. This could be because due to a variety of reasons. VGG is so big that it is indirectly capturing a lot of information that the other models discard, which makes generalization better despite worse task-specific performance. Also, VGGs have modular and tight features, so there is no big separation, i.e., residual in between layers, and also due to not reducing samples so abruptly, doing so only after two convolutions and then max pooling.

In order to transfer the style image to emphasize the obvious illness cues of the face, I implemented an extra step into the style transfer algorithm that aims to target the unusual dark red spots in the face (since these are the main signs of illness). This can be achieved through image processing techniques such as thresholding, masking, and segmentation. The first step was to grayscale the images to reduce the coloring bias, then apply a threshold value to the grayscale content image to create a mask. This mask will have high weighted values in the darker regions and low weighted values in brighter areas to then replicate the single-channel mask to match the shape of the original RGB content image. Lastly, I create a masked version of the generated image by applying the mask to the generated image. This operation allows the style transfer to affect only the darker regions of the content image, leaving brighter areas unchanged. Further, I reverted the image from grayscale to RGB to have a better look at the colored transfer. The style image I decided to use was a simple dark red image.

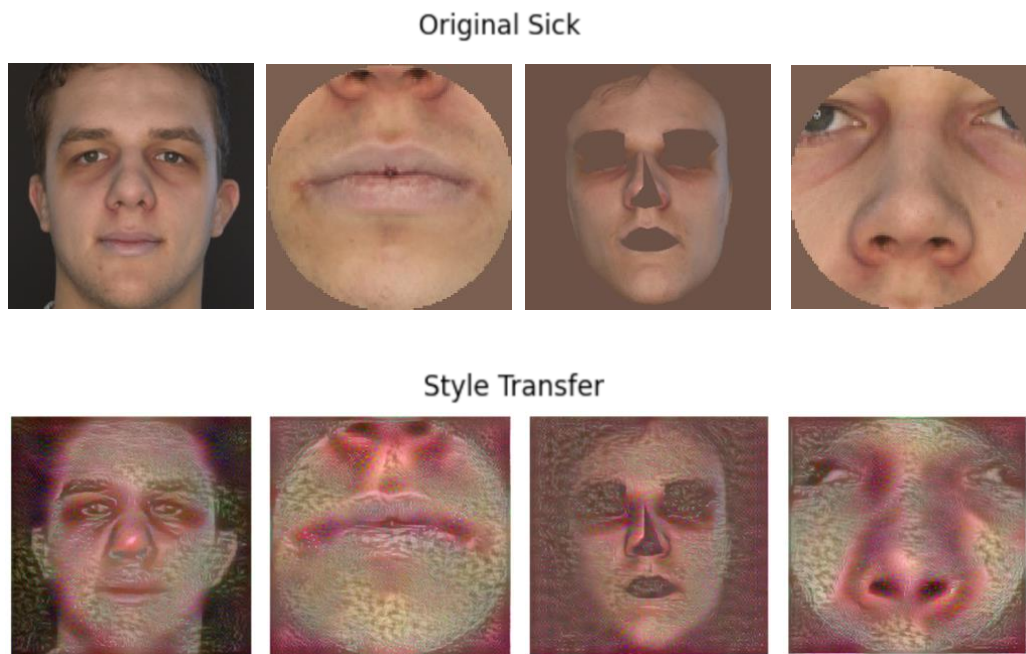


Figure 1: Style transfer on dark spots of the face indicating illness using a mask for the content image. Learning rate=.5, num_epochs=50, model=VGG19, objective=CrossEntropyLoss(), optimizer=Adam, Treshhold_mask=10, alpha=1e3, beta=1e-4.

From Figure 1, it is noticeable that not only the redness around the eyes and nose were captured, but shadows as well, like the jawline, bridge of the nose, mouth fissures, and other darker regions. To optimize the alpha and beta parameters for the weighted loss of the content and style image, as well as other arguments, I performed long grid searches, with the ones in the image being the best ones, saving checkpoints along the way for arguments like loss, epochs, etc. Once the styled images were obtained, I used transfer learning of a pre-trained CNN again (still using VGG19) by unfreezing layers to adapt the model to my specific labels ‘sick’ and ‘healthy’ and loading the checkpoints into this new model that will be tested on the validation set.

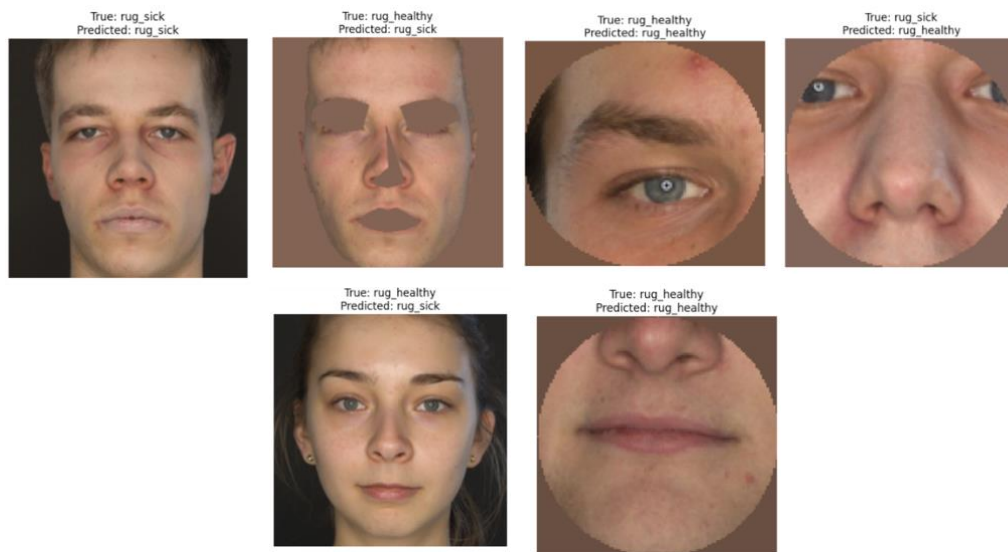


Figure 2: Results for classification based on style transfer, using transfer learning.

With an overall accuracy of 57%, the results show what was probably expected, that at times, the detection of the redness as dark spots would confuse the model. For instance, where the person is not sick it would label it as such because of other shadows in the face. This also occurs with images with redness on the skin that may not be signs of illness. As well as misclassification of an individual being healthy when they are actually sick, however, these are less prominent.

- *Color Segmentation Processing with OpenCV*

Lastly another approach I used to isolate the dark redness of the images was with the visualization package OpenCV. This library aids with detecting specific color shades of images and select or segment them. This time, I also highlighted the shade of dark red in the healthy images. This specific shade is assigned a number, in this case the number 8. It is important to note that this shade was generalize for all pictures, ignoring the possibility that the number may change due to the lighting and angles in different images.

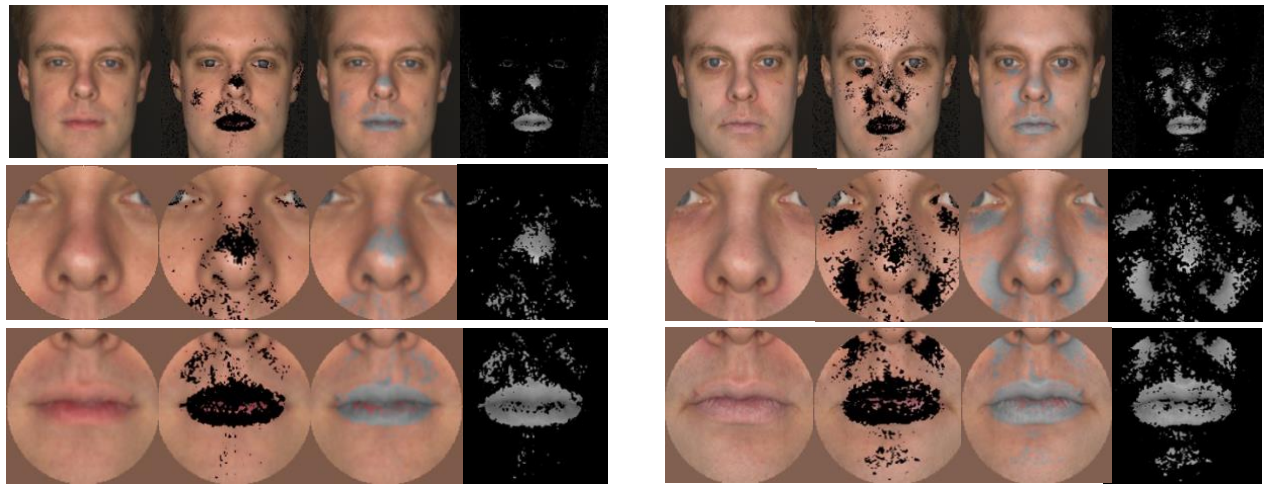
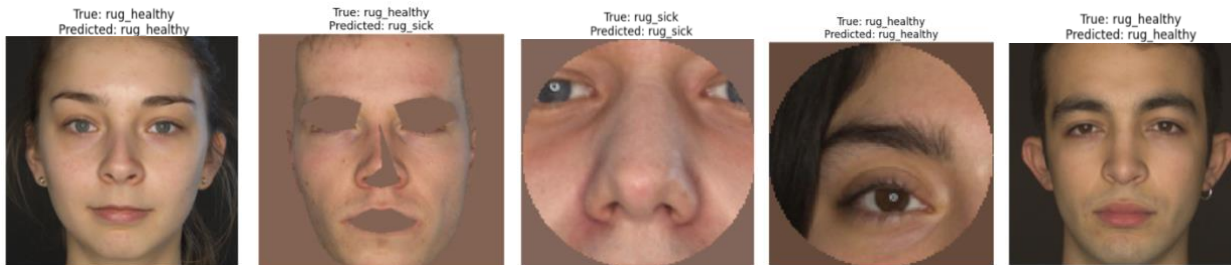


Figure 3: Color segmentation of shade 8 (dark red) using OpenCV for healthy (right) and sick (right) batches.

For training, a similar technique was taken as with style transfer. Using transfer learning on a pre-trained VGG19 model, once the segmented images were gathered, still using the dark mask as before on the gray scaled images that CV2 provides, which had to be inverted to weight more on the light regions instead since the black and white images had the highlighted regions in a light gray. Where later, the weighted values of the masking were applied to the regular RGB images.



The resulting classification was satisfactory with an overall accuracy of 78%. As seen in figure 4, the model was able to identify healthy faces more accurately than with the style transfer approach. The shading on other parts of the image were not considered when applying the mask. This was expected since color change is given by dots on very targeted places of the face, rather than a spread-out color on the whole image. Nevertheless, on images where the dotting was similarly placed for both sick and healthy, the results still differed. The model was kept with the same parameters as with the style transfer approach. The grid search performed in this model didn't affect drastically the overall performance. However, the layers of the network had to be heavy biased towards my images rather than with the dataset that is pre-trained with. The VGG architecture worked in favor for color feature detection.

Time-Log

12/7: Gathered datasets. Resized images to fit into CNN architecture. Around **2 hours**.

Started literature review about style transfer with weighted mask. Around **2 hours**.

12/8: Continued literature review about pre-trained nets that work best for style transfer. Around **2 hours**.

12/9: Started to implement style transfer algorithm and mask for content images. Around **4 hours**.

12/12: Finished implementation and debugging style transfer algorithm with checkpoints. Around **5 hours**.

12/13: Started to train and running grid searches. Around **5 hours**.

12/15: Finished training and running grid searches that gave best results . Around **6 hours**.

12/16 Gathered results for style transfer approach. Wrote draft for this section. Around **3 hours**.

Started literature review for OpenCV approach, mainly how to use the library. Around **1 hour** (it was more but I am not counting them).

12/18: Passed images through the cv2 package to get shade segmentations. Figured out the correct color for the red shade. Around **1 hour** (it was more but I am not counting it).

12/19: Started to train. Played around with the layering for transfer learning of the pre-trained model. Incorporated the dark mask and inverted it. Around **5 hours**.

12/20: Kept training and optimizing layers. Ran the largest grid search yet. Around **7 hours**.

12/21: Finished write up.

Total hours that are valid: **43 hours**.