CS-454 HW4
Assignment Report
Fall 2020

Mustafa Alper Sayan

S015674

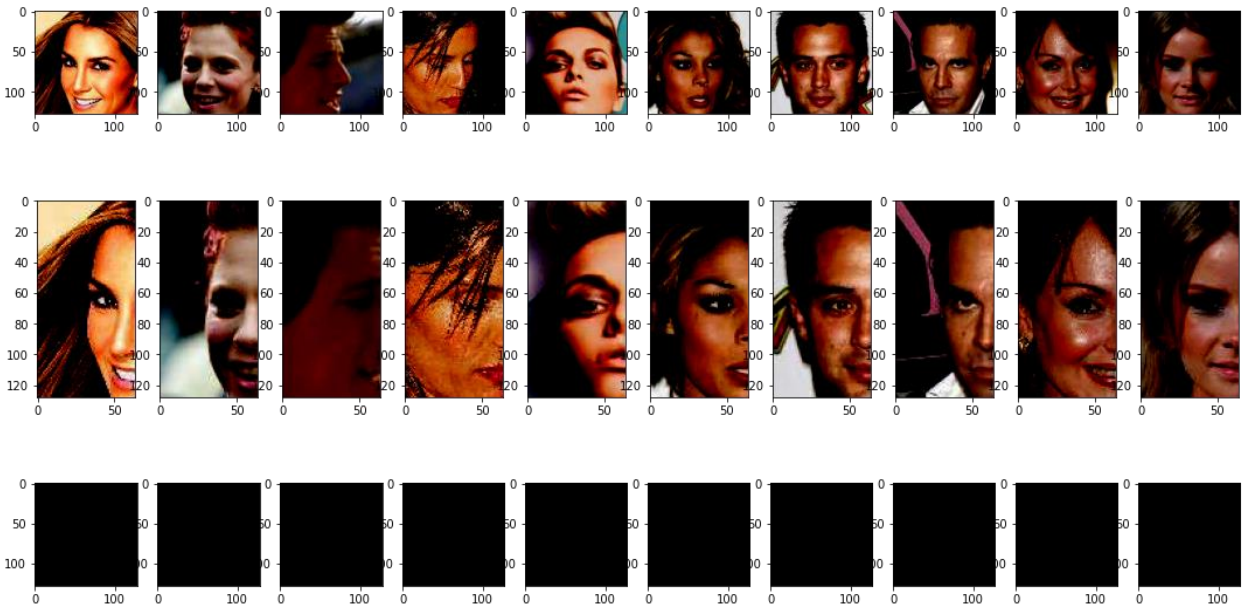

Figure 1: Shows the results of the simple encoder decoder. We can see that our model is too simplistic to capture the overlying problem

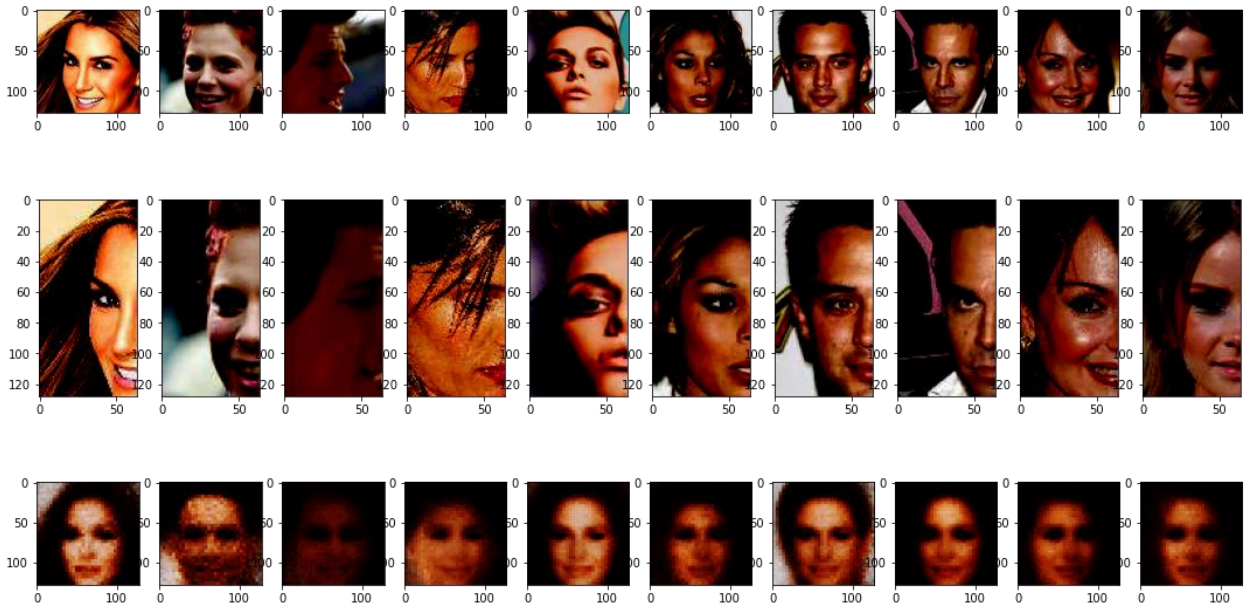

Figure 2: Shows the results of the encoder decoder model version 1. Even with the limited number of epochs my model produces reasonable outcomes. More epochs would have been helpful to obtain better results.

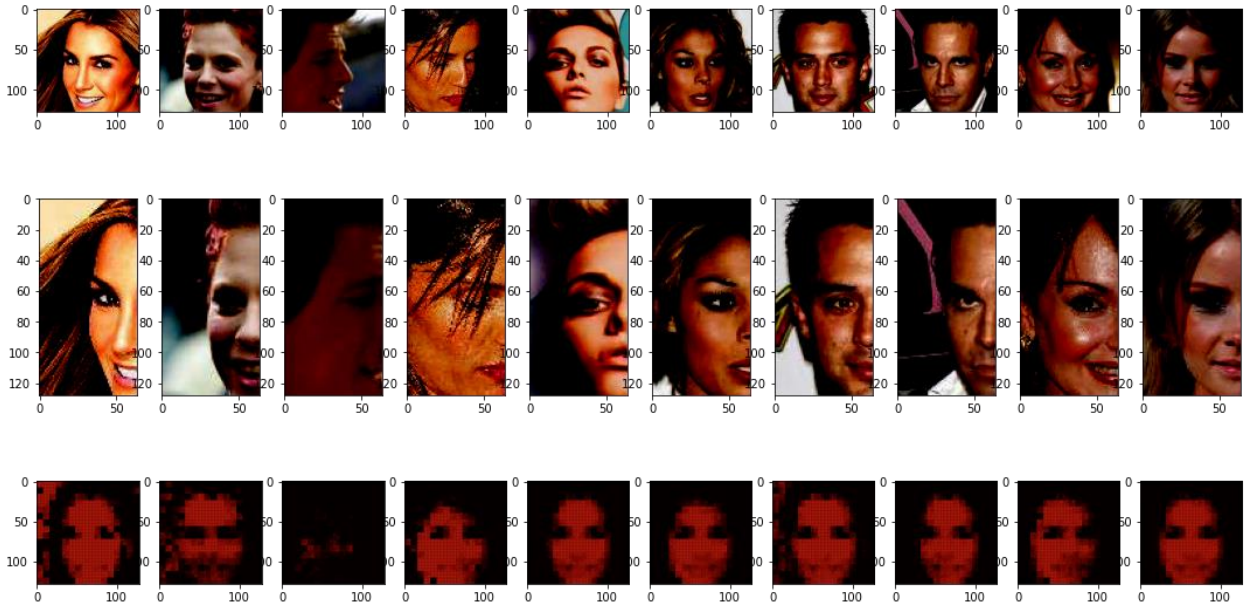

Figure 3: I added one more convolution layer to the model seen in figure two with different input and output channels. As can be seen from the figure this was not helpful in generating better results for the same training instances.

Figure 4: Shows the MSE scores of all the models. As can be seen from the plot simple encoder decoder model with one convolutional layer and 2 fully connected layers did not produce desirable outcomes. This tells us that our model was too simple to capture the underlying problem. Model v1 achieved the best result when compared to other models. This success can be seen in the figure 2 as well. Model 2 has 2 convolutional layers, 2 fully connected layers and two transposed convolutional layers. This model architecture achieved the best results out of the models I have tested. Model v2 has 1 more layer in each category compared to the model v1. As can be seen from the plot this model has underperformed compared to model v1. There can be too reasons for this behavior either the number of training instances the model saw was not enough therefore the model gave random outputs or the input channels and output channels of the models were not optimally tuned.

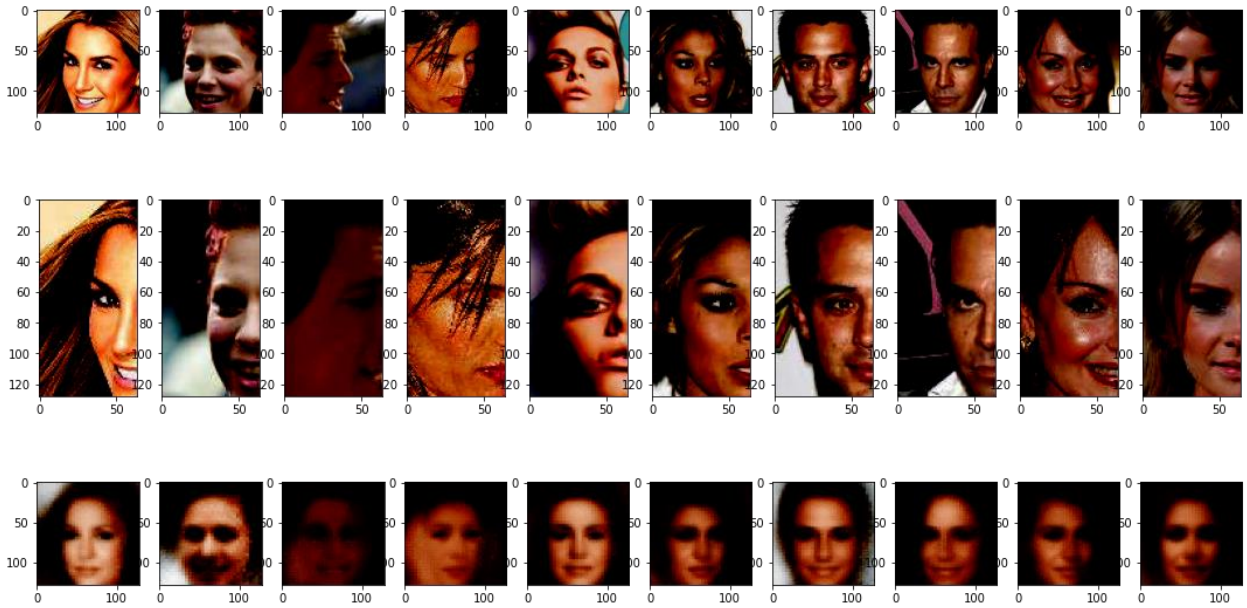

Figure 5: This is the same model shown in figure 2 but after 10 epochs. From this result we can determine that increasing the epochs would have increased our models performance. This results are much closer to optimal results

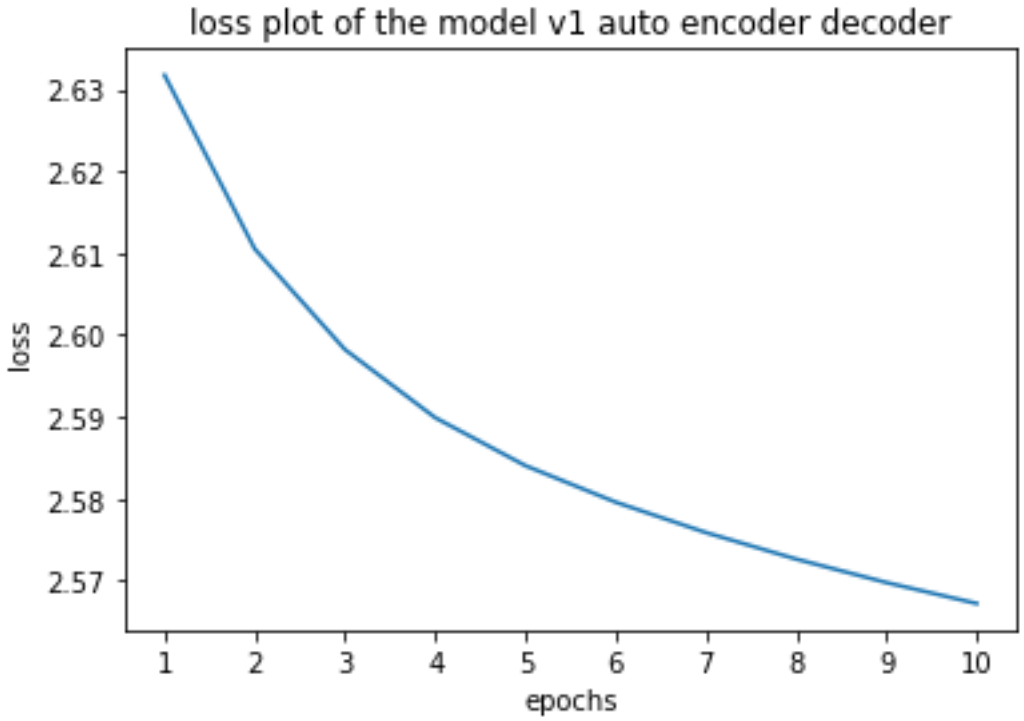

Figure 6: As can be seen from the figure we still needed to increase the number of epochs since we have not reached the elbow point yet. But due to time limitations and me using CPU as the training tool since I have exceeded the limits offered by Google Colab. This is the best result I have obtained