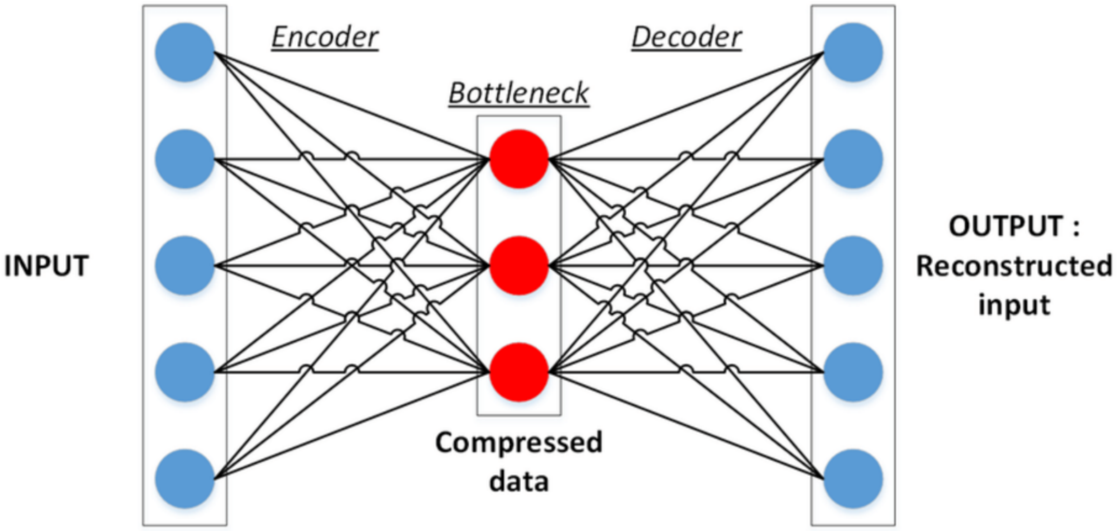
Name: Rahul Vijayvargiya
Student ID: 245784

Assignment 5: Neural Networks and Deep Learning

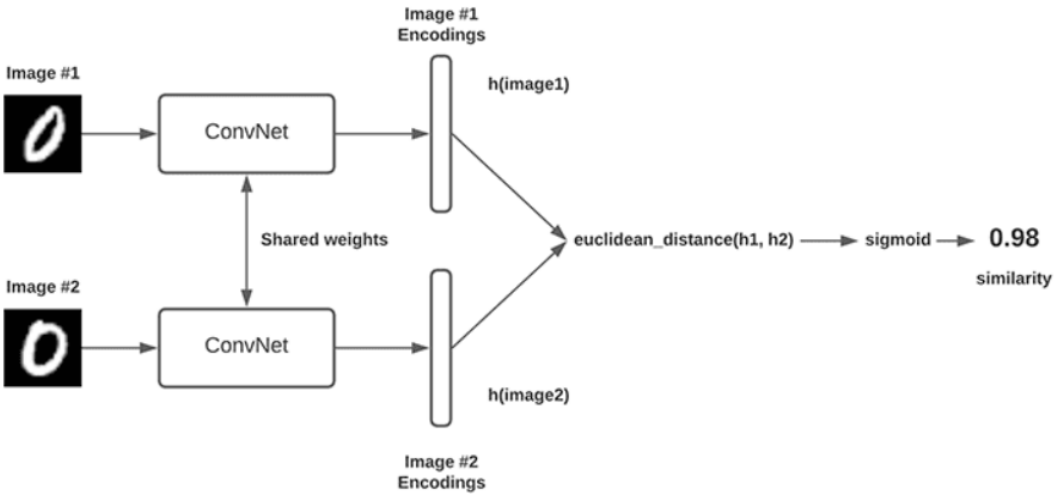
Autoencoders and Siamese Neural Networks:

Autoencoder is a technique in Deep learning which we gonna use for our assignment to decompress the image into latent view and decode the image from latent view based on encoding and decoding layers,



- Encoder: Learns how to compress the original input into a small encoding
- Decoder: Learns how to restore the original data from that encoding generated by the Encoder
- Bottleneck is the latent view, where compressed data is store in latent form

Siamese Neural Networks: is a technique in a deep learning to identify the similarities between images, face recognition,



Description about Data:

Source: [Totally Looks Like Dataset]

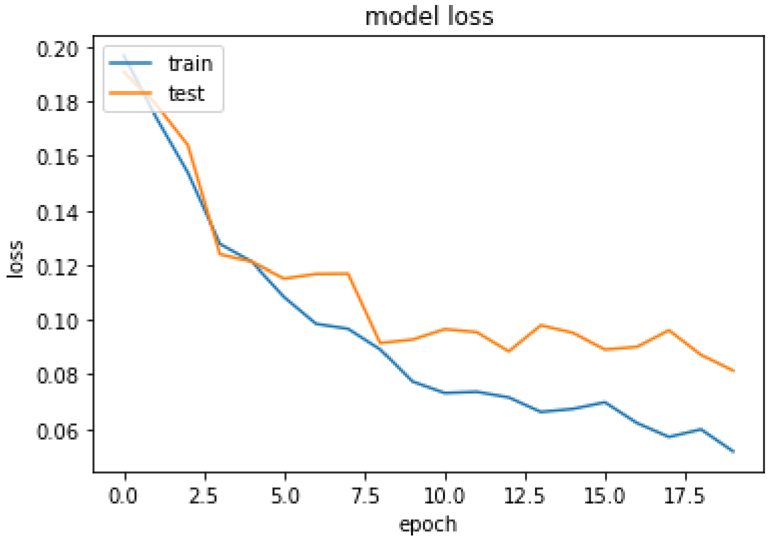
Data we have in image format, two folders left and right, more than 6000 images in jpg format available in the dataset, open source and free to use dataset, need to process the image in order to use with Autoencoder and Siamese Neural Network Algorithms

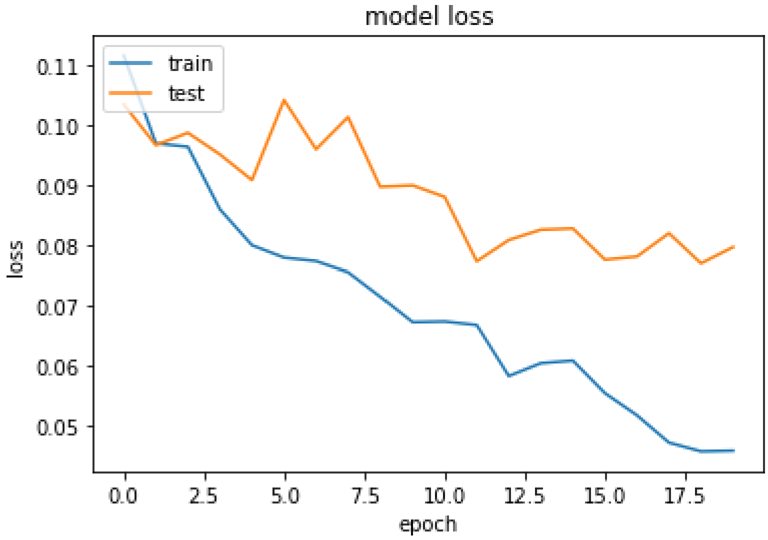
Data Preprocessing:

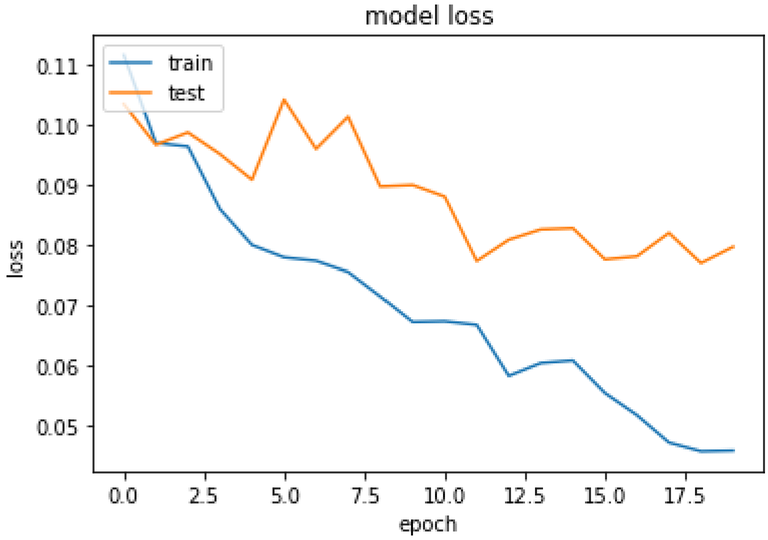
So our dataset is a bunch of images, i use glob and path functions to read all the images from the directory and combine both images from the folders left and right, read images through openCV imread method and decode into array and tensorflow decode to jpg and use tensorflow slices to use a slices of array

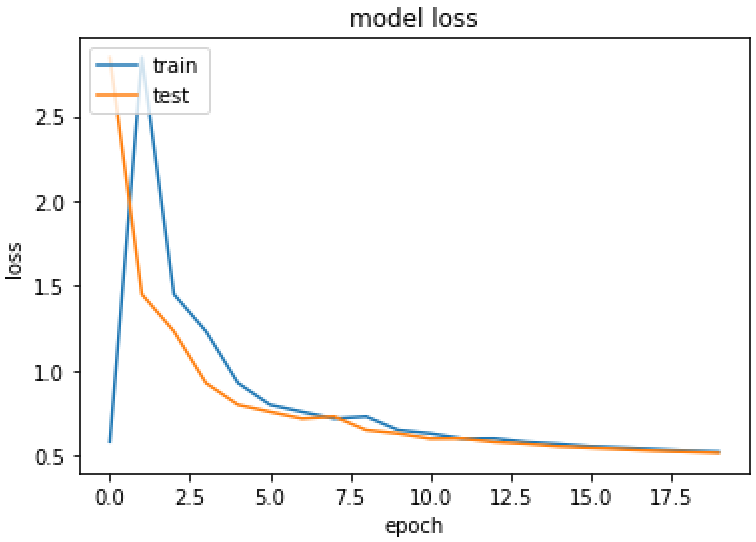
Experiments:

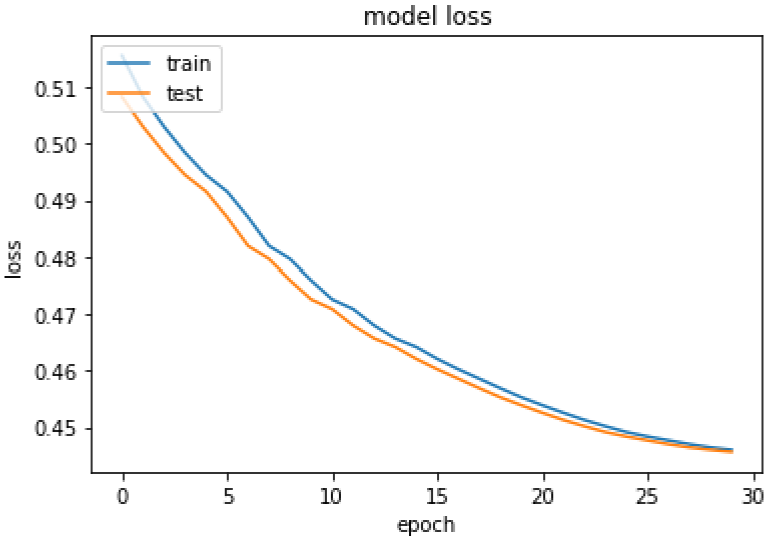
Using of different activations functions relu and sigmoid, selu, softmax, hyperbolic tangent activation function
training data with previously trained stacked autoencoder
changing epochs and batch size of the model

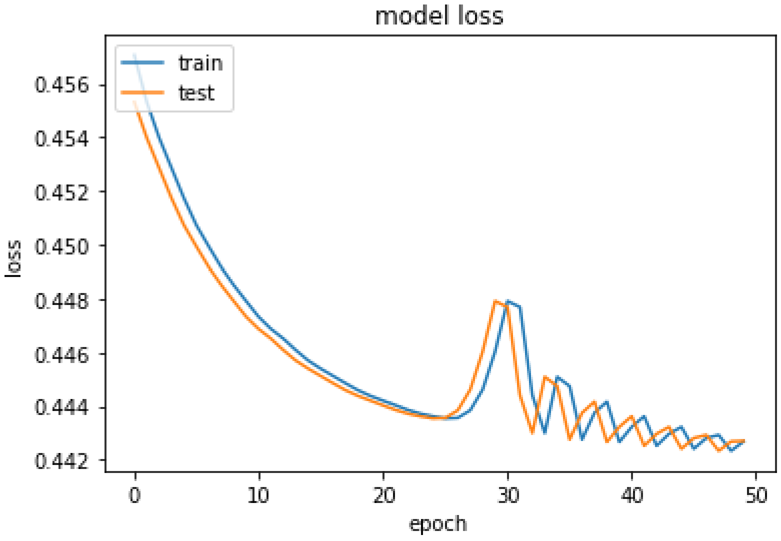

model loss
train
test
loss
0.20
0.18
0.16
0.14
0.12
0.10
0.08
0.06
epoch
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

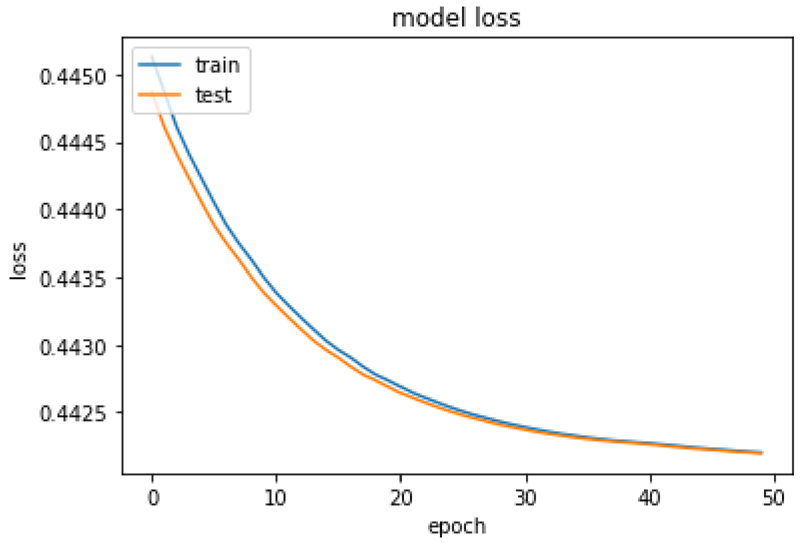

model loss
train
test
loss
0.11
0.10
0.09
0.08
0.07
0.06
0.05
epoch
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

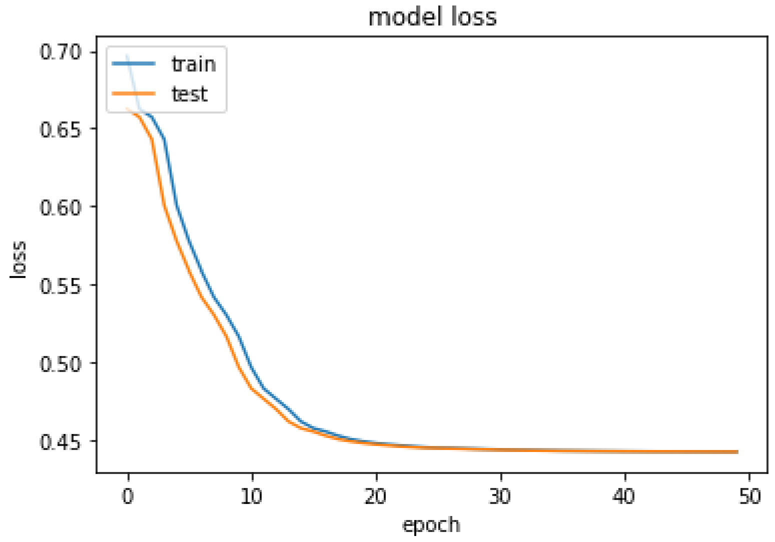

model loss
train
test
loss
0.11
0.10
0.09
0.08
0.07
0.06
0.05
epoch
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

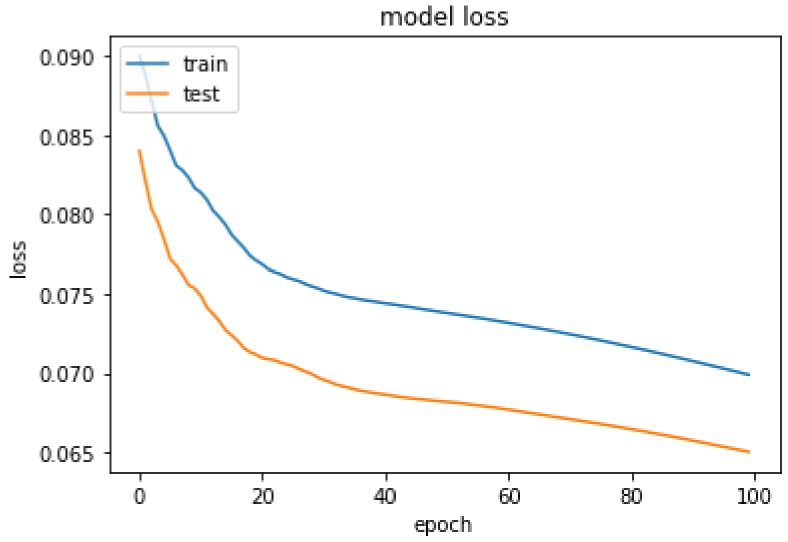

model loss
train
test
loss
2.5
2.0
1.5
1.0
0.5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
epoch

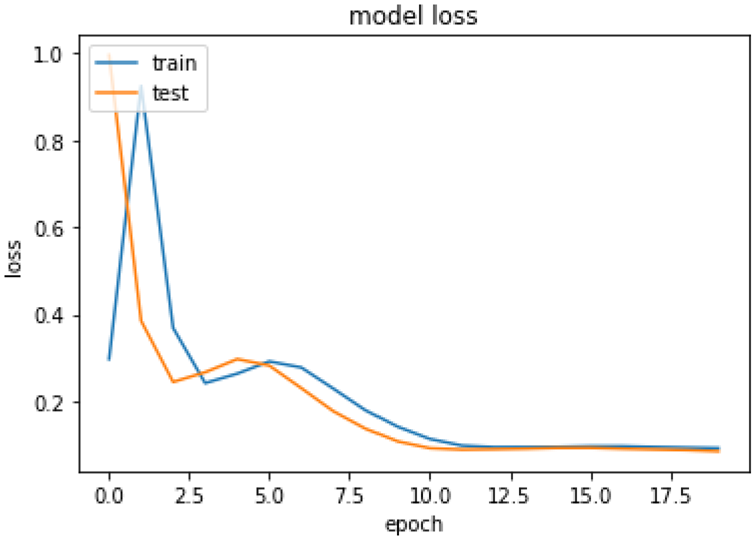

model loss
train
test
loss
0.51
0.50
0.49
0.48
0.47
0.46
0.45
0
5
10
15
20
25
30
epoch

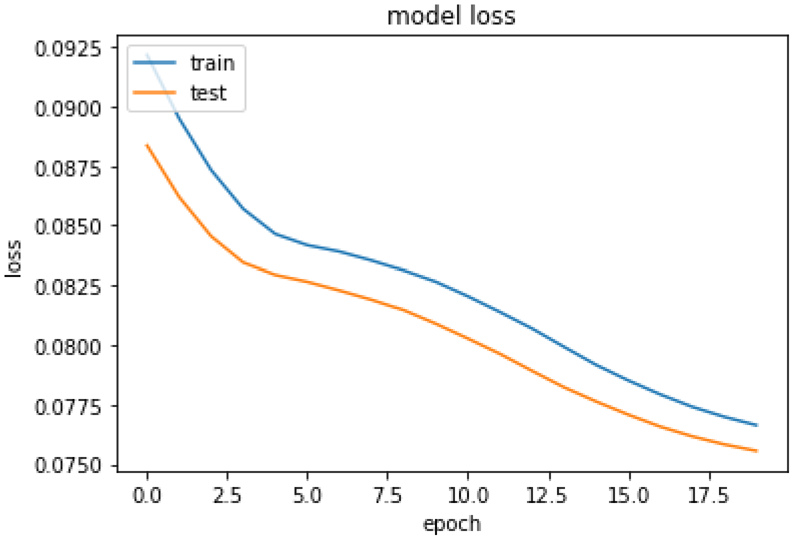

model loss
train
test
loss
0.456
0.454
0.452
0.450
0.448
0.446
0.444
0.442
0
10
20
30
40
50
epoch

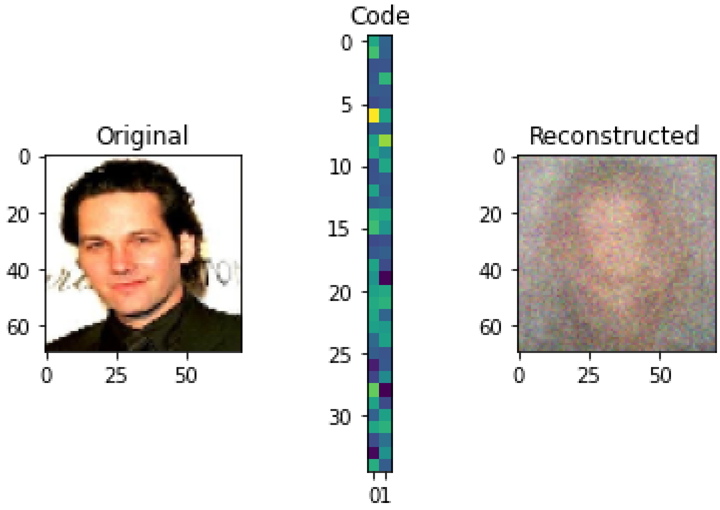

model loss
train
test
loss
0.4450
0.4445
0.4440
0.4435
0.4430
0.4425
0
10
20
30
40
50
epoch

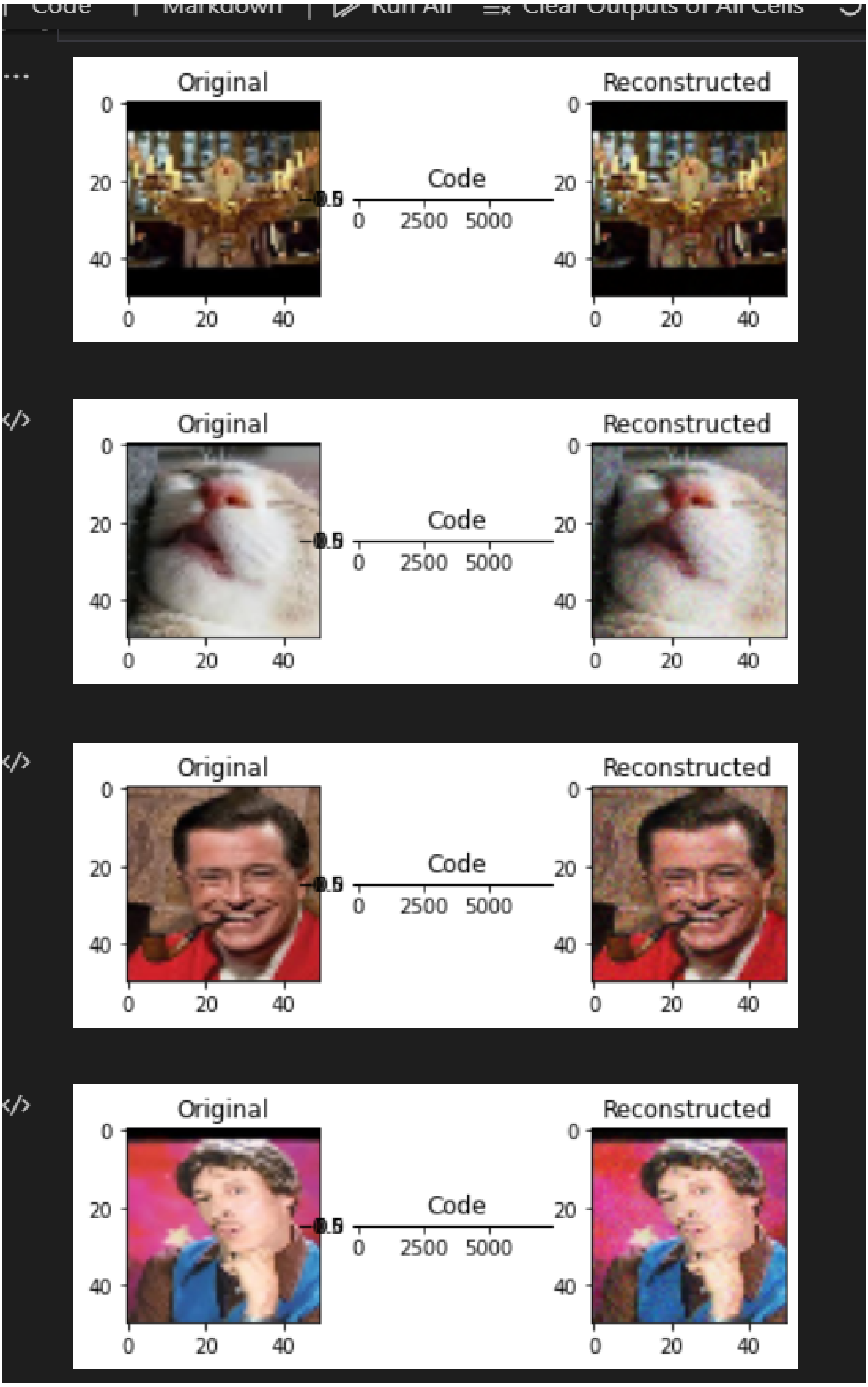

model loss
train
test
loss
0.70
0.65
0.60
0.55
0.50
0.45
0
10
20
30
40
50
epoch


model loss
train
test
loss
0.090
0.085
0.080
0.075
0.070
0.065
0
20
40
60
80
100
epoch


model loss
train
test
loss
1.0
0.8
0.6
0.4
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
epoch

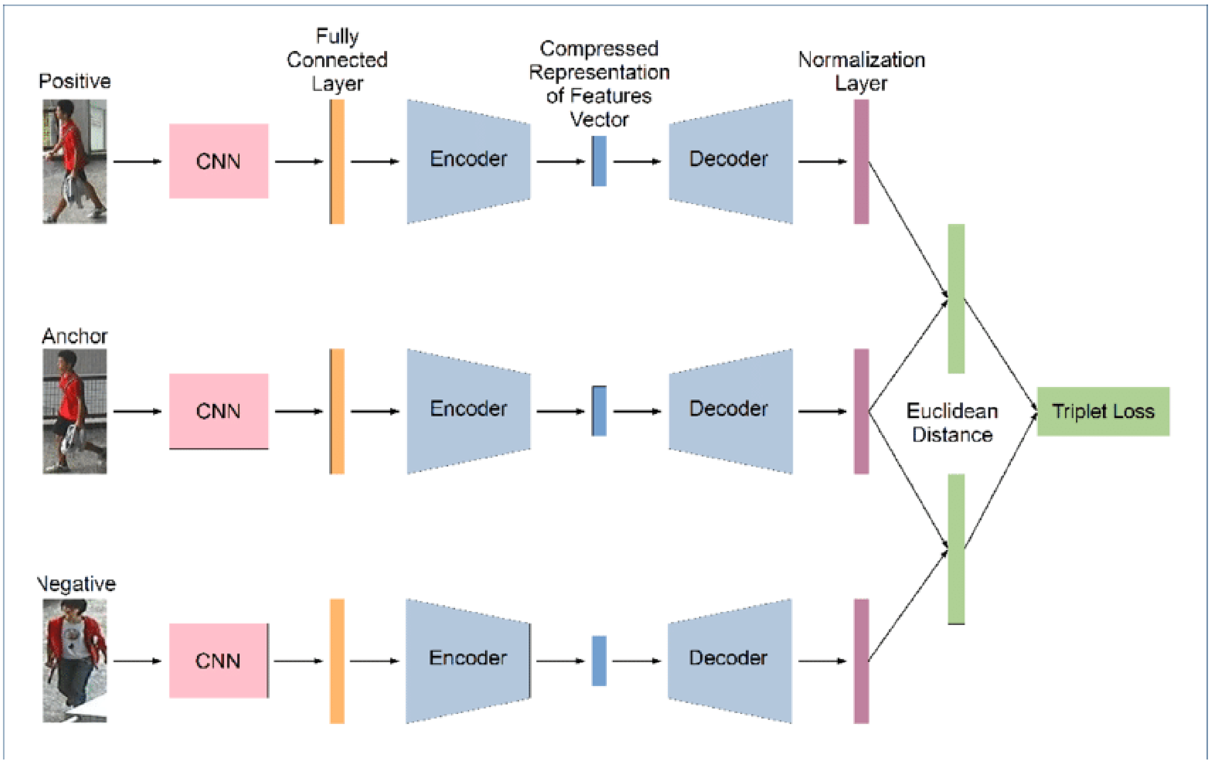

model loss
train
test
loss
0.0925
0.0900
0.0875
0.0850
0.0825
0.0800
0.0775
0.0750
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
epoch


Original
0
20
40
60
0
25
50
Code
0
5
10
15
20
25
30
01
Reconstructed
0
20
40
60
0
25
50


Code
Markdown
Run All
Clear Outputs of All Cells
Original
0
20
40
0
20
40
Code
−0.5
0.5
0
2500
5000
Reconstructed
0
20
40
0
20
40
Original
0
20
40
0
20
40
Code
−0.5
0.5
0
2500
5000
Reconstructed
0
20
40
0
20
40
Original
0
20
40
0
20
40
Code
−0.5
0.5
0
2500
5000
Reconstructed
0
20
40
0
20
40
Original
0
20
40
0
20
40
Code
−0.5
0.5
0
2500
5000
Reconstructed
0
20
40
0
20
40

Embedded layers we use Previous trained autoencoders, we train stacked autoencoder two time,
Different parameter gave us different result so far we didn’t obtain any concrete result for which we were using autoencoder

With the siamese neural network we did achieve the result, we compares images and found similarities in the images
Here we have anchor, positive and negative images,
We are using triplet loss to compute the distance with the images



Encoded and decoded images :

```
Positive similarity: 0.99747443
Negative similarity 0.996416
```

Overview:

So far i learned more about neural networks, how they worked, how to train autoencoders and siamese neural network using tensorflow and keras api,
How to preprocess images dataset, convert images into numpy arrays and decode images, playing with RGB scales and convert data type of images

Sources:

https://stackabuse.com/autoencoders-for-image-reconstruction-in-python-and-keras/

https://keras.io/examples/vision/siamese_network/

https://towardsdatascience.com/how-to-generate-images-using-autoencoders-acfbc6c3555e https://towardsdatascience.com/a-friendly-introduction-to-siamese-networks-85ab17522942