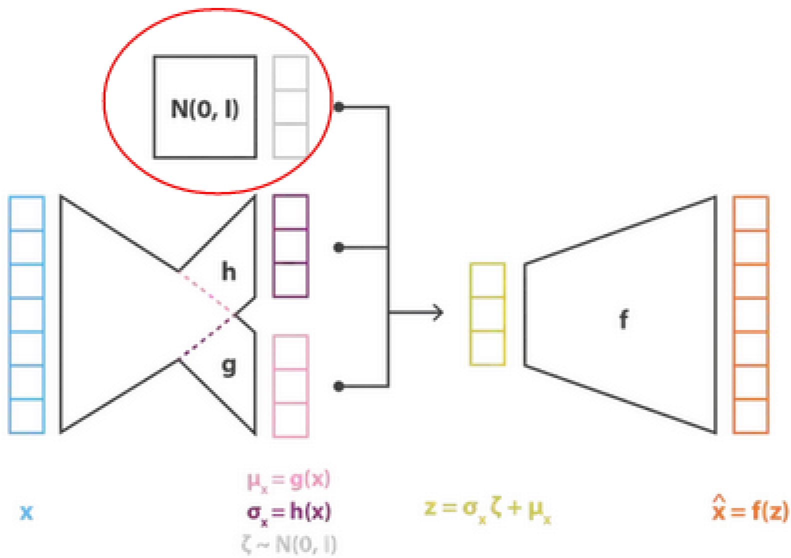
Carter Rogers
DATS6203

Final Project - Individual Report

Our final project was experimenting with different types of GAN architectures to generate facial images.We used the famous and widely available ‘Celeb-a’ dataset and trained three different types of networks using this dataset: GANs, WGANs, and VAEs.

I was responsible for the experimentation of the Variational Autoencoder (VAE), including the implementation, training, and interpretation of results. VAEs are similar to GANs in their ability to generate artificial image, text, and audio data, but the process by which the model generates the data is quite different. VAEs utilize an encoder/decoder structure similar to GANs, but a key aspect to VAEs is their dimensionality reduction by the encoder to preserve the maximum amount of information possible. Likewise, the object of the decoder is to minimize error in the reconstruction of the data (images, in the case of this project). The autoencoder (encoder/decoder) process is such that the data is fed through the encoder and decoder, comparing the output of the decoder to the input, and backpropagating the error over the network to update the weights accordingly. The autoencoder architecture ensures that only critical information is passed through and reconstructed through both Principal Component Analysis (PCA) and gradient descent to minimize error.

VAEs differ from traditional autoencoders in that the latent (encoded) space must be regularized in order to generate meaningful outputs. The autoencoder is trained to minimize loss, but does not organize the latent space during training. Unless the network is regularized, the autoencoder will overfit as much as possible. We define VAEs as “an autoencoder whose training is regularized to avoid overfitting” (Rocca). The autoencoder of a VAE is trained to minimize reconstruction error. This is accomplished by the encoder returning a distribution over the latent space instead of a single point.



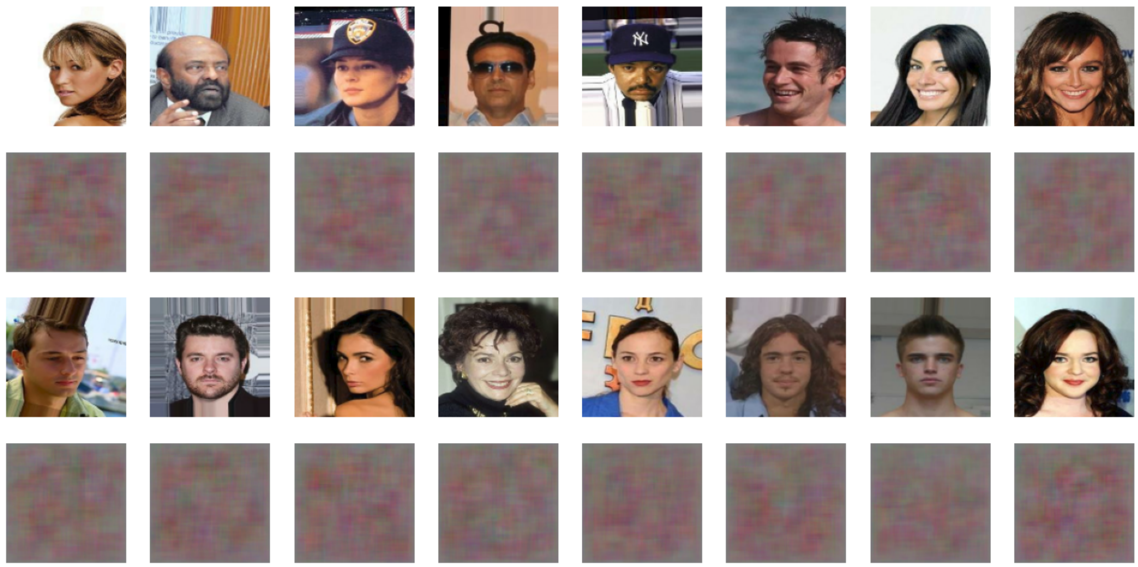
Architecture of a VAE network with Regularization term circled in red (Rocca).

Experiment

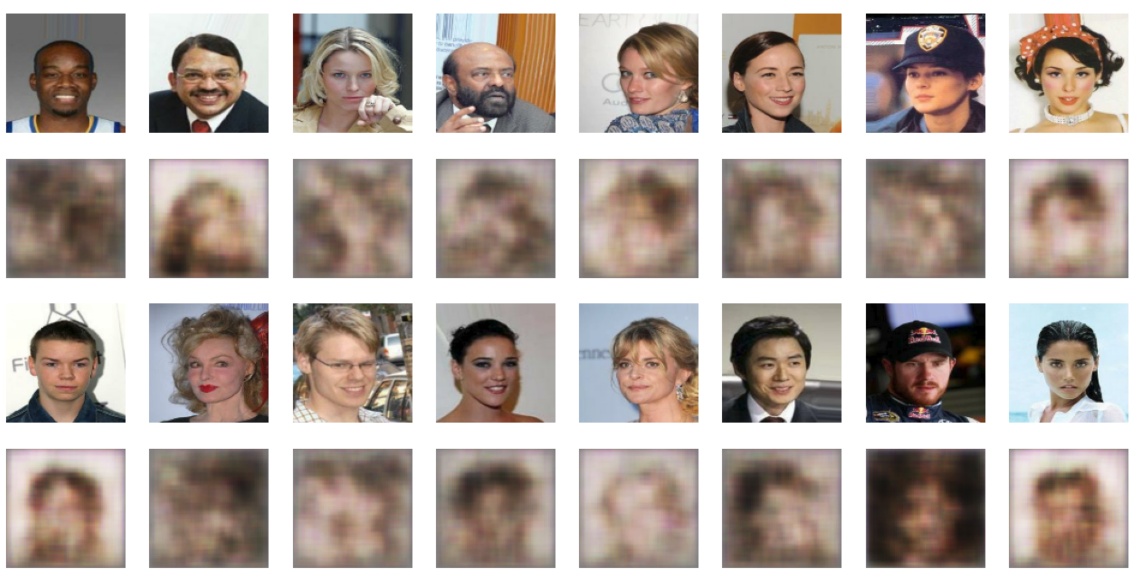
The experiment of generating facial images using VAEs follows a very similar process to that of using GANs. The images are resized to 112x112 with the pixels divided by 255. The training of the VAE took far less time than the GAN; using 50 epochs, the training was complete in roughly 100 minutes, for approximately 2 minutes per epoch. Compare this with the training time for the GAN, which took approximately 40 minutes per epoch. Thus, there was no need to downsize the images to a lower resolution, since training took a reasonable amount of time.

Results

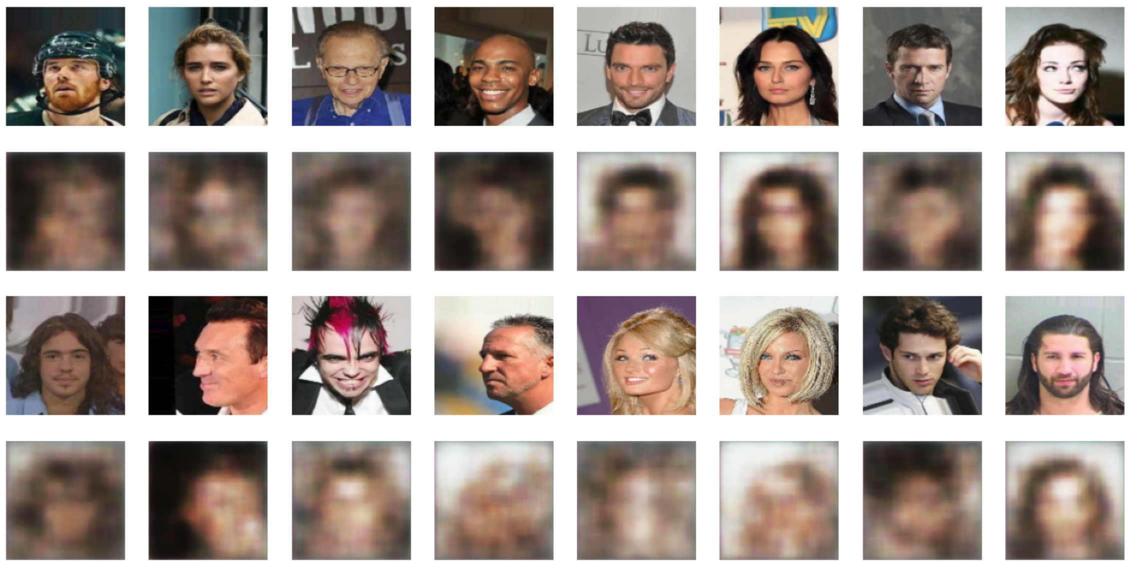
1 epoch:



20 epochs:



50 epochs:



One major benefit to using VAEs over GANs for image generation is the computational power needed for training. It is much easier to train a VAE network as opposed to a GAN. However, we observe that the images generated by the VAE network lack definition after 50 epochs of training. The outlines of the faces are visible, and hairstyles can be discerned, but the faces themselves appear uniform. This is one major weakness of VAEs as opposed to GANs; the images generated are less defined than those generated by the GAN. The regularization process of the VAE often leads to more "smoothing" of the images, resulting in faces with smooth outlines and less detail in the eyes, ears, nose, etc.

I implemented this portion of the group project in Tensorflow. With more time allotted, I would've liked to implement this network in PyTorch as well, for additional practice. We chose these three GAN architectures for their generative capabilities, but in the future we could consider other types of GANs as well. PixelCNNs are another popular type of generative network that we considered implementing, but did not due to time constraints.

Percentage of code found/copied from the Internet: $(100 - 40)/(100 + 30) * 100 = 46\,\%$

References

Cheong, Soon Yau. *Hands-On Image Generation with TensorFlow: A Practical Guide to Generating Images and Videos Using Deep Learning*. Packt Publishing, 2020.

Chollet, Francois. “DCGAN to generate face images.” *Keras*, 29 April 2019, https://keras.io/examples/generative/dcgan_overriding_train_step/. Accessed 12 December 2022.

Chollet, Francois. “WGAN-GP overriding `Model.train_step`.” *Keras*, 9 May 2020, https://keras.io/examples/generative/wgan_gp/. Accessed 12 December 2022.

Joseph, Rocca. “Understanding Variational Autoencoders (VAEs) | by Joseph Rocca.” *Towards Data Science*, 23 September 2019, https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73. Accessed 12 December 2022.

Liu, Ziwei. “CelebA Dataset.” *MMLab@CUHK*, https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html. Accessed 12 December 2022.