



Search for Similar Images

Islam Gamal - Karim Mohamed

Prof. Dr. Moustafa Youssef

Problem Statement

We are trying to implement a simple and efficient way to do the searching by image. Search by image can be useful in detecting rumors, finding the attached news or finding similar products. We chose AutoEncoders to implement this.

Discussion

Autoencoders can be used in this problem by training the Encoder Decoder model on the dataset, Using Encoder part to encode the whole dataset to 1D arrays, When an image query is received Just encode it and find the nearest neighbours to this array, Finally, Decode the candidates and return them as the search results.

Datasets

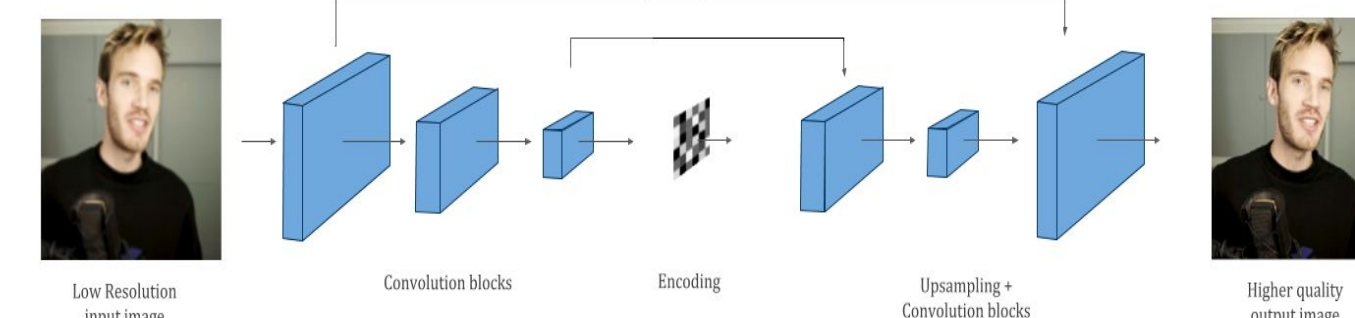
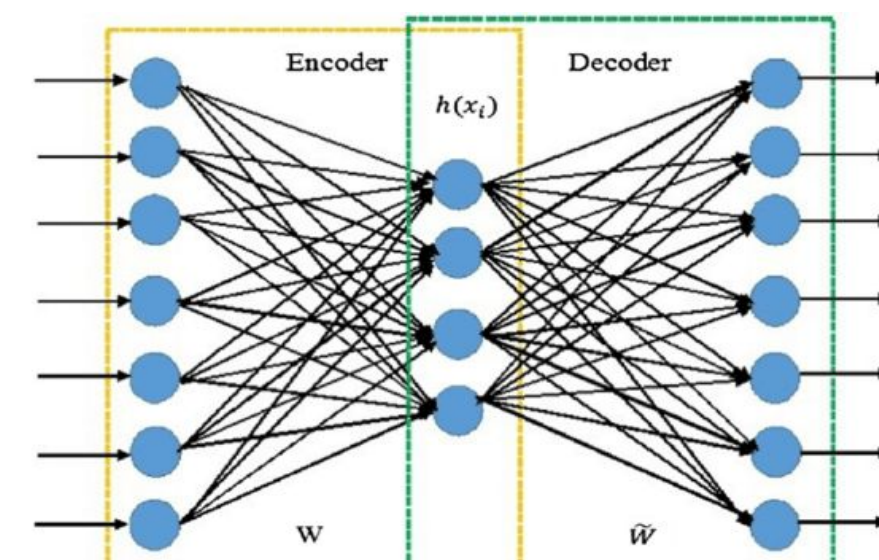
We used MNIST handwriting and Fashion.



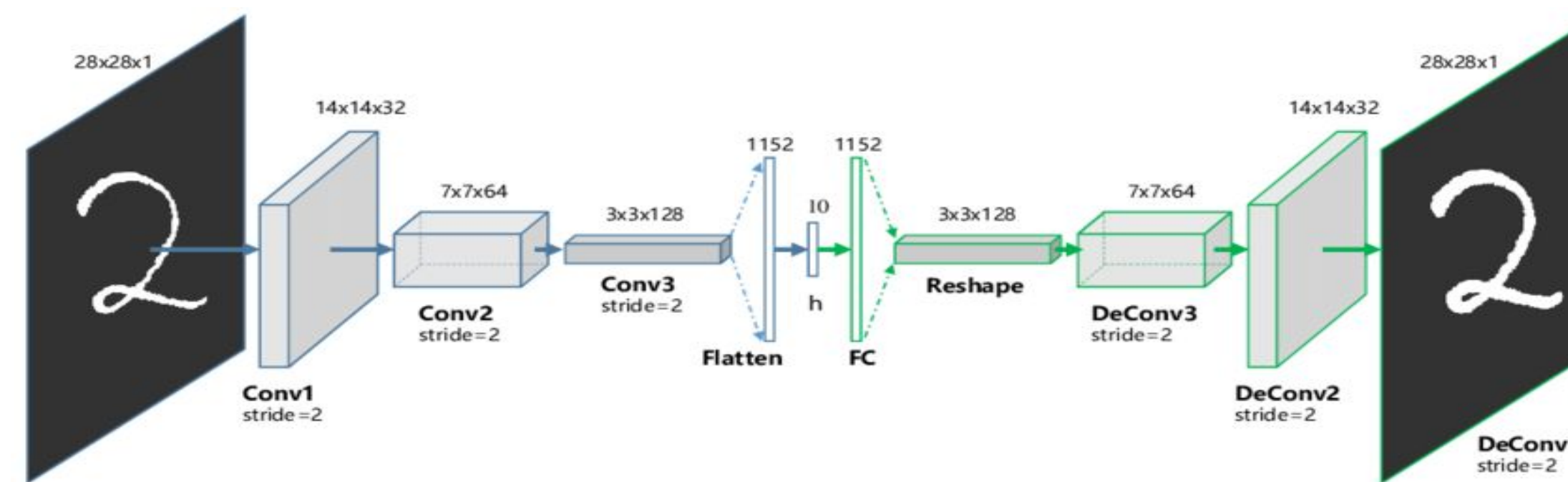
Model

Encoder Decoder Network, where we tried the following:

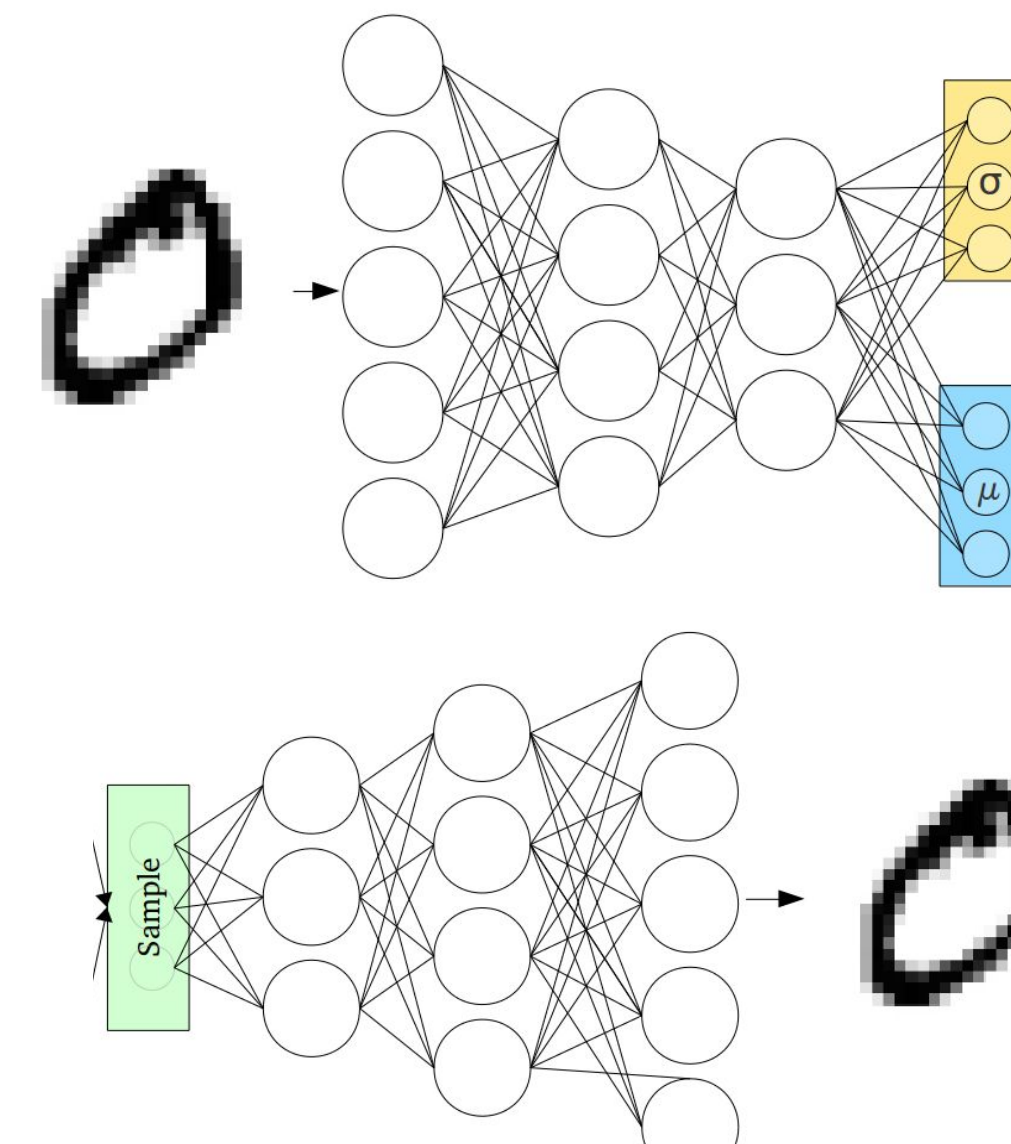
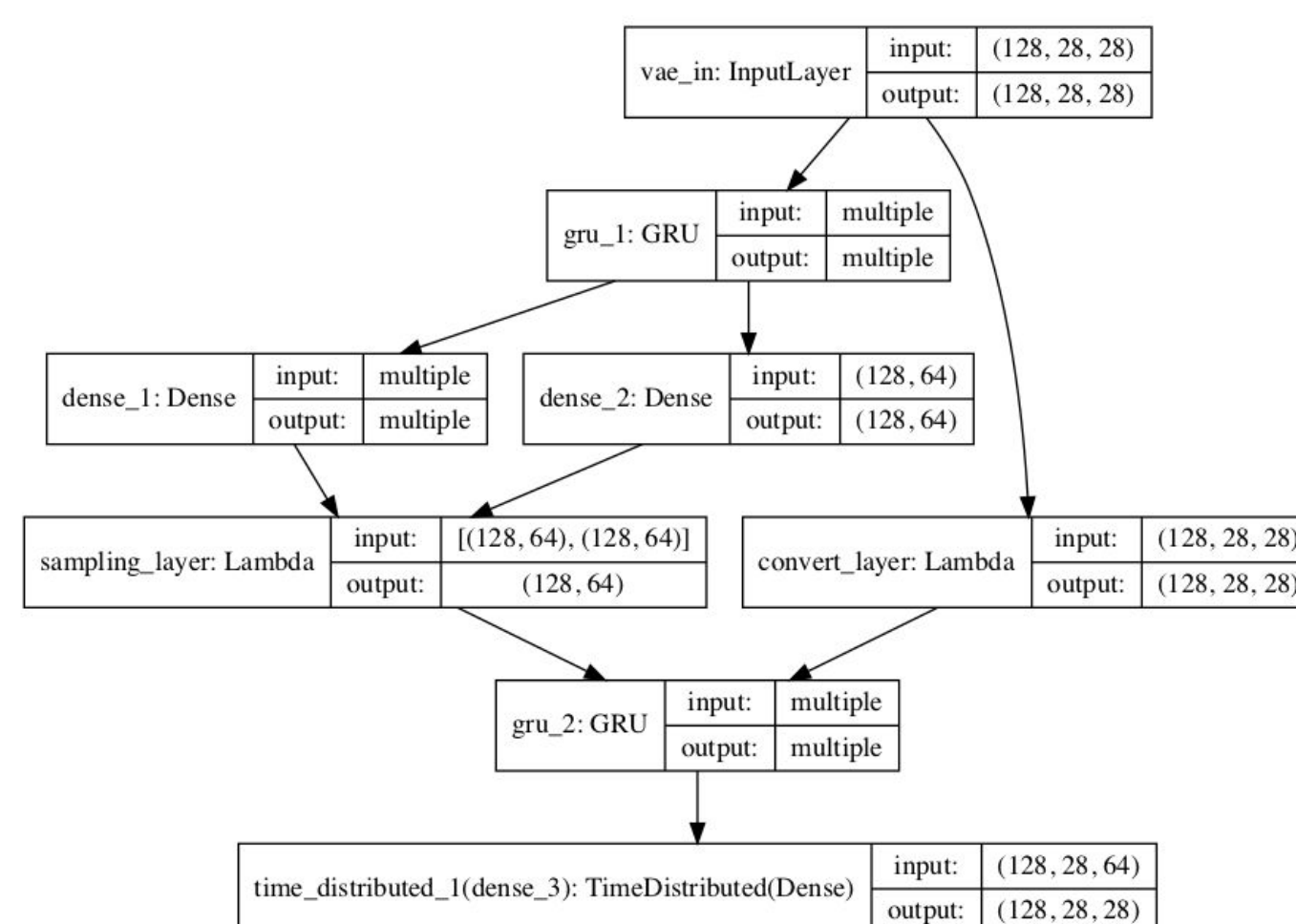
- Encoder: FC Layers
- Decoder: FC Layers
- Encoder: CNN.
- Decoder: UpSampling.



- Encoder: Convolutional NN + FC
- Decoder: FC + Deconvolution.

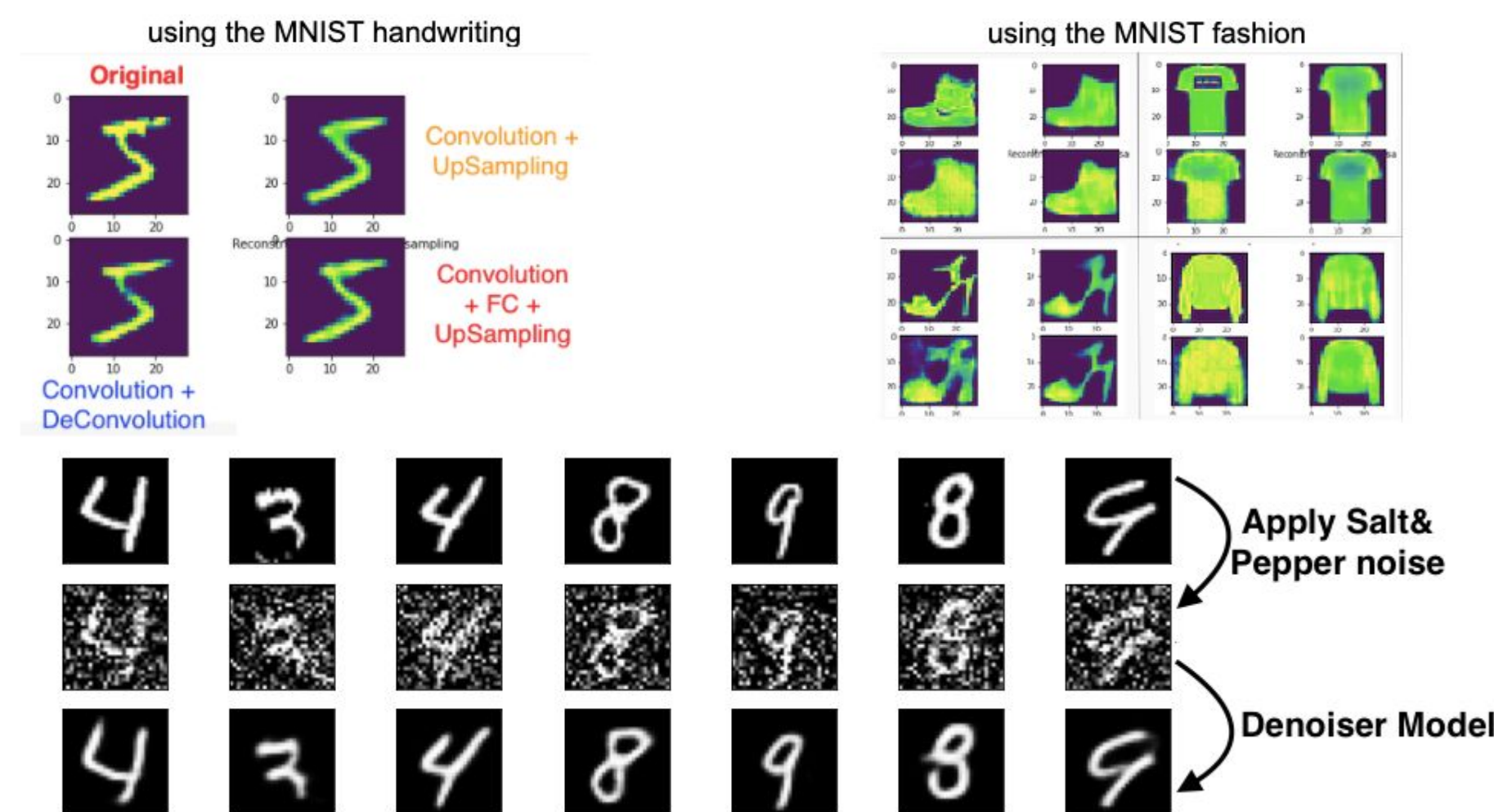


- Encoder: GRU+2 parallel Dense
- Decoder: Sampling+GRU+dense
- Encoder: CNN + mean,std dev
- Decoder: sampling + DeConv



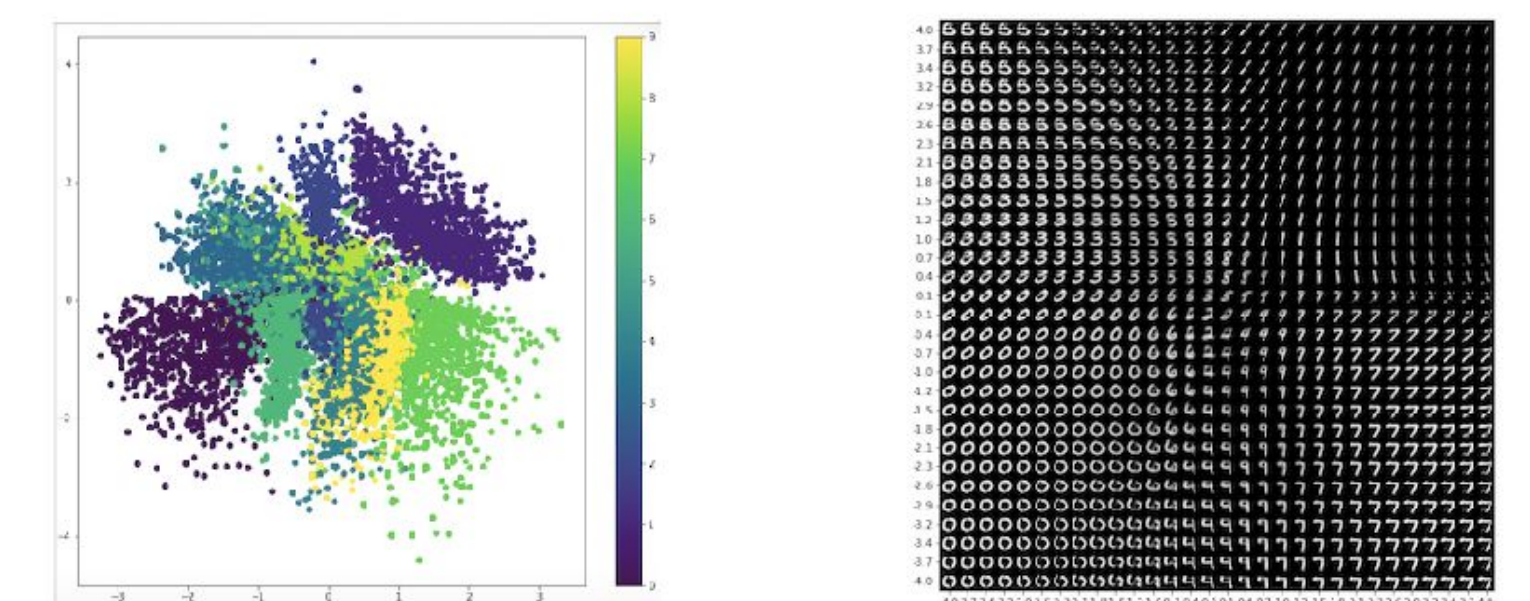
Evaluation & Results

Approach	MSE train loss by	Them	Us (Fashion MNIST)	Us (HandWriting MNIST)
Fully Connected Only		0.0214	0.0034	0.0013
Convolution + Up Sampling		0.0407	0.0073	0.0032
Convolution + DeConvolution		0.0297	0.014	0.024
Conv + FC + Up Sampling		0.0173	0.0068	0.0028



Dataset distribution Plots

Visualizing the effect of replacing AutoEncoders with Variational AutoEncoders



Conclusions

- Autoencoder is enhanced by adding FC layers.
- VAE is more powerful in our problem (Search).
- Encoding the dataset by VAE produces a continuous dataset while AE produces a clustered (discrete) one.
- Given a noisy input, AE will produce a 1D vector that is not similar to the original noisy image while VAE will.
- We can make AE reconstruct **noisy** images by originally training it on that (noisy vs denoised as input).
- Using RNN/Attention here is Not efficient.(No sequence)