Font generation

Alena Shilova, Darina Dvinskikh

Skoltech, Deep learning

May 25,2017

Outline

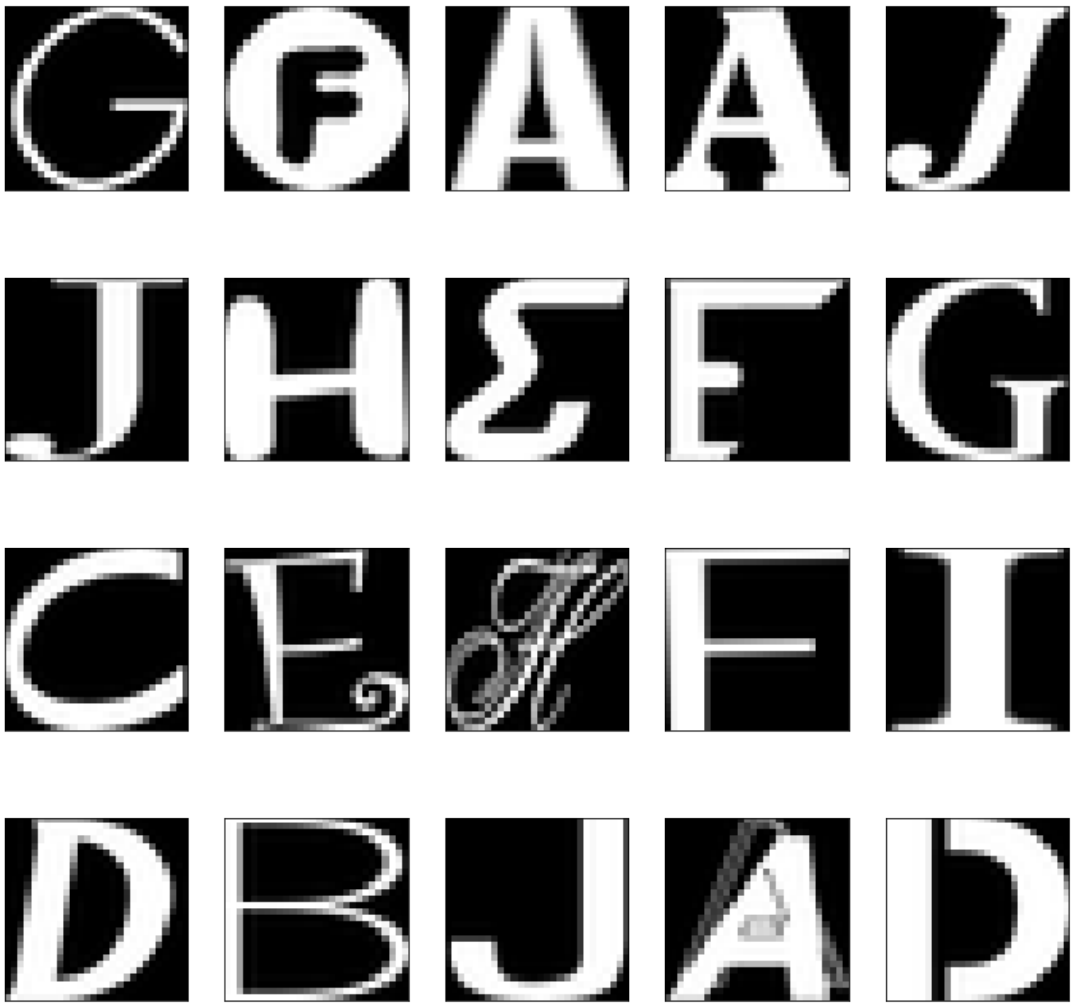
Problem statement

We had two goals which we wanted to achieve:

- to build and train neural networks which could generate already existing fonts for different letters
- to build and train neural networks which could create new fonts for different letters

Outline

Data presentation



Different letters in various styles

Data prepossessing

Convert letters by means of One Hot Encoder

- A — [1000000000]
- B — [0100000000]
- C — [0010000000]
- D — [0001000000]
- E — [0000100000]
- F — [0000010000]
- G — [0000001000]
- H — [0000000100]
- I — [0000000010]
- J — [0000000001]

In total, there are 1873 different fonts in train data set, for each of them we have also applied One Hot Encoder

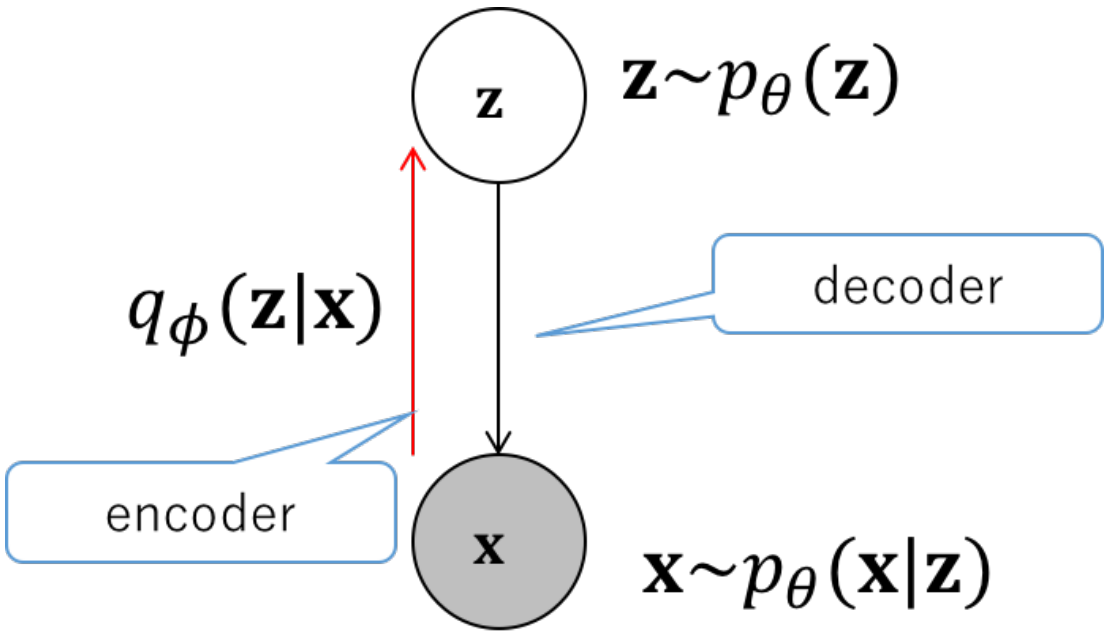
Outline

Neural Network with l2 loss

Architecture

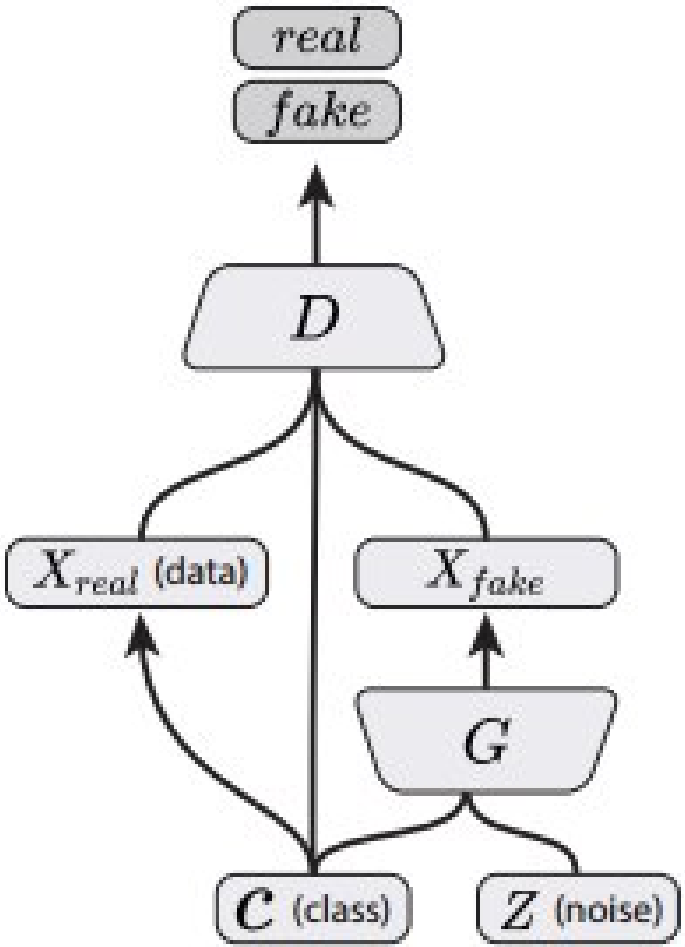
- 2 input layers for encoded letter and for encoded font style respectively
- 2 fully connected layers for each input layer correspondingly
- Layer for merging the layers mentioned above
- Reshape layer (to obtain 128 feature maps with the size 8×8)
- 2 combinations of layers Upscaling + Convolutional + Convolutional

Variational autoencoder



Variational autoencoder

Generative adversarial networks



Generator

Architecture

- 2 input layers for encoded noise and for encoded letter respectively
- 2 fully connected layers with 1024 neurons and tangent function of activation for each input layer correspondingly
- Concatenation layer
- Fully connected layer with 10*8*8 neurons and tangent function of activation
- Reshape layer (to obtain 10 feature maps with the size 8×8)
- Combination of 4 Deconvolution layers and 1 upscale layers
- Convolution layer with filter size 3

Discriminant

Architecture

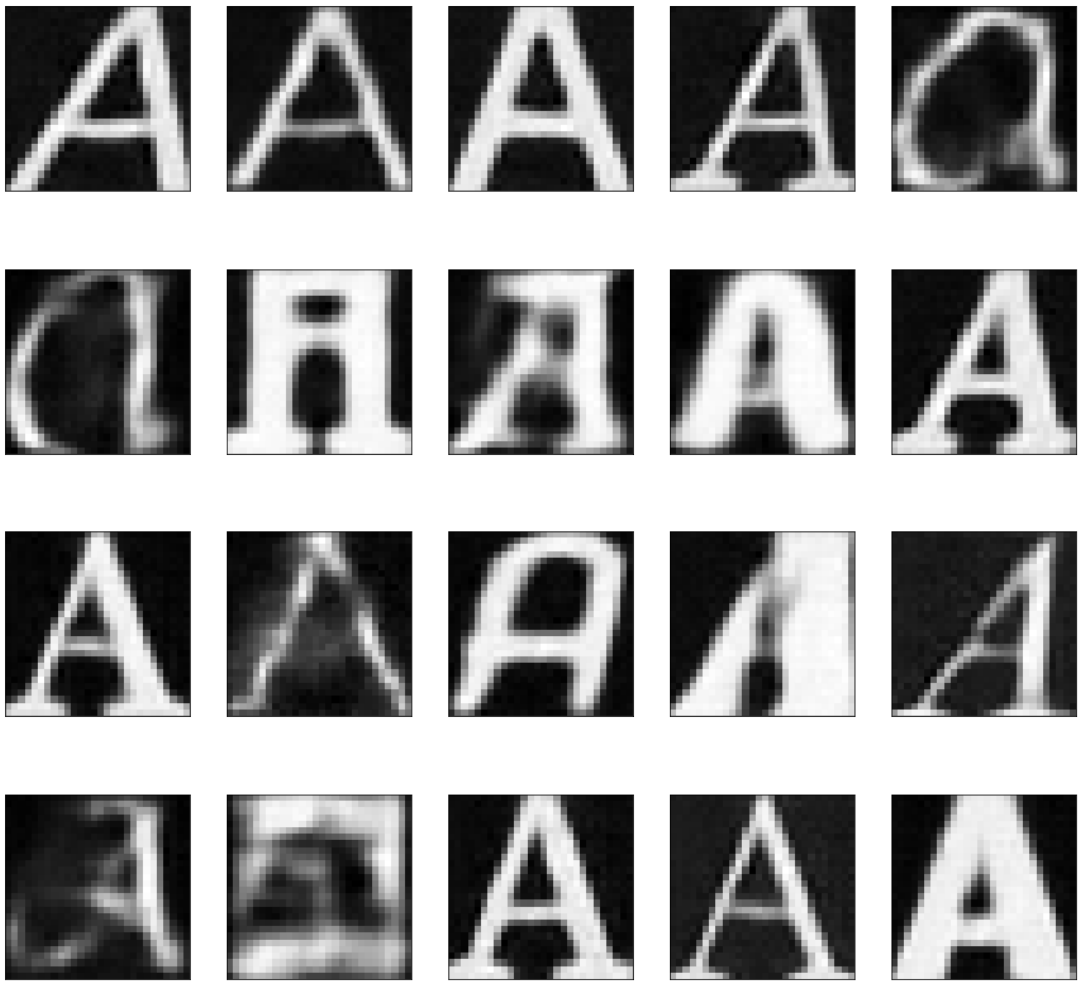
- Input layer for encoded real image
- Combination of 2 Convolution layers and Pooling
- 2 fully connected layers with 128 neurons and tangent function of activation applied to previous layers and letter input correspondingly
- Deconvolution layer with 128 neurons and filter size 5
- Concatenation layer
- 2 fully connected layers with sigmoid function of activation in the last layer

Outline

Results

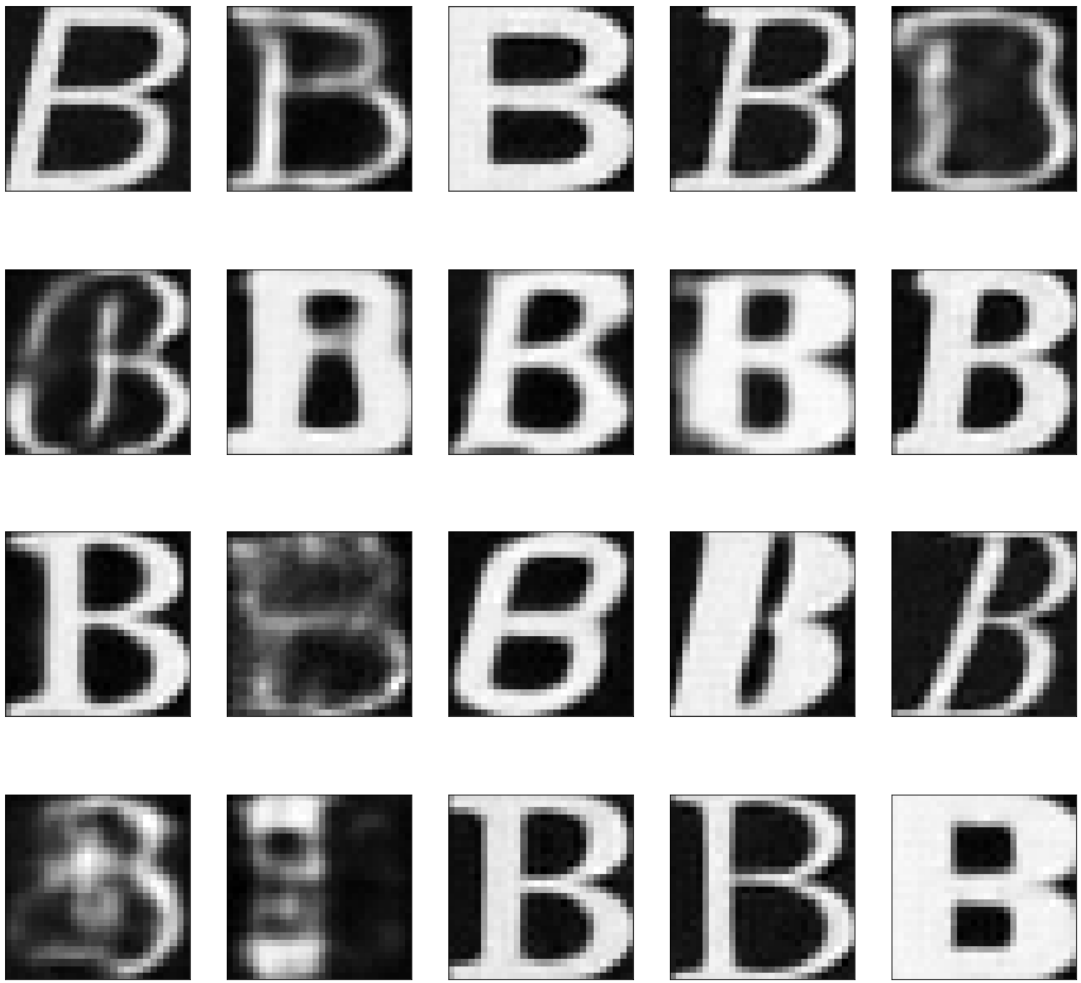
- Generative neural network with l2 loss

Generative neural network with l2 loss



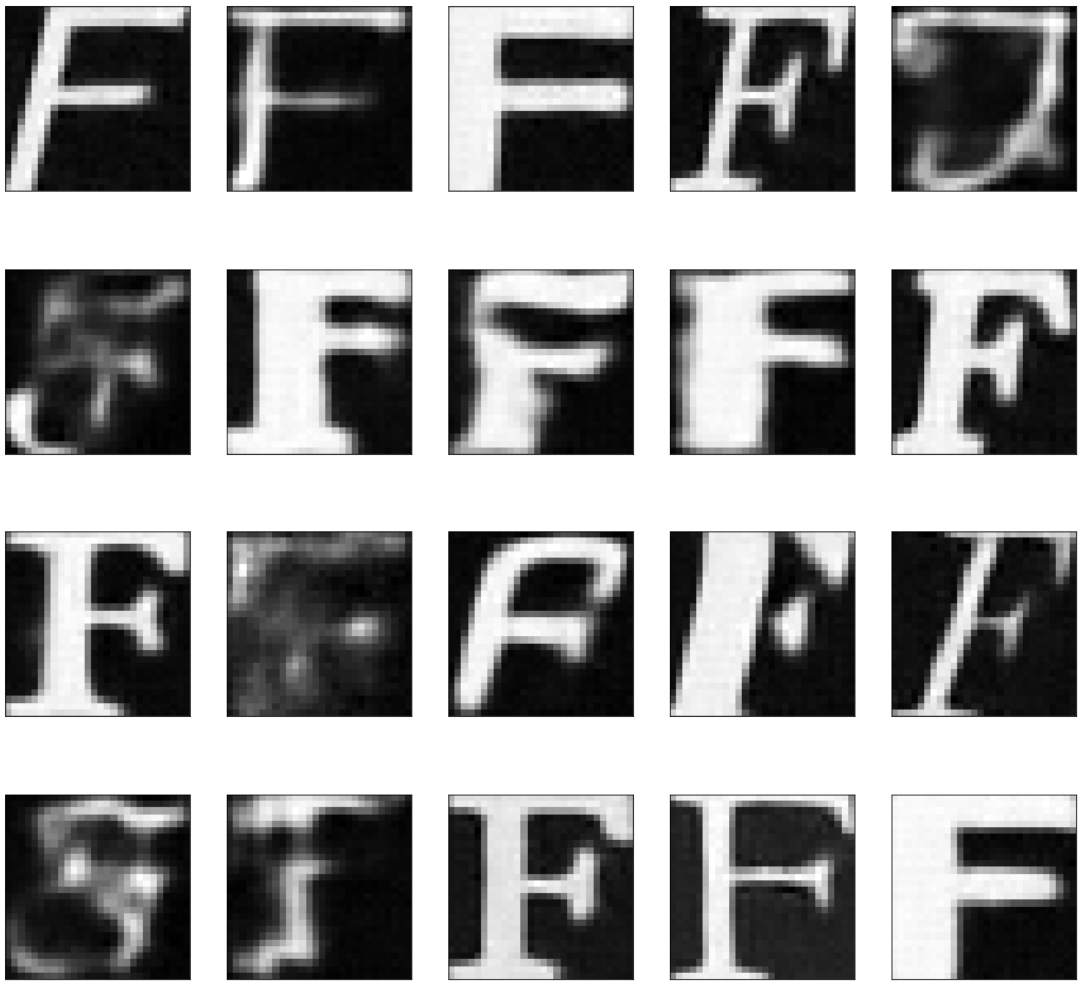
Generation of letter A

Generative neural network with l2 loss



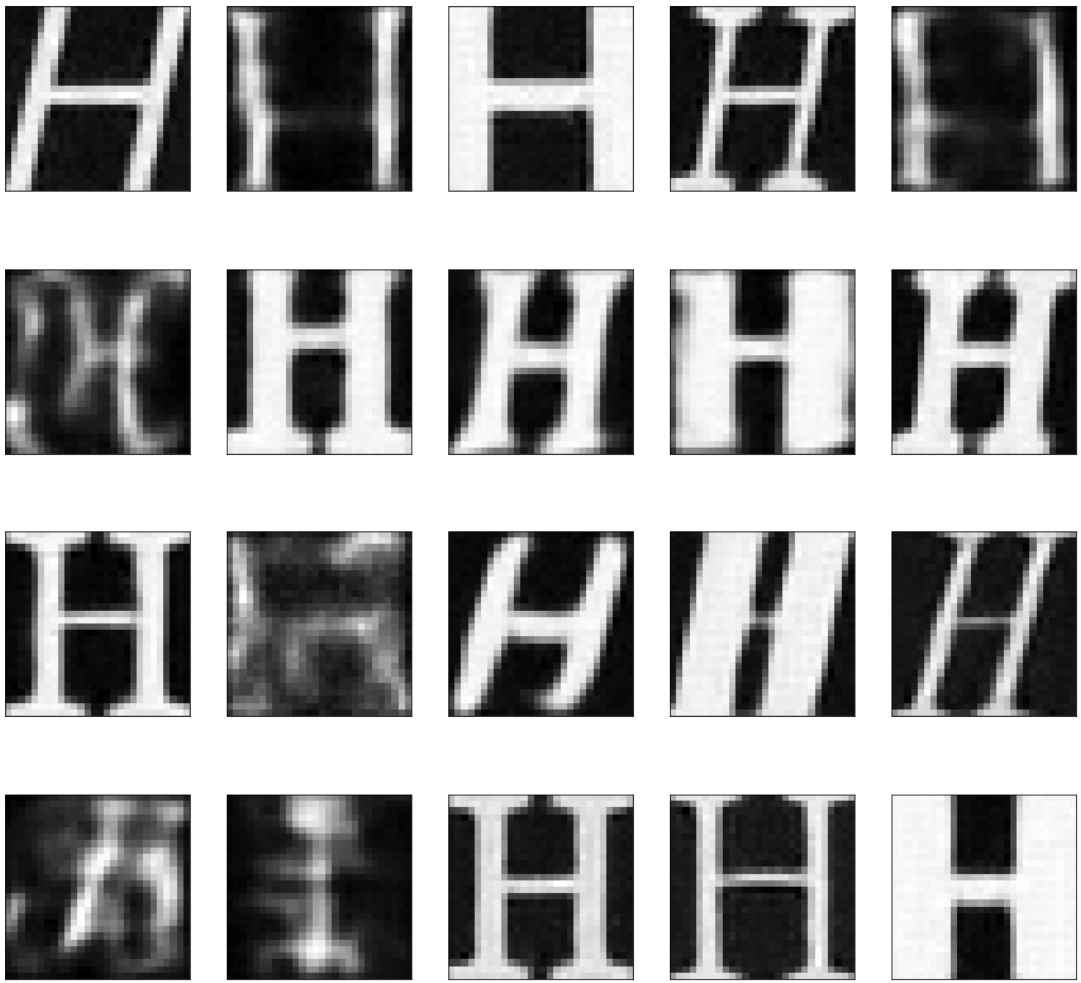
Generation of letter B

Generative neural network with l2 loss



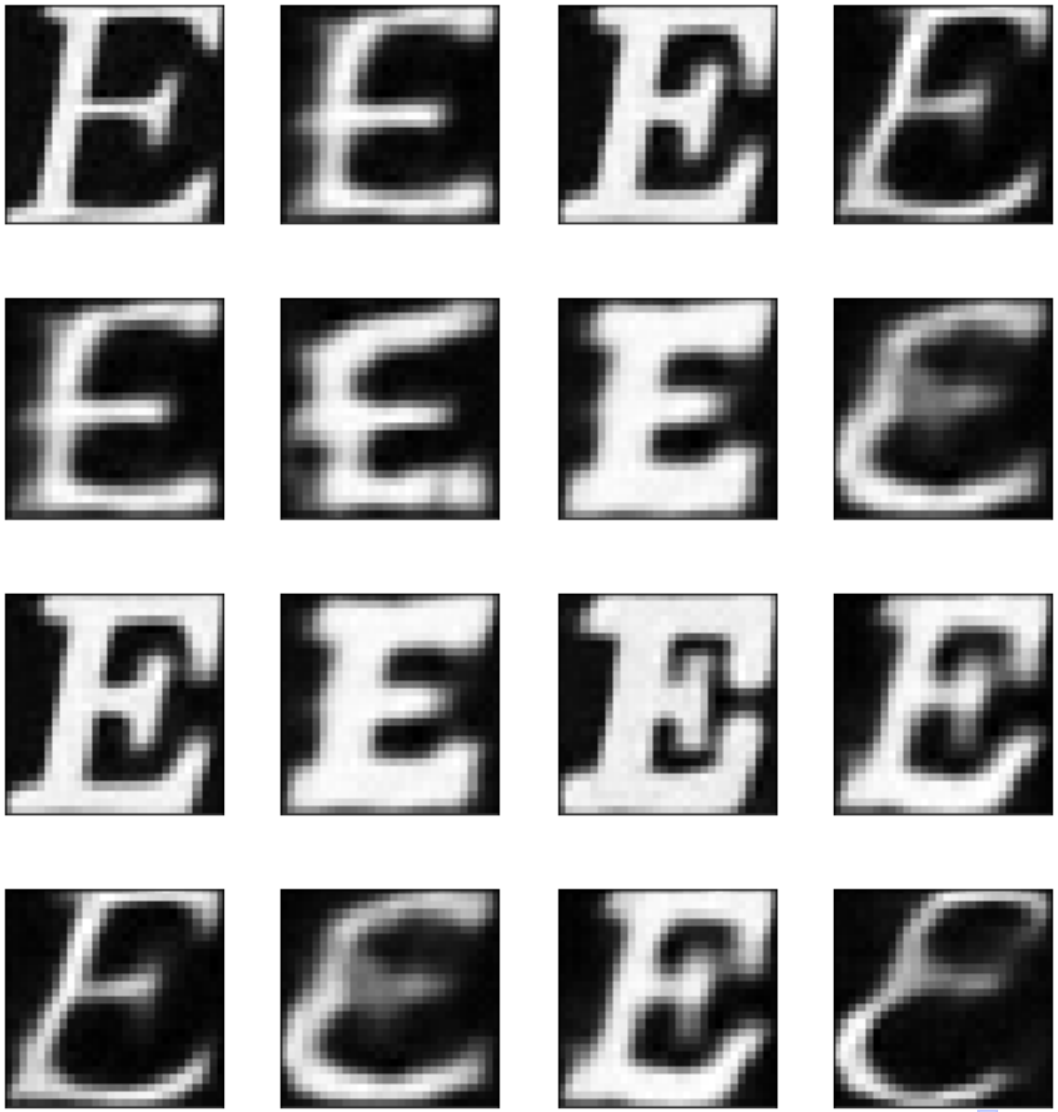
Generation of letter F

Generative neural network with l2 loss



Generation of letter H

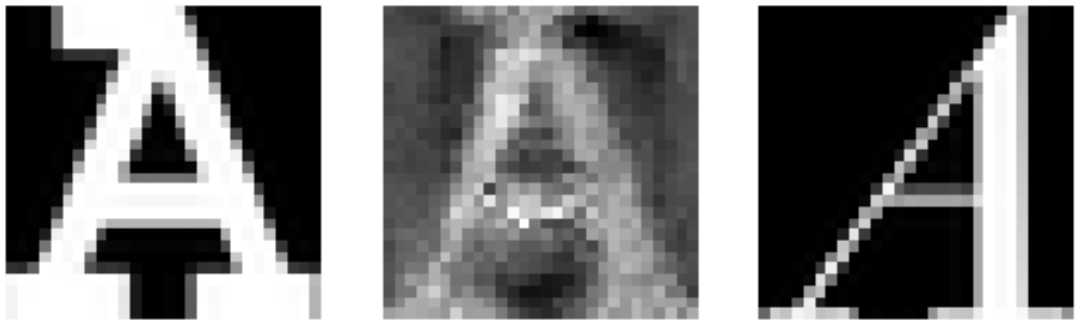
Mixture of fonts



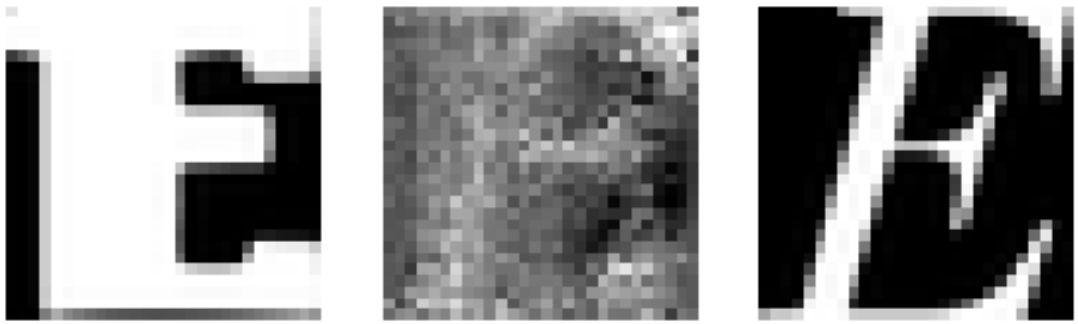
Results

- Variational autoencoder

Transformations using VAE



A transformation

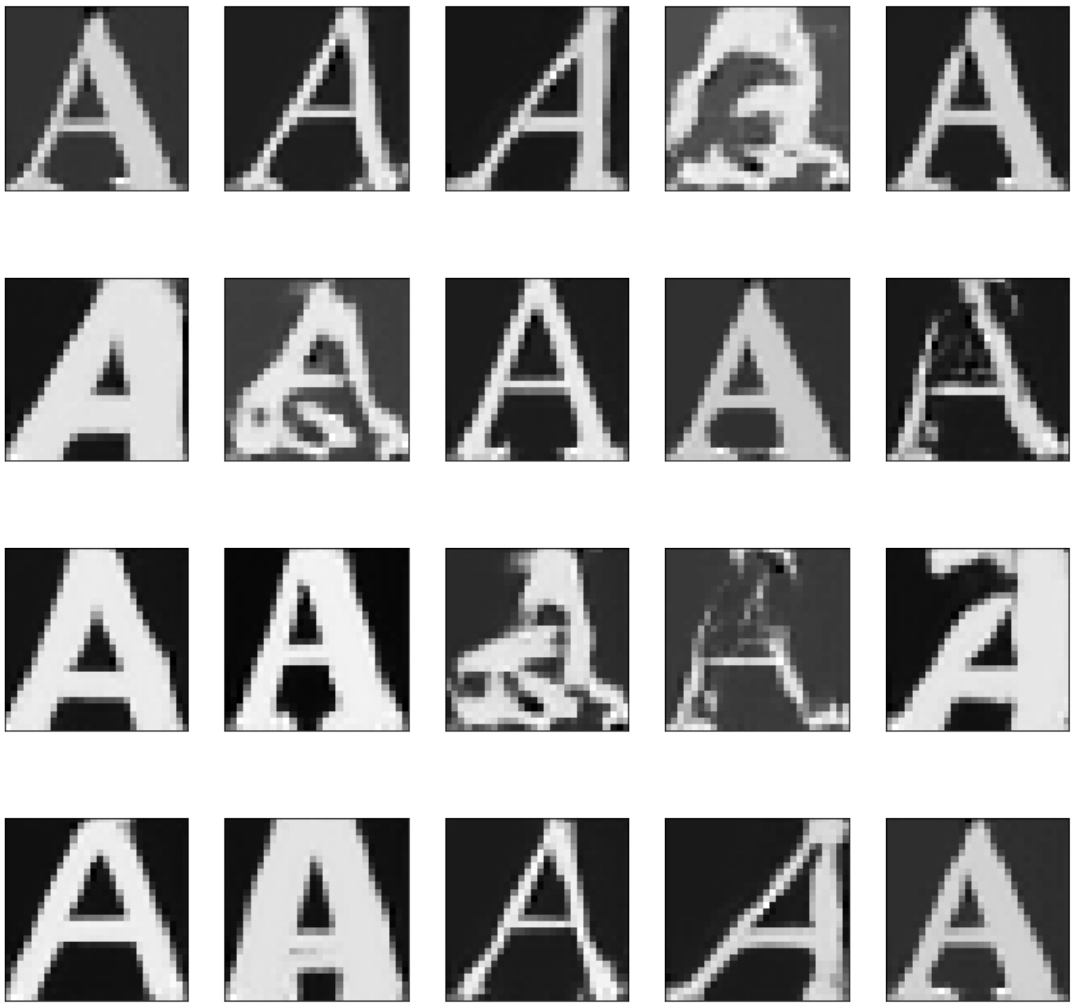


E transformation

Results

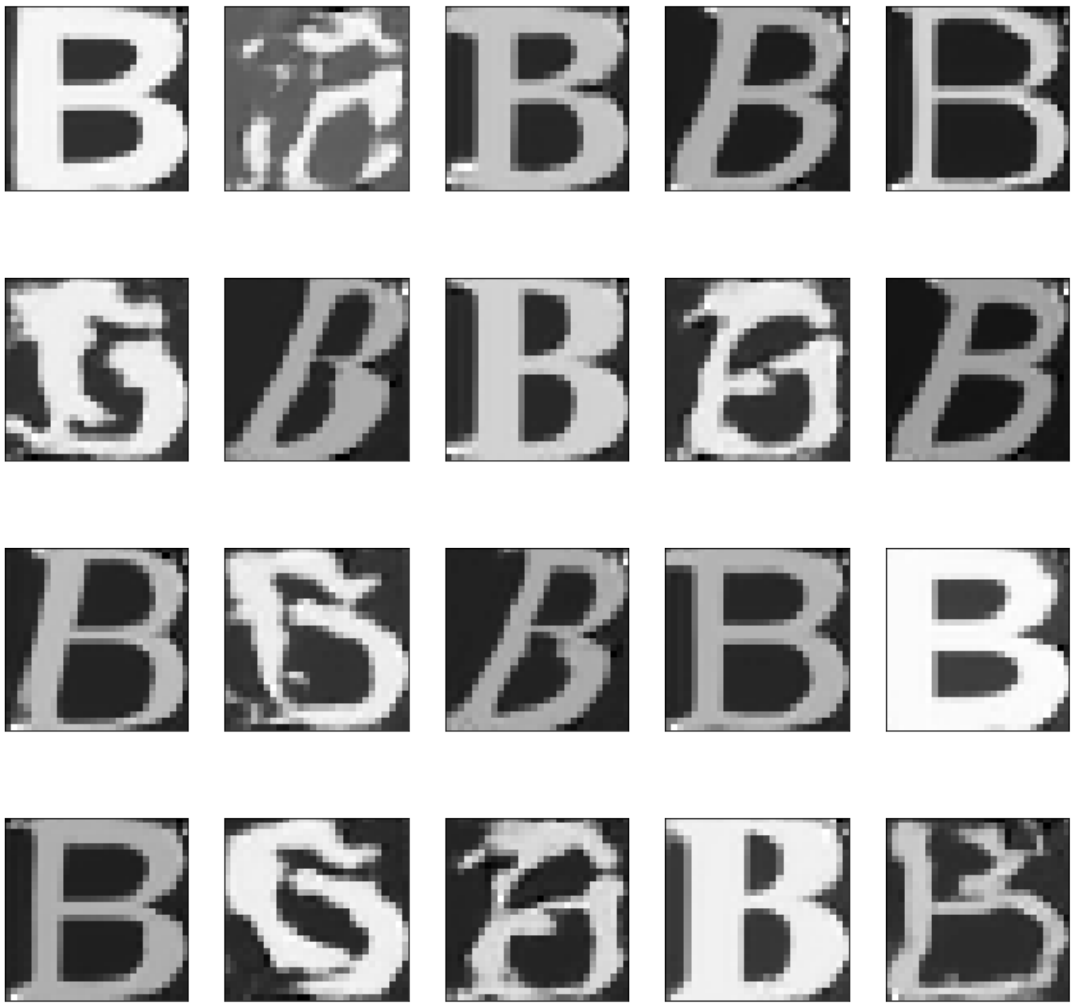
- Generative adversarial networks

Generative adversarial network



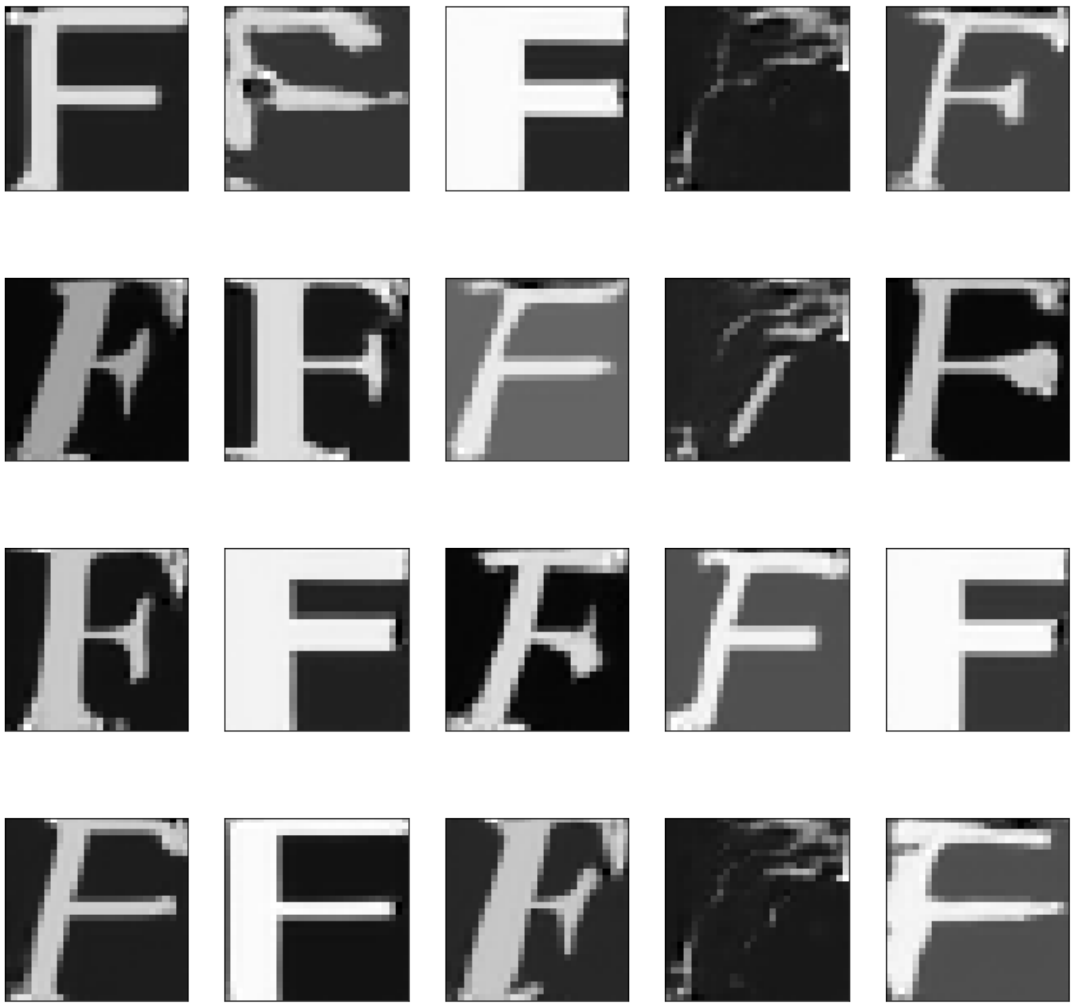
Generation of letter A

Generative adversarial network



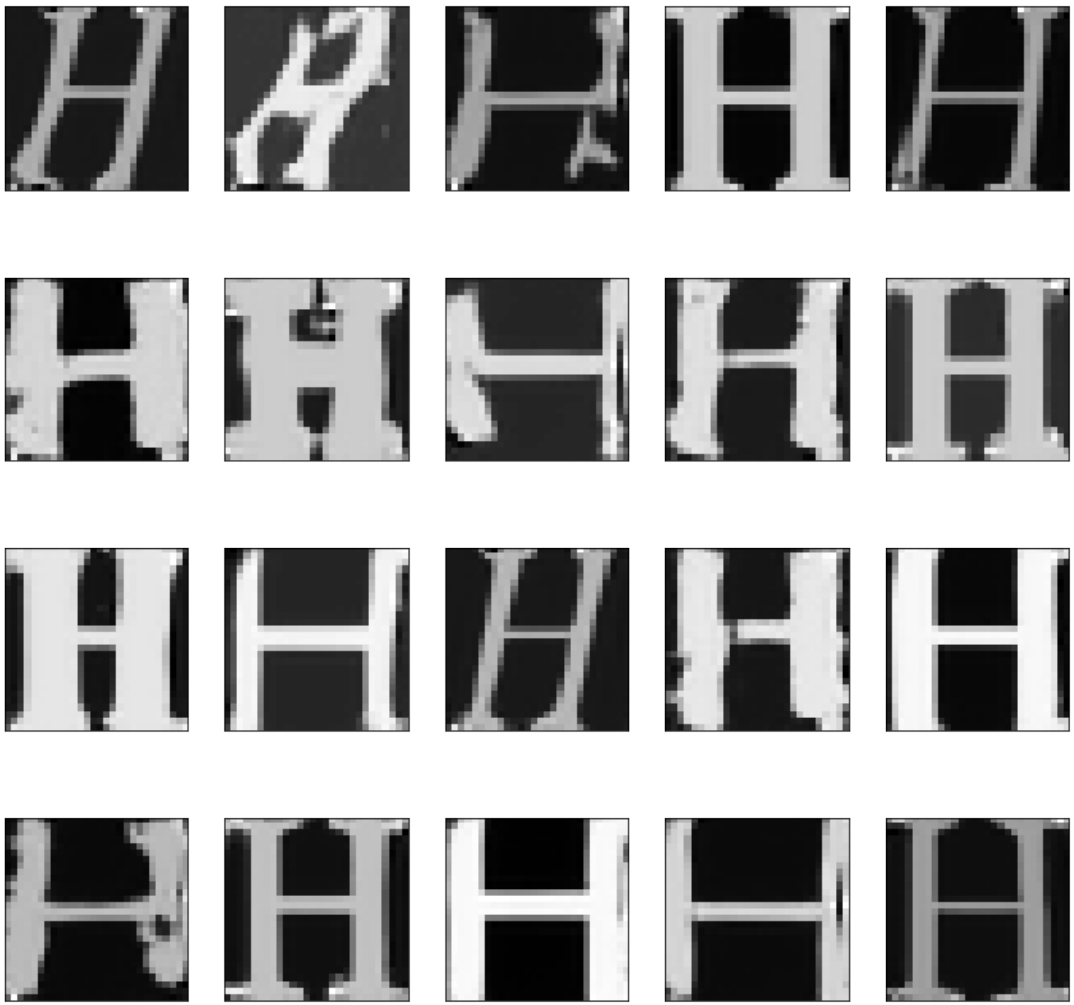
Generation of letter B

Generative adversarial network



Generation of letter F

Generative adversarial network



Generation of letter H

Generative adversarial network

Generative letters VS real letters

Conclusion

- VAE, GAN and other generative neural network with l2-loss were built and trained
- generative neural network with l2-loss was used to adapt a new font style for a fixed letter given some other letters with that style and also to mix different font styles to get a new one
- VAE was tested on ability "to subtract or to add"differnt styles to a fixed letter
- GAN is capable of generating new font styles which are noticeably different from existing ones