Semi-supervised training of CNNs

Student: Kevin Duarte
Mentors: Dr. Boqing Gong
Yang Zhang

VGG16

- Network structure
  - Input(3,96,96) - (64)3c - (64)3c - 2p - (128)3c - (128)3c - 2p - (256)3c - (256)3c - (256)3c - 2p - (512)3c - (512)3c - (512)3c - 2p - (512)3c - (512)3c - (512)3c - 2p - 4096fc -4096fc - 10fc
  - Each convolutional layer is followed by a batch normalization layer
  - Normally the input is (3,224,224), but since I am using the STL10 dataset and training the network using 224x224 images takes a very long time, I switched the input to (3,96,96)
- By training this network on the 5,000 labeled STL10 images, we get an accuracy of 50.30% on the 8,000 test images

VGG16 autoencoder

- Network structure (white text denotes the encoder):
  - Input(3,96,96) - (64)3c - (64)3c - 2p - (128)3c - (128)3c - 2p - (256)3c - (256)3c - (256)3c - 2p - (512)3c - (512)3c - (512)3c - 2p - (512)3c - (512)3c - (512)3c - 2p - (512)3c - (512)3c - (512)3c - 2up - (512)3c - (512)3c - (512)3c - 2up - (256)3c - (256)3c - (256)3c - 2up - (128)3c - (128)3c - 2up - (64)3c - (64)3c - 2up - (3)3c - Output(3,96,96)
  - Once again, batch normalization layers follow every convolutional layer (except for the last layer which has a sigmoid activation function)
  - Intermediate losses are calculated between the output of each pooling layer and the output of the convolutional layer right before the corresponding unpooling layer
- I am currently training two autoencoders:
  - An autoencoder with basic max-pooling and upscaling layers, and one final reconstruction loss function
  - An autoencoder with what-where pooling/unpooling, and multiple intermediate reconstruction losses

VGG16 autoencoder

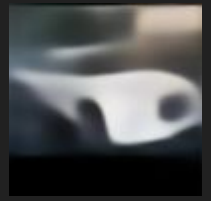
Original

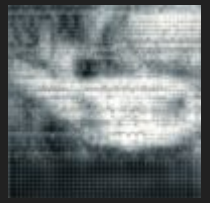
Basic Autoencoder
17 epochs

What-where Autoencoder
1 epoch

What-where Autoencoder
6 epochs

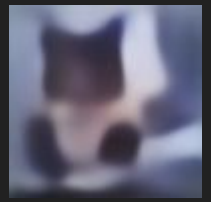


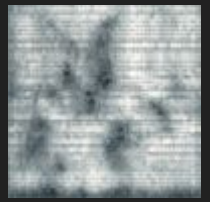














Coming weeks

- The SWWAE seems to have an bug since it only outputs grayscale images, so I will focus on fixing this issue first
- I will continue to train these autoencoders until they sufficiently reproduce the original image
- Then, I will use the autoencoders for semi-supervised training of the vgg16 network