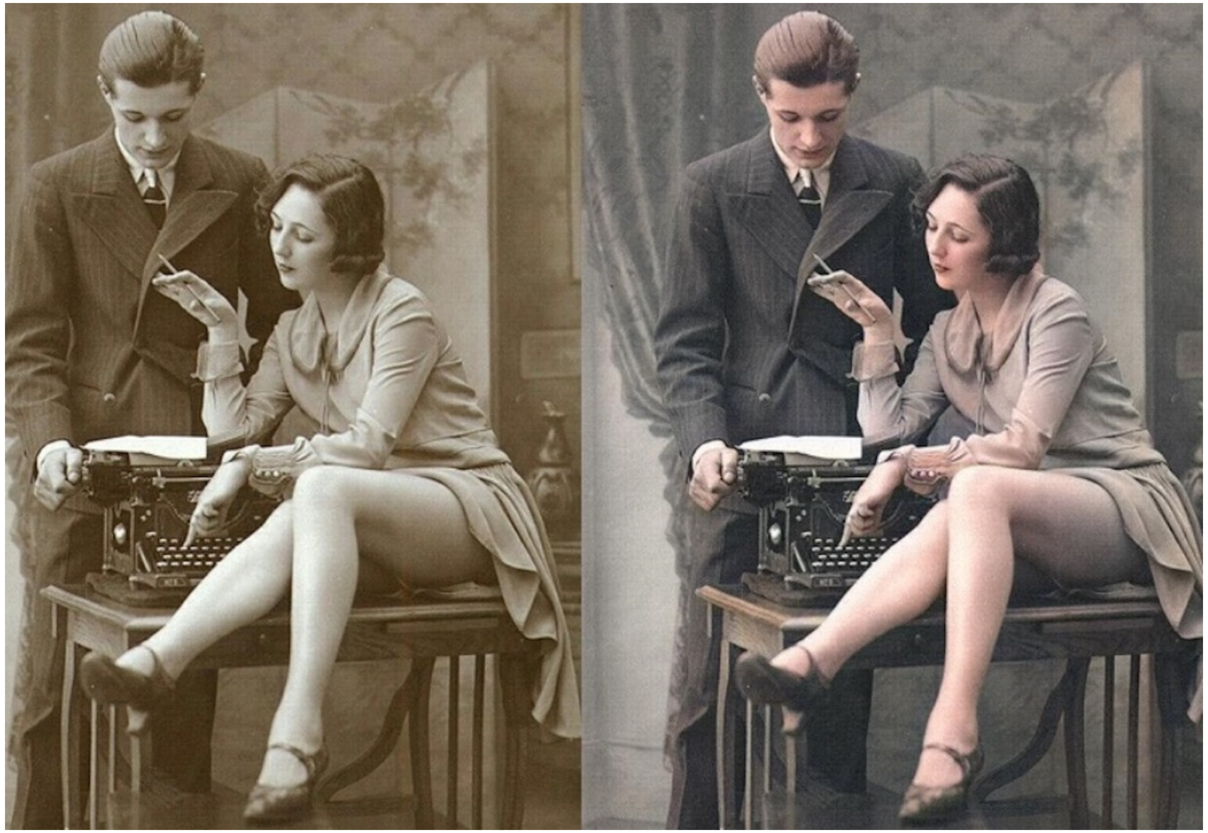
Colorize Grayscale Image

Shu Zhang, Yueqi Mao

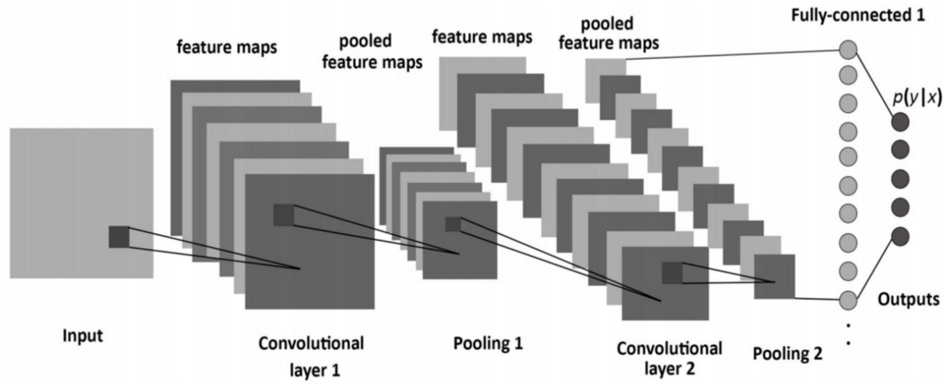
High Level Idea

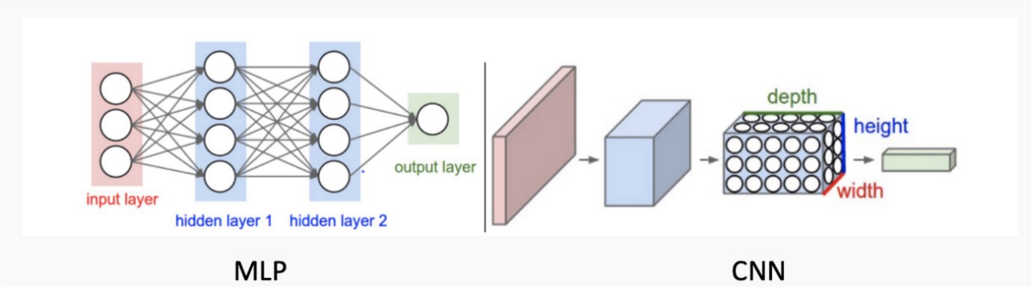
- What is automatic image colorization
- Why do we need automatic image colorization
- What are the challenges
- Our project goal: **Research and evaluate effective grayscale image colorization algorithms based on Convolutional Neural Networks (CNN)**



Model Architecture

- High level introduction to CNN and Generative Adversarial Network (GAN)
- How to apply CNN and GAN to greyscale image colorization





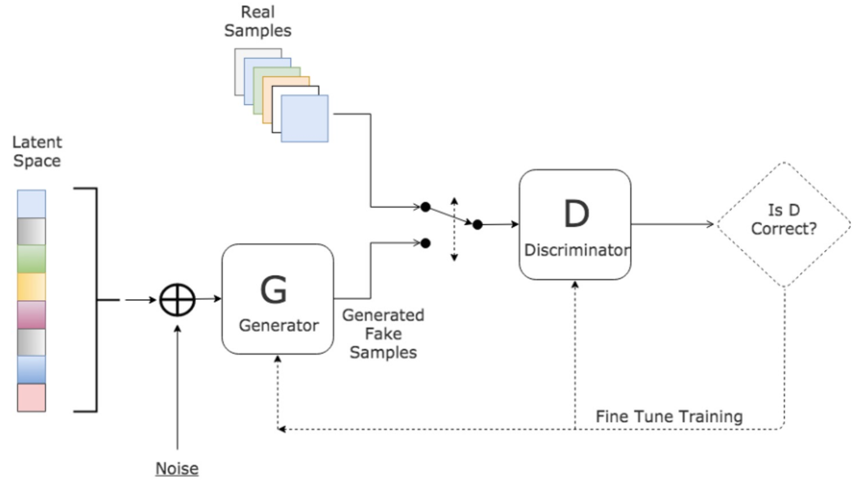
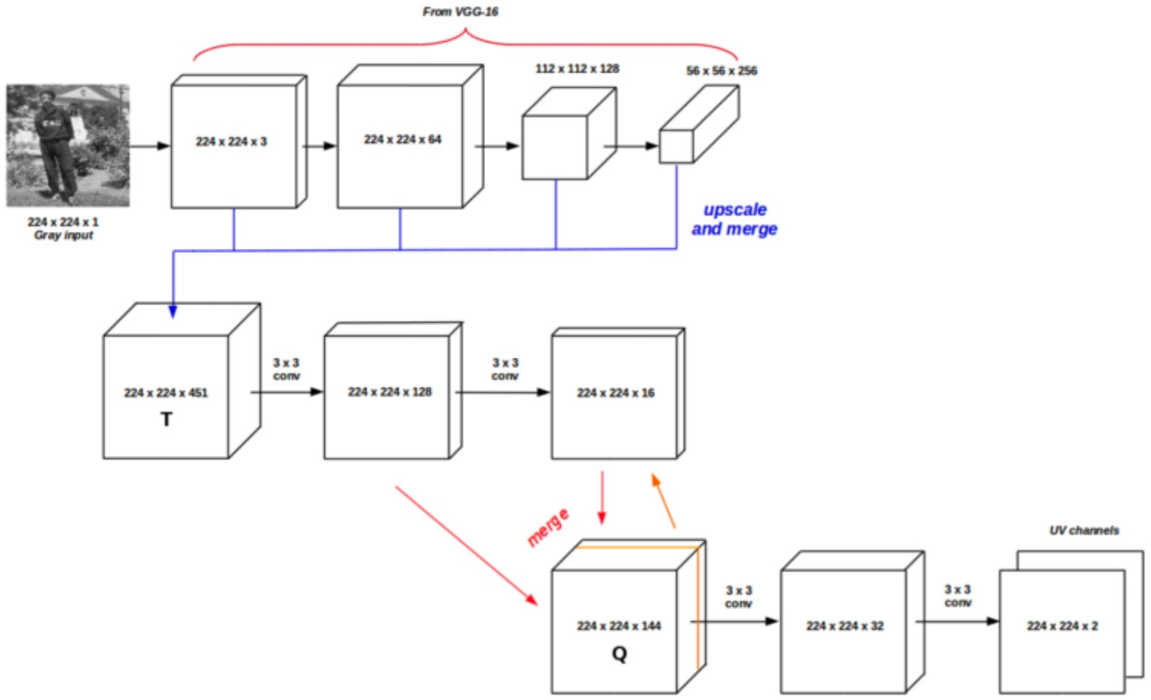
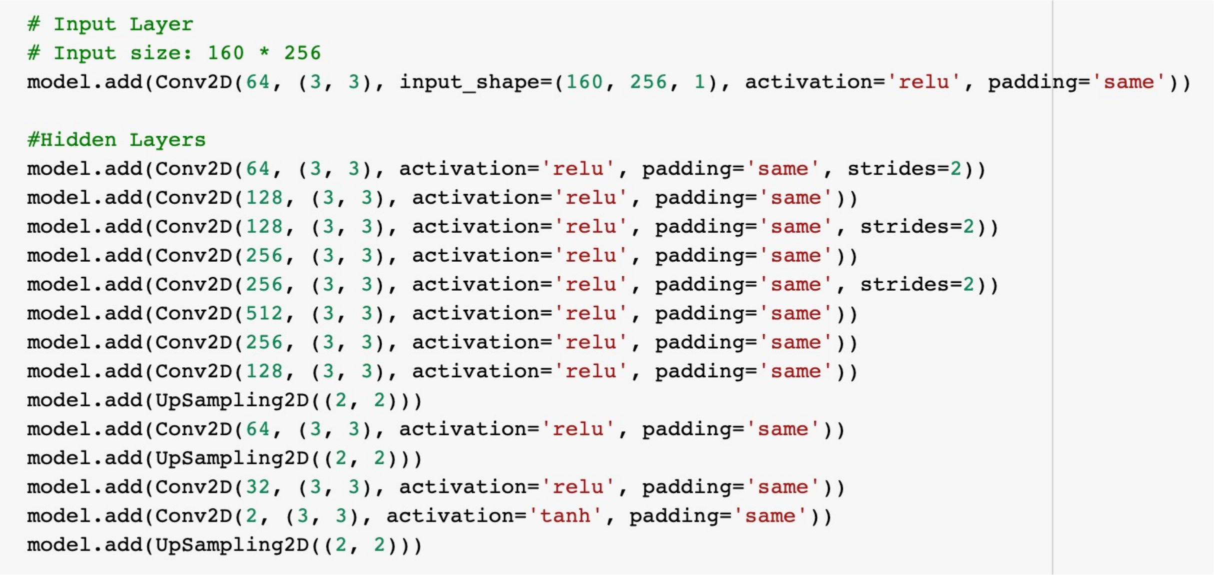


Figure 1. GAN architecture illustration (Hitawala, 2018).





```
# Input Layer
# Input size: 160 * 256
model.add(Conv2D(64, (3, 3), input_shape=(160, 256, 1), activation='relu', padding='same'))

#Hidden Layers
model.add(Conv2D(64, (3, 3), activation='relu', padding='same', strides=2))
model.add(Conv2D(128, (3, 3), activation='relu', padding='same'))
model.add(Conv2D(128, (3, 3), activation='relu', padding='same', strides=2))
model.add(Conv2D(256, (3, 3), activation='relu', padding='same'))
model.add(Conv2D(256, (3, 3), activation='relu', padding='same', strides=2))
model.add(Conv2D(512, (3, 3), activation='relu', padding='same'))
model.add(Conv2D(256, (3, 3), activation='relu', padding='same'))
model.add(Conv2D(128, (3, 3), activation='relu', padding='same'))
model.add(UpSampling2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu', padding='same'))
model.add(UpSampling2D((2, 2)))
model.add(Conv2D(32, (3, 3), activation='relu', padding='same'))
model.add(Conv2D(2, (3, 3), activation='tanh', padding='same'))
model.add(UpSampling2D((2, 2)))
```

- Our CNN model is based on the paper **Fully automatic image colorization based on Convolutional Neural Network** by Domonkos Varga, et al.
- Learning based colorization using CNN model
- Update wrights in the CNN model and make predictions of each U, V channel for every image

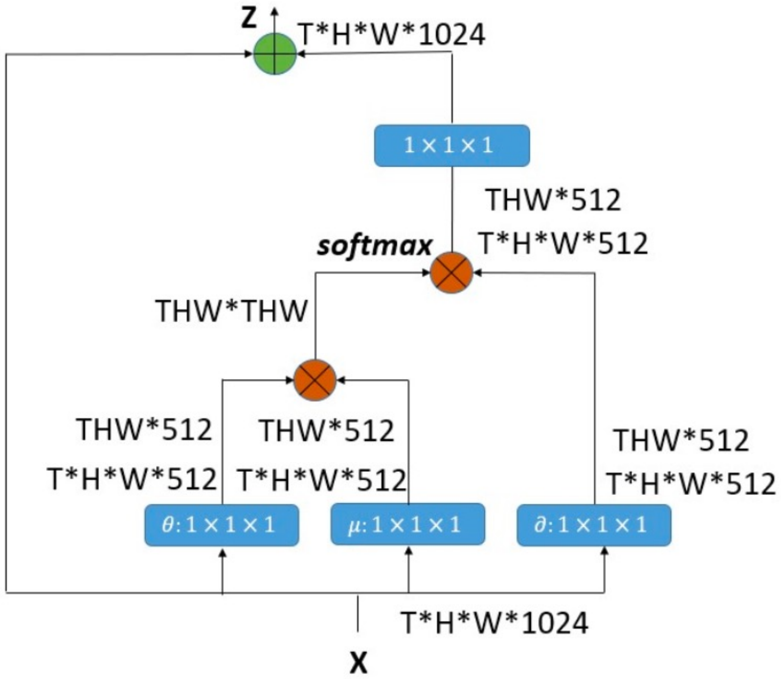


Figure 3. Optimized U-net structure

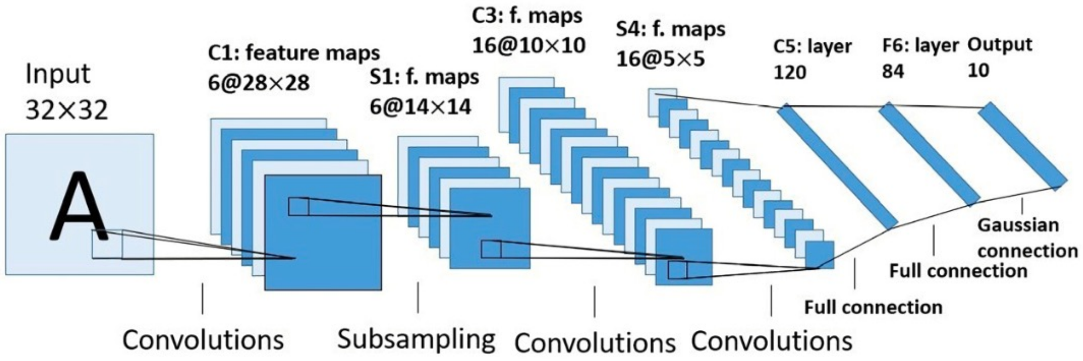
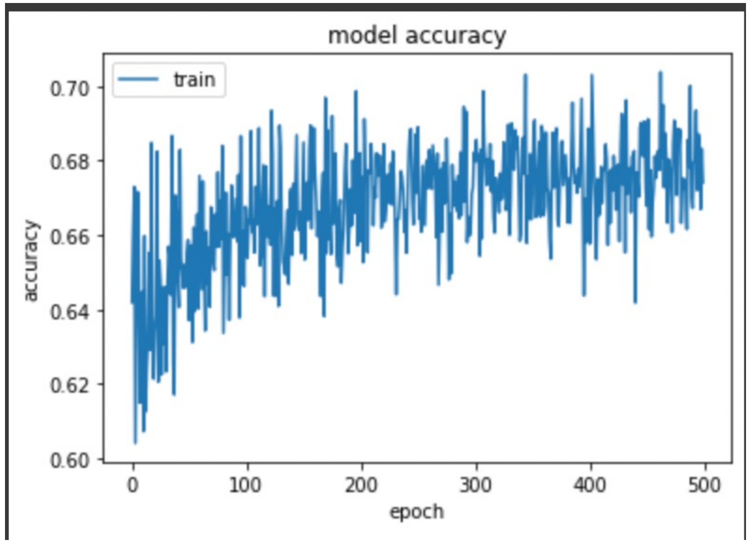
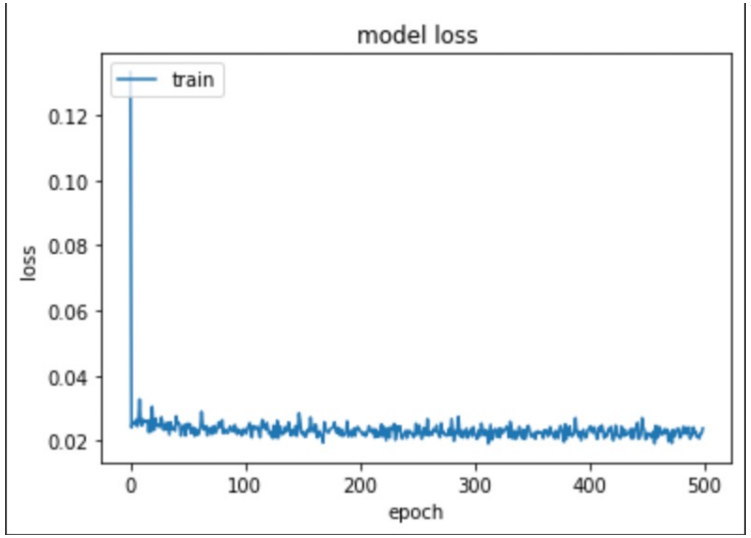


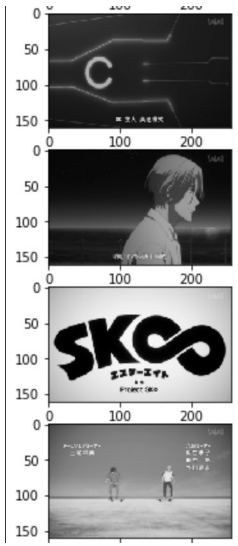
Figure 4. The components of Lenet-5

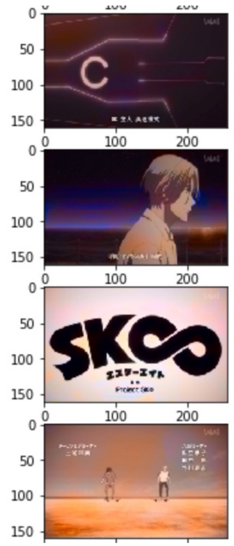
- Our GAN model is based on the paper **Colorization of anime gray images via generative adversarial networks** by Xinyu He, et al.
- Generator G is an optimization of U-Net (which is a type of CNN model)
- Discriminator D is a simple LeNet (which is another type of CNN model)

CNN Training Result (6000 images, 500 rounds)



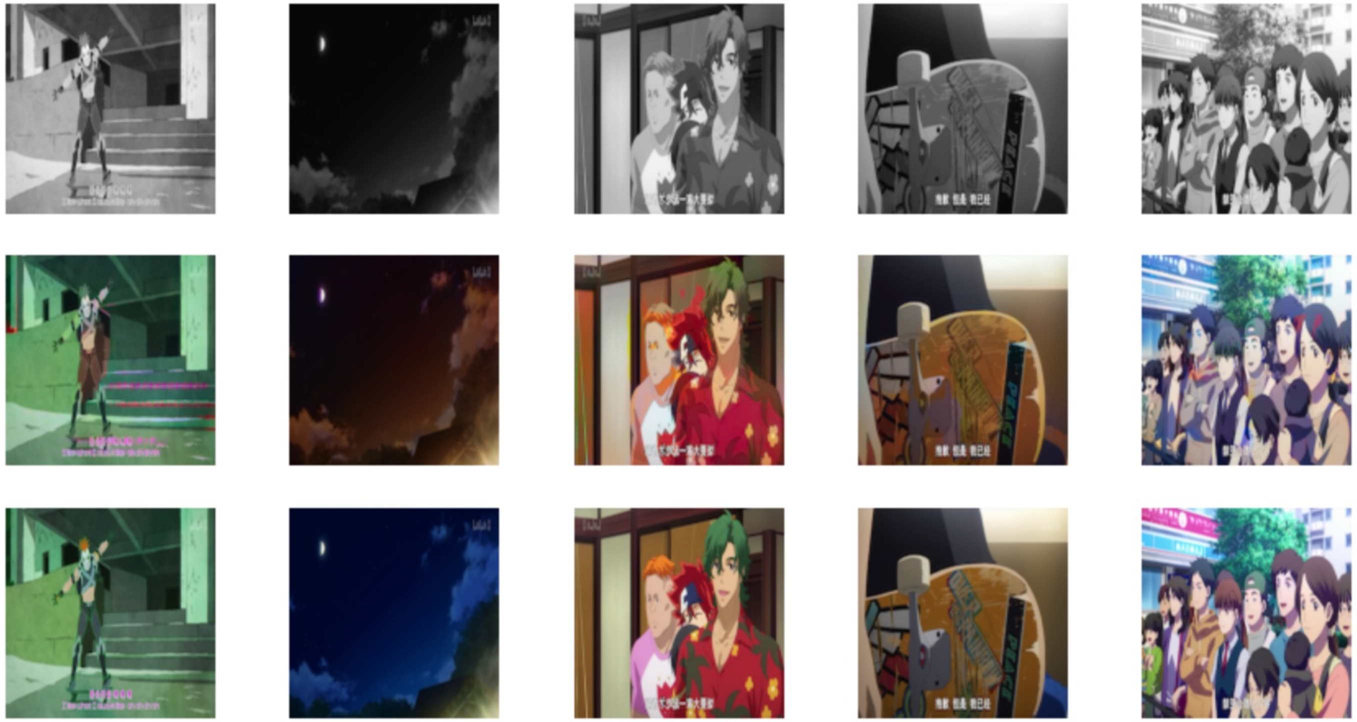






GAN Training Result (6000 images, 100 rounds)

```
Epoch 101/500
Iteration 280/282
loss_D_fake: 0.46282
loss_D_real: 0.45018
loss_D: 0.45650
loss_G_GAN: 1.66006
loss_G_L1: 4.92482
loss_G: 6.58488
```



Conclusion & Future Direction

- A brief summary of animation colorization using CNN based models
- Comparison between using CNN and GAN
- Problems we faced in the training process and the testing results
- Potential improvements
- Future research directions

Thank you!

Reference

- https://www.imcgrupo.com/review-of-the-vance-ai-photo-colorizer-tool/
- https://ieeexplore.ieee.org/document/9574551
- https://ieeexplore.ieee.org/document/7900208
- https://arxiv.org/abs/1406.2661