Individual Report of Final Project

He Huang

Introduction

According to Ledig et al. (2017), image super resolution refers to a task which turns low-resolution images into its high-resolution vision. Before deep neural networks, people use several traditional ways to do this task, including inter nearest method and other methods. After the development of computer vision, more and more deep neural network methods are created to implement image super resolution task. In this project, we first used traditional methods to check their performance, then built CNN network to test how deep neural networks perform on image super resolution. Then, we used autoencoder and GAN type model to help us implement image super resolution.

We separated the whole task into 4 parts, cv2 part, CNN model part, autoencoder part and GAN part. Li mainly took charge of the CNN model part; Liang mainly took charge of both cv2 part and autoencoder part.

Overall Description

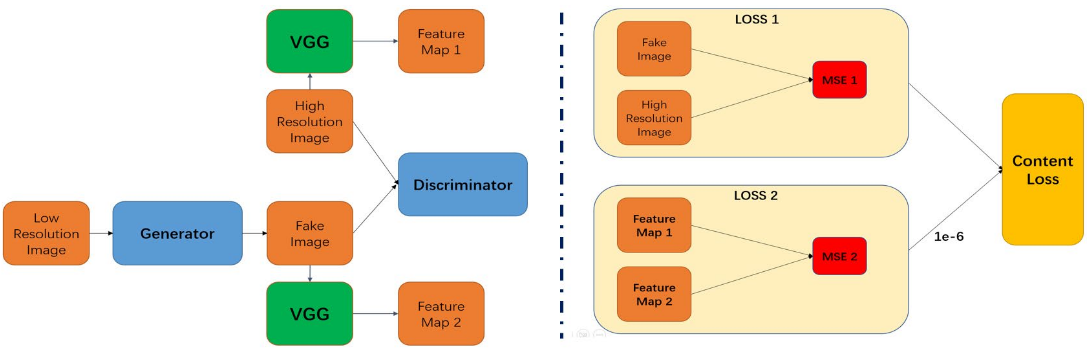
In coding section, I helped Liang to build and train the autoencoder model. Then, I built and trained the SRGAN model. In report section, after my teammates finished their part, I put all these 4 parts together and wrote the introduction, results and conclusion section.

Details in coding and training

After using autoencoder, I tried to use GAN on image super resolution task. After some research, I found that the first GAN model for image super resolution task is Super Resolution GAN(SRGAN). According to Ledig et al. (2017), the basic model and the calculation of loss function for SRGAN is shown in figure below. The basic structure is just like the normal GAN. The generator generates fake image with low resolution images, then the model feeds both fake images and high-resolution images to the discriminator to detect which image is real. The biggest difference is that the loss calculation of SRGAN. Not like other GAN only calculating MSE between fake images and high-resolution images (Loss 1), the SRGAN also uses VGG19 to create feature map with the fake images and calculate the MSE between the feature map between the low-resolution images and the high-resolution images (Loss 2). The reason why SRGAN adds loss 2 to total loss is that for image super resolution, low-resolution images and high-resolution images are strictly equivalent in shape and position, and all I need to do is to add more texture on the low-resolution images, which can be exactly shown in feature maps in CNN models.

Figure

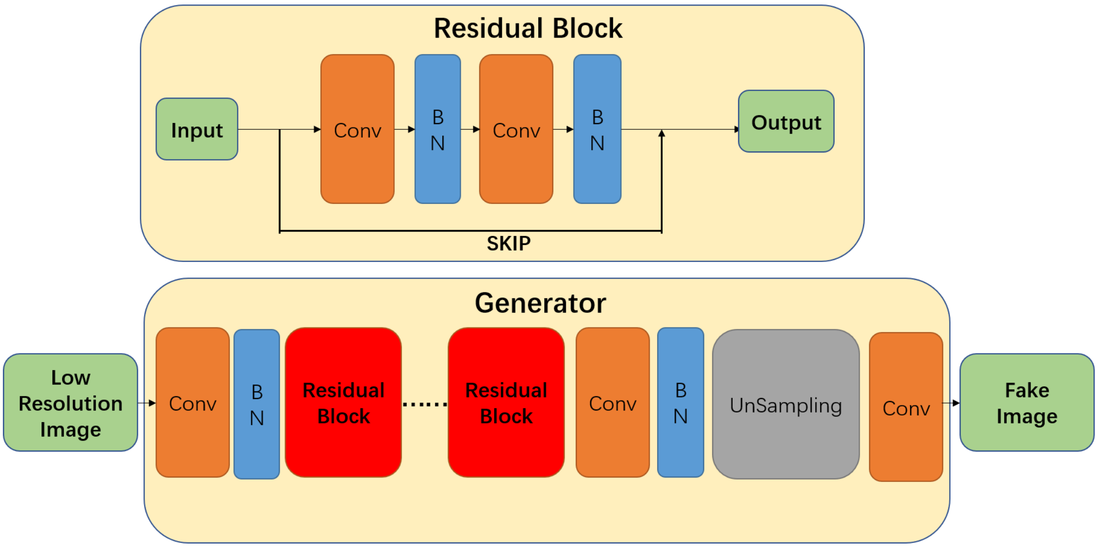
Model Structure and Loss Calculation for Generator



I built our own SRGAN model with the Generator shown below. All the activation function in this generator is PRelu function. In the SRGAN model I designed, there are total 8 residual blocks. For the discriminator, it is a simple CNN image binary classification model with LeakyReLU as activation function.

Figure

Structure of Generator



For training, I first freeze generator and train the discriminator, then unfreeze generator and freeze discriminator to train the generator. Every 5 epoch, I set the model output a prediction image (fake image) that the generator generates. The changes of the prediction images are shown below. From the figure, at the first several epochs, the prediction images generated by the generator only has the same shape as the original images, and the color is totally incorrect. However, after more training epochs, generator has learned how to get correct colors and add more details on the image, in other words, add more image texture that low resolution image doesn't have.

Figure

Changes of prediction image

Actually, except SRGAN, there are some other GAN models for image super resolution task. ESRGAN(Enhanced Super Resolution GAN) changes the structures and loss calculation of SRGAN to have a very good performance. However, building ESRGAN models and training them are beyond our capabilities. I didn't find how to build this model line by line. Also, new vision ESRGAN, real-ESRGAN, are constantly being updated. Now, the author is improving its performance on comic image.

For GAN models, I built and trained our own SRGAN model, and got very good performance on image super resolution. However, in the training, I found that GAN models always generate some random influences on image color or item position. Those random influences make the prediction image not like the high-resolution vision of the original low-resolution image, but like a brand-new image with high-resolution which is only influenced by the original image. Maybe GAN type models need to be adjusted to reduce this kind of random.

Results and conclusions

To sum up, SRGAN works really well on image super resolution tasks. However, there are still some questions in GAN type model. However, there are so many random influences occurred in generator. For me, GAN is more like creating a total new image having the same style, shape, and position of original low-resolution image, but not like increasing the resolution of the original image.

For future improvement, using more different image type will help GAN to handle more complex situation, including paintings and comic images. Also, image augmentation may help the models to handle complex environment, including dark environments.

Percentage

the percentage of the code that you found or copied from the internet:

(160-10)/(160+60) = 68.18%

Reference

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., ... & Shi, W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4681-4690).

Singh, S. (2022, November). *Image-Super-Resolution*. Kaggle. https://www.kaggle.com/code/shivam2111/image-super-resolution