Single Image Super Resolution Using GANs

Shaoyu Feng, Mengtong Zhang, Weile Chen

Georgetown University, School of Art and Science, Analytics

Abstract

Image super resolution can be defined as increasing the size of small images while keeping the drop in quality to minimum, or restoring high resolution images from rich details obtained from low resolution images. Traditional methods will yield distorted image. In this project, we explore the use of Super Resolution Generative Adversary Network(GAN) to perform the image restoration. SRGAN propose a perceptual loss function which consists of an adversarial loss and a content loss. The adversarial loss pushes the solution to the natural image manifold using a discriminator network that is trained to differentiate between the super-resolved images and original photo-realistic images. In addition, it use a content loss motivated by perceptual similarity instead of similarity in pixel space. The deep residual network is able to recover photo-realistic textures from heavily downsampled images on public benchmarks.

Introduction

Image super resolution can be defined as increasing the size of small images while keeping the drop in quality to minimum, or restoring high resolution images from rich details obtained from low resolution images. This problem is quite complex since there exist multiple solutions for a given low resolution image. Image restoration/enhancement has numerous applications like satellite image analysis, video/movie enhancement, crime scene analysis etc. It is the process of recover/restore high resolution image from low resolution image while keeping all details. This can be addresses using traditional image analytics like noise deduction, image dilation/erosion, image up-scaling and color adjustment. This is easy to use but this leads to distorted image or reduces the visual quality of image. Most common interpolation methods produce blurry images. In this project, we will explore the use of adversarial network (Generative Adversarial Networks) to produce high resolutions images. The major objective is to reconstruct high resolution image by up-scaling low resolution image such that texture detail in the reconstructed images is not lost.

Generative Adversary Network (GANs) : GANs are class of AI algorithms used in Unsupervised Machine Learning. GANs are deep neural network architectures comprised of two networks (Generator and Discriminator) pitting one against the other (thus the “adversarial”). GANs is about creating, like drawing a portrait or composing a symphony. The main focus for GANs is to generate data from scratch. Super-resolution GAN applies a deep network in combination with an adversary network to produce higher resolution images.

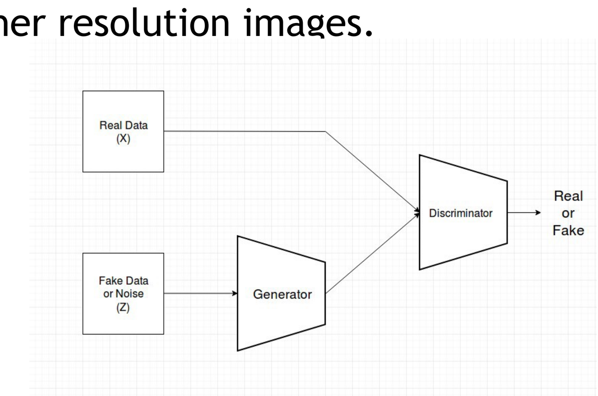

Figure 1 : GAN Architecture

Discriminator and Generator are both learning at the same time, and once Generator is trained it knows enough about the distribution of the training samples so that it can now generate new samples which share very similar properties.

SRGAN Model Architecture

SRGAN : Super-resolution GAN applies a deep network in combination with an adversary network to produce higher resolution images. Training procedure is shown in following steps:

- We process the HR(High Resolution) images to get down-sampled LR(Low Resolution) images. Now we have both HR and LR images for training data set.
- We pass LR images through Generator which up-samples and gives SR(Super Resolution) images.
- We use a discriminator to distinguish the HR images and back-propagate the GAN loss to train the discriminator and the generator.

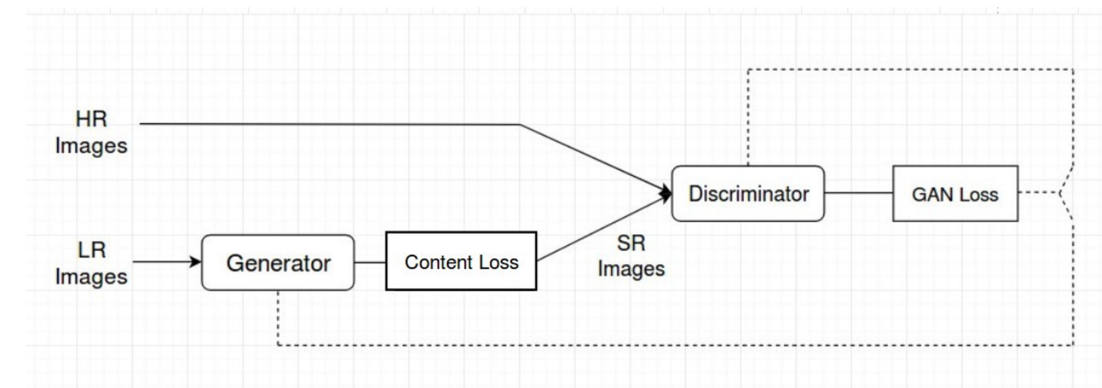

Figure 2 : SRGAN Architecture

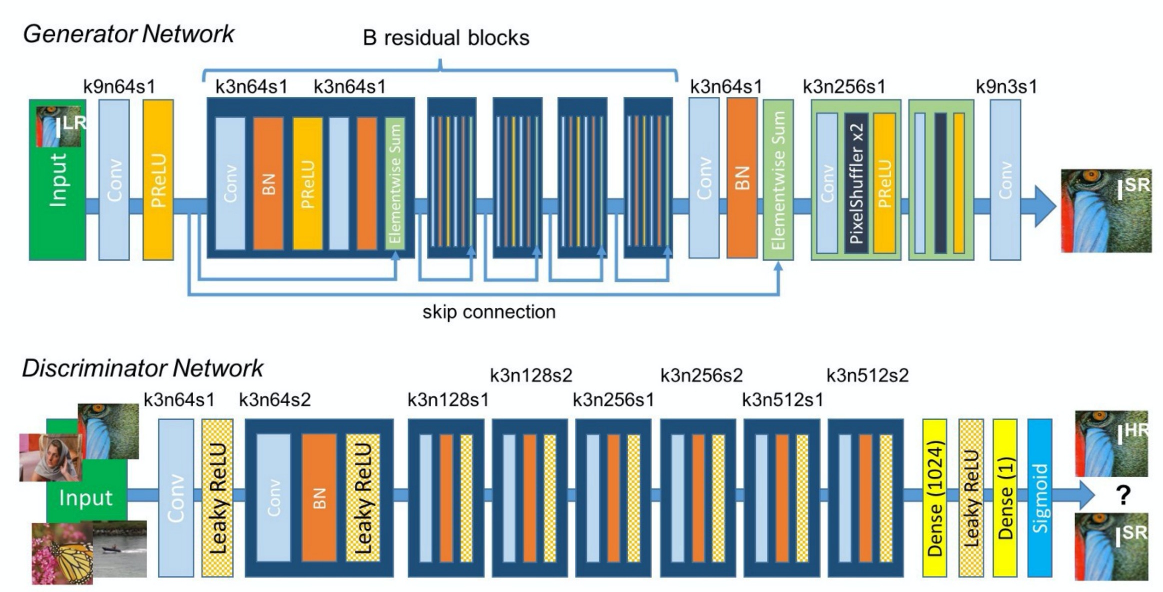

Figure 3: Generator and Discriminator Architecture

- Residual blocks: Since deeper networks are more difficult to train. The residual learning framework eases the training of these networks, and enables them to be substantially deeper, leading to improved performance. 16 residual blocks are used in Generator.
- PixelShuffler x2: This is feature map upscaling. 2 sub-pixel CNN are used in Generator.
- PRelu(Parameterized Relu): We are using PRelu in place of Relu or LeakyRelu. It introduces learn-able parameter that makes it possible to adaptively learn the negative part coefficient.
- k3n64s1 this means kernel 3, channels 64 and strides 1.

Data

The training data are collected and sampled from VOC 2012 dataset. There are about 16700 training images and 425 validation images.

The test image are sampled from common benchmark image data sets including Set 5 | Bevilacqua et al. BMVC 2012 | Set 14 | Zeyde et al. LNCS 2010 | BSD 100 | Martin et al. ICCV 2001 | Urban 100 | Huang et al. CVPR 2015.

The Loss Function

Loss Function: Perceptual loss is used in training. It comprises of Content(Reconstruction) loss and Adversarial loss.

- Adversarial loss: This pushes our solution to the natural image manifold using a discriminator network that is trained to differentiate between the super-resolved images and original photo-realistic images.
- Content Loss: Content loss is used to keep perceptual similarity instead of pixel wise similarity. This will allow us to recover photo-realistic textures from heavily down sampled images. Instead of relying on pixel-wise losses we will and use a loss function that is closer to perceptual similarity.

Results and Discussion

The model is being trained on Amazon-Web-Service(AWS) using a p2.xlarge instance, which means that we are training the model on a Single Nvidia K80 GPU. The entire implementation is written in PyTorch.

For every training image, we will firstly generate the low resolution image by using the RandomCrop function in Pytorch with a fixed image size of 88 pixels. And the batch size for training is 128, which takes about 6 min 30 seconds for each epoch on GPU instance. The default setting for number of epochs is 100.

Another important parameter is called upscaling factor, which describe the magnitude of the super resolution. The default setting is 4 in this project. Following shows the results for the test data sampled from benchmark datasets(sequence: LR Image, HR Image, SR Image).

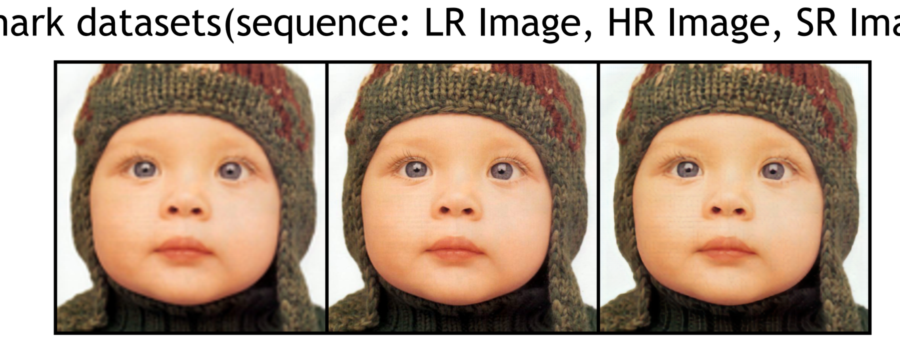

Figure 4: Set5_001_psnr_30.9192_ssim_0.8442

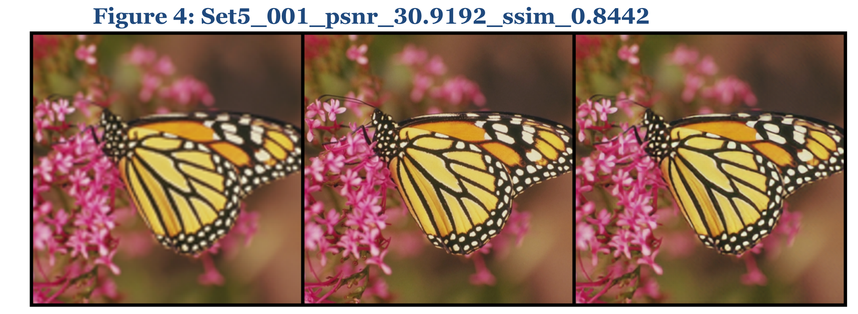

Figure 5: Set14_011_psnr_29.2762_ssim_0.9010

Figure 6: Urban100_075_psnr_28.0906_ssim_0.8191

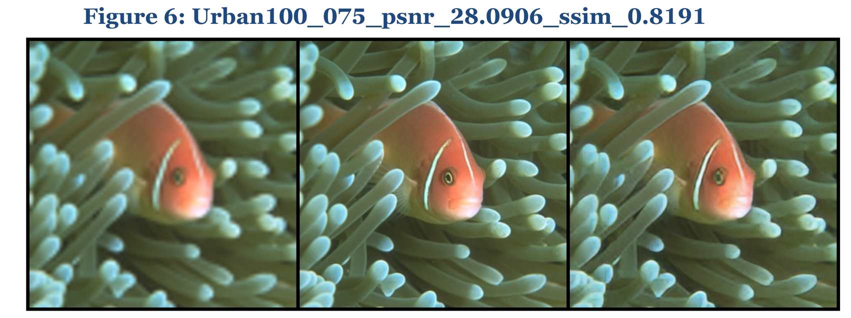

Figure 7: BSD100_040_psnr_31.2101_ssim_0.8991

Results and Discussion(cont’d)

As shown above, we can see that the SRGAN model is able to recover photo-realistic texture from low-resolution images. However, we can still observe blueness if we zoom into certain details.

Two major metrics **structural similarity** (**SSIM**) index and **Peak signal-to-noise ratio(PSNR)** are used as quantitative metric in measuring the model performance. Following table shows the model performance against baseline methods .

SET 5	nearest	bicubic	SRGAN
PSNR	26.26	28.43	28.54
SSIM	0.7552	0.8221	0.8263

SET 14	nearest	bicubic	SRGAN
PSNR	24.64	25.99	25.45
SSIM	0.7100	0.7486	0.7369

BSD 100	nearest	bicubic	SRGAN
PSNR	25.02	25.94	25.61
SSIM	0.6606	0.6935	0.6926

As shown above, we can see that SRGAN model performance is comparable with baseline model, which suggests that either model parameter is not optimal or the model architecture is not perfect. Due to computational constraint, we are not able to extensively optimize the training parameter, this is a future work we would like to continue exploring on.

Conclusions

In this project, we implemented a SRGAN deep neural network for restoring high resolution image from heavily downsampled images. The model features in using perceptual loss to measure the perceptual similarity between original image and trained image. The deep residual network is able to recover photo-realistic textures from heavily downsampled images on public benchmarks. However, we are not seeing significant difference in SSIM and PSNR as compared with baseline model.

The future work includes hyperparameter optimization, model architecture refinement and etc.

Reference

[1] C. Ledig, L. Theis, F. Husza´r, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a gener- ative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016

[2] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. 2016

[3] Single Image Super Resolution Using GANs — Keras https://medium.com/@birla.deepak26/single-image-super-resolution-using-gans-keras-aca310f33112