Super Resolution with Generative Adversarial Networks

Bobby Lv, Yaozhong Liu, Sherine Zhang, Hansi Zeng, Guangfei Zhu
Department of Computer Sciences, Department of Mathematics, Department of Mechanical Engineering
University of Wisconsin-Madison
{zlv7, yliu755, xzhang662, hzeng27, guangfei}@wisc.edu

Abstract

Recently, GANs can be widely used in image generation, feature extraction, image recovery and Image Super-Resolution. Single Image Super-Resolution (SISR) has a board range of applications, for specific fields including security video surveillance and medical imaging. In our project, we first present how can the Generative Adversarial networks can be applied on Image Super-Resolution. Then we modify the problem settings and generate results based on different problem settings. Finally, we evaluate and compare the different results based on several common measures to get a deeper insight of the Generative Adversarial Networks Image Super Resolution Algorithm. Our quality measures include metrics such as Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM), and human options to judge the results generated.

The goal of our project is to implement a Generative Adversarial Networks model for the image super-resolution and analyses its performance of image reconstruction with different parameter settings.

Introduction

Single Image Super-Resolution (SISR) is a topic with a board range of applications, for specific fields, including security video surveillance, cosmology and medical imaging. In security surveillance field, SISR can assist us to have more clear image that contain more important detail information. In the cosmology, the super-resolution algorithm can help the researchers to perceive a clear status of celestial bodies. Moreover, in the medical imaging area[12], the Image-Super resolution can help use to extract more information that hidden in the low-resolution medical images.

Meanwhile, the Generative Adversarial Networks has been a popular techniques on the Computer Vision and Machine Learning areas. Therefore, using the GANs for improving the current status of Image super resolution has attracted the interests of Machine Learning researchers[14]. The image super resolution problem is especially challenging on recovery of texture details. Thus, we first provide the GANs model that has several different loss functions. One of these loss functions can differentiate the super resolution generated by computer and original. Besides the differentiator, we have a loss function that measure the perceptual similarity. Therefore, our algorithm should generate the super solution image that with the detailed textures.

We modified the algorithm to test its performance with following conditions:

- applying a smaller upscaling factor
- combing the anti-aliasing approach with GANs

Then we evaluate the results generated under different scenarios with several common measurements, including PSNR (Peak Signal to Noise Ratio), SSIM (Structural Similarity), etc. But such measures have disadvantages that they may not capture the perceptual difference that human eyes can received. Therefore, the result with highest scores from these metrics may not be the best super-resolutions.

In our project, we will compare the super-resolution image with more deep insight comparison.

Related Work

Initial work on SR can be traced back to the beginning of the 80s with applications to image enhancement and restoration[13]. The goal of Single Image Super Resolution algorithms is to generate high-quality images from a single low resolution input image. With different priors each algorithm uses, we can classify the SISR algorithms into into several types of approaches.

Prediction Model

With the prediction models, the low resolution input images can be transformed into high resolution out images with a mathematical model instead of training data and trained model. For instance, the Image Registration [1] method introduced an algorithm that generate pixels by weighting the local neighbor pixels. But this method requires a pre-knowledge of edges of the lower resolution input image. Otherwise, edges on the result image would be wiggle and looks like mosaic.

Early Learning-based Methods

These methods generate the high resolution output image based on the classical Machine learning approaches and

training data. The VISTA(Vision by Image/Scene Training) [2] is the first learning-based super-resolution framework that learns relationships between low-resolution image patches and its high-resolution counterparts. The VISTA generate a combination of scenes and the rendered images to model the relationship with a Markov model. And as the input image is given, VISTA applies the Bayesian belief propagation to find the local optimal value of the posterior probability for the scene. However, such method requires scene patches for any given input low resolution image patch to generate the corresponding models, limiting the usage of such method.

Sparsity-based Methods

Sparsity-based methods [1,3,4,5] is based on a sparse representation for each patch of the low-resolution input. As suggested, a sparse linear combination of elements can represent the images patches of input image very well. Therefore, by choosing the appropriate weights of the linear combination, the method can result the high resolution output image with robustness to the noisy input image.

Self-examplars Methods

Super-resolution via self-examplars is combination of the classical model and learning-based model [6,7]. Such method exploits the statistical prior of natural image patches that tend to recur within and across scales of the same image [7]. And such method can handle the texture variation with additional affine transformations. However, such method can only give a small increment of resolution for the result image.

Deep Learning Methods

Deep learning methods are working as a magical tool on the area of Computer Vision. Recently, the deep Convolutional Neural Network has demonstrated its strength on image resolution [8]. Such architectures can generate excellent result because they capture both local and global information of image at same time by the very deep and complicated layers of CNNs.

Method

In Single Image Super-Resolution (SISR), the aim is to super-resolve an image I^{SR} from the image I^{LR}, where I^{LR} is the high resolution version of I^{SR}. Only high resolution images are available in the training time. Therefore, we obtain low resolution images I^{SR} by applying a guassian filter followed by downsampling R from high resolution images in the dataset. For the purpose of our paper R is defined to be 2 and 4.

We define a loss function l^{SR}. Our ultimate goal is to train a generator G, with parameter θ_G, to optimize the loss function l^{SR}, using a dataset of high resolution images I_i^{SR}, $i = 1, 2...N$:

$$\hat{\theta_G} = \frac{1}{N}\sum_{i=1}^{N} argmax_{\theta_G} l^{SR}(I_i^{SR}, I_i^{LR})$$

Adversarial network architecture

Following Goodfellow et al[9], we define our SRGANs formulation to be:

$$min_{\theta_G} max_{\theta_D} E_{I^{HR}\sim P_{train}(I^{HR})}[log(D_{\theta_D}(I^{HR})] + E_{I^{LR}\sim P_G(I^{LR})}[log(1 - D_{\theta_G}(G(I^{LR}))]$$

The key idea behind SRGANs is that the generator creates I^{HR} in order to fool the descriminator. The discriminator, however, tries to distinguish the real high-resolution images from the dataset from the generated images I^{HR}. Hence our model encourages the generator to learn the structure and the manifold of pixel distributions of the real image.

We follow the architecture introduced by C. Ledig et al[9], which is illustrated in Figure 1.

Loss function

The definition of loss function is key to the performance of our model. The l^{SR} is unsually based on MSE loss [10]. However, MSE loss often loses information of high frequency, and thus tends to make images blur and over-smooth. Hence following the l^{SR} introduced by [10], we replace the l_{MSE}^{SR} with l_{VGG}^{SR}, defined as following:

$$l^{SR} = l_{VGG}^{SR} + 10^{-3} l_{Gen}^{SR}$$

VGG loss

The VGG loss is based on the pre-trained VGG network introduced by Simonyan and Zisserman [11]. Unlike MSE method, which only considers image pixel-wise loss, VGG network also considers feature map loss. We define ϕ_{ij}, which is the ith convolution neural network before j^{th} max-pooling layer in VGG network map.

$$l_{VGG/ij}^{SR} = \sum_{x=1}^{W_{ij}}\sum_{y=1}^{H_{ij}}(\phi_{ij}(I^{HR}) - \phi_{ij}(G_{\theta_G}(I^{LR}))^2$$

where W_{ij}, H_{ij} represent width and height of the ij receptive field respectively.

Adversarial loss

The adversarial loss is part of GAN's architecture. Low adversarial loss means that the generator has stronger power to create I^{HR} that fools the discriminator.

$$l_{Gen}^{SR} = \sum_{i=1}^{N} log(1 - D_{\theta_D}(G(I_i^{LR})))$$

Implementation

Dataset

In this experiment, we use a dataset called RAISE dataset. This dataset includes in total 8137 high resolution raw images, mainly intended for digital image forensics. It is guaranteed that the images have not been compressed or

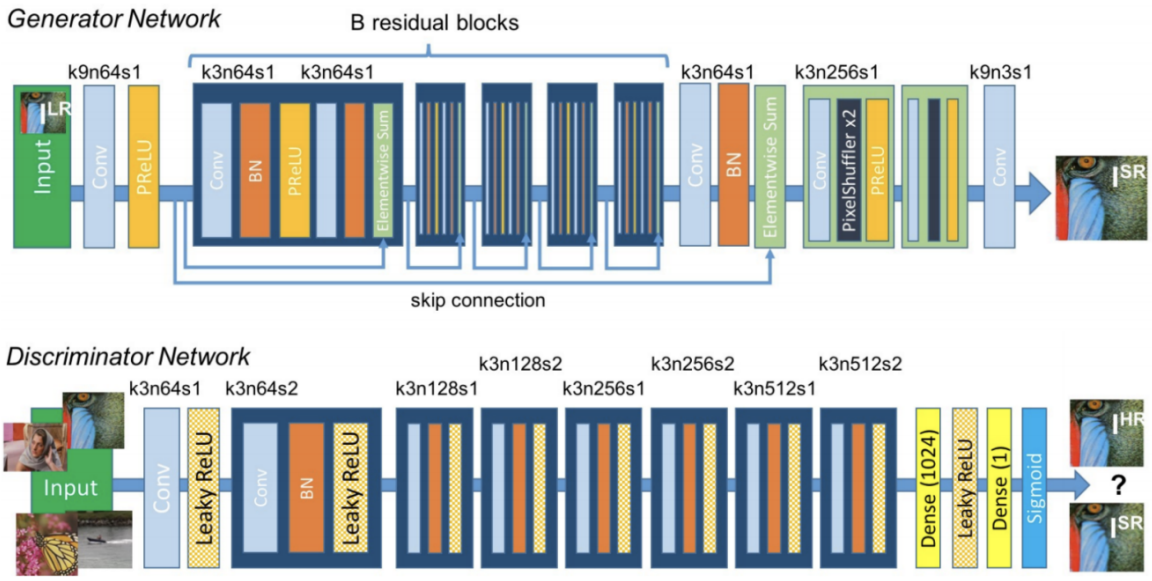

Figure 1: Structure of our network, taken from [15].

processed in any way. It can be easily downloaded and contains images with diverse color and scenery. More details on this dataset can be found in [14]. Figure 3 shows three randomly chosen images from the dataset.

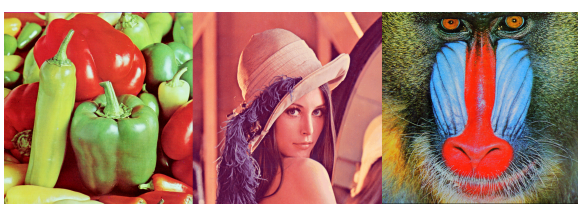

Figure 2: Examples of HR images from RAISE dataset.

In order to train our model, images from the RAISE dataset is not sufficient since both low resolution (LR) and high resolution (HR) images are needed in the training phase. To obtain low resolution images, we perform down-sampling on the HR images by a factor of 2 and 4.

SRGAN generally requires a larger dataset in the training phase. Therefore, we utilize two methods to increase dataset size. The first method is to randomly choose a fixed window size smaller than the image size. Random sub-images of that size are cut out from the original HR and LR images proportionally. In this paper, a window size of 24 by 24 is chosen. The second method is to create horizontal mirror images of the original images. This method increases the training set size without any loss of information on images' structure, and thus lowers the chance of overfitting while improves the model's robustness.

Training Details

All our training is done on our desktop with Intel i7-6700 and Nvidia GTX 1070. The machine learning framework we use is TensorFlow. We first run the MSE model of an iteration number of $500k$, which requires 40 hours to finish. Then we run the model of VGG with an iteration number of $20k$, which requires longer time per iteration and the total time is around 2 hours. So we get a quick intuition of the the VGG effects on the project. Then we run another VGG training of $100k$ to get the final training result.

Training Process

In order the get the loss status during the training, we monitor them by plotting out the metrics and loss value as shown in the figure 4.

Note: We have only implemented the MSE part. And the analysis of CNN results are based on the existing project[15]. y

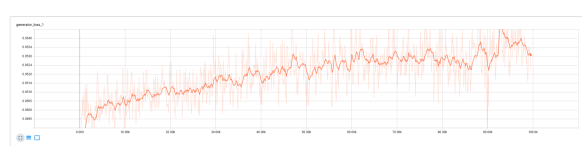

Figure 3: Generator Loss of the $10k$ VGG training process

Result

Analysis

First of all, we want to know how long we need to train for each setting of parameters, so we run under one setting for quite long time. However, out of our expectation, though

Figure 4: The screen shot on the left side shows PSNR and content loss after 20000 iterations with VGG loss function. The screen shot on the right side shows the same with 10000 more iterations. The content loss on the left side starts low because we trained on MSE loss prior to this training.

the content and adversarial loss is continuing decreasing, the PSNR stops decreasing pretty early. Then we use the trained model at different stage after PSNR stops decreasing to generate high resolution pictures. The outcome at different stage shows that they do vary in some way but have similar sharpness and similar degree of altered details. There is no leap of improvement in the quality of the output as shown in Figure 4.

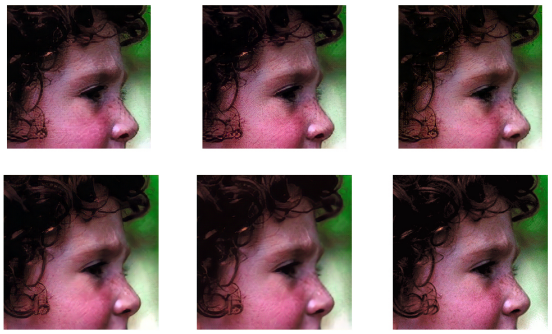

Figure 5: First row from left to right, the ratio of adversarial loss in the global loss is 0.0001, 0.0005, 0.003. Second row from left to right, the ratio is 0.01, 0.02, and the original HR image.

Because the limit of time and computational power, we continue our experiment with the model after the training iteration at which PSNR stops decreasing in the previous experiment. We want to know how adversarial loss function adds to our training process, so we adjust the ratio of adversarial loss in the global loss from 0.0001 to 0.003, and compare it with the result given by MSE loss. After the training phase, we observe that the blurriness in MSE output disappears when adversarial loss function is added. But as the ratio increases, the details of output become richer, but some of these new details are not from the original High resolution picture.

Why this is happening? As mentioned in some other articles, this problem itself is ill-posed in some degree. If the result looks similar to the original picture but the details have changed a lot, what is it's advantage considering wrong details may be more misleading than blurry ones? Can we find a better way to balance the sharpness of generated "high resolution pictures" and the new details which cannot be guaranteed to be true in theory?

Figure 6: The left most picture is the input (4x smaller than the HR image). The middle picture is our output. The right most picture is the original HR image.

The most result we see is doing the super resolution from $x * y$ resolution to $4x * 4y$ resolution, which means that we need to generate 16 pixels to replace one pixel in original low-resolution picture, what if we try 4 first? Following this though, we change our model and do training to get the "middle resolution" version. As expected, the result is very good both by data and by eye even though we did not do the MSE pre-training because it takes too long to do for the new model.

But this does not mean that we get a better solution yet because for same size output, this model requires more information than the previous one. We have tried three ways to achieve the original goal with this model:

- use model twice
- first use the model, then use anti-aliasing(AA) to double the size of picture
- first use AA then the model

The result is shown in Figure 7.

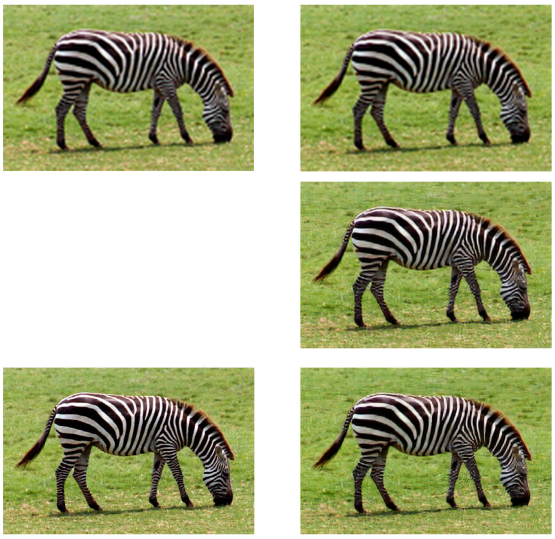

Figure 7: The top left image is the input. Bottom left image is the output. On the right side from the top to bottom, it is AA followed by model, model followed by AA, and model followed by model, respectively.

The result is a little bit blurrier but have better PSNR and do not have the misleading detail.

In this experiment we find that in the last method, namely, applying AA followed by the model, the model actually have learned "the degree of blurriness" because for an originally blurry input, the output from the model is enlarged but still blurry. In addition, the blurriness is all over the picture rather than over some specific area.

Discussion

We observe output images generated with VGG loss function to have a better result than the output images generated with MSE loss function. VGG loss function has a better human-observable result because it does better feature extraction. Therefore, to human eyes, it produces better outcomes. MSE loss function does more work on pixel comparisons of the LR and HR images, compared to VGG. Because of the nature of MSE function, the model that aims to minimize this error tends to move towards the average value of a pixel, and thus reducing its intensity in color. As a result, the output image has a relative sense of blurriness.

One confusing problem is that although VGG yields better results as we observe it, the values of PSNR do not match the human-observed result. As shown in Figure 8, the value of PSNR from VGG loss function is lower than the value from MSE loss. Generally higher PSNR value indicates better restoration of the target image.

We believe that this is because VGG generates many "fake" output pixels. Due to its nature, VGG tends to use its imagination to produce results that it thinks can best represent the input image. However, as for MSE, it follows truly the input image's pattern and structure and this is perhaps one of the main reasons behind this inconsistency between the PSNR value and the perception results.

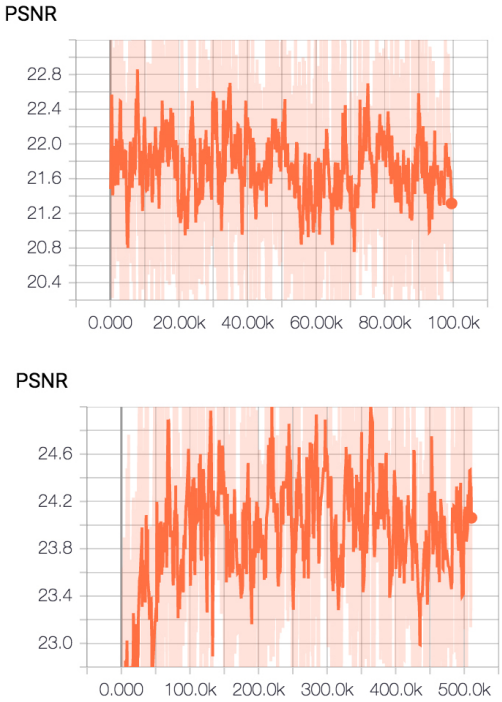

Figure 8: The top screen shot shows the value of PSNR with VGG loss function. The bottom shows MSE loss function.

Limitation

Our new approach, upscaling twice, has bad performance when dealing with clustered texture. The result would have many unrealistic textures.

Some of the results contain information that is guessed and created by our algorithm merely. This may bring better perceptual result. However, this problem limits the realistic application of our algorithm. For instance, in the medical imaging super resolution, the result may reflect some diseases that never exist.

Due to the limitation of time and computational power, we can not fully train each model. However, each model is given the same time so that the comparison results are still representative in some degree.

We believe that restoration of image details would be much better if we use Super Resolution on a specific class of pictures, instead of diverse images from the RAISE dataset, because image patterns and structures would be better learned given a training set with similar details.

Future Work

We also tried to use our project to do video resolution increment. But it is only theoretically approachable for us because of the running time. Our method super resolution the input video frame by frame, but we believe there exists a cleverer approach that generate a resolution matrix as the continuous frames tend be similar.

And currently, we can only gain some unclear information about how the parameters influence the result images. But the theoretical foundations of how GANs work is still vague. Then, in the future, we want have more mathematical details about the GANs; so we can improve the parameters and results much better.

Conclusion

In this project, we learned how super resolution with GAN works.We saw the limitation of existing algorithm, and try to propose a new approach that balance the illness of problem and our requirement of resolution increase. Given limited computation power, our method gives better result(in terms of pant) than existing method by sacrifice a little sharpness and avoid a lot fake details that was learned from a general training set. And our second method gives sharp output picture without that many fake details on pictures that don't have detailed textures. Furthermore, we learned a lot about images processing, including the storage and manipulation of images. And we have also learned how to use CNN as a fancy image processor.

Acknowledgements

We would like to thank Professor Craven and Professor Page for this wonderful semester and the dedication given by teaching assistants. The course has been an extraordinary experience for us!

References

[1] J. Yang. Image Super-Resolution Via Sparse Representation, University of Illinois at Urbana-Champaign, 2010.

[2] W. Freeman, E. Pasztor, O. Carmichael. Learning Low-Level Vision, 2000.

[3] R. Zeyde, M. Protter and M. Elad. On Single Image Scale-up Using Sparse-Representations, International conference on curves and surfaces, 2010.

[4] W. Dong, L. Zhang, G. Shi, and X. Wu, Image Deblurring and Super-resolution by Adaptive Sparse Domain Selection and Adaptive Regularization, TIP, 2011.

[5] T. Peleg and M. Elad, A Statistical Prediction Model Based on Sparse Representations for Single Image Super-Resolution, TIP, 2014.

[6] D. Glasner, S. Bagon and M. Irani, Super-Resolution from a Single Image, ICCV, 2009.

[7] J. Huang, A. Singh, and N. Ahuja, Single Image Super-Resolution from Transformed Self-Exemplars, CVPR, 2015.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems, NIPS, 2014.

[9] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. CoRR, 2016.

[10] C. Dong, C. Loy, K. He, and X. Tang. Image Super-Resolution Using Deep Convolutional Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2):295307, 2016.

[11] K. Simonyan and A. Zisserman. Very Deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations. ICLR, 2015.

[12] A. Gholipour, J. Estroff, S. Warfield, Robust super-resolution volume reconstruction from slice acquisitions: application to fetal brain MRI., IEEE Trans. Med. Imaging 29(10), 17391758, NIH Public Access, 2010.

[13] T. Huang, R. Tsay. Multiple frame image restoration and registration. Adv. Comput. Vis. Image Process. 1, 317339, JAI, Greenwich, 1984.

[14] D. Dang-Nguyen and C. Pasquini and V. Conotter and G. Boato. RAISE: A Raw Images Dataset for Digital Image Forensics, 2015.

[15] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, 2017.