IJTRET

International Journal of Trendy Research in Engineering and Technology
Volume 4 Issue 5 Proceedings of NCCETCC'20
ISSN NO 2582-0958

Proceedings of National Conference on Cutting Edge Technologies in Computing & Communications held at Sriram Engineering College, Chennai on 6th March 2020

UPSCALING IMAGE USING GENERATIVE ADVERSARIAL NETWORKS NETWORKS

Venkatesh B, Nithesh kumar J S, Praveen G, Murugan V
Dept. of Computer science and engineering
*Kingston engineering college,*Vellore, TamilNadu, India
rajuvenkatesh98@gmail.com,nitheshkumar9991@gmail.com,
praveenengg1999@gmail.com,muruganv0307@gmail.com

***Abstract*—We enhancing low resolution images by applying deep network with adversarial network (generative adversarial networks) to produce high resolutions images.Our main target is to reconstruct super resolution image or high resolution image by up-scaling low resolution image such that texture detail in the reconstructed SR images is not lost. In GANs , there is a generator and a discriminator. The generator generates fake samples of data(be it an image, audio, etc.) And tries to fool the discriminator. The discriminator, on the other hand, tries to distinguish between the real and fake samples.**

I. INTRODUCTION

There are various ways of enhancing image quality. One of the most commonly used technique is interpolation. This is easy but this leads to distorted image or reduce the visual quality of the image. So we use generative adversarial network (GAN) are the tySpe of deep neural network that are used to generated images by using networks. Deep learning produce better solution to get optimized images. Here we discuss about SRGAN. The generator and the discriminator. The generator attempts to generate images from an array of often smaller size, and the discriminator determines whether the generated images are real or fake by finding the Nash equilibrium of a game. In this paper we explores effective way of training to achieve super-resolution and other application, using various dataset in SRGAN. Generative modelling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate new examples that plausibly could been drawn from the original dataset.

The basic idea of a generative model is to take a collection of training example and from a representation of their probability distribution. And the usual method for it was to infer a probability density function directly. GAN is a piping hot topic in deep learning because of their power. The real power in GANs comes from the training style they follow. The Generator networks weights are learned based on the loss of the discriminator. Thus, the generator is pushed to be trained in such a way that for the images it generates, it is very hard to tell if they are real or not. At the same time that these images are looking more and more real.

II. DESIGN OF GENERATIVE ADVERSARIAL NETWORK

In machine learning, the two main classes of models are generator and discriminator. The generator tries to produce some data from probability distribution and discriminator act like a judge. Discriminator decides whether input is coming from true training data set of fake generated data. Generator tries to optimize data so that it can match true training data. Or we can say discriminator is guiding generator to produce realistic data. They just work like encode and decode. 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.

The generative network generates candidates while the discriminative network evaluates them. The generative network learns to map from a latent space to a data distribution of interest, while the discriminative network distinguishes candidates produced by the generator from the true data distribution. The generative network's training objective is to increase the error rate of the discriminative network.

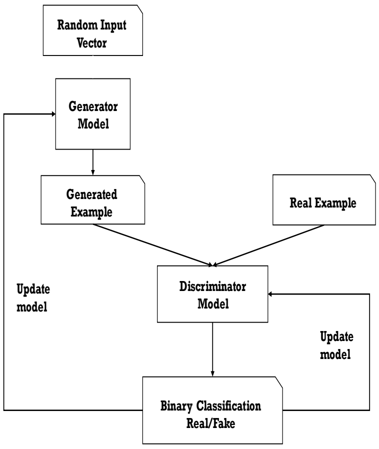


Figure 1. Architecture design of Generative Adversarial Network.

III. SUPER RESOLUTION GAN

we use single image super resolution GAN. It applies a deep network in combination with an adversary network to produce higher resolution images. In this method still there is a problem which is not solved. The result have high peak signal to noise ratio means we have good image quality results, but they are often lacking in high frequency details and are perceptually unsatisfying as they are not able to match the fidelity excepted in high resolution images.

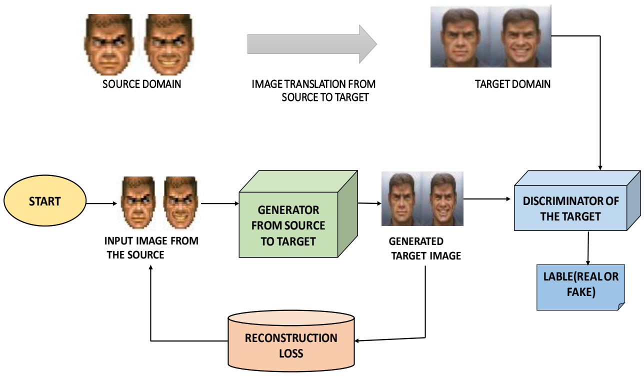


Figure 2. Detailed design of super resolution GAN

There is low resolution image(source domain) and high resolution image (target domain). Now we have both HR and LR images for training data set. We pass LR images through generator which up samples and give SR image. Then discriminator used to distinguish the HR images and back propagates the GAN loss to train the discriminator and the generator.

A. Generator network

Generator is trained while discriminator is idle. After the discriminator is trained by the generated fake data of the generator, we can get its prediction and use the results for training the generator and get better from the previous state to try and fool the discriminator. The above method is repeated for a few times and then manually check the fake data if it seems genuine. If it seems acceptable, then the training is stopped, otherwise, its allowed to continue for few more times.

B. Discriminator network

The discriminator is trained while the generator is idle. In this phase, the network is only forward propagated and no back propagation is data. The discriminator is trained on real data for n times, and see if it can correctly predict them as real. Also, in this phase, the discriminator is also trained on the fake generated data from the generator and see if it can correctly predict them as fake.

IV. THINGS TO NOTE NETWORK ARCHITECTURE

a) 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. More about residual blocks are used in generator.

b) PixelShuffler x2: This feature may upscaling. 2 sub pixel CNN are used in generator. Upscaling or Upsampling are same. There are various ways to do that. In code kerasinbuild function has been used.

c) Loss Function: This is most important part. As discussed we will be using perceptual loss. It comprises of content loss and adversarial loss.

$$l^{SR} = \underbrace{\underbrace{l^{SR}_{X}}_{\text{content loss}} + \underbrace{10^{-3} l^{SR}_{Gen}}_{\text{adversarial loss}}}_{\text{perceptual loss (for VGG based content losses)}}$$

d) 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.

$$l^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I^{LR}))$$

Adversarial Loss

e) Content Loss: Content loss we are using so that we can keep perceptual similarity instead of pixel wise similarity. This 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.

$$l^{SR}_{VGG/i.j} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^{HR})_{x,y} - \phi_{i,j}(G_{\theta_G}(I^{LR}))_{x,y})^2$$

Content Loss

f) Peak signal to noise ratio : PSNR is a term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signal have a very wide dynamic range, PSNR is usually expressed terms of the logarithmic decibel scale.

	Bicubic	EDSR	VDSR	SRGAN(ours)
SF-4	46.41	47.78	43.04	49.06
SF-8	41.16	41.64	37.92	44.06
SF-16	37.11	29.91	33.43	38

Table 1. PSNR(Peak Signal to Noise Ratio)

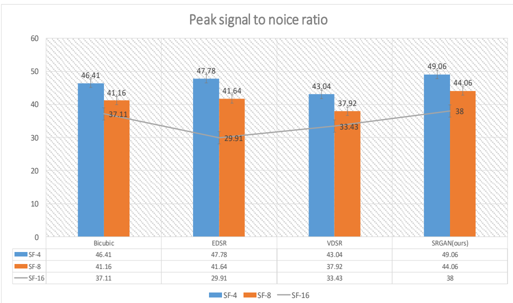


Figure 3. Peak signal to noise ratio flow chart

g) Structural Similarity index: It is a perceptual metric that quantifies current documentation image quality degradation caused by processing such as data compression or by losses in data transmission. It is a full reference metric that requires two images from the same image capture a reference image and a processed image.

	Bicubic	EDSR	VDSR	SRGAN(ours)
SF-4	0.9992	0.9997	0.9991	0.9995
SF-8	0.9975	0.9993	0.9975	0.9982
SF-16	0.9943	0.9967	0.9994	0.9968

Table 2.SSIM (Structural similarity index)

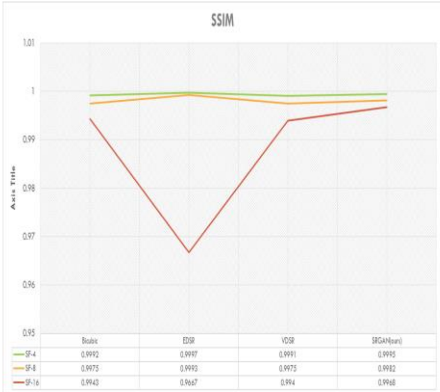


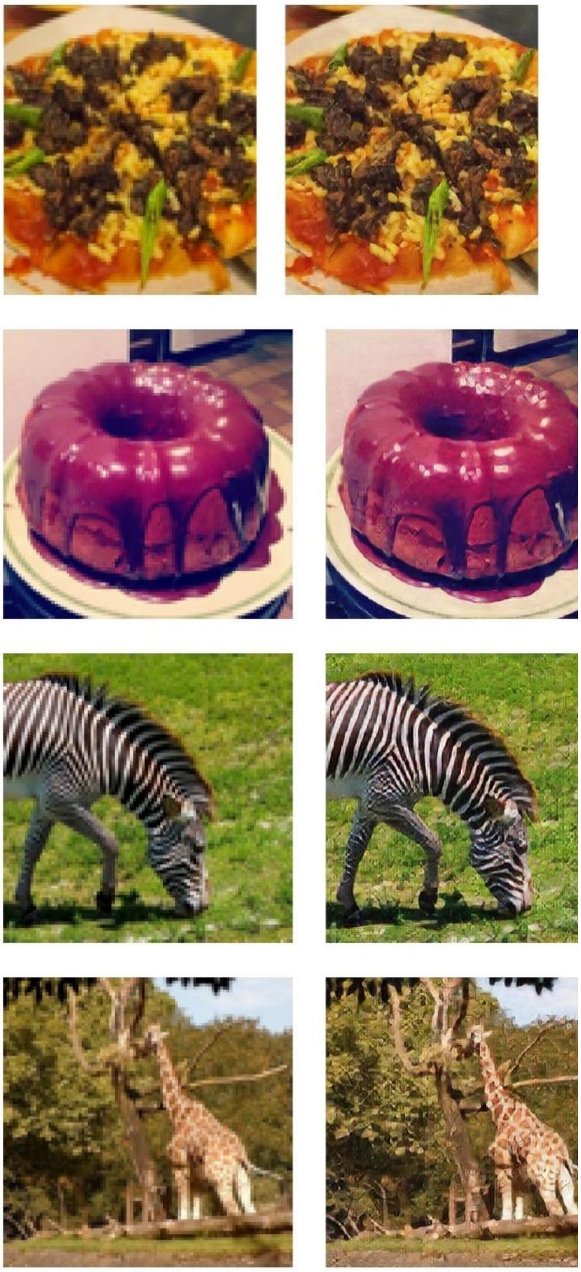
Figure 4. graph of Structural Similarity index

V. CONCLUSION

In this paper, by embedding improved SR blocks in the generator and the discriminator of the GAN, and by using new perceptual loss function, we have presented an effective SR-GAN model. The experimental results on PSNR and SSIM graph have shown that our model outperforms existing model (Bicubic, EDSR, VDSR). Moreover, our method can reconstruct images with more detail structures for higher scaling factors. Let's conclude that we don't know if the idea of GANs is really "the most interesting idea in the previous years in machine learning".

Here are few results :



REFERENCE

- *1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. Generative adversarial nets, NIPS (2014).*
- *2. Radford, A., Metz, L. and Chintala, S., Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434. (2015).*
- *3. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. Improved techniques for training gans. NIPS (2016).*
- *4. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., &Abbeel, P. InfoGAN: Interpretable Representation Learning by Information Maximization*
- *Generative Adversarial Nets, NIPS (2016).*
- *5. Mirza, Mehdi, and Simon Osindero.Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014).*
- H. S. Hou and H. Andrews, "Cubic splines for image interpolation
- and digital filtering," *IEEE Trans. Acoust., Speech, Signal Process.*,
- vol. ASSP-26, no. 6, pp. 508–517, Dec. 1978.
- [2] T. M. Lehmann, C. Gonner, and K. Spitzer, "Survey: Interpolation
- methods in medical image processing," *IEEE Trans. Med. Imag.*, vol. 18,
- no. 11, pp. 1049–1075, Nov. 1999.
- [3] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive
- modeling and soft-decision estimation," *IEEE Trans. Image*
- *Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.
- [4] X. Liu, D. Zhao, R. Xiong, S. Ma, W. Gao, and H. Sun, "Image
- interpolation via regularized local linear regression," *IEEE Trans. Image*
- *Process.*, vol. 20, no. 12, pp. 3455–3469, Dec. 2011.
- [5] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE*
- *Trans. Image Process.*, vol. 10, no. 10, pp. 1521–1527, Oct. 2000.
- [6] S. Dai, M. Han, W. Xu, Y. Wu, and Y. Gong, "Soft edge smoothness
- prior for alpha channel super resolution," in *Proc. IEEE Conf. Comput.*
- *Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
- SONG *et al.*: FAST IMAGE SUPER-RESOLUTION VIA LOCAL ADAPTIVE GRADIENT FIELD SHARPENING TRANSFORM 1979
- [7] J. Xie, R. S. Feris, and M.-T. Sun, "Edge-guided single depth
- image super resolution," *IEEE Trans. Image Process.*, vol. 25, no. 1,
- pp. 428–438, Jan. 2016.