Grenze International Journal of Engineering and Technology, Jan Issue

Image Super Resolution and Deblurring using Generative Adversarial Networks

Iruvanti Gautam[1], K P Arjun Rajesh[2], Krishna Sidharth Sagere[3] and K V Badri Prasad[4]
[1-3]PES University/CSE,Bangalore,India
Email: {gautams818,arjun.rajesh1886,krishna.sid123}@gmail.com
[4]PES University/CSE,Bangalore,India
Email: badriprasad@pes.edu

***Abstract*—As deep learning methodologies exhibit strong advantages in the vast domain of feature extraction, it has been used widely in the computer vision area, and gradually replaced traditional machine learning algorithms. This paper describes in detail the development, creation and Analytics of a Deep learning pipeline which is comprised of two components firstly a Deblur Generative Adversarial Network (DeblurGAN) and secondly a Super Resolution Generative Adversarial Net-work (SRGAN) which work in unison with one another in order to convert and transform Low Quality Blurred images into a higher quality Super resolved de-blurred image. In our solution we are aiming to tackle various types of blurs such as linear blurring, Gaussian Blurring and Media Blurring. Generative adversarial networks (GANs) are algorithmic architectures that primarily use two neural networks, pitting one against the other (hence the "adversarial") in order to produce new ,enhanced, synthetic data instances that can pass for real data, using this principle we are generating de-blurred and super-resolved images. The Trained Discriminator acts as an accurate classifier and has the main functionality of differentiating and distinguishing between the multiple generated images of the GAN and the real images. At the specific point in time when the discriminator component is unable to differentiate between the images that are created and the actual images, it is safe to say that we have generated the most de-blurred and Super-resolved image possible. Afterwards the results (i.e. Peak Signal to Noise Ratio aka PSNR) are compared in different scenarios to observe which type of blur is best tackled by the deep learning pipeline.**

***Index Terms*— Generative Adversarial Networks, Super-resolution, Deblurring, Content Loss, Adversarial Loss and PSNR value.**

I. INTRODUCTION

Image processing is a common technique for performing different manipulative operations on an image to produce an enhanced image, or to extract some essential information from it. It is a type of signal processing in which the input is an image and the output may be a reconstructed image, characteristics, or even features associated with that image. The problem we will be focusing on is image reconstruction. Generative Adversarial Networks (GANs) [4] are widely used neural networks for image restoration and provide state of the art results for image super resolution and image de-blurring. Image Super-Resolution is an image restoration technique of reconstructing a high resolution image from the observed low-resolution image. Most of the approaches for Image Super-Resolution till now used the mean squared error (MSE) as a loss function, MSE presents a problem

Grenze ID: 01.GIJET.7.1.19_1

where the image's high texture information is averaged to create a smooth rebuild. GANs overcome this issue by using the perceptual loss that guides the reconstruction of the picture towards the manifold of the natural world, generating solutions that are perceptually more realistic and convincing. Image Deblurring is an image restoration technique of recovering a sharp latent image with required edge structures and information from the noisy input image. GANs [4] are known for the ability to preserve texture details in images, create solutions that are close to the real image manifold and look perceptually convincing and hence are used in the image de-blurring problem. Our goal is to combine these two techniques to generate images which are more realistic and have less noise per unit pixel.

II. Related Work

A. Image to Pixel Mapping

Image de-blurring has become a major focus in computer vision and image processing for many years. The goal of de-blurring is to recover a sharp latent image with required edge structures and information, due to a motion-blurred or focal-blurred input image induced by camera shakes/disturbance, sudden object movement or even out-of-focus bodies. Single image deblurring is rather un-positioned. In current approaches, various constraints are applied to the blur model characteristics (e.g. uniform / non-uniform / depth-aware), different natural image priors are used to regularize the Solution space. Some of these approaches require complex computing, often heuristic, parameter-tuning and are costly. However, the simplistic assumptions on the blur model also hamper their success on real-word instances, where blur is much more dynamic than models and is entangled with the image processing system in the camera. History of image processing methodologies shows that the most popular machine learning technique for image processing and image analysis was Convolutional Neural Network (CNN) [9]. A CNN works by extracting image attributes, it removes the need for any manual feature extraction. The features are not trained in any form, they are learned while training on a collection of images by the network. This makes deep learning models highly effective for tasks involving computer vision. CNNs learn to detect features across tens or hundreds of layers in hiding. The complexity of the learned features increase as you go to deeper layers.

B. Photo-Realistic Single Image Super-Resolution Using A Generative Adversarial Networks

This paper defines a super-resolution generative adversarial network which employs a deep residual network (ResNet) with skip-connections and diverges from MSE as the sole optimization target. To achieve a high level of super-resolution, the authors 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 the original photo-realistic images. The authors also say that the aim is to use a content loss driven by perceptual similarity instead of similarity in pixel space. The deep residual network that has been proposed for super resolution is able to recover photo-realistic textures from heavily down sampled images. The authors claim that with the super-resolution architecture the results are closer to the original high-resolution images when compared to those obtained with any other state-of-the-art method.

III. Proposed Method

There are three stages of implementation in our project:

1. Creation of the DeblurGAN architecture and code, so that the various blurs present in the image can be removed (Gaussian, Lateral etc.).
2. Creation of the Super Resolution GAN architecture and code for extracting the essential features of the image so that a higher resolution output image can be synthesized.
3. Connecting the two components of the pipeline so that the output of the DeblurGAN becomes the input of the Super Resolution GAN. Integration testing these two components was done so that no errors and bugs are present.

A. Deblur GAN

The architecture consists of a generator and discriminator.

1. The discriminator(critic) is trained to assign a realness measure to every image to symbolize whether the image is an actual image or a de-blurred image. It is Implemented using CNN blocks.
2. The generator is used to take input blurry images and generate de-blurred output images which are sent to the critic for evaluation. It is Implemented using ResNet blocks [5].

3. The loss function is a multi component loss function consisting of wasserstein loss [6] and the perceptual loss [4].

The generator architecture consists of nine residual blocks [3] and two transposed convolution blocks. Each ResNet block consists of a CNN layer, batch normalization layer and activation rectified linear activation unit (RELU). Each ResNet block consists of a dropout of 0.5 after the first CNN layer [4]. A skip connection on a global level is introduced which results in a faster training process and better generalizations. The CNN learns a residual correction (IR) to the blurred image (IB), so image output, i.e IS = IB + IR [4]. The discriminator consists of CNN blocks similar to Wasserstein GAN (WGAN) [7] but with gradient penalty. Each layer of the discriminator consists of Batch normalization and a leaky RELU (alpha = 0.2) [8], except the last layer which is flattened to produce a 1D output which is a critic score between 0 and 1.

B. Super Resolution GAN

The SRGAN has two components: a discriminator and a generator which compete with each other.

1. The discriminator which is also the critic is trained to discriminate between the real High Resolution (HR) image and the image generated by the generator. The Discriminator is implemented using CNN blocks.
2. The Generator is used to take input Low Resolution images and generate High Resolution output images which are sent to the critic for evaluation. It is implemented using ResNet blocks.
3. The loss function is a multi component loss function consisting of adversarial loss and content loss.

We are implementing the architecture defined in the following paper [9]. The generator network G is implemented using B residual blocks. The blocks consist of two convolution layers followed by batch-normalization layers [10] and the Parametric-ReLU function [5] as the activation function. Resolution is increased using 2 trained sub-pixel convolution layers proposed in [12] The discriminator network consists of 8 convolution layers with 3x3 filter kernels, which are increasing by a factor of 8 from 64 to 512 kernels as mentioned in the VGG network [13]. We implement the network using strided convolutions to reduce the image resolution each time the number of features is doubled. The feature maps are then succeeded by two dense layers and a final sigmoid activation function.The sigmoid function outputs the probability of sample classification.

C. Loss Function

The DeblurGAN and SRGAN are trained using the same loss function.
The total loss function is defined as the sum of adversarial loss and content loss as shown in (1).

$$\mathscr{L} = \underbrace{\underbrace{\mathscr{L}_{GAN}}_{\text{adv loss}} + \underbrace{\lambda \cdot \mathscr{L}_{X}}_{\text{content loss}}}_{\text{total loss}} \tag{1}$$

Adversarial loss : The loss is the Wasserstein loss performed on the outputs of the whole model. It takes the mean of the differences between the two images. It is known to improve convergence of generative adversarial networks. In (2), D is the Discriminator, G is the generator and I is the blurry image [4].

$$\mathscr{L}_{GAN} = \sum_{n=1}^{N} -D_{\theta_D}\left(G_{\theta_G}(I^B) \right) \tag{2}$$

Content loss : It is a perceptual loss [14] computed directly on the generator's outputs. This first loss ensures the GAN model is oriented towards a de-blurring task. It compares the outputs of the first convolutions of VGG [13] for the original image and the de-blurred image [4].

$$\mathscr{L}_{X} = \frac{1}{W_{i,j} H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} \left(\phi_{i,j}(I^S)_{x,y} - \phi_{i,j}\left(G_{\theta_G}(I^B) \right)_{x,y} \right)^2 \tag{3}$$

In (3), φ(i,j) is the feature map obtained by the jth convolution (after activation) before the ith max-pooling layer within the VGG19 network [4], pre trained on ImageNet, W(i,j) and H(i,j) are the dimensions of the feature maps. VGG(3,3) convolution layer is used in our network [4]. Higher abstraction of the features is represented

by the activations of the deeper layers [16] [15]. The perceptual loss helps in restoring general content [17] [15] and the adversarial loss is used to restore texture details in the image [4].

D. Pipeline

In this section we will define the flow of data and processing in our architecture, i.e in both the DeBlurGAN and SRGAN components as shown in Fig. 1.

The input to the DeBlurGAN is a 256X256 image, the image is de-blurred and down sampled to a 64X64 image.

This de-blurred image is then up-sampled to a 96X96 image and is then fed into the SRGAN which generates the super-resolved image.

The final image is now a de-blurred and super-resolved image which has been fed through the aforementioned pipeline.

E. Dataset

We perform the experiments on the DIV2K and GO PRO dataset

Dataset used for DeblurGAN (Go Pro Dataset): We downloaded the GoPro data set and it contained blurred and original images. The blurs were a combination of artificially applied linear Blur and motion blur due to movement of the objects in the image at the time of capturing. The images were taken from a street view camera. We then organised the data set into training and testing sets and then trained the model.

Dataset used for SRGAN (DIV2K Dataset): We trained the SRGAN on the DIV2K dataset, it consisted of 800 HR images. The images are down sampled and used for training. The dataset has a large variety of images.

F. Evaluation Metric

We use Peak Signal to Noise Ratio (PSNR) to evaluate our model's performance

PSNR ratio is used as a measure of the consistency between the original and compressed picture. The PSNR value is a ratio of the maximum possible value of a signal (power) to the intensity of noise distorting the representation of the overall image. Higher the PSNR value we have the better the overall representation of the output image.

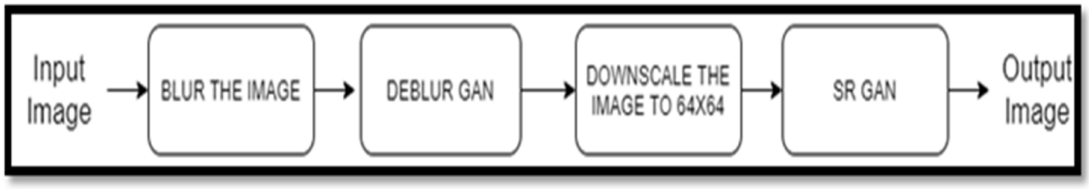

Figure 1 Workflow Diagram Providing a General view of the process

III. CONCLUSIONS

Fig. 2 shows the result of the overall pipeline subjective to the eye, Table 1 shows the PSNR values

1. The DeBlurGAN gives better PSNR values across all the types of blurry images by an average of 2.3 compared to the DeBlurGAN+SRGAN pipeline.
2. For input images taken from the out of focus camera of a phone, the DeBlurGAN gives better PSNR values across all the types of blurry images by an average of 1.7 compared to the DeBlurGAN+SRGAN pipeline.
3. For input images of more blur length and nonuniform blur the DeBlurGAN+SRGAN pipeline gives better PSNR values across all the types of blurry images by an average of 0.6 compared to the DeBlurGAN alone.
4. As illustrated in Table 1 while averaging blur is applied (in our case tested with 15 random images) the DeBlurGAN+SRGAN pipeline gives better PSNR values across all the types of blurry images by an average of 0.5 compared to the DeBlurGAN alone.

From these values we can observe that for averaging blur and non uniform types of blur the combination of the DeBlurGAN+SRGAN works better than the DeBlurGAN alone. Other types of blur and more uniform blur is better augmented by the DeBlurGAN.

A. Future Work and Enhancements

1. Create a dataset of de-blurred images obtained from the DeblurGAN. Use that as the input to train the SRGAN to improve the pipeline's results.
2. Incorporate image segmentation in the generators of both DeBlur and SRGAN to improve performance on varying blur lengths i.e more non-uniformly blurry images.

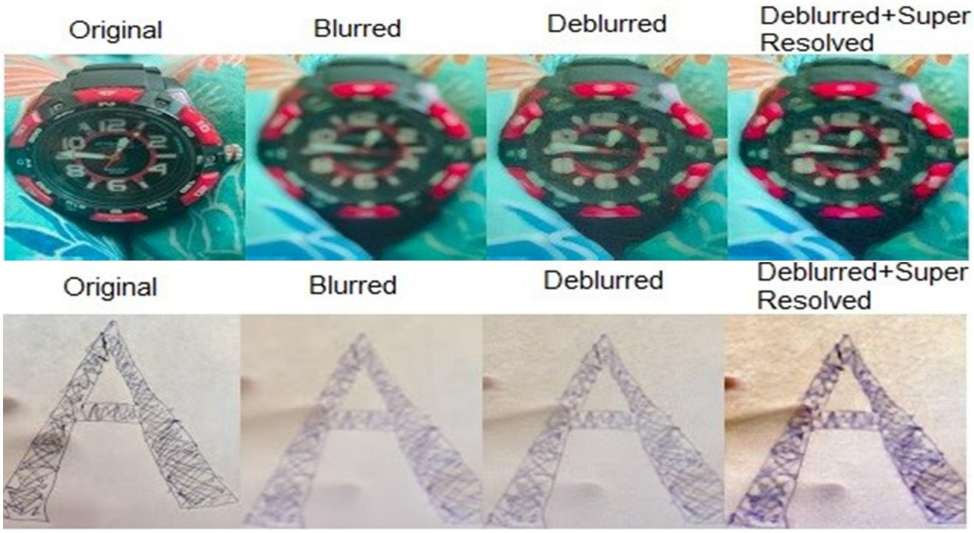

Figure 2 Results of The Overall Pipeline

TABLE I. RESULTS OF THE TESTING PROCESS AFTER TESTING WITH A SERIES OF VARIOUS IMAGES

Types of Blur	PSNR Values	
	PSNR(after deblurring)	PSNR(after deblurring and SR)
Averaging	29.37	32.4
Bilateral	31.71	29.9
Median	31.26	27.2
Gaussian	32.4	30.1

REFERENCES

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks,"June 2014.

[2] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR*, abs/1311.2901, April 2013.

[3] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Transactions on Image Processing*, 20(7):1838– 1857, May 2011.

[4] Kupyn, O., Budzan, V.,Mykhailych, "DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks," *In:Computer Science - Computer Vision and Pattern Recognition* , May 2017.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint* arXiv:1512.03385, May 2015.

[6] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs," *ArXiv e-prints*, March 2017.

[7] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *ArXiv e-prints*, January 2017.

[8] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint* , arXiv:1505.00853,January 2015.

[9] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, "Photorealistic single image super-resolution using a generative adversarial network," *In: CVPR*, March 2017.

[10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *In Proceedings of The 32nd International Conference on Machine Learning (ICML)*, pages 448–456, 2015.
[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint*, arXiv:1512.03385, 2015.
[12] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," *ArXiv e-prints*, September 2016.
[13] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv e-prints*, September 2014.
[14] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real time style transfer and super-resolution," In European Conference on Computer Vision, 2016.
[15] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR*, abs/1311.2901, 2013.
[16] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," *In International Conference on Machine Learning (ICML)*, pages 807–814, 2010.
[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arxiv*, 2016.