Image Inpainting using GAN's

Sisir Nalla
Computer Science and engineering
Shiv Nadar University
Delhi NCR
sn178@snu.edu.in

***Abstract*—This paper examines the existing image inpainitng techniques and proposes a new technique using a GAN consisting of a context encoder like generator and global and local discriminators.**

I. Introduction

Image Inpainting is the process of reconstructing missing parts of an image so that observers are unable to tell that these regions have undergone restoration. Inpainting can be used to restore damaged photos, removing unwanted regions from an image or even filling missing pieces in an artwork. There has been considerable amount of research going into making Inpainting possible with context encoders and patch based techniques being quite successful. But this task remains a challenging problem because it often requires high-level recognition of scenes. Not only is it necessary to complete textured patterns, it is also important to understand the anatomy of the scene and objects being completed but, the techniques proposed so far haven't been successful in solving these challenges.

In this paper I have built upon the context encoder proposed in (1) using adversarial loss. I leverage a fully convolutional network as the basis of our approach, and propose a novel architecture that results in both locally and globally consistent natural image completion.

My architecture comprises of 3 networks - a generator network, a local discriminator network and a global discriminator network out of which the discriminators are exclusively for training. The architecture is similar to that of a conditional GAN (2) but we have multiple discriminators instead of just one.

During each training iteration, the generators and discriminators are updated simultaneously to make image inpainting possible. the discriminators are updated so that they correctly distinguish between real and completed training images. The completion network is updated so that it fills the missing area well enough to fool the discriminator networks.

In Summary, I have presented -

- An efficient network that can fill randomly missing pieces of a photograph
- A training approach that respects the anatomy and local information of the image
- Results of the proposed techniques

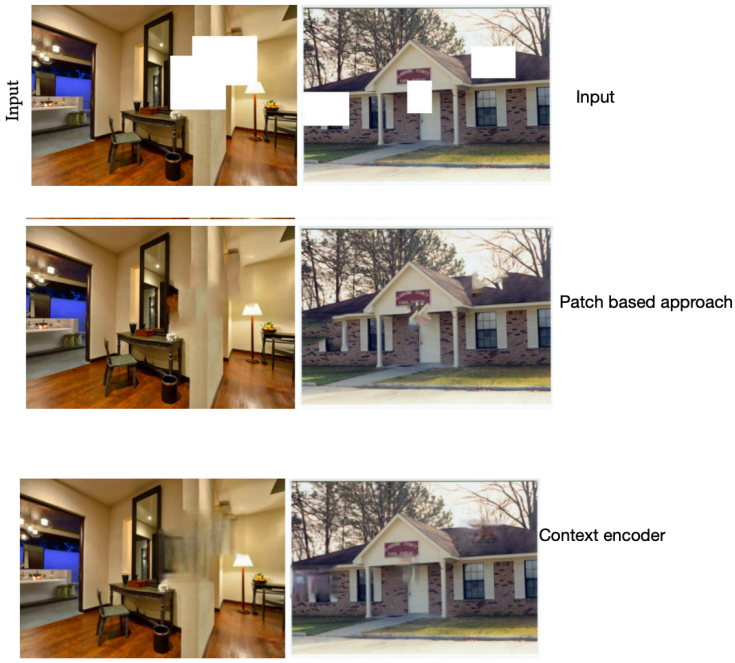
II. Literature Review

A. A diffusion based technique

There have been many attempts to solve the image completion task and one of the traditional ways is to use the diffusion based image synthesis. This techniques attempts to fill the holes in the image by propagating the local image information around these holes. However, it only possible to fill small holes and the anatomy of the image is completely ignored.

B. A patch based approach

Patch-based approaches have been able to perform more complicated image completion tasks that can fill large holes in natural images. In this completion technique texture patches are sampled from a source image and then pasted into a target image. The patch based methods depend on low-level features such as the sum of squared differences of patch pixel values, which are not effective to fill in holes on complicated structures. Furthermore, they are unable to generate objects that are not in the source image or the source data set i..e it fails when completion requires textures that are not found in the input image.



C. *Context encoder*

An improvement to the patch based technique was using a context encoder. Converting the image from a high dimensional space to a lower dimension and again back to its original size has shown to preserve the anatomy of the image and even generate objects that were never seen before. The only drawback of this approach was that, it didn't respect the local information around the patches of the image and sometimes generated entirely different images to the source image.

	Patch-based	Context encoder	Ours
Image size	**Any**	Fixed	**Any**
Local Consistency	**Yes**	No	**Yes**
Semantics	No	**Yes**	**Yes**
Novel objects	No	**Yes**	**Yes**

We have extended context-based inpainting to large masks, and proposed a context encoder to learn features by inpainting, based on Generative Adversarial Networks (GAN). The Generator network learns to fool the discriminator network while both the networks are updated in parallel. A mean squared error (MSE) loss has been used with a GAN loss.

One of the main issues of GAN is the instability during learning, this is avoided by not training purely generative models and tuning the learning process to prioritise stability. The architecture is heavily optimised for image completion by using 2 discriminators, a global and a local network. This has proven to produce locally and semantically coherent image shown in the later sections.

This approach overcomes the drawback of the previous approaches and has been successful in filling even arbitrarily large holes.

III. Data Description

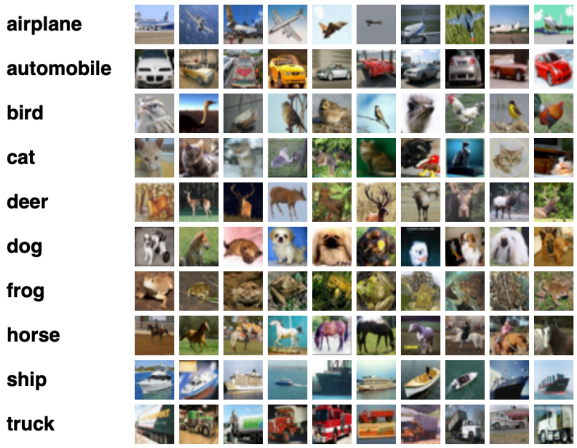
The CIFAR10 dataset has been used to train the network. The dataset consists of 32X32X3 images of 10 classes with 6000 images of each class. 90% the data has been allocated for training and 5% of data has been allocated for testing and validation each.

This dataset has been selected considering the CPU and GPU capabilities of the software to handle smaller images and the generality of the images.

Before we pass the image through the network a rectangular mask is applied onto the image at random location and the original image along with the masked image is fed through the GAN.

.

Identify applicable funding agency here. If none, delete this text box.



IV. Approach

CNN's have been used for the image completion task. A single generator network has been used for the completion task and a local and global discriminators have been used to verify if the image has been completed correctly. During training, a masked image is fed through the generator and completed by it. The completed image is fed through the discriminators and the errors are calculated and backpropagated. Only by training the networks simultaneously it is possible to realise a generator that can complete the task of inpainting.

A. *The Architecture*

The Architecture of the network consists of 3 neural networks - A generator, a local discriminator and a Global discriminator. All the neural networks are using convolutional layers.

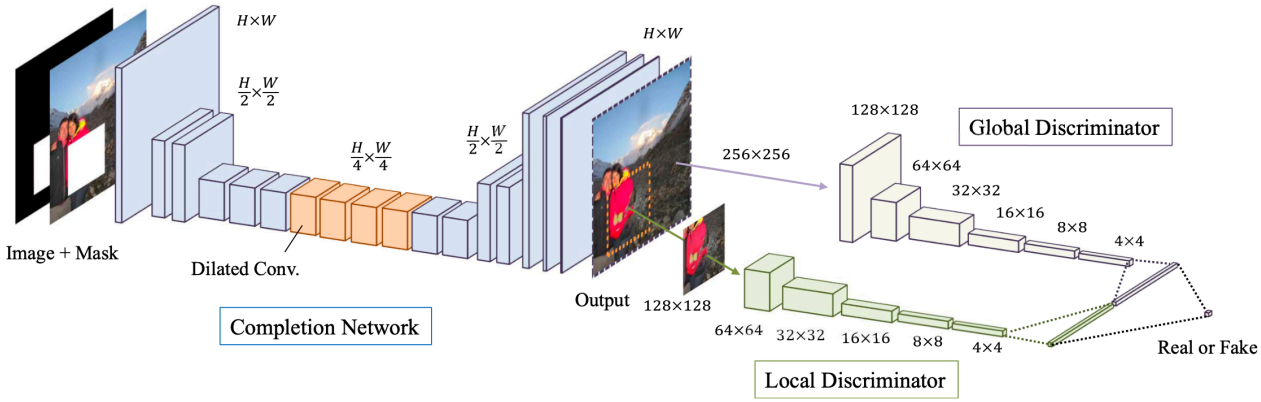
1) The generator -

The general architecture of the generator has an encoder-decoder structure. This allows for reducing memory usage and computational time as the resolution is decreased before further processing. The image is regenerated by deconvoluting the encoded output.

The generator only reduces the resolution of the image twice using strided convolutions of 1/4th the size of the original image. This is being done to generate non-blurred texture in the missing regions.It is recommended to use some

Type	Kernel	Dilation (η)	Stride	Outputs
conv.	5×5	1	1×1	64
conv.	3×3	1	2×2	128
conv.	3×3	1	1×1	128
conv.	3×3	1	2×2	256
conv.	3×3	1	1×1	256
conv.	3×3	1	1×1	256
dilated conv.	3×3	2	1×1	256
dilated conv.	3×3	4	1×1	256
dilated conv.	3×3	8	1×1	256
dilated conv.	3×3	16	1×1	256
conv.	3×3	1	1×1	256
conv.	3×3	1	1×1	256
deconv.	4×4	1	$1/2 \times 1/2$	128
conv.	3×3	1	1×1	128
deconv.	4×4	1	$1/2 \times 1/2$	64
conv.	3×3	1	1×1	32
output	3×3	1	1×1	3

dilated convolutional layers as it allows to compute each pixel output of the missing region using a much larger neighbourhood of pixels.



2) The Discriminators

As established before we have 2 discriminators - local and global discriminators, whose output is concatenated to verify the generated image.

a) The Global Discriminator: The global context discriminator takes as an input the entire image rescaled to 256 × 256 pixels. It consists of six convolutional layers and a single fully-connected layer that outputs a single 1024- dimensional vector. All the convolutional layers employ a stride of 2 × 2 pixels to decrease the image resolution while increasing the number of output filters. In contrast with the completion network, all convolutions use 5 × 5 kernels..

b) The Local Discriminator: The local context discriminator follows the same pattern, except that the input is a 128 × 128-pixel image patch centered around the completed region. The output is a 1024-dimensional vector representing the local context around the completed region.

(a) Local Discriminator

Type	Kernel	Stride	Outputs
conv.	5 × 5	2 × 2	64
conv.	5 × 5	2 × 2	128
conv.	5 × 5	2 × 2	256
conv.	5 × 5	2 × 2	512
conv.	5 × 5	2 × 2	512
FC	-	-	1024

(b) Global Discriminator

Type	Kernel	Stride	Outputs
conv.	5 × 5	2 × 2	64
conv.	5 × 5	2 × 2	128
conv.	5 × 5	2 × 2	256
conv.	5 × 5	2 × 2	512
conv.	5 × 5	2 × 2	512
conv.	5 × 5	2 × 2	512
FC	-	-	1024

c) Finally, we get a 2048D vector by concatenating the outputs of both the discriminators and this vector is fed through a single layer neural network to produce a continuous output between [0,1] by taking the sigmoid of the output. A value close to 1 means that the completed image is a valid image or real and a value close to 0 means that the completed image is not valid or unreal.

B. Training

In order to train the network two error functions have been used - the mean squared error and the GAN loss to improve the realism of the results. Using a combination of the error functions allows for stable training of the high performance model.

The first step of training involves masking the incoming image with a random rectangle and feeding it through the generator network. The network outputs an image which is checked with the original and the error is back propagated through the generator. The discriminator is frozen and the generator is updated during the process. MSE is used as the loss function

Next, the discriminator network is fed original image for which the output is supposed to be 1. Error is calculated and back propagated. The generator is frozen and only the discriminator is updated here. MSE is used as the loss function

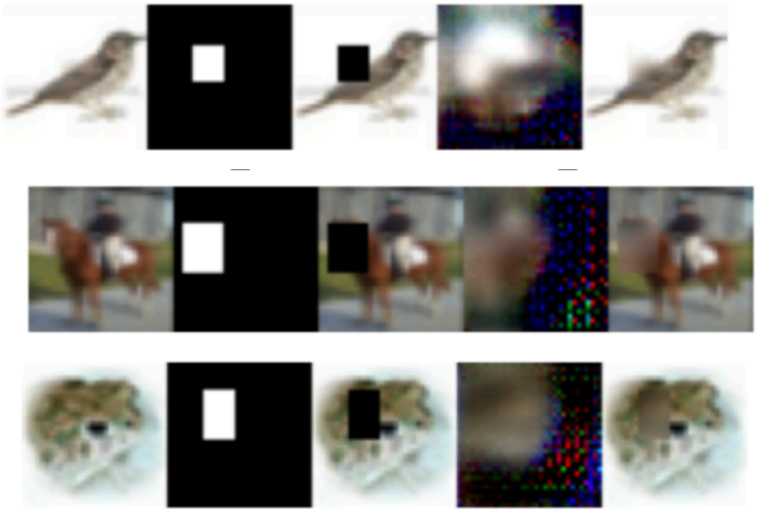
Next, the completed or generated image is fed through the discriminator network. The local discriminator gets the masked part of the generated image and the global discriminator gets the entire image. The ideal output is supposed to be 0 as any image generated by the generator network is a fake one and the error is calculated accordingly and backpropagated . The generator is frozen and the discriminator is updated. MSE is used as the loss function.

Finally, the GAN loss is calculated for the generator. We use the output at the discriminator to find this error. For a generated image w.r.t the generator, the output at the discriminator is supposed to be 1 as it's job is to fake the discriminator. The error is calculated and back propagated through the discriminator and then through the generator. During this back propagation the weights of the discriminator are frozen and the weights of the generator are updated.

As the epochs progress, the generator loss goes down as it learns to fake the discriminator and the discriminator loss goes up

V. RESULTS

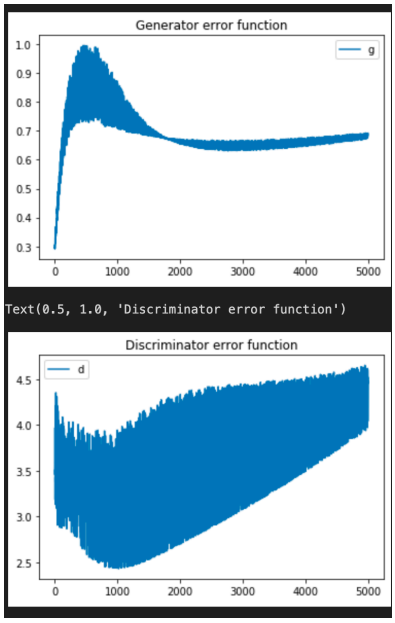
It can be seen that the images have been generated pretty well esp the sparrow and the toad. But the image in which the horse's head was masked wasn't generated well. This is because of a number of reasons firstly, the network hasn't been trained for long because of the computational constraints. Secondly, it's being trained only on a general dataset which is not good enough for the generator to learn to generate the face of a horse specifically.



(a) Input (b) Ours (ImageNet) (c) Ours (Places2)

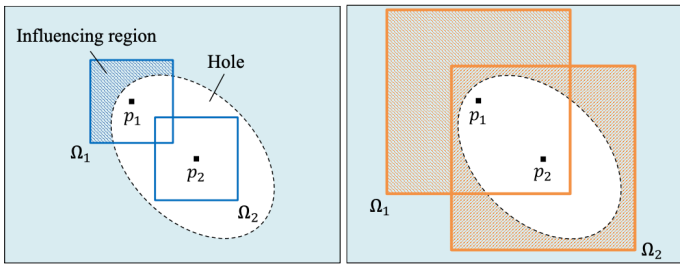
Above are some results of using different datasets to train the GAN. It can be seen that different datasets produced different kind of results. So training the GAN with a specific dataset for a specific purpose is essential to producing good results.

The Discriminator's error doesn't necessarily decrease and we can see that it started going up after a few epochs. The generators error goes down after a few epochs.



VI. LIMITATIONS AND FUTURE SCOPE

Although the model can handle various images with arbitrarily sized holes, it fails when the holes are too big. Consider the image below -



In the first case if we try to generate the second pixel p2, it can be seen that the influencing neighbourhood is basically the hole itself which doesn't have much information. Any attempt to generate p2 will fail cause of this bad spatial support. A possible solution for this would be to use more dilated convolution layers. In the second case, dilated convolution layers have been used which have greatly enhanced the neighbourhood of the p2 which can be used to re-generate the pixel.

Another limitation of this model is regenerating an image whose mask is at the border of the image. Re-generating these pixels is difficult because of the lack of a neighbourhood at the border.

VII. CONCLUSION

In this paper I have shown that by using global and local discriminators it is possible to train a GAN to complete images realistically and unlike other approaches it is even possible to generate novel objects that don't appear anywhere in the image.

1. Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell and Alexei A. Efros. **Context Encoders: Feature Learning by Inpainting.** In *CVPR 2016*. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
2. https://machinelearningmastery.com/how-to-develop-a-conditional-generative-adversarial-network-from-scratch/
3. Iizuka, S., Simo-Serra, E., & Ishikawa, H. (2017). Globally and locally consistent image completion. ACM Transactions on Graphics (TOG), 36, 1 - 14.