A Project Report
on

Image Restoration

carried out as part of the course CS1634 Submitted by

By Manan Uppal

179301112

vi-CSE

Towards the partial fulfilment for the Award of the Degree

of

BACHELORS OF TECHNOLOGY

In

Computers Science and Engineering

2020-2021

MANIPAL UNIVERSITY
JAIPUR

INSPIRED BY LIFE

Under the guidance of **Mr. Sandeep Chaurasia**

Department of Computer Science and Engineering

School of Computing and Information Technology
Manipal University Jaipur
Jaipur, Rajasthan

<u>CERTIFICATE</u>

This is to certify that the project entitled **Image Restoration** is a bona fide work carried out as part of the course ***Minor Project CS-1634***, under my guidance by ***<u>Manan Uppal</u>*** student of Bachelor Of Technology (B.Tech.) in Computer Science & Engineering (CSE) at the Department of Computer Science & Engineering , Manipal University Jaipur, during the academic semester *VI of year 2019-20.*

Place:CSE DEPARTMENT

Date: Signature of the Instructor (s)

ABSTRACT

I want to find out a way to restore old images.now i had two ways first to choose from either take pre-determined machine learning algorithms for the desired the result or use neural networks, the latter one is better and gives far better results because deep networks have remarkable classification performance that can often surpass human accuracy. Generative adversarial networks are typically used for this sort of implementation, given their ability to “generate” new data, or in this case, the missing information.Now we can use already present algorithms like FMN etc. these algorithms give results but are not as good as the results received by neural networks.So the question now arises what neural networks to use, the answer to it was CNN and the results received by the CNN’s accompanied with denoisers give great results.But they had difficulty where the images were almost faded where more guessing work was required than repairing work so I turned to GAN networks.GAN’s was the best choice in pictures where hope of restoration was bleak but GAN’s worked almost upto the expectation.So what I did was I present a generative image in-painting system to complete images with free-form mask and guidance. The system is based on gated convolutions learned from millions of images without additional labelling efforts. The proposed gated convolution solves the issue of vanilla convolution that treats all input pixels as valid ones, generalises partial convolution by providing a learnable dynamic feature selection mechanism for each channel at each spatial location across all layers Results on automatic image in-painting and user-guided extension demonstrate that this system generates higher-quality and more flexible results than previous methods. This system helps user quickly remove distracting objects, modify image layouts, clear watermarks and edit faces hence **Restoring** the image.

TABLE OF CONTENTS

TABLE OF FIGURE

1.INTRODUCTION

We've all heard the saying A picture is worth a thousand words. But is a tarnished image with gaping holes or splotches or blurs worth a few hundred? What if you just found an age-old photograph of your grandparents' wedding, but the surface was so worn that you could barely make out their faces. Or perhaps you got photobombed in what would otherwise have been the perfect picture.

In-painting is the process of reconstructing lost or deteriorated parts of images and videos.

Traditional forms of image restoration usually evolve around some simple concept. Given a gap in pixels, fill the gap with pixels that are the same as, or similar to, neighbouring pixels. These techniques are generally dependent on various factors and are more efficient for removing noise or small defects from images. They will most likely fail when the image has huge gaps or a significant amount of missing data.so I want to use latest approach using neural network to reduce noise and other defects to bring out the best picture.The approach started with trying to attempt CNN based denoiser and using it to analyse how much better is the image is restored .it worked nicely with images which have few fades but with images that were too old CNN didn't work as good.So I planned to use GAN's cause they "generate" and hence better guess and fill the blanks.You can use these to test on normal images by blanking any portion of images but it should not be more than 265x265 cause bigger gap than this causes loss.

1.1 Scope of work

The final aim of this project is to have a web application with the neural network built in so that people can directly visit the site and restore their images.

- ✦ For the above to happen we need to work on improving our model as the research in this field progresses so will the model improve and Better results with more accuracy and detailing.
- ✦ This can have other application like removing watermarks and those corporates who import their watermark for earning more money this tool can help them fix their document images etc.
- ✦ We all have images that could have been perfect if it wasn't for the stranger standing or our friend photobombing a perfect picture.Our model can try removing distraction or purposely added noise right now the final result in this field can be improved but it still gives satisfactory results

2.**MOTIVATION**

My grandparents have lots of old memories captured in photos.but even those can't withstand the test of time.due to past technology the paint fades away leaving us with a splotch image where we can't figure the background or the person.old images of lost loved ones ,victim of time.Random passerby photobombing what could be a perfect picture, all these problems exist and with the help of neural networks I want to tackle these problems

3. LITERATURE REVIEW

Well it started with searching about what tools are available for manipulating an image and what to use to restore images.

- The first thing found was machine learning which uses ML algorithms to calculate the faded pixel by calculating the nearest pixel and their value the closer the pixel the more effect It had on the faded pixel these were classified under discriminative method for solving problem those were fast but were not flexible at all

- **1.**Then on further research I find that CNN can be used for this purpose these are kind of model based optimisation which are flexible but they are time consuming so, the question was can I incorporate best of both world such that they are fast and flexible as well this was the purpose of the research for kai shang and his associates and I read what he had to say was trying to train CNN based denoisers that can be attached as a module to model based optimisation to achieve the speed as well as the flexibility. This project was in MATLAB so I tried running his theory what I did was I trained a set of fast and effective CNN denoisers. With variable splitting technique, the powerful denoisers can bring strong image prior into model-based optimisation methods.

From the study it was concluded that The learned set of CNN denoisers are plugged in as a modular part of model-based optimisation methods to tackle other inverse problems. experiments on classical IR problems, including deblurring and super-resolution, have demonstrated the merits of integrating flexible model-based optimisation methods and fast CNN-based discriminative learning methods for in-painting.It was noted that using half-quadratic splitting technique technique to make a modular CNN denoisers worked really well in removing noises and hence better in-painting this was also chosen because we needed to make the model to work on RGB images and previous techniques were only successful only on grey images so this was chosen. Why CNN were chosen were because First, the inference of CNN is very efficient due to the parallel computation ability of GPU. Second, CNN exhibits powerful prior modelling capacity with deep architecture. Third, CNN exploits the external prior which is complementary to the internal prior of many existing denoisers such as BM3D.So all this was great here is the figure of the structure below

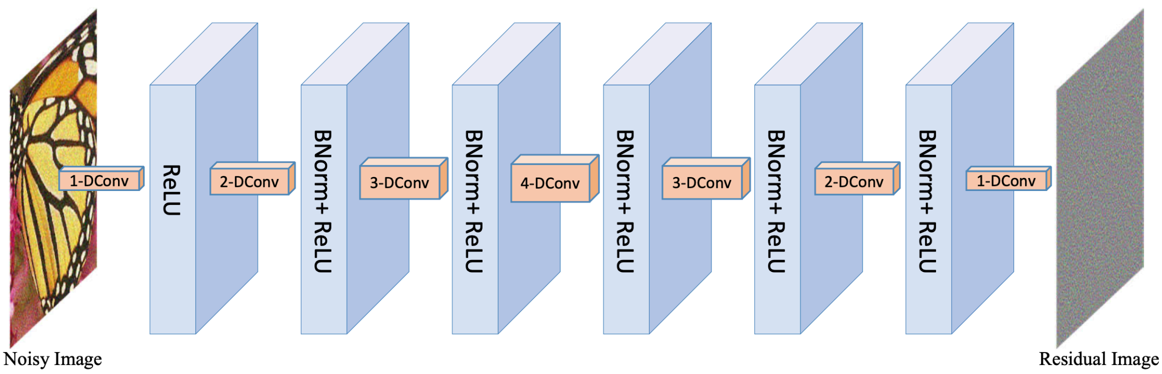


fig.1

The architecture of the proposed CNN denoiser is illustrated in Figure 1. It consists of seven layers with three different blocks, i.e., “Dilated Convolution+ReLU” block in the first layer, five “Dilated Convolution+Batch Normalisation+ReLU” blocks in the middle layers, and “Dilated Convolution” block in the last layer. The dilation factors of (3×3) dilated convolutions from first layer to the last layer are set to 1, 2, 3, 4, 3, 2 and 1, respectively. The number of feature maps in each middle layer is set to 64.this structure provided a fast flexible model but the issue with it was its accuracy greatly decreases when the area faded is too big in that cases it fails to give results

2.Next now we had 2 major problem with the above 1st was the easy one the project was made on MATLAB though MATLAB is good it still lacks the functionality and better maths implementation so we had to switch to python the second major drawback was that the above model failed to reach tackle images with bigger fades or wholes for this we realised that CNN weren’t sufficient we wanted something that can guess the missing piece of the image and guessing means self-generate so we needed a neural network that could learn to generate the missing piece that is when I study about GAN(generative Adversarial Neural Networks) that generates after being trained on a dataset.now the question was is GAN a better option over CNN. On researching it was found that the texture the realness that GAN’s bring about is far superior to the one generated by CNN’s but at the cost of huge computation time.But I think the compromise is beneficial that is why GAN was the better choice than CNN so I opted for GAN’s now the question was what was GAN and how gan worked. So gan has a generator part and a discriminator part and both are actually CNN’s the role of generator is to create new images and the goal of discriminator is to identify wether the image generated is fake or real During the training process, weights and biases are adjusted through back propagation until the discriminator learns to distinguish real images from fake images. The generator gets feedback from the discriminator and uses it to produce images that are more ‘real’. The discriminator network is a convolutional neural network that classifies the images as either fake or real. The generator produces new images through a de-convolutional neural network.

3.1 Outcome of Literature Review

The conclusion drawn is GAN’s are better suited for restoring image as they have the ability to repair images that are on the verge of fading to oblivion and the resulting images are better than those with CNN denoisers

3.2 Problem Statement

Restoring old images or new images removing noises and doing pixel restoration so that we can forever preserve images that were clicked before the digital age and are now degrading because of time

3.3 Research Objective

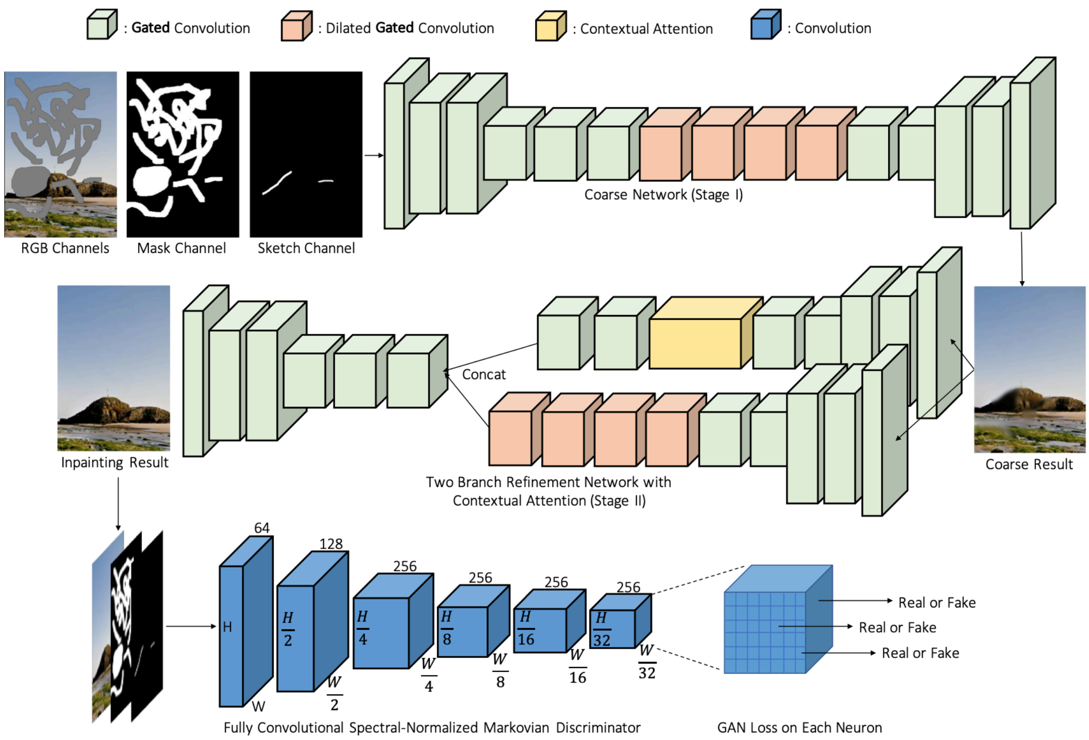
The objective of this research is to try implementing GAN's for pixel restoration for RGB images. We want to see how SnGAN's can be used to implement for pixel restoration.People have used various GAN I want to try using snGAN's which have never been tried for pixel restoration

4. METHODOLOGY AND FRAMEWORK

4.1 System Architecture

We customise the generative in-painting network by a gated convolution as suggested by few research paper and along with SN-patchGAN loss.We need something for refining the model so we use a simple encoder-decoder.We then faced a problem with additional parameters that didn't had any effect on the results so I tried to slim the model by approximately 20% and there was no obvious performance drop both quantitatively and qualitatively the model can be divided into two parts 1st a coarse stage which generates random mask on the image and then the image is passed in the coarse network which is combination of gated convolution and dilated gated convolution it gives out a result that is coarse and needs refinement that is when this coarse result goes to second stage in the refinement network it is a two branch concatenated structure which

fig.2



4.2 Algorithms,Technique etc.

The algorithm to automatically generate free-form masks is important and non-trivial. The sampled masks, in essence, should be

- Similar to mask drawn in real use-case
- Diverse to avoid over-fitting
- Efficient in computation and storage
- Controllable and flexible

A simple algorithm to automatically generate random free-form masks on-the-fly during training. For the task of hole filling, users behave like using an eraser to brush back and forth to mask out undesired regions. This behaviour can be simply simulated with a randomised algorithm by drawing lines and rotating angles repeatedly. To ensure smoothness of two lines, we also draw a circle in joints between the two lines.below is example of how masks can be drawn over the images.One side it is human made mask on the right its mask generated by the free-form mask generator

fig.3

fig.4

- SN-PatchGAN is proposed for the reason that free-form masks may appear anywhere in images with any shape. Previously introduced global and local GANs [15] designed for a single rectangular mask are not applicable. We pro- vide ablation experiments of SN-PatchGAN in the context of image in-painting in SN-PatchGAN leads to significantly better results, which verifies that (1) one vanilla global discriminator has worse performance, and (2) GAN with spectral normalisation has better stability and performance

- Previously, Miyato et al proposed a normalisation technique called spectral normalisation (SN). In a few words, SN constrains the Lipschits constant of the convolutional filters. Spectral norm was used as a way to stabilise the training of the discriminator network. In practice, it worked very Yet, there is one fundamental problem when training a normalised discriminator. Prior work has shown that regularised discriminators make the GAN training slower. For this reason, some workarounds consist of uneven the rate of update steps between the generator and the discriminator. In other words, we can update the discriminator a few times before updating the generator. For instance, regularised discriminators might require 5 or more update steps for 1 generator update.To solve the problem of slow learning and imbalanced update steps, there is a simple yet effective approach. It is important to note that in the GAN framework, G and D train together. In this context, Heusel et al introduced the two-timescale update rule (TTUR) in the GAN training. It consists of providing different learning rates for optimising the generator and discriminator. The discriminator trains with a learning rate 4 times greater than G - 0.004 and 0.001 respectively. A larger learning rate means that the discriminator will absorb a larger part of the gradient signal. Hence, a higher learning rate eases the problem of slow learning of the regularised discriminator. Also, this approach makes it possible to use the same rate of updates for the generator and the discriminator. In fact, we use a 1:1 update interval between generator and discriminator.
- Here is the algorithm for SN-GANs

- Initialize $\tilde{\boldsymbol{u}}_l \in \mathcal{R}^{d_l}$ for $l = 1, \ldots, L$ with a random vector (sampled from isotropic distribution).
- For each update and each layer l:
  1. Apply power iteration method to a unnormalized weight W^l:
  $$\tilde{\boldsymbol{v}}_l \leftarrow (W^l)^{\mathrm{T}} \tilde{\boldsymbol{u}}_l / \|(W^l)^{\mathrm{T}} \tilde{\boldsymbol{u}}_l\|_2 \tag{20}$$
  $$\tilde{\boldsymbol{u}}_l \leftarrow W^l \tilde{\boldsymbol{v}}_l / \|W^l \tilde{\boldsymbol{v}}_l\|_2 \tag{21}$$
  2. Calculate $\bar{W}_{\mathrm{SN}}$ with the spectral norm:
  $$\bar{W}^l_{\mathrm{SN}}(W^l) = W^l / \sigma(W^l), \text{ where } \sigma(W^l) = \tilde{\boldsymbol{u}}_l^{\mathrm{T}} W^l \tilde{\boldsymbol{v}}_l \tag{22}$$
  3. Update W^l with SGD on mini-batch dataset $\mathcal{D}_M$ with a learning rate α:
  $$W^l \leftarrow W^l - \alpha \nabla_{W^l} \ell(\bar{W}^l_{\mathrm{SN}}(W^l), \mathcal{D}_M) \tag{23}$$

✦ When we use SnGAN here is the code for it we directly call it in our research but here is the algorithm/code that runs behind

```python
def _l2normalizer(v, epsilon=1e-12):
    return v / (K.sum(v ** 2) ** 0.5 + epsilon)

def power_iteration(W, u, rounds=1):
    '''
    Accroding the paper, we only need to do power iteration one time.
    '''
    _u = u

    for i in range(rounds):
        _v = _l2normalizer(K.dot(_u, W))
        _u = _l2normalizer(K.dot(_v, K.transpose(W)))

    W_sn = K.sum(K.dot(_u, W) * _v)
    return W_sn, _u, _v

def compute_spectral_normal(self, training=True):
    # Spectrally Normalized Weight
    if self.spectral_normalization:
        # Get kernel tensor shape [batch, units]
        W_shape = self.kernel.shape.as_list()

        # Flatten the Tensor
        W_mat = K.reshape(self.kernel, [W_shape[-1], -1])  # [out_channels, N]

        W_sn, u, v = power_iteration(W_mat, self.u)

        if training:
            # Update estimated 1st singular vector
            self.u.assign(u)

        return self.kernel / W_sn
    else:
        return self.kernel
```

5. WORK DONE

* Thanks two various online I tried the code for in-painting with code being available for SnGANs online . I had to add a file for the mask generator as stated above and then I used the model and trained it on images of celebrities using free online dataset named Celebhq A for training my model to fill in the faces if something is missing, as shown in the figure below
* The next step was to train it to fill in the faded background in those old images that is why I used online data set places which has images of backgrounds with different landscapes
* I also had to use toolkit neuralgym downloaded from GitHub its a toolkit for Tensorflow library
* The dataset all were used in flint format so we had to write code for converting your .jpg and .png to flist format
* The model was trained for 1,00,00,000 it took 10 hrs approx. to train this model to expect first result it was a 2 day training session
* The masks should not be more than 128x128 cause bigger than that and the results deppriciates

Below are the results obtained by my model:-

fig.5.1

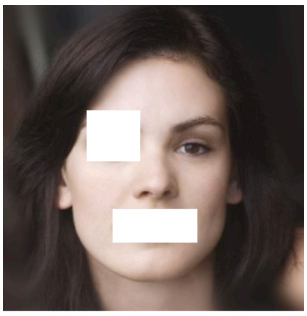


fig.5.2

fig5.3

Fig.6.1

fig6.2

fig.6.3

Fig.7.1

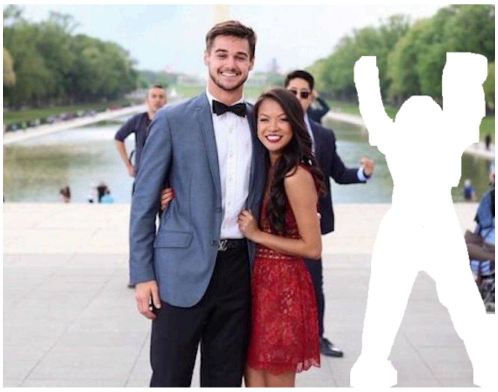


fig.7.2

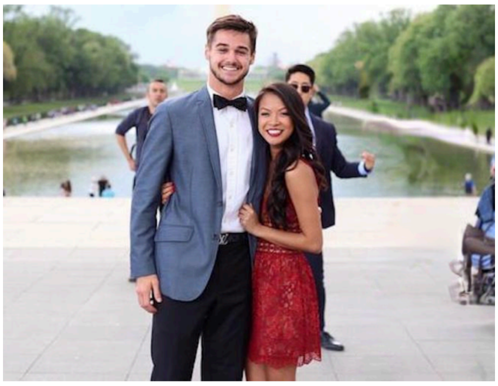


fig.7.3

Fig.8.1

fig8.2

fig.8.3

6. CONCLUSIONS AND FUTURE

We demonstrated that gated convolutions significantly improve in-painting results with free-form masks and user guidance input. We showed user sketch as an exemplar guidance to help users quickly remove distracting objects, modify image layouts, clear watermarks, edit faces and interactively create novel objects in photo it is seen that the final image has better quality and better results compared to vanilla discriminator or partial convoluted models.This model shines particularly well incase of free hand drawing of mask where the error chances are 2% although its not the best which is 1.2% but it still gives good results and in future can be used as web application or an app for fixing our images

6.REFERENCES

[1] Alec Radford & Luke Metz, Soumith Chintala (2016). “ UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS ”. https://arxiv.org/pdf/1511.06434.pdf

[2] Christian Ledig, Lucas Theis, Ferenc Husz ́ar, Jose Caballero, Andrew Cunningham,Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe ShiTwitter (2017) . “ Photo-Realistic Single Image Super-Resolution Using a Generative AdversarialNetwork ”. https://arxiv.org/pdf/1609.04802.pdf

[3] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, Bryan Catanzaro (2018) . “ High- Resolution Image Synthesis and Semantic Manipulation with Conditional GANs ”.

https://arxiv.org/pdf/1711.11585v2.pdf

[4] Talha Iqbal, Hazrat Ali (2018). “ Generative Adversarial Network for Medical Images(MI-GAN) ”.

https://arxiv.org/pdf/1810.00551.pdf

[5] Salome Kazeminia , Christoph Baur, Arjan Kuijper, Bram vanGinneken, Nassir Navab, Shadi Albarqouni, Anirban Mukhopadhyay (2019) . “ GANs for Medical Image Analysis ”. https://arxiv.org/pdf/1809.06222.pdf

[6] Jason Liu, Max Spero, Allan Raventos (2017). “ Super-Resolution on Image and Video ”.

http://cs231n.stanford.edu/reports/2017/pdfs/312.pdf

[7] Krupali Ramavat, Prof. Mahasweta Joshi, Prof. Prashant B. Swadas (2016) . “ A survey of Super resolution Techniques ”.
https://www.irjet.net/archives/V3/i12/IRJET-V3I12238.pdf

[8] Sapan Naik, Nikunj Patel (2013). “ SINGLE IMAGE SUPER RESOLUTION IN SPATIAL AND WAVELET DOMAIN ”.
https://arxiv.org/ftp/arxiv/papers/1309/1309.2057.pdf

[9] Nao Takano, Gita Alaghband . “ SRGAN: Training Dataset Matters ”.

https://arxiv.org/ftp/arxiv/papers/1903/1903.09922.pdf

[10] Boris Kovalenko (2017). “ Super resolution with Generative Adversarial Networks ”.

http://cs231n.stanford.edu/reports/2017/pdfs/17.pdf

[11] Ian Goodfellow (2017). “ NIPS 2016 Tutorial:Generative Adversarial Networks ”.

https://arxiv.org/pdf/1701.00160.pdf