Generative Image Inpainting using Deep Learning

A project report submitted in partial fulfillment of the requirements for B.Tech. Project

Integrated Post Graduate

by

Ujjwal Sharma (2017IMT-108)

विश्वजीवनामृतं ज्ञानम्

ABV INDIAN INSTITUTE OF INFORMATION TECHNOLOGY AND MANAGEMENT
GWALIOR-474 010

2020

CANDIDATES DECLARATION

We hereby certify that the work, which is being presented in the report, entitled **Generative Image Inpainting using Deep Learning**, in partial fulfillment of the requirement for the award of the Degree of **Bachelor of Technology** and submitted to the institution is an authentic record of our own work carried out during the period *July 2020* to *November 2020* under the supervision of **Prof. Rajendra Sahu**. We also cited the reference about the text(s)/figure(s)/table(s) from where they have been taken.

-

Date: 7th*Novemeber*, 2020 Signatures of the Candidate

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Date: Signatures of the Research Supervisors

ABSTRACT

Image Inpainting has been one of the most ancient problems in the field of Computer Vision. The Utility of this problem lies from the field of surveillance to creation of competent datasets . In real life applications, the complexity of some computer vision tasks is increased due to some corrupted or missing values of pixels in images. On this scale, it is very difficult to estimate the value of pixels in the missing regions. A lot of models have been proposed to solve the problem of missing pixels and address the problem of large missing values. However in this field, getting satisfactory results is somewhat complex. In this Btech. Project, We have proposed a solution approach that use architectures based on Deep learning that helps to solve this problem. Generative Adversarial Networks are highly supportive and helpful for the major task of image completion. Therefore, in GANs,a trained Least Squares GAN (LSGAN) architecture has been utilized for completion of missing parts of images which recreates the missing portion by generating the image closest to the corrupted image in coarse patches. This is done by subjecting the image generation of LSGAN to Perceptual and Contextual Loss which generates a realistic looking image similar to the data distribution of image. After this a Refinement network is trained on images with noise to remove the noise which additionally enhances the quality of resultant images utilizing an Auto-Encoder Network procedures and hence provide complete and enhanced pictures for computer vision applications. The AutoEncoder Model also helps to generalize some outlier results generated by the Generative Network. This model is inspired by Context Encoders and Progressive Inpainting approach.The Experimental outcomes show that the proposed approach improves the Peak Signal to Noise ratio and Structural Similarity Index values by 2.5% and 2% than the existing Techniques in use.

Keywords: Auto-Encoder Network, Deep learning, Image Completion,Image Enhancement, Least Squares GAN .

ACKNOWLEDGEMENTS

We are gravely indebted to **Prof. Rajendra Sahu** and we are extremely obliged for giving us the autonomy of functioning and experimenting with our ideas. We would like to take this opportunity to express our profound gratitude to him not only for his academic guidance but also for their personal interest in our project and constant support coupled with confidence boosting and motivating sessions which proved very fruitful and were instrumental in infusing self-assurance and trust within us. The nurturing and blossoming of the present work is mainly due to their valuable guidance, suggestions, astute judgment, constructive criticism and an eye for perfection. Our mentor always answered myriad of our doubts with smiling graciousness and prodigious patience, never letting us feel that we are novices by always lending an ear to our views, appreciating and improving them and by giving us a free hand in our project. It's only because of their overwhelming interest and helpful attitude, the present work has attained the stage it has.

Finally, we are grateful to our Institution and colleagues whose constant encouragement served to renew our spirit, refocus our attention and energy and helped us in carrying out this work.

(Ujjwal Sharma)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

AUC	Area under the curve
CNN	Convolutional Nueral Network
Conv2DT	Convolution 2 Dimensional Transpose
CVAE	Conditional Variational Auto Encoders
dB	Decibels
drop	Dropout
GAN	Generative Adversarial Networks
GPU	Graphical Processing Unit
HQ	High Quality
lrelu	Leaky Relu
LSGAN	Least Squares Generative Adversarial Networks
MSE	Mean Squared Error
NSGAN	Non Saturated Generative Adversarial Network
PCR	Principal Component Regression
PSNR	Peak Signal to Noise Ratio
SSIM	Structural Similarity Index
VAE	Variational Auto Encoders
WGAN	Wasseteranian Generative Adversarial Network

CHAPTER 1

INTRODUCTION AND LITERATURE SURVEY

This section covers the detailed description of image inpainting and enhancement, wherein we predict what the missing pixels in the image using Deep Learning Methods. We also discuss the literature survey associated with this work.

1.1 Introduction

Some of the most common issues in the field of Deep Learning and Processing of images are Image Completion and Image Enhancement. Image Completion is getting a lot of attention due current trend in usage and requirement for this technology. Although Completion of Images has been researched a lot in the past 2 decades, there is still a long path to travel to achieve perfectness in image completion. Work is required not only on filling the missing image but also make it as similar as the existing image. So, we can say that this process is highly Subjective in nature. Many techniques exist that take care of low level features and try to handle Image Completion and Image Enhancement separately [24]. However, these issues may occur concurrently. Solving these tasks separately causes poor results and results in an increase in complexity.

Generative Adversarial Networks[6] are one of the best methods for Generative modeling involving Deep Learning methods like Convolutional Neural Networks(CNNs)[21]. Finding patterns in data and picking out relevant points from the latent space to recreate a distribution that has to be modeled is one way of explaining Modeling using Generative Networks [23]. GANs work in an Unsupervised way that uses 2 models. The generator and the discriminator. The Task of the generator model is to generate samples by picking up points from latent space in a particular manner. The function of Discriminator is to label the fake examples generated by the generator network. The Discriminator can be trained separately but the generator is trained using the output of the discriminator as the goal of generator is to fool the discriminator that its images are real. When the generator does this without errors, we can say that the generator has been trained. Now, this generator is used to generate fake samples which appear real.

In order to measure GANs for image completion, we use 2 metric functions.These are Structural Similarity Index(SSIM)[10] and Peak Signal to Noise Ratio(PSNR) [16].The better the value of these metrics the better is the image completion model. On one hand, Image completion tries to fill the data in the missing part correctly and on the other hand, the refinement networks focuses on improving the quality of images. Least Squares Generative Adversarial Network (LSGAN)[18] has been used for image completion and enhancement network used is inspired by Auto-Encoder Network which refines the quality of the image. LSGAN tries to recreate the image closest to the given missing image which is done by relating the image to least square loss function which is in turn refined by the Auto-Encoder[19] refinement network.

Generative Networks pick local points from latent space which are basically random points relating to some distribution. A Trade off between 2 models that are Generator and Discriminator leads to filling of the missing pixels.So, inevitably, we see that loss is Incorporated in the outputs of the generator model. This is one of the main reasons we need a Refinement network that tries to minimize this loss using the metric Mean Square Error(MSE)[8]. Gradually, we get a refined and smooth image that can be related to the original existing image.

All of this is based on probabilistic frameworks. These models have been trained on CelebA HQ Database [17] and All the models work in unison to provide the best results of Completion. We need to decrease the least square distance between the 2 images. The lesser the distance, the more the generated image is similar to the given image.

If we carefully go through the intertwining between Image Completion and Image Enhancement, we would conclude that advancements in one of these fields affect the progress of the second one. We can say that better Image Completion lead to more realistic Image Enhancement Techniques as the distribution of data gets better and the noise generated is less.

Generative Image Inpainting basically is combination of the following tasks together :-

1. Analysing and Converting the Given Incomplete Image and generating Feature Maps and Importance Matrix.
2. Using LSGAN to generate the image closest to the importance matrix
3. Minimizing the Contextual Loss
4. Using Auto-Encoders to generate the final completed image

As the modern methods of Image Completion strongly use Deep Learning, the most common model used in most of the techniques is NSGAN[6] which is the normal and the most initial GAN created. The Methods also use Context Encoder Models to complete images. This is done using an Encoder and Decoder model.

Image Completion is a very difficult task even for humans when there is no context present. The Image Completion is an active research area with a lot of interesting techniques addressing the problems and sub problems of the domain.Most of these techniques are overwhelmingly used in recreation/ enhancement/ restoration of art paintings. These maybe degraded or even damaged by excessive aging, moisture and temperature and other natural factors which leads to a great cultural loss. This can be both Supervised and Unsupervised.

The Previous work in the domain of Image Inpainting included background completion or the main image completion. However, current methods in production suffer from some major issues like generation of distorted images, object removal problem , blurry results for completed part, inconsistency for extra large regions and coarse patches, inconsistent image distribution identification and lack of target specific generation.

In our thesis, we propound a deep learning model compromising of Convolutional Layer based models which are Least Square Generative Adversarial Networks which extracts the important features from the background and generates an image according to the importance matrix. The Image generated is the most contextually image near to the given incomplete image.The Auto-Encoder model further makes enhancement to the given importance matrix and corrects the given outlying features to minimize the perceptual loss.

The Contributions made by our work that make the proposed model unique from the current existing methods and induces the sense of Novelty are given as follows :-

1. It completes the images solely based on visual input provided by using Extracted Features and deriving Importance Matrix of the incomplete image.

2. The Model maintains greater and better variational diversity for Image Completion for the given target distribution.

3. New structure that incorporates the image completion of image using LSGAN combined with the output of Auto-Encoder Model.

4. Better Pose Detection and Better Quality Image completion over existing GANs and stabler learning and adaptation process.

5. Clubbing the problems of Image Completion and Image Refinement together using multipurpose models with better Quality of Image Completion than the current models using Deep Learning to deal with this problem.

1.2 Literature Review

Image Inpainting has been a field of Brisk research since its discovery. Since then, Some previous works have been done which brought major changes in the field of Image Completion. The existing work in the field uses some features embedded in the image and normally work for a single mask. On Such works has been proposed by Chen and Fu who put forward an approach for image inpainting which has been of significance in terms of methods. This model is also called Progressive Inpainting using Generative Models [4] where the models first predicted the full missing region. This missing region was then completed using a Refinement network to make image quality better. But this model could not handle the issue of large missing regions.

Ahuja and Jia-Bin have proposed an advance-knowledge approach which applied contextual information for image completion titles "Image Completion using Planar Structure Guidance" [11] .But this also shows poor results if the image is of very high quality and if the missing region is very large. This approach tried to use structural cues to complete missing region using patch generation and image transformation. This was also not a generative approach to complete images.

Zhao, Liu, and Hiang proposed deep neural networks to inpaint and denoise the corrupted image. This was titled "“A deep cascade of neural networks for image inpainting, deblurring and denoising" [26]. However, this model failed to produce good similarity in structure. Even the texture of the generated image was not upto the mark.

A good model was also developed which used Context Encoders [19]. This model was developed by Deepak Pathak. This method helped to complete image using Encoding and Decoding where first the images were encoded to a linear vector and then decoded again to the final completed image. However the images generated were noisy and not clear. They were also blurry but this was a good breakthrough.

Introduction of GANs and VAEs [22] has led to improvement in the field of image inpainting. CVAEs [2] are also used for image completion but these also use conditional labels. These labels are used to interpret what the completed images look like and this conditional generation generates better outputs. This technique is called Variational Image Completion. However, on a practical scale there is an absence of labels and the results produced were blurry for very large regions.

Another approach developed by Chuanxia Zheng,Tat-Jen Cham,Jianfei Cai regarding the pluralistic inpainting [27] has been developed that creates multiple possible solutions for the existing missing regions using GANs. However, this method is a bit slow and the training needs a lot of existing data.

The Recent Methods proposed using image to image translations can produce different results for the same image however, in such cases, mappings are required to get pixel to pixel values and better representation.However, getting such mapping is not possible to be obtained for image inpainting where the labels that represent the missing portions are the missing regions themselves. So, different outputs for the same image have to be generated using extreme targeting for domain specific image completion].

1.3 Gaps in literature

I. **Structural Continuity** - The existing models face the problem of maintaining sensible continuity in the images such that the completed image is comparable to the original image. As the problem is unsupervised in nature, There is a limitation of this type because the models take very varying amount of structures to base the pixels.

Although this may not seem significant but this can often lead to a lot of discontinuity and thus result in image with pixel distribution very different from the image's distribution. Out approach focuses on correct pose generation that results in structural continuity for the image.

II. **Calculating correct Distribution for Pixels** - A large amount of existing models fail to estimate the correct distribution of the missing areas because of generalizing nature over a large context. The more the model is generalized, the more we have a hard time to calculate the distribution of pixels
Our LSGAN model does directional generation where we train the model for a certain set of distribution and this helps our model to create the missing pixels better and more near to the correct Distribution by targeting a problem specifically.

III. **Dealing with Large Missing Patches** - This remains one of the most complex parts of image inpainting. The larger the missing region, more is the probability of different type of distributions that are in it [9]. It is virtually impossible to get a sense of exact distribution for the pixel and this creates a very unsolvable problem.
Large Patches may even contain some objects that cannot be identified by the model for completion as this objects information is completely lost. We have tried to solve the problem of large patches relative to the other works done.

IV. **Time Required for Inpainting** - For the inpainting of different parts of video, we actually need to cut the video into frames and inpaint each and every frame. The frames need to be presented in real time and so we need a method which is faster to converge and faster to generate the missing regions.

1.4 Problem Statement

Current methods of image inpainting suffer from major issues like large size of corrupted images or if the image of very high dimensions which resulted the algorithms to generate low quality and unsatisfactory images. Image inpainting is also difficult because a corrupted image may contain pixels which are of drastically different distributions than the pixels of non corrupted part of the image. As the information is permanently lost,it is impossible to generate exact missing portions. So, the results provide a resemblance.

1.5 Motivation

Image Completion is an essential problem when it comes to restoration of art forms or for surveillance. Though some methods have been developed and are currently in use for image completion, the results generated may not be accurate or even adequate to solve the modern problems [13].

The field is currently undergoing different approaches so as to get a better understanding on how we can deal with such problems and how we can complete images with industry level competency. This too is a hard task as we are dealing with the terms of complete data loss and we don't know what has been lost exactly. So, the issue to determining the complexity of the underlying image and the quality is also a grave issue.

Some practical areas where image inpainting is overwhelmingly being used are removal of unwanted images from background, basic implementation in leading soft wares are Photo-shop and Canva, Professional Photography, Restoration and Recreation of Art Work. The main aim is to get images as closest to the ground truth as possible.

This research aims at the Generative Inpainting of Images using Deep Learning approaches. For this the proposed research work will try to improve the stated gaps in the literature and come up with an optimum solution for the same. Keeping in mind that it will decrease the respective loss functions and increase the accuracy and resemblance of images to the original images. The Model will train on the samples as well as penalize samples even for correct convergence.

1.6 Objective

The literature on image inpainting suggests that overall performance of generative models can be improved by applying appropriate architectures and loss functions and work can be done to further enhance the generated image to look more perceptual and contextual. The Aim is to further improve the performance the Generative Adversarial Networks to reduce Perceptual and Contextual Losses by adding networksfor refining images.

The generated image most similar to the ground truth is enhanced using Deep Convolution Networks to improve SSIM and PSNR than the current exisiting standards. To maximize this resemblance is the main objective of our Model.

CHAPTER 2

DESIGN DETAILS, IMPLEMENTATION AND TESTING

2.1 Schematic Diagram

The sequential steps involved in the process are shown here. The flow diagram, Figure 2.5, encapsulates the different stages involved in the work.

Table 2.1 encapsulates the dimensions of the inputs and outputs of each layer.

2.2 Implementation Details

2.2.1 Dataset

The dataset used to train the model is the CelebA Dataset [17] which contains images of Faces of a wide variety of images. It consists of more than 200,000 images of celebrity

Table 2.1: Model Architecture

Layer	Size/Stride/Pad	Levels	Output
Generator Conv2DT 1024/lrelu/drop	32,3,3	1	4x4x1024
Generator Conv2DT 512/lrelu/drop	32,3,3	1	8x8x512
Generator Conv2DT 256/lrelu/drop	32,3,3	1	16x16x256
Generator Conv2DT 128/lrelu/drop	32,3,3	1	32x32x128
Encoder		4	1x8092
Decoder		4	64x64x3

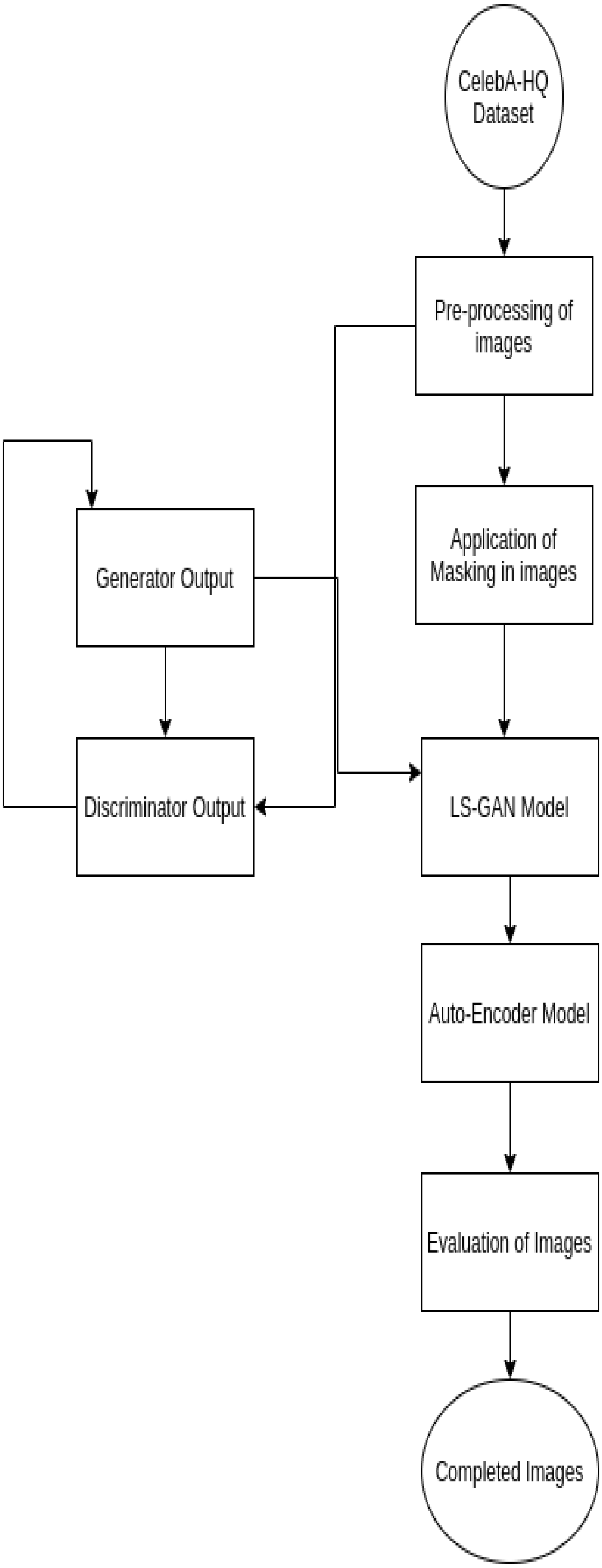

Figure 2.1: Proposed work-flow diagram

faces that provides the vast feature scales. The images are of size 64*64*3. The total size of the Dataset is 1.40 GB and contains High Quality Images. This is also called Celeb-A HQ Images.

To train the AutoEncoder model, the LSGAN model is run in parallel and the completed image produced by LSGAN is sent and refined. This image is compared to the original image in order to reduce the mean squared error between the 2 images.

2.2.2 Pre-Processing

Before feeding the Images into our LSGAN [18] model,the images are first normalized in the range of (-1 , 1).Out of these 200,000 images, only 20,000 images have been used to train and test the models. The shape of training set is (20000,64,64,3) and the test set is (100,64,64,3).

Thus in this pre-processing step,Image is broken into a suitable range and the importance matrix is also derived which is thereby input to the LSGAN. The Image generated by AutoEncoder model is modified according to the missing portions and only the missing part is pasted in the actual image to give the complete refine image.

2.2.3 LSGAN Model

Generator and Discriminator models are 2 different Convolutional Neural Networks which help to learn features of the given images and predict according to the desired answer which is ensured by applying a Contextual Loss (Based according to the pixels surrounding the missing regions) and Perceptual Loss (The filled position results are Normal and not too far from expectation from human understanding).

This LSGAN is trained on the CelebA-HQ Dataset while minimizing the loss function. The benefits of LSGAN is that unlike the normal Generative Adversarial Networks, it penalizes samples even when they are correctly [18]. Now, if we fix the discriminator during the training of the generator, it means that the decision boundary is also fixed. When we train the discriminator as well as the generator we are basically shifting the decision boundary and this means we get more closer to the desired results.

Generator and Discriminator models are 2 different Convolutional Neural Networks which help to learn features of the given images and predict according to the desired answer which is ensured by applying a Contextual Loss(Based according to the pixels surrounding the missing regions) and Perceptual Loss (The filled position results are

Normal and not too far from expectation from human understanding).

The images that are in the training set are first passed to the pre-trained LSGAN. The images in training set are first masked before feeding into the LSGAN and then the masked images are passed to the model. The Model fills up the importance matrix and tries to create an image which is closest to the generated image. Satisfactory results are shown in 100 epochs and takes less time with GPU. The image is recreated by minimizing the square loss function which is also clipped for better results. The learning rate is kept low for the model to learn at better rate and set proper bench marks. The loss is measured in each iteration and we can see that the loss flattens after 100 iterations ie. there is no improvement in the model.

2.2.3.1 Generator Model

Generator model takes an input of 100 latent points. It provides an output of 64*64*3. In this model, UpSampling techniques have been used using Conv2D Transpose Layers. Batch Normalization has been applied to improve the scalability and add regularization effect. We basically Upsample 1024*4*4 to 64*64*3 step by step that is by converting to 1024,512,256,128 and then 64. This output is similar to a fake generated image and this is passed to the discriminator which tries to classifies this as real or fake. This way, Contextual loss is reduced by the generator.

2.2.3.2 Discriminator Model

Discriminator model is a normal neural network that aims at binary classification of the image. It reduces the binary cross entropy as metric 4 Conv 2D Layers, The image that is 64*64*3 is broken down step by step into a linear vector space of 8192 size. These features are now used to find whether the given image is a real or a fake image. This is usually trained separately but can also be trained along with the generator. Shuffled images are passed of the generator as well as train set along with labels to that Discriminator identifies the images as real or fake. The below given is the Loss function of Least Squares GAN as cited in the research paper 'Least Squares Generative Adversarial Networks'.

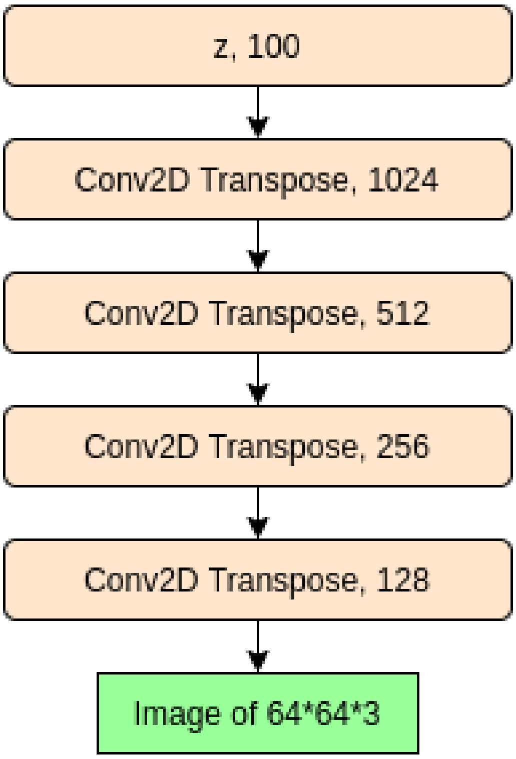

Figure 2.2: Architecture of Generator for LSGAN

$$minV_{LSGAN}(D) = \frac{E_x[D(x) - b^2]}{2} + \frac{E_x[D(G(x)) - a^2]}{2} \tag{2.1}$$

$$minV_{LSGAN}(G) = \frac{E_x[D((G(z))) - c^2]}{2} \tag{2.2}$$

D : Discriminator

G : Generator

V$_{LSGAN}$: $LSGANLossFunction$

E[] : Expectation Function

x : value of x

z : Value of latent points

When both of these models are clubbed together, we desire convergence according to the loss function provided above. So, LSGAN continually penalizes the samples irrespective of whether they are converging correctly or not. This helps LSGAN create better quality images as this is the main requirement for LSGAN. To get the model get the nearest image as per the background distribution of image, we use clipping of the perceptual and the contextual loss. The trained LSGAN is provided with this image,

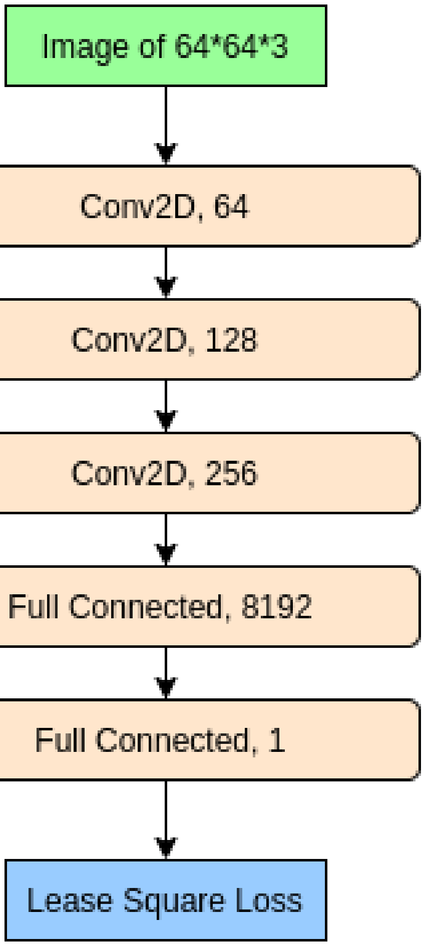

Figure 2.3: Architecture of Discriminator for LSGAN

importance matrix, all the latent points as well as the masked images to find the loss for each iteration. When we run the session with these images, we see that the loss is clipped as we do this by calculating an additional *zhat* for each iteration. This is clipped according to the learning rate and the desired momentum. So, *zhat* is updated in each iteration. Slowly, the graph flattens out and there is no improvement. We have compiled the GAN using Adam Optimization Algorithm [15] and this is applied for the Stochastic Gradient Descent.

The Final Cost function used for the models are given below which is the normal output of the specific loss function error. In this case, this is the least squares error which are clipped together with the help of $zhats$. The technique that mainly differentiates GAN from Variational AutoEncoders and Boltzmann machine [20] is the estimation of unique ratios that help to determine the problem of Mode Collapse. This ratio is $\frac{P_{data}(x^i)}{P_{model}(x^i)}$. This is repeatedly calculated by the Discriminator and is collaborated with the generator which helps the loss of both the models to converge. This is the main reason, we call the discriminator as the Adversary.:-

$$J^{(D)}(\theta^G, \theta^D) = \frac{E_x * log(D_x) - E_z * log(1 - D(G(z)))}{2} \tag{2.3}$$

By penalizing every loss, the Discriminator loss makes sure that there is no Overfitting and Generator loss makes sure the there is no overfitting in this case. This Also prevents the chances of Mode Collapse [25] and hence LSGAN can produce high quality images.

2.2.3.3 Minimum Maximum Game in LSGAN

We need to train the discriminator in order to accurately predict whether the given image is modelled by the generator or not. At the same instant, we train the generator to make images that easily fool the discriminator [5]. So, the discriminator and the generator are in constant tussle and we need to maintain proper losses in order to make sure that the model is being trained correctly. If the loss of Discriminator goes very low, this means we have overfit the data. If the loss of Generator goes very low, this means that the generator is easily fooling the discriminator by making garbage images and hence less training. This is somewhat easier for LSGAN as the given equation is easy to maintain and mostly gives very stable results.

$$min_G max_D V(D, G) = \frac{E_{x\epsilon data}[D(x) - b^2]}{2} + \frac{E_{z\epsilon data}[D(G(z)) - c^2]}{2} \tag{2.4}$$

The reason there is a need to discuss mode collapse and training of LSGAN in detail is because the training of GANs is done on the basis of an optimization game which can have effects on training of the data [3]. This led to the search of efficient methods that have better and more stable training's like LSGAN and WGAN-GP [7]. Due to randomization, it is impossible for the data distribution generated by LSGAN to reach the exact resemblance of the exact data distribution of data. This is the main reason there is a bit of noise in the image generated by the generator. Addition of more noise leads to garbage images which are of no use practically and for training purposes. So, this can be a failure mode for GAN which is easily overcome using LSGAN. While choosing the GAN, we tried different types of GANs and out of all these GANs, LSGAN showed the most promise. One of the special features of LSGAN is that when highly trained, it can predict the pose of the person in the image and this will lead to better structural and perceptual similarity based on the unmasked portion of the image. This is the power of this technique.

2.2.4 Auto-Encoder Model

The Images produced by Generative Adversarial Networks are generated from latent points which are random points in space as per some distribution. So, inevitably, some noise is added in the images produced by LSGAN. Now, it is the responsibility of Auto-Encoder model to refine the image so that it resembles the ground truth. An image is a set of pixels where some set of pixels together represent some features that are interpreted by human eye to form an understanding. Convolutions are performed of these input frames followed by pooling layer which is then passed through ReLu layer. For classification, this result is passed to activation layer to obtain the final result.

This model takes input in the form of image ie. 64*64*3 and responds with a new and refined image of the same dimensions ie. 64*64*3. This includes 2 parts and these are Encoder and Decoder. The function of Encoder part is to break the image of (64,64,3) into linear dimensions which means basically encoding all the data of the image to a relevant layer of size 8192. Now, all of these 8192 features are picked up by the Decoder Model which tries to recreate a refined image by using UpSampling. This means that 8192 linear dimensions are converted back to (64,64,3) and this image is more refined and smoothed. This process is called Decoding. The Auto-Encoder model basically aims at minimizing the Mean Square Error and produce images near to our understanding and hence helps to reduce the Perceptual Loss. This model is one of the most important part of this process as it can correct some vague results produced by LSGAN model and derive features from it which can be correctly decoded.

The AutoEncoder network [1] has been trained and it can easily make the importance matrix and enhance the given images. The main function of AutoEncoders is dual in nature. They help to learn better visual structural cues and help to reduce noise and treat the outlying images made by the generator. It makes sure that the final image generated has the least perceptual loss. The final image generated is made by taking only the portion of incomplete image and is pasted on the missing part. This gives way better Structural Similarity Index(SSIM).

$$n_{out} = [\frac{n_{in} + 2p - k}{s}] + 1 \tag{2.5}$$

n_{in}: number of input features

n_{out}: number of output features

K : size of Kernel

p : padding size

s : stride

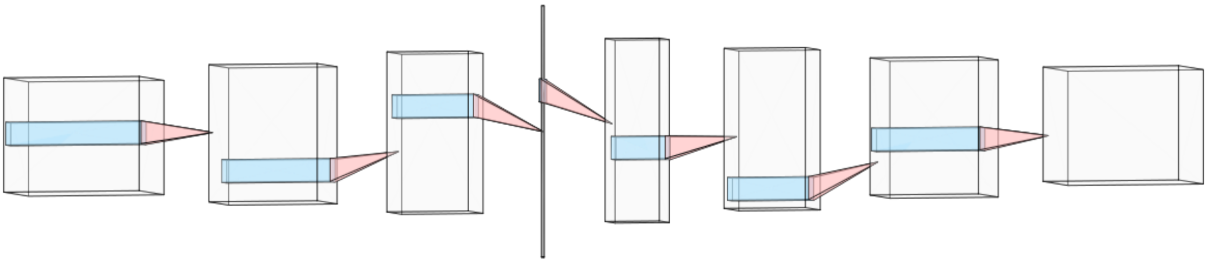

Figure 2.4: Proposed Encoder Network

When the image is totally encoded, we move above the sense of all missing portions. The major enhancement is applied when we are decoding the encoded image. The learnt rates help to make the image similar to the distribution of data using Convolution Operations and it is these weights that make the Enhancement possible [14]. As the data is learnt to a great extent by the AutoEncoder, so it helps to deal with the outlier issue and even corrects it.

2.3 Image Completion Methodology

The Main feature that separates our work from the existing techniques is the combination of Least Squares Generative Adversarial Network along with AutoEncoder network both of which work toward reducing the Contextual and Perceptual Loss and provide realistically completed images. This type of image completion is called Generative Inpainting as we first develop the image closest to the existing image and then refine it. The Existing Methods use normal GAN [12] that produces low quality images for image completion and stop just after single generation. There is still noise present and hence result in poor performance when the metrics are calculated. The Existing models use Context Encoders to find the distribution of data which results in blurry results.

However we combine these 2 models and get great results. The LSGAN provides high quality image and better convergence with penalization at each step and AutoEncoder removes the existing noise. This is the Salient Feature of our thesis as no existing technique combines these 2 models for the same purpose.

For Image completion technique we quantify the existing loss functions and make a final function that helps in converging the input. Clipping additionally helps to reach the find the final distribution closest. Clipping can also be represented as follows :-

$$\hat{z} = argmin_z(Loss_{contextual}(z)) + \lambda Loss_{perceptual}(z) \tag{2.6}$$

The value of is chosen so as to get the best results. In our example, we have chosen as 10. This can be seen as a hyper parameter and can be tuned as per the requirement. As the LSGAN works in an unsupervised fashion and the whole model requires the complete image for training, so the domain that we put this whole model is "Semi-Supervised Learning". So, out model provides probability distributions of missing data in this way.

$$x_{completed} = Mask * y + (1 - Mask) * (Y_{AutoEncoder}) \tag{2.7}$$

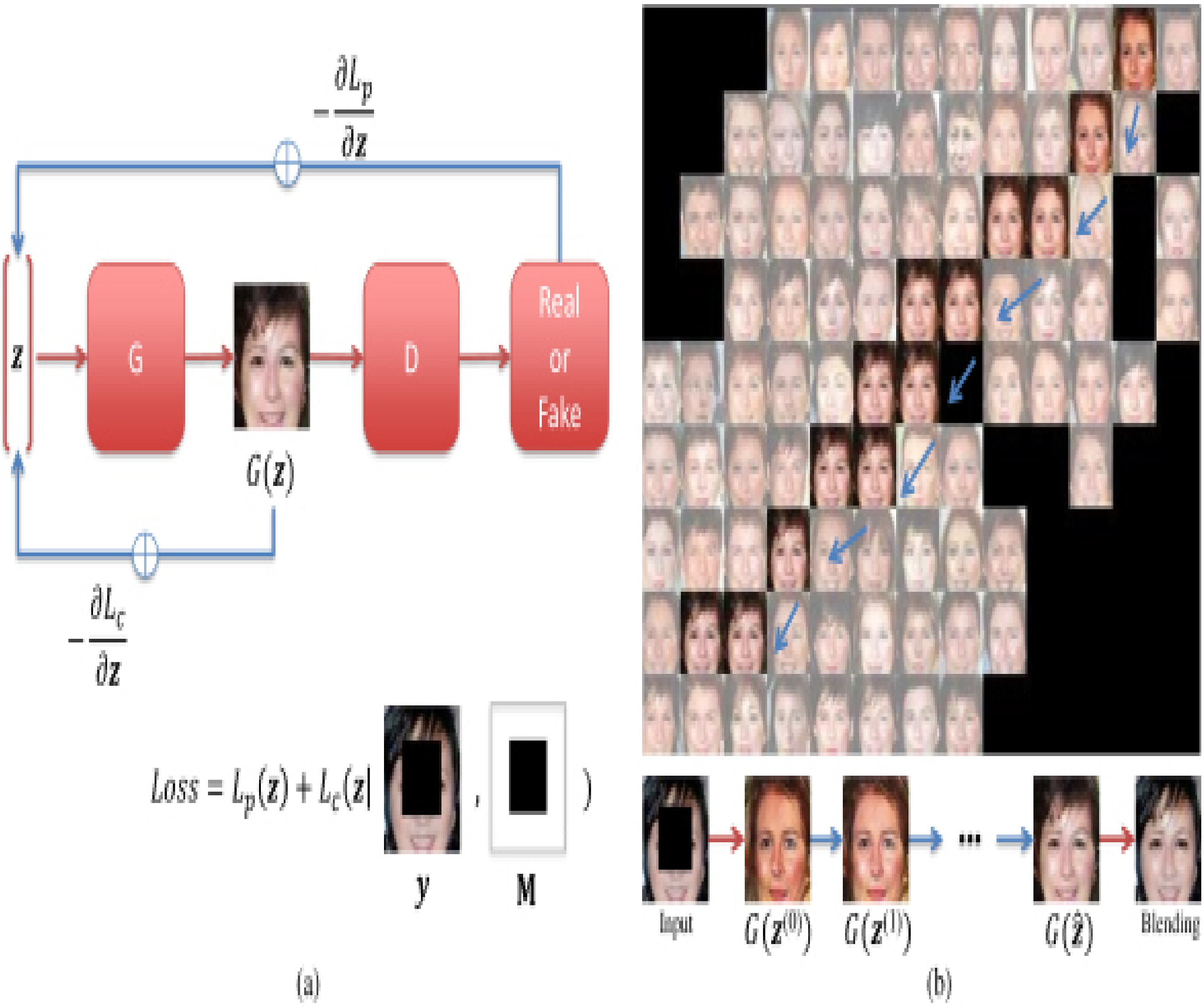
$-\frac{\partial L_p}{\partial z}$
z
G
G(z)
D
Real or Fake
$-\frac{\partial L_c}{\partial z}$
Loss = L_p(z) + L_c(z| ,)
y
M
(a)
Input
G(z^{(0)})
G(z^{(1)})
...
G(\hat{z})
Blending
(b)

Figure 2.5: Proposed Image Completion Architecture in "Semantic Image Inpainting with Perceptual and Contextual Losses." using Perceptual and Contextual LossesChen and Hu (2018)

2.4 Output

A completed and refined image with minimized Contextual and Perceptual loss by minimizing metrics of Structural Similarity Index(SSI) and Peak to Noise Signal Ration(PSNR). This is close to the actual ground truth and human perception.

CHAPTER 3

RESULTS AND DISCUSSION

3.1 Results

We compared LSGAN + AutoEncoder Models with the currently existing models. The results are summarized in Table 3.1.

Our model completes the images and also enahnces the quality by reducing the existing noise. Based on the below metrics, the LSGAN model tries to generate image closest to the given image. This generated image is then enhanced using the Auto-Encoder Network. This is because of the noise induced in the generated images of the latent points. This noise has an effect on the structural similarity of the generated images which is around 0.80 - 0.82 along with a PSNR in the range of 0.20-0.22. When this is passed to the trained AutoEncoder model, we see a spike in the SSIM as well as PSNR which is around 0.87-0.89 and 0.21-0.23 respectively.

The Existing Industry Standards provide the Structural Similarity 0.85 and 0.87 and a PSNR near 0.21 and 0.22. So, this model performs at par with the current industry standard and in some cases crosses the set bar. So, we saw an improvement in the image quality by around 1.5% - 3% when compared to Pluralistic inpainting and Context Encoders.

We can see how the graph is obtained for the total loss by combining the contextual and the perceptual loss as well as for the training of AutoEncoder Network. The Models converge fine and so we can say that they have been trained in a proper fashion.

Below We can see the comparison between some established techniques and our model and we can compare the performance based on these metrics. :-

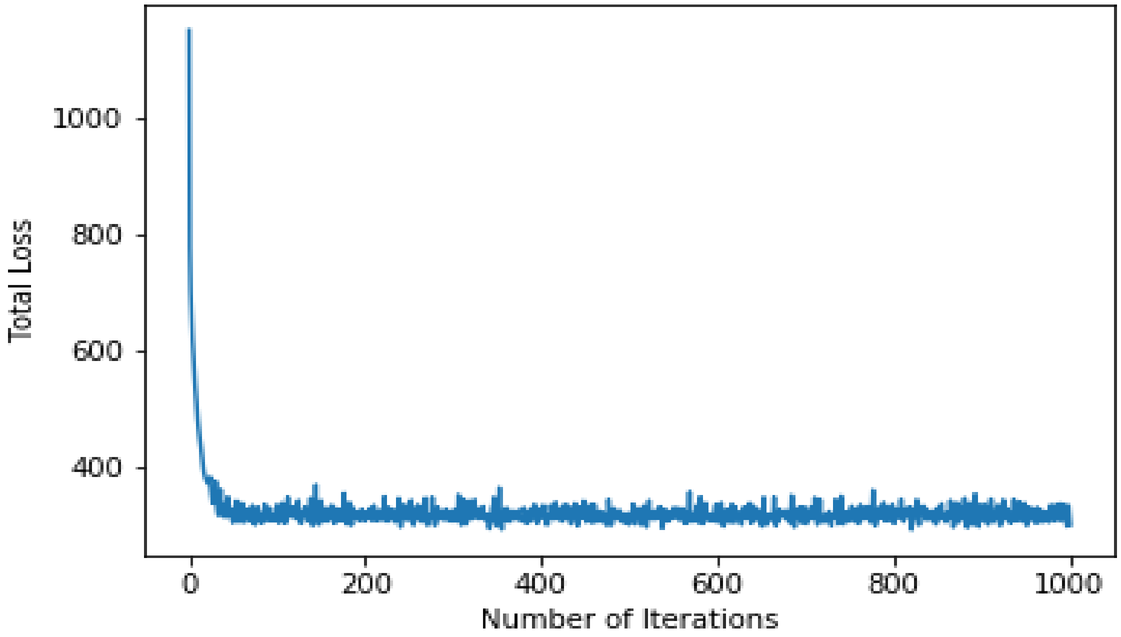

Figure 3.1: Total Performance Loss of LSGAN in terms of Perceptual and Contextual Loss to calculate the nearest neighbouring image using the Least Squares Distance

3.2 Discussion

3.2.1 Contextual Loss

We need to make sure that the generated image and the refined image has the same state of affairs. This is done to make sure that the generated pixels by the generator (G(z)) are as similar as possible to the actual missing or uncorrupted pixels. To do this, we simply take a pixel wise difference between the 2 images that are the generated ones and the actual complete image which is not corrupted.

$$Loss_{contextual} = ||M.G(z) - M.y||_1$$

And here $||x||_1$ is the l1 normalization of some vector.

3.2.2 Perceptual Loss

The main focus of this loss is to ensure that the output looks real. This means that the output has to be near the ground truth. The image generated can be similar the actual image but it can have a variant distribution which is very different from the actual image

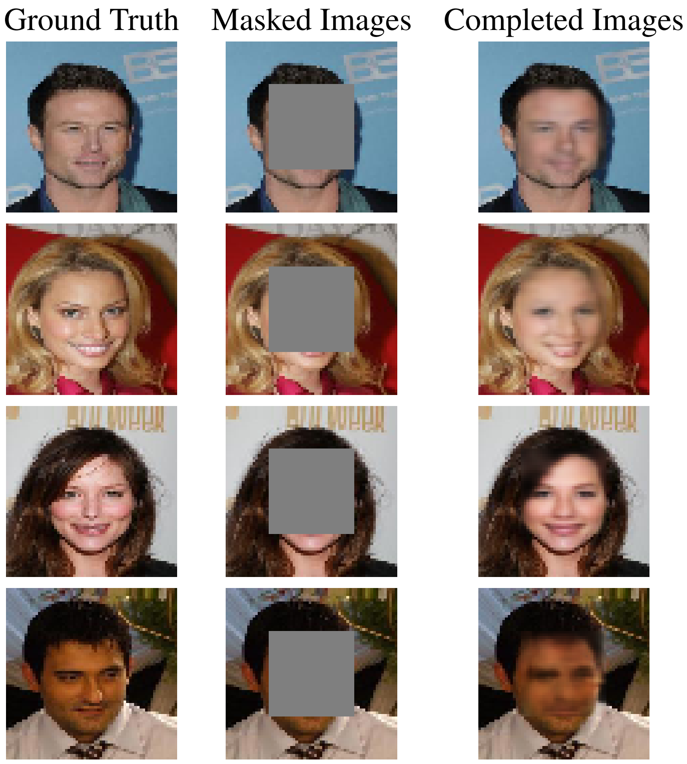

Table 3.1: Errors for Test and Validation Set

Inspection of Metrics		
Model	**SSIM**	**PSNR**
Context Encoders	0.872	22.85
Pluralistic Inpainting	0.856	21.79
LSGAN	0.817	19.30
LSGAN+AutoEncoder	**0.883**	**22.30**

Table 3.2: Evaluation of Metrics for different models

in perception. So, this type of perceptual loss is used and the results due to this loss makes the model more presentable.

$$Loss_{perceptual} = log(1 - C(G(z)))$$

3.2.3 Total Loss

This is the final loss which is calculated by adding the weighted perceptual loss and contextual loss.

$$Loss_{z} = Loss_{contextual}(z) + QLoss_{perceptual}$$

3.3 Understanding the Evaluation Metrics

3.3.1 Peak Signal to Noise Ratio

Peak Signal to Noise Ratio is an evaluation metric to measure how good our model performs.The more we achieve the PSNR score, the image quality is more superior. The measure of PSNR is carried out in decibels(dB). It is also based on MSE and calculated by taking log of MSE. This ensures that the value is in decibels.

$$MSE = \frac{1}{mn}\sum_{m}\sum_{n}(x_{mn} - y_{mn})^2$$

m number of real pixels

n number of generated pixels

x_{mn} pixel value

y_{mn} pixel value

$$PSNR(x, y) = \frac{10\log_{10}(\max(\max(x), \max(y))^2}{|x - y|^2}$$

3.3.2 Structural Similarity Index

This metrix can depend on the complexity of the data. It can depend on constrast(C),structural term(S), luminance(I).

$$SSIM(x, y) = [C(x, y)]^a * [S(x, y)]^b * [CIx, y)]^c$$

These approaches of metrics perform very good on the CelebA-HQ dataset when we apply LSGAN along with Auto-Encoder for image completion.

3.3.3 Architecture Details

We implement our model using TensorFlow 1.5. We do not apply layer normalization as it tends to destailize the LSGAN.We use Leaky-Relu Activation function which is the most effective in this case. The sequential architectural layers are given in Table 2.1. The model uses a mini-batch size of 32. The LSGAN has been trained for 100,000 epochs to provide sharp results and the AutoEncoder Model has been trained till 2000 epochs. Batch Normalization has been used in the AutoEncoder Model to prevent the extent of overfitting.

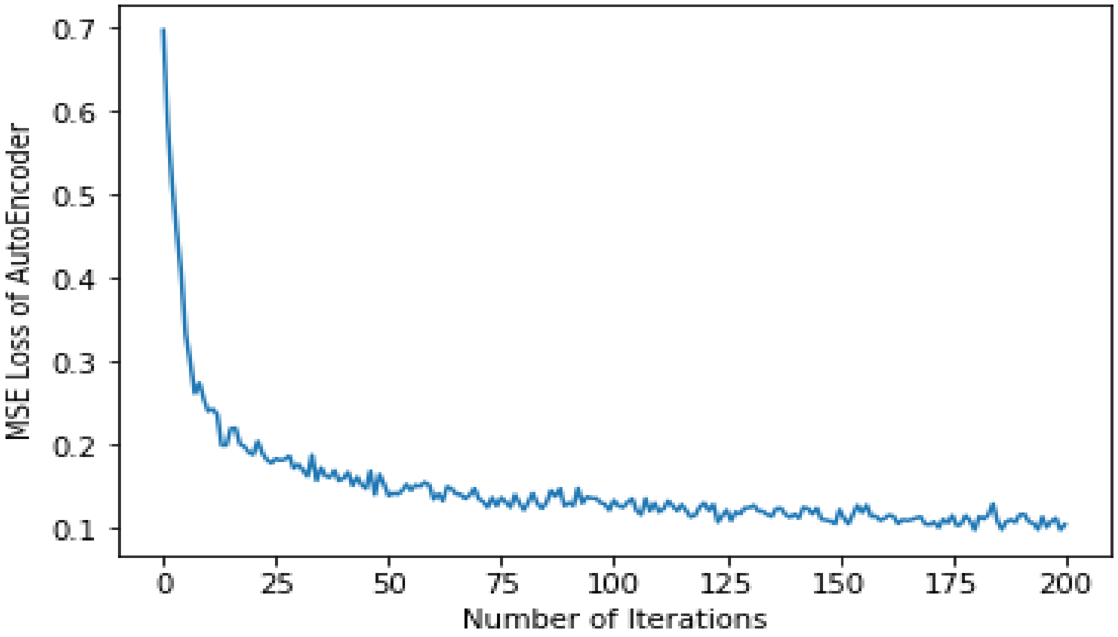

Figure 3.2: Initial Training Graph for the AutoEncoder Model

We compare our model with recent state of the art models: Contextual Encoders, which was one of the breakthrough model for the purpose of Image Completion and it also used the CelebA dataset for training. Along with this we also compare our model to Progressive Inpainting as it used the Contextual and Perceptual loss idea for the first time.

In (Fig 3.1), the minimised objective loss function is shown. Different values of λ between 5 and 10 were tested and the results from the optimal value are presented in the paper. We get the final result through the joint model of LSGAN and AutoEncoder Network which separates our work from the existing techniques.

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 Conclusion

In our thesis, we present LSGAN + AutoEncoder architecture which is an end to end network model, for the purpose of completion of images by generative inpainting. The potency of LSGAN can be demonstrated from the decrease in the error rates and an overall increase in the accuracy. The use of AutoEncoder enables the encoding of spatial and temporal features of the input images which in turn helps to enhance the quality of generated images . LSGAN solves the problem of vanishing gradients encountered and provides a guarantee of better quality image generation and stable training and decrease in loss. Conditional dependence is given by AutoEncoder network which corrects if there are some outlying samples generated by the network.

4.2 Future Work

In future, we plan to extend our work by exploring the following possibilities:

- We intend to extend our existing model not only for completion of faces but for different domains of images present.
- We plan to implement the model on a dataset with more extensive range of facial features to ensure even better results .
- The input images used for training and testing have been taken in a consistent environment. We intend to implement the model on a more noisy environment

and check the accuracy.

REFERENCES

[1] Bank, D., Koenigstein, N. and Giryes, R.: 2020, Autoencoders.

[2] Bao, J., Chen, D., Wen, F., Li, H. and Hua, G.: 2017, Cvae-gan: Fine-grained image generation through asymmetric training, *Proceedings of the IEEE International Conference on Computer Vision (ICCV).*

[3] Barnett, S. A.: 2018, Convergence problems with generative adversarial networks (gans), *CoRR* **abs/1806.11382**.
URL: *http://arxiv.org/abs/1806.11382*

[4] Chen, Y. and Hu, H.: 2018, An improved method for semantic image inpainting with gans: Progressive inpainting, *Neural Processing Letters* **49**, 1355–1367.

[5] Dong, H. and Yang, Y.: 2019, Towards a deeper understanding of adversarial losses, *CoRR* **abs/1901.08753**.
URL: *http://arxiv.org/abs/1901.08753*

[6] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: 2014, Generative adversarial nets, *in* Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger (eds), *Advances in Neural Information Processing Systems 27*, Curran Associates, Inc., pp. 2672–2680.

[7] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A.: 2017, Improved training of wasserstein gans.

[8] Haoyu Ren, J. L. and El-khamy, M.: 2018, Dn-resnet: Efficient deep residual network for image denoising.

[9] He, K. and Sun, J.: 2014, Image completion approaches using the statistics of

similar patches, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **36**, 2423–2435.

[10] Hore, A. and Ziou, D.: 2010, Image quality metrics: Psnr vs ssim, *International Conference on Pattern Recognition* .

[11] Huang, J., Kang, S., Ahuja, N. and Kopf, J.: 2014, Image completion using planar structure guidance.

[12] Ji, J. and Yang, G.: 2020, Image completion with large or edge-missing areas, *Algorithms* **13**(1).
URL: *https://www.mdpi.com/1999-4893/13/1/14*

[13] Jiang, J., Kasem, H. M. and Hung, K.: 2019, Robust image completion via deep feature transformations, *IEEE Access* **7**, 113916–113930.

[14] Kim, J., Song, S. and Yu, S.: 2017, Denoising auto-encoder based image enhancement for high resolution sonar image, *2017 IEEE Underwater Technology (UT)*, pp. 1–5.

[15] Kingma, D. P. and Ba, J.: 2017, Adam: A method for stochastic optimization.

[16] Liu, G., Yang, R., Li, S., Shi, Y. and Jin, X.: 2018, Painting completion with generative translation models, *Springer* .

[17] Liu, Z., Luo, P., Wang, X. and Tang, X.: 2015, Deep learning face attributes in the wild, *Proceedings of International Conference on Computer Vision (ICCV)*.

[18] Mao, X., Li, Q., Xie, H., Lau, R. Y. K. and Wang, Z.: 2017, Least squares generative adversarial networks.

[19] Pathak, D. and Krahenbuhl, P.: 2016, Context encoders: feature learning by inpainting.

[20] Salakhutdinov, R. and Hinton, G.: 2009, Deep boltzmann machines, Vol. 5 of *Proceedings of Machine Learning Research*, PMLR, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, pp. 448–455.
URL: *http://proceedings.mlr.press/v5/salakhutdinov09a.html*

[21] Thompson, R.: 1 April, 2019, Action detection using deep neural networks: Problems and solutions.
URL: *https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529*

[22] Yi, K., Guo, Y., Fan, Y., Hamann, J. and Wang, Y. G.: 2020, Cosmovae: Variational autoencoder for cmb image inpainting.

[23] Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X. and Huang, T. S.: 2018, Generative image inpainting with contextual attention, *CoRR* **abs/1801.07892**.
URL: *http://arxiv.org/abs/1801.07892*

[24] Zarif, S., Faye, I. and Awang Rambli, D.: 2014, Image completion: Survey and comparative study, *International Journal of Pattern Recognition and Artificial Intelligence* **29**, 1554001.

[25] Zhang, Z., Li, M. and Yu, J.: 2018, On the convergence and mode collapse of gan, pp. 1–4.

[26] Zhao, G., Liu, J., Jiang, J. and Wang, W.: 2017, A deep cascade of neural networks for image inpainting, deblurring and denoising, *Multimedia Tools and Applications* **77**.

[27] Zheng, C., Cham, T.-J. and Cai, J.: 2019, Pluralistic image completion, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1438–1447.