Unpaired Image to Image Translation using modified Gnerative Adversarial Network

A Thesis Submitted in
Partial Fulfilment of the Requirements for the Degree of
Master of Technology
in
Artificial Intelligence

by

Sudhir Panda
Registration No. 21-2-2-105

Under the Supervision of

Dr. Dalton Meitei Thounaojam

Department of Computer Science & Engineering
NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR
May, 2023

COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR
(An Institute of National Importance)
SILCHAR, ASSAM, INDIA – 788010
Fax: (03842) 224797 **Website:** http://www.nits.ac.in

Declaration

Thesis Title: **Unpaired Image to Image Translation using modified Gnerative Adversarial Network**
Degree for which the Thesis is submitted: **Master of Technology**

I declare that the presented thesis represents largely my own ideas and work in my own words. Where others ideas or words have been included, I have adequately cited and listed in the reference materials. The thesis has been prepared without resorting to plagiarism. I have adhered to all principles of academic honesty and integrity. No falsified or fabricated data have been presented in the thesis. I understand that any violation of the above will cause for disciplinary action by the Institute, including revoking the conferred degree, if conferred, and can also evoke penal action from the sources which have not been properly cited or from whom proper permission has not been taken.

Signed: **Sudhir Panda**
Date: **21-2-2-105**

COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR
(An Institute of National Importance)
SILCHAR, ASSAM, INDIA – 788010
Fax: (03842) 224797 **Website:** http://www.nits.ac.in

Certificate

It is certified that the work contained in this thesis entitled "**Unpaired Image to Image Translation using modified Gnerative Adversarial Network**" submitted by **Sudhir Panda**, Registration no **21-2-2-105** for the M.Tech is absolutely based on his own work carried out under my supervision.

Place:
Date:

Dr. Dalton Meitei Thounaojam
Computer Science & Engineering
National Institute of Technology Silchar

Abstract

Imagine a world where you can turn a daytime scene into a night time scene, or even change a photo of a horse into a photo of a zebra. Such a world is made possible by image-to-image translation, a task which is the focus of computer vision research for years having a rich history. Over the years, researchers have explored various techniques to achieve this task, from handcrafted feature extraction methods to deep learning-based approaches. Recent breakthroughs in deep learning have enabled the development of powerful generative models that can perform image translation without relying on explicit supervision. In this thesis, we present an overview of the history and evolution of image-to-image translation, from its basic roots to the most recent cutting-edge methods. This thesis focuses on CycleGAN, an unsupervised image translation framework that uses cycle-consistency to discover mappings between two domains in absence of paired data. The goal is to replicate the results of the baseline ResNet-based generator model and subsequently replacing the ResNet block with a U-Net block. ResNet and U-Net generator models produce feature maps as an intermediate, hidden layer output which are abstract representations encoded in a high-dimensional space and cannot be directly visualized or interpreted. This thesis aims to take the advantage of both ResNet and U-Net by concatenating the feature maps obtained from ResNet and U-Net for generating the translated image. Concatenation of feature maps helps in improving the information flow, better localization of the region of interest and helps in obtaining an enhanced and more comprehensive representation of the image. To assess the effectiveness of the suggested system, the work incorporates a convolutional neural network (CNN) model that is trained on real images. This CNN model is then used to classify the translated images generated by the system. The experimental results manifest that the proposed system outperform the existing image translation techniques.

Acknowledgements

I would like to seize this moment to extend my genuine appreciation and warm thanks to my supervisor, **Dr. Dalton Meitei Thounaojam**, who is affiliated with the Department of Computer Science and Engineering at the National Institute of Technology Silchar. I am deeply grateful for his constant inspiration and invaluable guidance throughout every phase of my research project. I have acquired a wealth of knowledge from his mentorship, and I consider myself fortunate to have had the opportunity to work under the guidance of such a supportive and kind individual.

I would like to thank my Panel members for their continuous evaluation and valuable constructive suggestions during this work. I would like to also thank all the faculty members of the Computer Science And Engineering of National Institute of Technology Silchar, for their administrative support during various phases of this work.

I would like to thank my parents, my sister and friends, who always kept faith on me and staying distant apart, it is only their love and blessings, which never made me feel like staying away from them.

Date: Sudhir Panda

Place: Reg. No. 21-2-2-105

COMPUTER SCIENCE & ENGINEERING DEPARTMENT
NATIONAL INSTITUTE OF TECHNOLOGY SILCHAR
(An Institute of National Importance)
SILCHAR, ASSAM, INDIA – 788010
Fax: (03842) 224797 **Web Site:** http://www.nits.ac.in

RECOMMENDATION SHEET

Submission Type (For Evaluation/Record)	:	**Record**
Title of the Thesis	:	**Unpaired Image to Image Tanslation using modified Gnerative Adversarial Network**
Degree (Ph.D./M.Tech/M.Sc./MBA)	:	**Master of Technology**
Specialization:		**Artificial Intelligence**
Name of the Student	:	**Sudhir Panda**
Registration no.	:	**21-22-105**
Name of the Department	:	**Computer Science And Engineering**
Name of the Supervisor	:	**Dr. Dalton Meitei Thounaojam**

(Signature of the Student)

Recommendation of the Examiners
Above thesis is recommended for submission.

Prof. Ajoy Kumar Khan (External Examiner) :

Dr. Gaurav Singh Baghel (Expert from other Department) :

Dr. Partha Pakray (Expert from the Department) :

Dr. Biswajit Purkayastha (Chairperson) :

Dr. Dalton Meitei Thounaojam (Supervisor) :

Signature with Seal of the Head of the Department

For Academic/Department office use
Thesis received on
Signature of the Dealing Assistant/AR

Contents

List of Figures

List of Tables

List of Abbreviations

cGAN Conditional Generative Adversarial Network. 14

CNN Convolutional Neural Network. 9, 17

CRF Conditional Random Field. 9

CycleGAN Cycle-Consistent Generative Adversarial Network. 14, 27

GAN Generative Adversarial Network. 2, 9, 13, 18

ITIT Image to Image Translation. 7, 9–14

MRF Markov Random Field. 10

MUNIT Multimodal Unsupervised Image-to-Image Translation. 15

RBM Restricted Boltzmann Machines. 11

VAE Variational Auto Encoder. 11, 12

CHAPTER 1

Introduction

Images have the power to capture our imagination, evoke emotions, and transport us to different worlds. From paintings and photographs to digital images, the art of visual storytelling has evolved over time. Image-to-image translation is a captivating and demanding domain within computer vision, which involves the task of converting an image from a specific domain into another domain, all while retaining its underlying content and structure.

Imagine you are a painter who has been commissioned to create a portrait of a famous person, but you only have access to a collection of pictures of them in different settings and with different expressions. You could try to manually combine these images to create a single portrait. However, doing so would be tiresome and time-consuming, and the outcome might not exactly capture the core of the subject.

Similarly, consider yourself as a wildlife photographer on an expedition in Africa, and you capture some stunning images of horses running through the Sahara desert. Now you want to turn those horse images into zebras, without losing any of the details present so that it makes the original photo much more captivating.

Imagine you're flipping through your photo album and come across a stunning picture you took during a summer vacation in the mountains. You wish you could see how it would look in the winter, with snow-capped peaks and icicles hanging from the trees. But you didn't take a winter picture, and going back to the same location during a different season is not always an option.

This is where the field of computer vision comes in, where we seek to translate pictures present in one domain to another, such as converting a summer picture to a winter one, or converting images of horses to zebras, or converting photos to portraits. This is a challenging task, but one that is becoming increasingly feasible thanks to the advancement of computer vision techniques.

1.1 Image Translation

Image translation in the field of computer vision is a problem that refers to the mapping of an input image present in domain X to an equivalent output image in a different domain Y. The task is challenging because it requires learning complex mappings between domains. One popular approach for image translation is to use the Generative Adversarial Network (GAN) [1]. The capacity of GAN to acquire intricate domain-to-domain mappings has led to its widespread application for image translation jobs. The GANs model are a deep neural network architecture consisting of two neural networks blocks called generator and discriminator that are trained together in a mini-max game fashion competing against each other all the time. The generator is responsible for producing an image from a random noise vector z, and the discriminator duty is for predicting whether an incoming input image is fake or real. The generator attempts to beat the discriminator by mapping the random input z to an image that the discriminator will predict to be real.

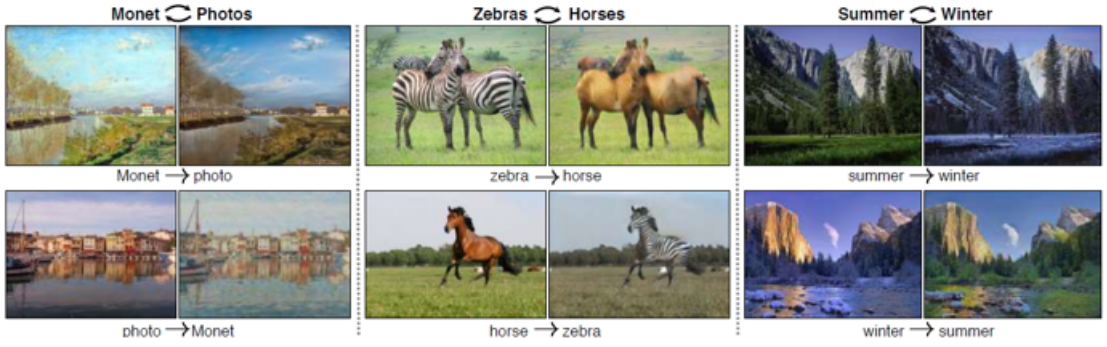


FIGURE 1.1: Image to Image Translation of images from Domain X to Y and vice versa.Dataset taken from [48]

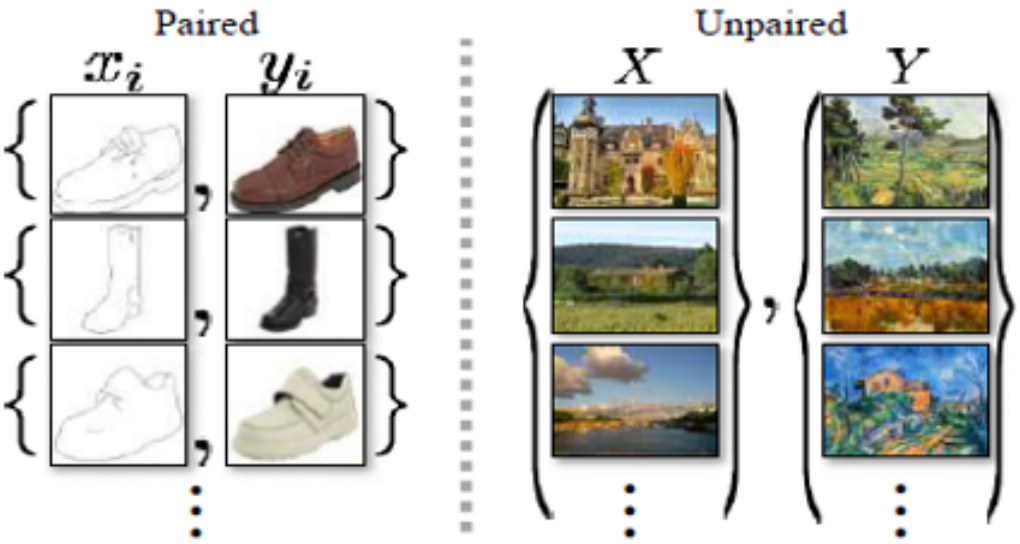


FIGURE 1.2: Unpaired Images(right) and Paired Images(left).

There are basically two approaches for carrying out the image translation task namely Paired Image Translation and other being the Unpaired Image Translation as shown in Figure 1.2.

1.1.1 Paired Image Translation

Paired image translation, also known as supervised image translation, is a technique in computer vision that involves training a machine learning model to translate images present in one field to another using paired data. Paired training data comprises of training examples in the form $\{x_i, y_i\}_{i=1}^{N}$ where the relationship between y_i and x_i exits. In other words, for each image present in the input or source domain, there belongs an equivalent image in the output or target domain, forming pairs of images.

The paired data provides explicit supervision for the model during training. During training, The model gains the ability to translate images present in the source domain to their comparable images in the target domain. This type of image translation is commonly achieved using conditional Generative Adversarial Networks (cGANs), e.g pix2pix [2] where an image corresponding to the source domain X passes through the generator and it generates an image belonging to the target domain Y as output. The

discriminator network then assesses the resulting image's quality and offers suggestions to the generator for further development.

Supervised image translation using paired data has demonstrated impressive results in various applications such as in the field of [2–5]. The paired data gives the model a powerful signal that it can use to precisely investigate the conversion between the input and output domains. However, the need for paired data can be a significant limitation as it can be time-constraint and extravagant task to acquire. Additionally, in some cases, paired data may not even be available, which limits the applicability of supervised image translation.

Conversely, unsupervised image translation, which does not need paired data, has grown in popularity in recent years.

1.1.2 Unpaired Image Translation

Unpaired image translation, also known as unsupervised image translation, is a task of translating images present in one field to another field all in deficit of paired training data. Unpaired training data includes a source set $\{x_i\}_{i=1}^{N}$ $(x_i \in X)$ and a target set $\{y_i\}_{i=1}^{N}$ $(y_i \in Y)$ with no knowledge of whether for every $(x_i \in X)$ there is a match $(y_i \in Y)$ or not. This task is challenging because there is no ground-truth data to supervise the mapping between the two domains. The absence of paired training data poses a significant challenge in developing unsupervised image translation models. One way to approach this challenge is the use of GANs, e.g CycleGAN.

CycleGAN is a popular unsupervised image translation method that uses GANs. It works by training two GANs, one for each domain, and introduces a cycle consistency loss term on top of the adversarial loss. The term "cycle consistency loss" is utilized to guarantee that when an image belonging to domain X is translated into domain Y and subsequently back into domain X, it will exhibit similarity to the original image that belonged to domain X. Similarly, if an image from domain Y is translated to domain X and then back to domain Y, it should bear resemblance to the initial image from domain Y. The cycle-consistent loss ensures that there is a singular correspondence between the input and output domains, while also maintaining the identity of the

input image. This regularization term effectively reduces the risk of producing visually unrealistic or distorted images and helps to maintain consistency across the image domains. Another significant advantage of cycle-consistent loss is that it allows for bidirectional image translation, enabling the model to learn the mappings between two domains in both directions.

There are several benefits of using unsupervised image translation with unpaired training data. First, it eliminates the need for expensive and time-consuming manual annotation of paired training data. Second, It permits the use of enormous volumes of unpaired data, which may improve the model's generalisation and scalability. Thirdly, it makes translation between domains conceivable that would not otherwise be possible, For instance, transforming summer photos into winter scenes or transitioning from daytime to nighttime, can be achieved without relying on paired data.

Despite the potential benefits, unsupervised image translation with unpaired training data still faces several challenges. The lack of ground-truth data makes it difficult to evaluate the quality of the generated images objectively. Additionally, the design of the GANs architecture and the choice of hyper-parameters utilised have a significant impact on the quality of the images that are produced.

In conclusion, unsupervised image translation with unpaired training data is a challenging but promising task. CycleGAN is a popular method for unsupervised image translation that uses GANs and a cycle consistency loss term. Employing unpaired training data offers numerous advantages, such as dispensing with the requirement for paired training data, and facilitating the translation between domains that would otherwise be unfeasible. However, several challenges still need to be addressed in order to improve the quality and scalability of unsupervised image translation models.

1.2 Motivation

The domain of computer vision has faced a persistent difficulty regarding the translation of images to other forms, particularly when there is no available paired training data, which limits the performance of existing methods. Impressive image-to-image translation outcomes are shown by paired GAN models, such as conditional GAN

models. For these techniques to work, it is required to have a ground-truth translation for each of the images in the training set. This ground-truth offers supervision since it allows for easy comparison of each translation to its corresponding ground-truth, which greatly simplifies the issue. The requirement for paired data is highly constricting, and obtaining it may be exceedingly expensive and time consuming. The model's depth and usefulness are increased by its capacity of converting images across domains without relying on ground truth, or unpaired data. Cycle consistency loss is added to boost the model performance in maintaining the real picture identity while changing only the domain of the picture.

CycleGAN has emerged as a popular technique that can learn the conversion across two separate domains without paired data. The potential applications of CycleGAN are numerous, ranging from artistic style transfer and image colorization to domain adaptation and image manipulation. In various fields, such as art, fashion, and advertising, the benefits of CycleGAN have already been demonstrated. In this thesis, we aim to further explore the capabilities of CycleGAN in generating high-quality images and advancing the field of computer vision. Specifically, we will investigate novel architectures, loss functions, and training strategies to improve the stability and quality of CycleGAN. Moreover, we will study the interpretability and robustness of CycleGAN. By conducting rigorous experiments and analyzing the theoretical underpinnings of CycleGAN, In doing so, we intend to significantly advance the field of computer vision research and open the door to the development of fresh approaches for image synthesis.

1.3 Problem Statement

Earlier several different models existed for performing different type of image translation tasks, like Style Transfer, Photo enhancement, Season Transfer and Paired Image to Image Translation. This thesis presents a single novel model for performing all type of image translation task. The presence of non-convergence problem, mode collapse, and hyper parameter selection in the existing GAN models continue to be a major obstacle for GAN development. Besides assessing the performance of Generative Adversarial Network(GAN) models is a complex and challenging task.

1.4 Objectives of the Thesis

The objectives of the thesis are :

i. To design a model for image translation task where one model will be using an UNET-based generator and the other model using a RESNET block-based generator.

ii. To design an architecture where we will be concatenating the feature maps generated by UNET generator and that of the RESNET generator and train the model for image translation task.

1.5 Organization of the Thesis

The main focus of the thesis is to solve the task of Image to Image Translation (ITIT). It has six chapters, which are listed below with a brief overview of what each chapter contains.

Chapter 1: Introduction
In this chapter a brief introduction about ITIT has been given. In this chapter we have discussed about what is Image to Image translation task and it's various types.

Chapter 2: Literature Review
In this chapter detailed survey about all the recent state-of-art techniques that are available for ITIT is given. The Literature survey is given on the basis of the available Image to Image Translation approaches.

Chapter 3: Background
In this chapter the background knowledge required to comprehend the remaining portions of the thesis is provided.

Chapter 4: Proposed System
In this chapter we have discussed about our proposed system. The proposed system consists of two distinct stages. The initial stage focuses on analyzing the existing

architecture, while the subsequent stage delves into the discussion of our proposed architecture.

Chapter 5: Experimental Results and Discussion

In this chapter, we've provided specifics on the experimental dataset and assessment criteria we utilised to assess the system's performance. We also talk about the results that were attained.

Chapter 6: Conclusion

The overall work and future scope is summarized in this chapter. Here, we have concluded that concatenating the feature maps helps in generating better translated images and outperforms the existing standalone generator based models.

CHAPTER 2

Literature Review

The field of computer vision has undergone a rapid evolution in recent years with the emergence of deep learning techniques. These techniques, such as Convolutional Neural Network (CNN) and Generative Adversarial Network (GAN), have demonstrated remarkable success in a wide range of computer vision tasks, including image classification, object detection, and image segmentation. However, prior to the widespread adoption of deep learning, researchers relied on conventional methods such as Conditional Random Field (CRF) and patch-based methods for performing image-to-image translation tasks.

2.1 Traditional Methods for image to image translation task

2.1.1 Conditional random field

The Conditional Random Field (CRF) is a versatile probabilistic graphical model that finds application in various image processing tasks., including image segmentation, denoising, and restoration. Given an input picture and a collection of labels, CRF may be used to determine the most likely output image in the context of ITIT.

In [6], the authors propose a method for multi-class image segmentation and labelling using fully connected CRF models with dense pairwise connectivity at the pixel level. The authors introduce an extremely efficient approximate inference algorithm for these models, where pairwise edge potentials are defined using a linear combination of Gaussian kernels. Through experimentation, they demonstrate that dense connectivity at the pixel level enhances the accuracy of segmentation and labeling.

2.1.2 Image Analogies

The Image Analogies algorithm uses a patch-based approach to perform ITIT. The functioning of this approach involves the identification of corresponding patches within the input and output images. Subsequently, it transfers the visual appearance and texture of these patches from the input image to the output image. The algorithm uses a Markov Random Field (MRF) to ensure consistency and coherence in the translated image.

In [3], a framework based on "image analogies" for processing images by example is proposed. A pair of pictures, one of which is a filtered form of the other, are supplied as training data during the design step, and the learnt filter is then exercised on a fresh target image to produce an equivalent filtered output during the application step. The framework is based on a simple multi-scale auto-regression and supports a wide variety of image filter effects, including texture synthesis, super-resolution, texture transfer, artistic filters, and texture-by-numbers.

2.1.3 Seam Carving

Seam Carving another method proposed in [7], for image processing is based on Seam Carving. Seam carving is a technique for image resizing that preserves important regions of the image by removing less important regions. The algorithm works by iteratively finding and removing "seams" of pixels that are least important to the image content, i.e.,those that contribute least to the energy of the image. The seam carving algorithm has been shown to be effective in reducing image size while preserving

important content, and has been compared favourably to other resizing methods such as cropping and scaling.

While the traditional image translation algorithms were able to generate visually pleasing results, it had limitations in terms of scalability and flexibility.These traditional methods, often rely on handcrafted features and heuristics, making them limited in their ability to handle complex and diverse image datasets. Furthermore, they are often computationally expensive and require significant human intervention for parameter tuning and optimization. These algorithms were developed with the goal of texture synthesis and had limited applicability to other ITIT tasks that needed to learn the complex representation of one image domain to another domain.

For these complex and high level image translation tasks several deep learning methods were proposed among which the use of generative models like the Restricted Boltzmann Machines (RBM), Variational Auto Encoder (VAE) and the GANs have achieved remarkable success in this area. The operation of these generative models and their application to image-to-image translation tasks will be briefly discussed in this section.

2.2 Generative Models for image to image translation task

2.2.1 Restricted Boltzmann machines

Restricted Boltzmann Machines (RBM) as proposed in [8] are a kind of generative model that can learn the underlying probability distribution of input data. RBMs are composed of two layers, a visible layer and a hidden layer, with binary units in each layer. However, it was not until the work of Hinton and Salakhutdinov in [9] that RBMs were shown to be effective in learning complex probability distributions for unsupervised learning tasks. Finding the weights that maximise the probability of the training data is the aim of training an RBM. This is done through a process called contrastive divergence, which involves running a Gibbs sampling chain to estimate the distribution of the hidden units given the visible units Subsequently, the weights are adjusted by considering the disparity between the actual distributions and the

estimated distributions, and then making updates accordingly. New samples can be produced by randomly initialising the hidden units, performing Gibbs sampling, and then sampling the visible units.

RBMs are useful for unsupervised learning tasks where labelled data is scarce or expensive to obtain. Additionally, these techniques have demonstrated strong performance in situations involving semi-supervised learning, which involve a scenario where there is a limited amount of labeled data available alongside a substantial volume of unlabeled data. However, RBMs have some limitations, such as being computationally expensive to train and limited to modelling binary data, which may not be suitable for some applications.

2.2.2 Variational autoencoders

Variational Auto Encoder (VAE) [10] are a type of generative model that learn a low-dimensional representation of input data by training an encoder and decoder network. In VAEs, the encoder network takes an input image and maps it to a distribution of latent variables, which are then fed into the decoder network to generate an output image. The objective of VAE training is to reduce the disparity among the input and output images, while also imposing a restriction on the distribution of latent variables to adhere to a predetermined prior distribution, commonly in the form of a Gaussian distribution.

VAEs have been successful in image generation, but they have several limitations when it comes to image-to-image translation tasks. One major drawback is that they use mean squared error (MSE) for the measuring the reconstruction loss, which does not capture the perceptual quality of the images. This is because MSE tends to produce blurry images due to its averaging effect over the pixel values. They also rely on the assumption that the latent variables follow a Gaussian distribution, which can lead to the loss of important information during the encoding process.

In [11], the authors proposed a method for unsupervised ITIT using VAEs. The authors use a cycle-consistent adversarial network (CycleGAN) with VAEs to translate images between two domains without paired training data.

In [12], a new approach called auto-encoding generative adversarial networks is introduced. This approach combines the standard GAN algorithm with an auto-encoder that provides a reconstruction loss to prevent mode collapse. The authors propose a principle for linking auto-encoders with GANs, based on the hierarchical structure of the generative model. They propose the utilization of variational inference, where the impractical likelihood is substituted with a synthetic likelihood, and the unknown posterior distribution is subsituted by an implicit distribution. These replacements are achieved by employing discriminators, which are trained to learn both the synthetic likelihood and the implicit distribution. The resulting approach combines the strengths of both methods and is assessed through a series of tests.

In [13] the authors suggest a method for multimodal ITIT using VAEs. The authors employ a conditional VAE (Variational Autoencoder) to produce a range of distinct outputs for an individual image, enabling the existence of multiple potential translations across different domains.

In [2], the paper proposes a method for ITIT using a conditional GAN with a VAE-based encoder. The authors show that this approach produces more realistic results than traditional VAEs or GANs alone. These papers demonstrate the potential of VAEs for image translation tasks, but also highlight some of their limitations and challenges, such as the tendency to produce blurry or low-resolution images.

The outcomes of ITIT challenges using GANs have been proven to be more aesthetically pleasing than those using VAEs.

2.2.3 Generative Adversarial Network

In the field of computer vision, Generative Adversarial Network (GAN) have emerged as a powerful technique for a variety of tasks such as image editing [14, 15], image generation [1, 16] and representation learning [17–19]. One of the main advantage of GANs is their potential to generate diverse images of high-quality that closely mimic real-world examples.

The approach presented in reference [1] introduces a method where two models, namely a generative model (G) and a discriminative model (D), are trained simultaneously in

a two-player game with a minimax objective. The generative model (G) is responsible for capturing the distribution of the data, while the discriminative model (D) aims to estimate the probability that a given sample originates from the training data. The system can be trained using backpropagation and does not require any Markov chains or unrolled approximate inference networks.

The work discussed in reference [20] introduced a concept known as Conditional Generative Adversarial Network (cGAN). This approach extends the functionality of GANs by incorporating a conditional model, where both the generator and discriminator are influenced by additional information denoted as 'y.' This extra information, which can take the form of various auxiliary details such as class labels, serves as a conditioning factor in the cGAN framework.

In [2], the authors propose a cGAN network "Pix2Pix", suitable for ITIT tasks for paired images. It uses a UNET based generator [21] and PatchGAN based discriminator [22]. Similar concepts have been used for a variety of purposes, including creating photos from drawings [23] or from attribute and semantic layouts [24].

In [25], the authors propose a new method called CoGAN has been developed that can learn a joint distribution of images from different domains without requiring any pairs of corresponding images. CoGAN works by sharing weights between two generative adversarial networks (GANs), which forces the model to learn a combined joint distribution rather than a product of marginal distributions. CoGAN has been successfully applied to several joint distribution learning tasks, including learning a joint distribution of color and depth images, and learning a joint distribution of face images with different attributes.

In [26], the authors make use of Cycle-Consistent Generative Adversarial Network (CycleGAN) for unpaired translation of images-to-images. It seeks to translate images from different domains, without the need of paired training data. The architecture comprises of two generators and two discriminators, which work together in a cyclic process to translate images among different domains and vice versa, while ensuring consistency between the original and the translated image. The proposed method is demonstrated to be effective in various tasks, such as transforming horse images into zebras, and turning summer images into winter.

In [27], the authors introduce a new task of unsupervised video-to-video translation, which presents unique challenges in learning realistic motion and transitions between frames. The authors propose a spatio-temporal 3D translator as a more effective solution. They compare the performance of their 3D technique to frame-wise translation and show that it performs better, delivering more accurate results across the board.

The authors of [28] introduce a neural algorithm capable of extracting and merging the content and style components of natural images. This algorithm enables the creation of novel images by combining the content from any photograph with the visual characteristics of various renowned artworks. The outcomes of this research shed light on the intricate image representations acquired by convolutional neural networks (CNNs) and highlight their potential for advanced image synthesis and manipulation.

In [4], an approach for training feed-forward convolutional neural networks for image transformation tasks using perceptual loss functions based on high-level features is suggested. To produce excellent results in real-time, the authors combine the advantages of the conventional per-pixel loss and perceptual loss functions. They demonstrate that their method is quicker and more aesthetically pleasant than optimization-based methods by showing results for picture style transfer and single-image super-resolution.

Multimodal Unsupervised Image-to-Image Translation (MUNIT) [29] is a model based on GANs (Generative Adversarial Networks) that has the ability to carry out unsupervised translation of images in a multimodal manner. This implies that from a single input image, it can generate various outputs that correspond to different styles or attributes. MUNIT achieves this by using an encoder-decoder architecture with shared encoders and multiple decoders that correspond to different styles or attributes. Tasks like style transfer and face attribute manipulation have witnessed outstanding and notable application of MUNIT.

In [30], the authors introduce a new layer called spatially-adaptive normalisation, which can be applied to generate photorealistic images based on a given input semantic layout.The conventional approach of directly feeding the input semantic layout to the deep network is not optimal because the normalisation layers tend to remove semantic information. This approach adjusts the activations within normalization layers by employing a learned, spatially-adaptive transformation that is influenced by the input

layout. It surpasses current methods in terms of visual accuracy and alignment with the input layouts.

In reference to the work described in [31], the authors introduced StarGAN, an innovative and scalable approach for translating images across multiple domains using a single model. Unlike other methods, StarGAN employs a unified model architecture that allows for simultaneous training on multiple datasets containing different domains within a single network. This approach enables the translation of input images to any desired target domain, resulting in high-quality translated images.

DualGAN [32] is another GAN-based approach that addresses the problem of unpaired image-to-image translation. Instead of using cycle consistency loss, DualGAN introduces a dual learning scheme, where two GANs are trained simultaneously to learn the mappings from both domains to a shared latent space. This approach has been shown to outperform CycleGAN in certain tasks, such as image colorization.

In [33], the authors propose a method called Domain Transfer Network (DTN) to transfer a training specimen from one domain to an analog specimen in another domain. Their objective is to acquire knowledge about a generative function G, capable of transforming input samples from one domain to another, while ensuring that the output of a given function f remains unaltered, regardless of the input domain. To achieve this, they employ unsupervised training data comprising samples from each domain and a compound loss function. This loss function consists of multiple components, including a multiclass GAN loss, an f-constancy component, and a regularization component that motivates G to maintain the mapping of samples from one domain to themselves. They substantiate the effectiveness of their approach by applying it to digit and face images, demonstrating the generation of convincing and original images while preserving their inherent characteristics.

CHAPTER 3

Preliminaries

This chapter provides the background knowledge required to comprehend the remaining portions of the thesis.

3.1 Convolutional Neural Networks

Convolutional Neural Network (CNN) are a type of deep learning neural network that are primarily used for image and video processing tasks.Yann LeCun first introduced CNN in 1998 [34]. CNNs have revolutionised the field of computer vision and have achieved state-of-the-art performance in various image classification [35–37], object detection [38–40], face recognition [41, 42], and segmentation tasks [43, 44]. This section will cover the details of how a CNN model operates.

Convolutional Neural Networks (CNNs) as shown in Figure 3.1 consist of multiple layers, such as convolutional layers, pooling layers, and fully connected layers. The input image is represented by a three-dimensional tensor, which includes its height, width, and depth. In the initial stage of a CNN, there exists a convolutional layer comprising a set of adaptable filters. These filters convolve with the input image, resulting in the generation of a series of activation maps. Each filter in the convolutional layer is specifically designed to identify distinct features present in the input image. For example, the first layer of a CNN can specialize in the detection of edges, while subsequent layers

can develop expertise in recognizing more complex features like corners, shapes, and textures.

A pooling layer is used after the convolutional layer to reduce the spatial dimensionality of the activation maps. This is accomplished by choosing the highest or average value found within a constrained area of the activation map. Pooling layers help the network become less computationally demanding while increasing its resistance to slight changes in the input picture.

The final layers of a CNN are typically fully connected layers, which are used to map the features learned by the convolutional layers to the output classes. The output of the final layer is passed through a softmax function to obtain the probability distribution over the output classes. Some of the popular CNN architectures include AlexNet [35], VGGNet [36], GoogleNet [45], ResNet [46], and DenseNet [47]. These networks have achieved state-of-the-art performance in various image classification, object detection, and segmentation tasks.

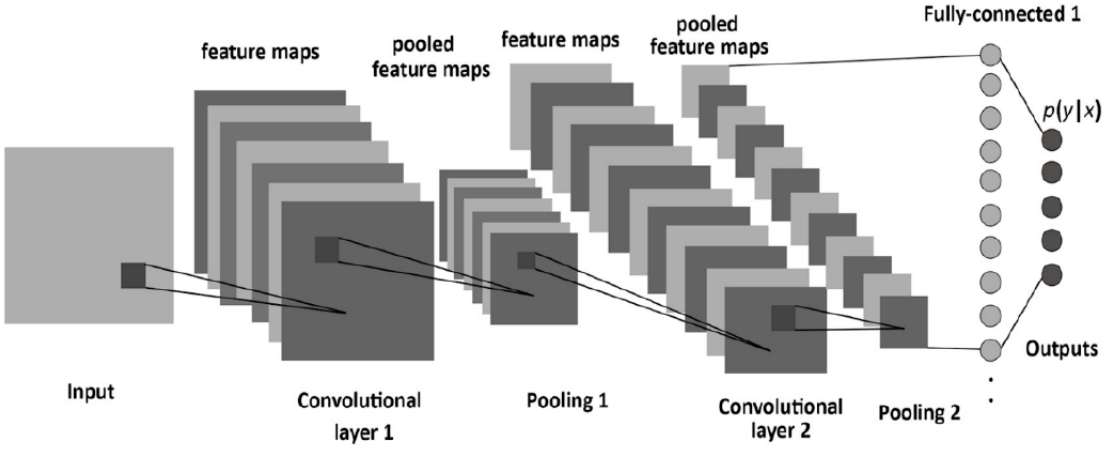


FIGURE 3.1: A typical CNN architecture, from [34]

3.2 Generative Adversarial Networks

In 2014, Goodfellow et al. [1] published the initial study on Generative Adversarial Network (GAN). GANs belong to a category of deep learning models employed for unsupervised learning purposes, specifically in tasks like image generation [1, 16]. Two

neural networks, namely a generator and a discriminator, constitute the composition of GANs. The purpose of training the generator is to generate data that closely relates a particular dataset, whereas the discriminator is trained for differentiating between the generated data and the original data obtained from the dataset. The training process of these two networks involves an adversarial approach, where the generator aims to produce realistic data that cannot be distinguished from genuine data by the discriminator. Simultaneously, the discriminator strives to correctly identify the generated data as counterfeit.

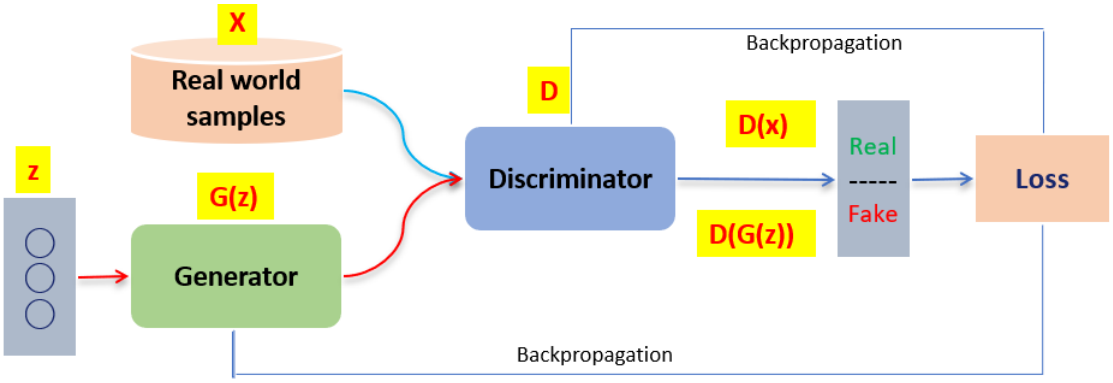


FIGURE 3.2: Working of GAN, from [1]

The mathematical notation for GANs can be described as follows:

Let x be a vector that represents the input data, and z be a vector that represents the noise input. G here is the generator which is a function responsible for converting the noise input z to a generated output x'. D which is the discriminator, is basically a function that receives an input x and outputs a scalar value between 0 and 1, representing the probability that the input is real (1) or fake (0). The training procedure may be described as a two-player minimax game in which the generator attempts to minimise the subsequent loss function given by Equation 3.1:

$$\mathcal{L}_G = \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{3.1}$$

and the discriminator tries to maximize the following loss function given by Equation 3.2:

$$\mathcal{L}_D = \mathbb{E}_{x \sim p_{\text{data}}\,(x)}[\log(D(x))] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{3.2}$$

In the above equations, $p_{data}(x)$ represents the probability distribution of the real data, and $p_z(z)$ represents the probability distribution of the noise input. During training, the generator and discriminator networks are updated alternatively. At first, training of the discriminator is done for one step by minimizing $\mathcal{L}_D$ with respect to its parameters. Then, training of the generator is done for one step by minimizing $\mathcal{L}_G$ with respect to its parameters. This process is repeated until convergence.

The overall objective function of the GANs is the combination of the above two loss functions 4.1 and 4.2 given by Equation 3.3:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))] \tag{3.3}$$

Observing the two log functions used in Equation 3.3, it is possible to understand the GAN loss function. Plots of the used log terms are presented in Figure 3.3 to further understand the goal function.

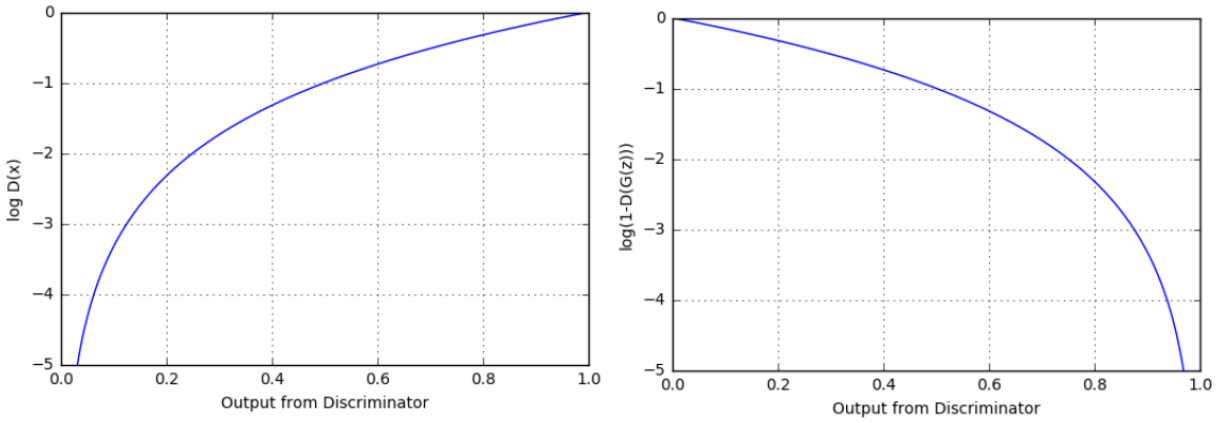


FIGURE 3.3: Loss function components of GAN, from [1]

The main goal of the discriminator is to optimize the two functions. When presented with a genuine image as input, the objective is to maximize the logarithm of the discriminator's output, denoted as $\log D(x)$. This maximization aims to correctly identify and classify the image as real, corresponding to an output value of 1. Conversely, when provided with a fabricated image generated by the generator, the discriminator's objective is to maximize the logarithm of the complementary output, $\log(1 - D(G(z)))$. In this case, a discriminator output of 0 signifies the image as fake.

On the contrary, the main goal of the generator is to produce synthetic images that convincingly deceive the discriminator into perceiving them as genuine. To achieve this, the generator aims to minimize the logarithmic function $\log(1 - D(G(z)))$, which is visually depicted in the right graph of the figure. By doing so, the generator prompts the discriminator to output a value of 1 when presented with a fake image generated by the generator, indicating that it is authentic. In essence, the discriminator's role is to discern between real and fake images, while the generator's objective is to generate images that trick the discriminator into believing they are real.

Eventually, the generator learns to produce samples that are indistinguishable from real data, and the discriminator's accuracy approaches 0.5 (i.e., random guessing). At this point, the GAN has converged, and the generator can be used to generate new data samples that are similar to the original dataset.

3.3 PatchGAN

PatchGAN is a type of discriminator architecture first introduced in [2] commonly used in image translation tasks, particularly in conditional adversarial networks (cGANs). It is a type of convolutional neural network (CNN) that operates on small patches of an image to predict whether or not each patch is real or fake.

A patch of a picture, usually 64x64 or 128x128 pixels in size, serves as the input for the PatchGAN architecture, which processes the patch via a number of convolutional layers. The convolutional layers acquire the ability to extract characteristics from the input patch, and the ultimate outcome of the last convolutional layer is a solitary numerical value which signifies the likelihood that the input patch is genuine.

One of the main benefits of the PatchGAN architecture over a conventional discriminator is that it provides high-resolution feedback to the generator. In a conventional discriminator, the output is a single scalar value that represents the probability that the entire image is real or fake. This makes it difficult for the generator to learn how to produce realistic images with high-resolution details.

In contrast, the PatchGAN architecture provides feedback on a per-pixel basis, allowing the generator to learn how to produce realistic details at a high resolution. By operating

on small patches of the image, the PatchGAN can provide feedback at multiple scales and capture both global and local structure information.

Another advantage of the PatchGAN architecture is that it can encourage the generator to produce images with more diverse and realistic details. By forcing the generator to produce realistic details at the patch level, the PatchGAN can help to prevent the generator from producing overly smooth or blurry images that lack fine details.

All things considered, the PatchGAN architecture has gained popularity for translation of images to images, especially in uses like picture colorization, style transfer, and super-resolution. It's a good option for producing high-quality photos since it may give the generator high-resolution input and promote the creation of more varied and realistic elements.

3.4 U-Net

The U-Net architecture is a convolutional neural network (CNN) based structure specifically developed for segmenting biomedical images. It was first presented in a research paper published by Ronneberger et al. in 2015 [21]. An encoder network and a decoder network are the two fundamental parts of the U-Net architecture. A standard Convolutional Neural Network (CNN) is used in the encoder network to extract pertinent features from the input picture and minimise the size of the image. The decoder network, on the other hand, upsamples the feature maps and creates a segmentation map. The term "U-Net" is derived from the unusual U-shaped link between the encoder and decoder network.

The U-Net architecture as shown in Figure 3.4 incorporates a skip connection that links the encoder and decoder paths together. This connection enables the decoder path to utilize the high-resolution feature maps obtained from the encoder path. This utilization of feature maps helps retain spatial information and prevents the loss of intricate details during the upsampling procedure. The skip connection works by combining the feature maps from the encoder path with the feature maps from the corresponding decoder layer.

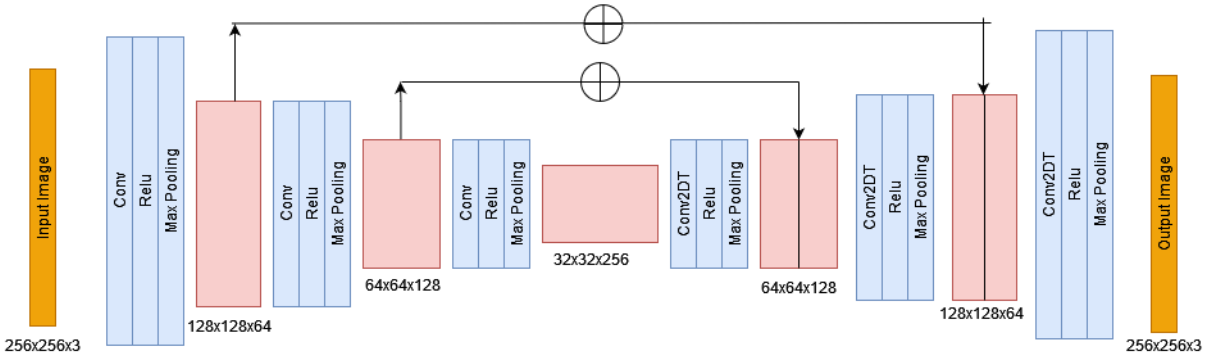


FIGURE 3.4: Working of U-Net

The U-Net architecture incorporates a modified variant of the cross-entropy loss function. This adjusted loss function is characterized by the following definition:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} w_j \left[y_{i,j}\log\left(p_{i,j}\right) + \left(1 - y_{i,j}\right)\log\left(1 - p_{i,j}\right)\right] \tag{3.4}$$

where N is the number of samples, M is the number of classes, $y_{i,j}$ is the ground truth label of the j-th class for the i-th sample, $p_{i,j}$ is the predicted probability of the j-th class for the i-th sample, and w_j is a weight for the j-th class. The weight for the j-th class is defined as:

$$w_j = \frac{\sum_{i=1}^{N}\sum_{k=1}^{M}\left[y_{i,k} = j\right]}{\sum_{i=1}^{N}\sum_{k=1}^{M} y_{i,k}} \tag{3.5}$$

The weight assigned to the j-th class is determined by its inverse relationship with the frequency of the j-th class within the training set. This approach aims to tackle the issue of class imbalance, which arises when certain classes contain significantly fewer samples compared to others.

The U-Net architecture has been widely used for various image segmentation tasks, including biomedical image segmentation, satellite image segmentation, and road segmentation. Additionally, it has been employed as a fundamental component in the development of more intricate structures like the DeepLab series and the Mask R-CNN. In general, the U-Net architecture has demonstrated its efficacy in tasks related to image segmentation and has established itself as a widely adopted model in the realm of biomedical image analysis.

3.5 ResNet

Residual Networks (ResNets) as shown in Figure 3.5 are a type of deep neural network architecture that have been highly successful in computer vision tasks such as image classification and object detection. ResNets were introduced by He et al. in [46].

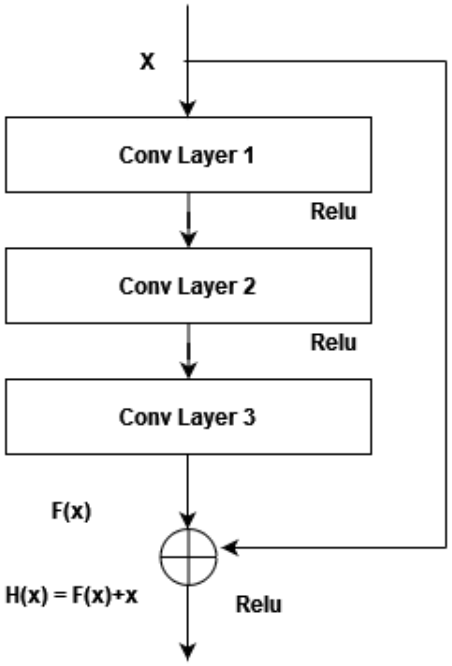


FIGURE 3.5: Working of ResNet

ResNets are built upon the fundamental concept of incorporating shortcut connections, which are alternatively referred to as skip connections, connecting the input and output of a layer. ResNets are used in deep neural networks to address the vanishing gradient problem, which occurs when gradients in initial layers become too small to learn meaningful representations. ResNets use skip connections that allow information to bypass one or more layers and be directly fed into later layers, enabling the network to learn the residual mapping. This approach mitigates the vanishing gradient problem by allowing the gradient to flow more easily through the network, facilitating training of deeper networks.

The residual mapping of a layer is given by the Equation 3.6:

$$\mathbf{y} = \mathcal{F}(\mathbf{x}) + \mathbf{x} \tag{3.6}$$

where $\mathbf{x}$ denotes the input to the layer, $\mathcal{F}$ is the residual function that the network needs to learn, and $\mathbf{y}$ is the output of the layer. The addition of $\mathbf{x}$ to $\mathcal{F}(\mathbf{x})$ creates the shortcut connection. The ResNet architecture consists of several residual blocks, each containing multiple layers with shortcut connections. The ResNet blocks can be divided into two types: identity blocks and convolutional blocks. Identity blocks are employed when a block's input and output contain an equal number of channels. On the other hand, convolutional blocks are utilized when the input and output of a block have varying numbers of channels.

The ResNet architecture includes extra elements like a global average pooling layer and a fully connected output layer. The purpose of the global average pooling layer is to calculate the average of the feature maps obtained from the last convolutional layer, creating a combined feature vector. Afterwards, the output layer utilizes this feature vector to produce predictions.

ResNets have been shown to outperform other deep neural network architectures on various image classification tasks. They have also been implemented in several other computer vision tasks such as semantic segmentation, object detection and image restoration.

CHAPTER 4

Proposed System

In this chapter, we will discuss our proposed approach for performing the image-to-image translation task. First, we will discuss the workings of the existing CycleGAN model, and then later, we will discuss our proposed architecture of the modified CycleGAN for the above-mentioned image-to-image translation task.

4.1 Cycle Consistent Generative Adversarial Network (CycleGAN)

Cycle-Consistent Generative Adversarial Network (CycleGAN) is another type of GAN based deep learning model first proposed in [26], used for unsupervised translation of image-to-image. Without the need for paired training data, the model may learn to transform pictures from one domain X to another domain Y. For instance, it can be trained to convert images of horses to images of zebras without the need for matching pairs of horse and zebra images.

The model as represented in Figure 4.1 shows the basic CycleGAN that consists of two generators, G and F, which are responsible for mapping images from the corresponding domain X to equivalent domain Y and from corresponding domain Y to equivalent domain X, respectively. In addition, the model includes two discriminators, D_X and D_Y. These discriminators undergo training to discriminate actual images from domain

X from synthetic images created by generator G and actual images from domain Y from synthetic images produced by generator F.

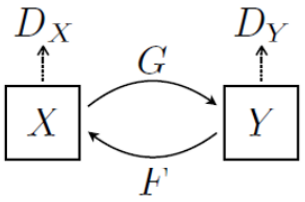


FIGURE 4.1: Functions F: $Y \rightarrow X$ and G: $X \rightarrow Y$ along with adversarial discriminators D_X and D_Y, from [26]

The two generators represent the functions: $G_X : X \rightarrow Y$ $G_Y : Y \rightarrow X$

The two discriminators represent the function: $D_X : X \rightarrow \mathbb{R}$ $D_Y : Y \rightarrow \mathbb{R}$

The training of the CycleGAN model involves the utilization of the conventional adversarial loss, complemented by supplementary parameters. Among these parameters is the Cycle consistency loss, which guarantees that the reconstructed images, after passing through both generators and returning to their original domains, maintain similarity to the initial images. The cycle consistency loss is determined as the difference between the original image and the image that has undergone translation to the alternate domain and subsequently been translated back. The Cycle consistency loss is given by the Equation 4.1

$$\text{Cycle consistency loss } = \|F(G(x)) - x\|1 + \|G(F(y)) - y\|1 \tag{4.1}$$

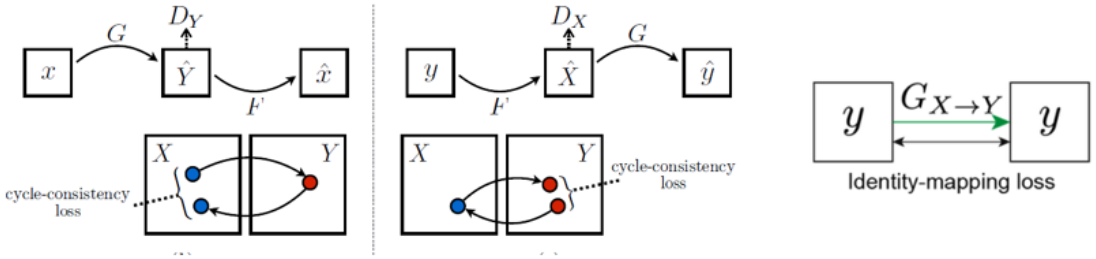


FIGURE 4.2: Calculation of identity loss and cycle consistency loss, from [26].

Here, x represents an image from domain X, y represents an image from domain Y, $G(x)$ represents the image translated from domain X to Y, $F(y)$ represents the image translated from domain Y to X, and $\|\cdot\|_1$ represents the L1 norm. The cycle consistency loss encourages the generators to produce images that maintain their original characteristics after being translated to the other domain and back again. This helps to prevent the generators from overfitting to the training data and producing unrealistic or inconsistent results.

Along with the cycle consistency loss, CycleGAN implements the identity loss shown in Figure 4.3 which encourages the generators to preserve the original content of the input images. The identity loss is given by the Equation 4.2:

$$L_{id}(G,F) = \mathbb{E}_{y\sim p_{\text{data}}\,(y)}\left[\|F(y)-y\|_1\right] + \mathbb{E}_{x\sim p_{\text{data}}\,(x)}\left[\|G(x)-x\|_1\right] \tag{4.2}$$

Thus the overall objective function of the CycleGAN is given by the Equation 4.3:

$$\begin{aligned}\mathcal{L}\left(G,F,D_X,D_Y\right) &= \mathcal{L}_{\text{GAN}}\left(G,D_Y,X,Y\right) + \mathcal{L}_{\text{GAN}}\left(F,D_X,Y,X\right) \\ &\quad +\lambda_1\mathcal{L}_{\text{cyc}}(G,F) + \lambda_2\mathcal{L}_{\text{identity}}(G,F)\end{aligned} \tag{4.3}$$

where $\mathcal{L}_{\text{GAN}}$ is the adversarial loss, which encourages the generator to produce images that are indistinguishable from real images by the discriminator. $\mathcal{L}_{\text{cyc}}$ is the cycle consistency loss, $\mathcal{L}_{\text{identity}}$ is the identity loss . The hyperparameter λ controls the trade-off between the two losses.

4.2 Existing architecture of the CycleGAN

In the original CycleGan model proposed by Zhu.et.al in [26] the two generators implemented use the architectures from Johnson et al. [4]. The generator uses 6 residual blocks for 128x128 training images, and 9 residual blocks for 256x256 or higher-resolution training images.

The six residual block network is composed of:

C7s1-64, D128, D256, R256, R256, R256, R256, R256, R256, U128, U64,C7s1-3

The nine residual block network is composed of:

C7s1-64, D128, D256, R256, R256, R256, R256, R256, R256, R256, R256, R256, U128, U64,C7s1-3

The CycleGAN model's generator architecture is made up of a number of levels. The layers are represented by notations such as C7s1-k, Dk, Rk, and Uk. The C7s1-k notation implies a 7x7 Convolution-InstanceNorm-ReLU layer with k filters and stride 1. Similarly, Dk represents a 3x3 Convolution-InstanceNorm-ReLU layer with k filters and stride 2. The residual block is denoted by Rk, which contains two 3x3 convolutional layers with the same number of filters. Lastly, uk stands for a 3x3 fractional-strided-Convolution-InstanceNorm-ReLU layer with k filters and stride $\frac{1}{2}$.

The model's PatchGAN discriminator architecture features a receptive field with a size of 70x70 pixels [10]. A 4x4 convolutional layer, an instance norm layer, and a leaky ReLU activation function with a slope of 0.2 make up each layer of the discriminator. Each layer's stride is 2, which results in a two-fold downsampling of the feature maps. A 1-dimensional convolutional layer with a scalar output makes up the last layer. Four convolutional layers with 64, 128, 256, and 512 filters each make up the discriminator.

In the later stage of training process the residual blocks used in the generator part of the CycleGan are replaced by the U-Net block while keeping the discriminator part as it is. The U-Net block consists of a series of downsample layers that take the input image, perform the convolutional operation, and generate the feature map, followed by the same number of upsample layers connected with the corresponding downsample layers through the skip connections that take the generated feature map, perform the trans-convolutional operation, and produce the desired output image.

4.3 Proposed architecture of the modified Cycle-GAN

This thesis discusses the idea of taking advantage of both the U-Net and ResNet blocks for the image translation task in an ensemble fashion. The generator section of the model has been revised with the following modifications, where the input to the proposed generator takes the image X. The proposed generator block in turn consist of two

separate blocks, one U-Net block and the other being the ResNet block. Each of the two blocks will take the incoming input image, then perform the series of convolutional operations on the image at their respective levels and will return the corresponding feature maps as displayed in Figure 4.3.

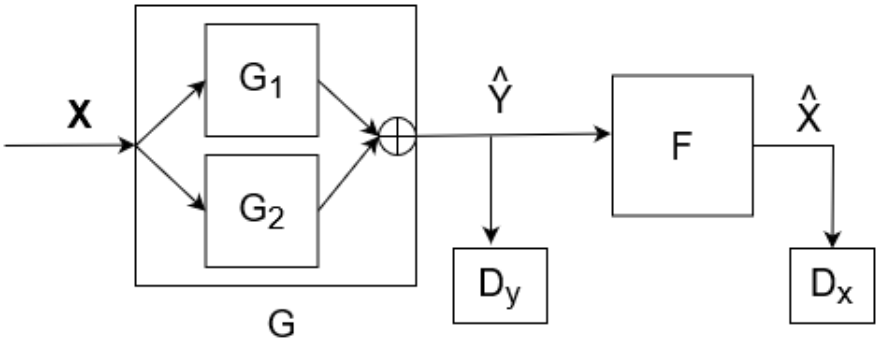


FIGURE 4.3: Overview of the proposed Modified Model

The corresponding feature maps thus obtained from the U-Net and ResNet block will be concatenated and passed to the up-sampling block where a series of trans-convolutional operations will be performed on the merged feature map to generate back the desired output image Y. This output image Y is then passed to the normal ResNet block based generator model to get back the original image X.

An algorithm of the proposed system is given below:

1. The image X is supplied as input to both Generator G_1 and Generator G_2.
2. The bottleneck Feature map is taken from the Unet Generator G_1.
3. The bottle neck Feature Map is taken from the Resnet Generator G_2.
4. Both the Feature Maps extracted are merged to obtain a single Feature Map.
5. Now Up-sampling is performed on this combined Feature Map.
6. The resulting output image with dimensions as that of input image is obtained.
7. The following Generator receives the output image.
8. The cycled input Image is obtained as the final result.

4.3.1 Feed the input image X to the Generator G_1 and G_2

In this first step the input image X is passed to both the U-Net based Generator and ResNet based Generator. Both the Generators performs the respective convolutional operations on the input image.

4.3.2 Extraction of Feature map from the Unet Generator G_1

In the second step the input image which is passed to the Unet generator undergoes a series of downsampling operations where at each stage of downsample the convloutional tasks followed by pooling are performed on the image thereby reducing the size of the input image to obtain the feature map. Finally at a certain stage the bottle neck feature map is obtained which is the required criteria from this step.

4.3.3 Extraction of Feature map from the Resnet Generator G_2

In the third step the input image which is passed to the Resnet generator undergoes a series of downsampling operations where at each stage of downsample the convloutional tasks followed by pooling are performed on the image thereby reducing the size of the input image to obtain the feature map. Finally at a certain stage the bottle neck feature map is obtained which is the required criteria from this step.

4.3.4 Concatenation of the Feature Maps

In this stage the bottle neck Feature Maps thus obtained from the Unet and ResNet in the above stages are concatenated to obtain a single feature map. This concatenated feature map is then later on passed to the next stages.

4.3.5 Up-sampling of the combined Feature Map

In this stage the combined feature map obtained is taken and a series of up-sampling operations are performed, where in each stage of up-sampling operation trans-convolutional tasks are performed on the feature map which increases the size of the feature map.

4.3.6 Getting the output Image

In this stage the resulting output image having dimension as that of input image is obtained, after the series of up-sampling operations are being performed on the feature map. This output image is the translated version of the actual input image.

4.3.7 Passing of output Image to the next Generator

In this stage the translated output image thus obtained in the above stage, is passed on to the next Generator to get back the original cycled version of the input image.

4.3.8 Getting the Cycled Image

At this concluding phase, we acquire the version of the input image that has undergone the aforementioned sequence of steps, resulting in a translated image that has completed a full cycle.

The detailed architecture of the proposed Generator model is shown in Figure 4.4

cPs1-k : a (PxP) Convolution layer consisting of k filters and stride 1.
dk : a (3x3) Convolution layer consisting k filters and stride 2.
Rk : a residual block that comprises of two (3x3) Convolution layers having k no of filters on each of the layers.
uk : a (3x3) Transposed Convolution layer having k filters.

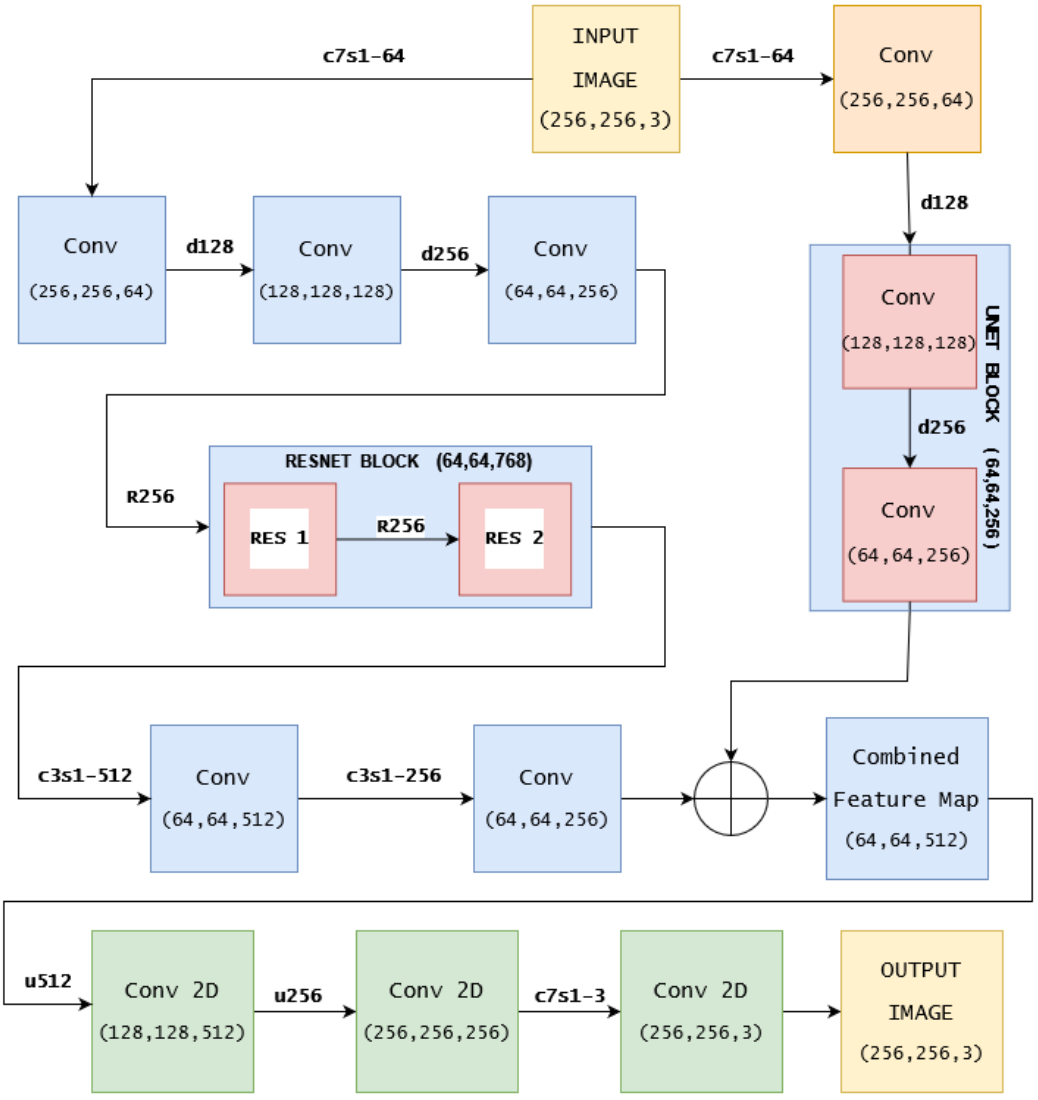


FIGURE 4.4: Proposed architecture of the Generator Model

CHAPTER 5

Experimental Results and Discussions

5.1 Dataset Used

To analyse the effectiveness of the proposed approach and to prove its advantages over the existing techniques, the proposed approach is tested over the dataset obtained from [48] . The Horse2zebra dataset obtained already consists of training and testing data for both the Horse and Zebra class. Each image is of size (256,256).

The training data contains Train-horse : 1067 images and Train-Zebra : 1334 images, whereas the Testing data contains Test-horse : 120 images and Test-zebra : 140 images respectively.

The images are pre-processed before the start of training phase. The images are first resized to (286,286) and then randomly cropped back to (256,256) and then randomly flipped. The Images are then normalised, for the pixel values to range from (-1,1).

(a) Sample Horse

(b) Sample Zebra

FIGURE 5.1: Sample images of Horse and Zebra

5.2 Training Details

The training process of the model spans 200 epochs, where each epoch represents a single iteration of training that covers the entire dataset. To increase the training data size, an additional 1000 augmented images have been added to each of the original training datasets for horses and zebras. The Adam optimizer is employed in the model with specific parameter values: β_1 set to 0.5, β_2 set to 0.5. Furthermore, the learning rate, denoted as η, is established at the value of 0.0002. Additionally, when computing the cycle loss and identity loss, the model utilizes weighting factors of $\lambda_c = 10$ and $\lambda_i = 5$ respectively. When training the discriminator, the loss is reduced by half, and The weights are initially set using a normal distribution characterized by a mean of 0 and a standard deviation of 0.02. To train the discriminator, a set of 50 generated images is maintained in a buffer, and the batch size is adjusted to be 1. Before each epoch, the training set is shuffled and partitioned into subsets that correspond to the minibatch size. Each image in the minibatch is resized to 158x158 and randomly cropped to 128x128 to introduce more variety into the training data.

5.3 Evaluation Parameters

To assess the efficacy of the model in transforming horse images into zebra images, we leverage a Convolutional Neural Network (CNN). The CNN model is trained using the train-horse and train-zebra dataset which contains the real horse and zebra images. The horse images are labelled as 0, while the zebra images are labelled as 1. Once the CNN model is fully trained, we make use of the CNN model to classify the fake images of horses and zebras that are being generated by our CycleGAN model.

The performance evaluation of the proposed system is done by using *accuracy* (Acc), *recall* (R), *precision* (P), $F1$ *score* ($F1$). All the parameters are evaluated using Equation 5.1.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.1}$$

TP represents the count of correctly classified zebra images (true positives), TN represents the count of correctly classified horse images (true negatives), FP represents the count of incorrectly classified horse images as zebra images (false positives), and FN represents the count of incorrectly classified zebra images as horse images (false negatives).

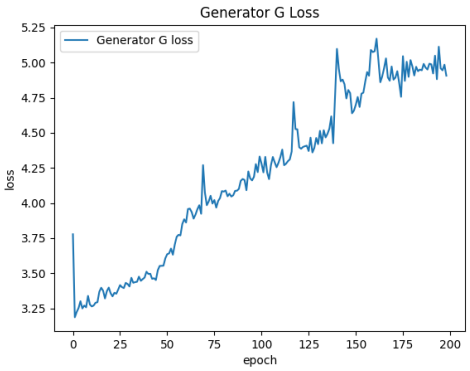
5.4 Experimental Results

The outcomes achieved are displayed in the accompanying Figures 5.2 and 5.3.

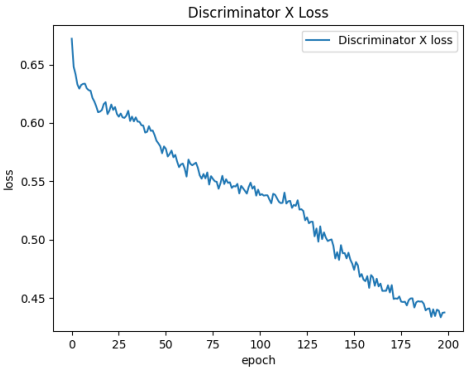
The various losses being the Total Cycle loss, Generator G loss, Discriminator X loss, Generator F loss and Discriminator Y loss that are calculated during the training process of the model are recorded and displayed in the corresponding Figures 5.4, 5.5 and 5.6.

FIGURE 5.2: Cycled Translation of Horse-Zebra-Horse using proposed model

FIGURE 5.3: Cycled Translation of Zebra-Horse-Zebra using proposed model

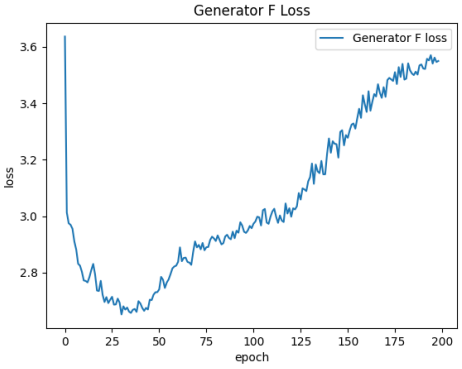


(a) Generator G loss

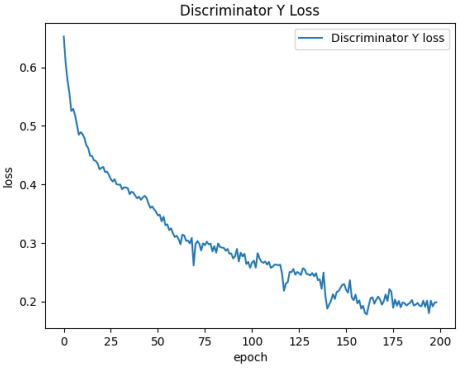


(b) Discriminator X loss

FIGURE 5.4: Generator and Discriminator losses of the first GAN model



(a) Generator F loss



(b) Discriminator Y loss

FIGURE 5.5: Generator and Discriminator losses of the second GAN model

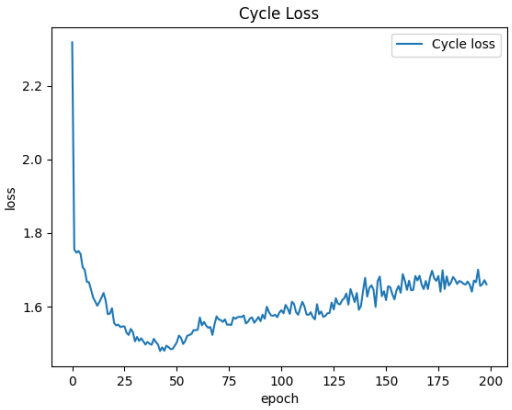


FIGURE 5.6: Total Cycle Loss of the entire model

5.5 Discussion

The graphs shown in Figures 5.4 and 5.5 demonstrate that during the training process, the generator loss tends to increase while the discriminator loss decreases steadily over the range of epochs. This is because as the generator improves and produces more realistic images, the discriminator becomes better at distinguishing them from the real images. This means that the generator needs to work harder to produce even better images that fool the discriminator, which can result in an increase in the generator loss.

The loss function utilized to train the discriminator gets minimized as the discriminator improves its ability to differentiate between authentic and counterfeit images. This indicates that the discriminator is making fewer mistakes in its classification task, which is evident through the progressive decrease in the discriminator loss as time progresses.

The loss of the discriminator indicates its ability to distinguish between genuine and counterfeit samples, while the loss of the generator reflects its proficiency in producing samples that can deceive the discriminator into perceiving them as authentic. The primary objective of the GAN model is to achieve a state in which the generator produces samples that are impossible to differentiate from real samples, while simultaneously minimizing the loss of the discriminator.

From the figure 5.6 it can be observed that the total cycle loss of the model first steadily decreases over the time upto to a certain time and then steadily increases afterwards, this is reasonable as the cycle loss is the sum total of the generator loss and discriminator loss which shows similar behaviour over the time.

5.6 Comparison

In this section the results obtained from the proposed model are compared against the basic CycleGAN model that uses the ResNet based Generator and also with the model where the Generator block is replaced by U-Net in place of ResNet.

Figure 5.7 shows a comparison of the three models, which are used to translate some of the test images.

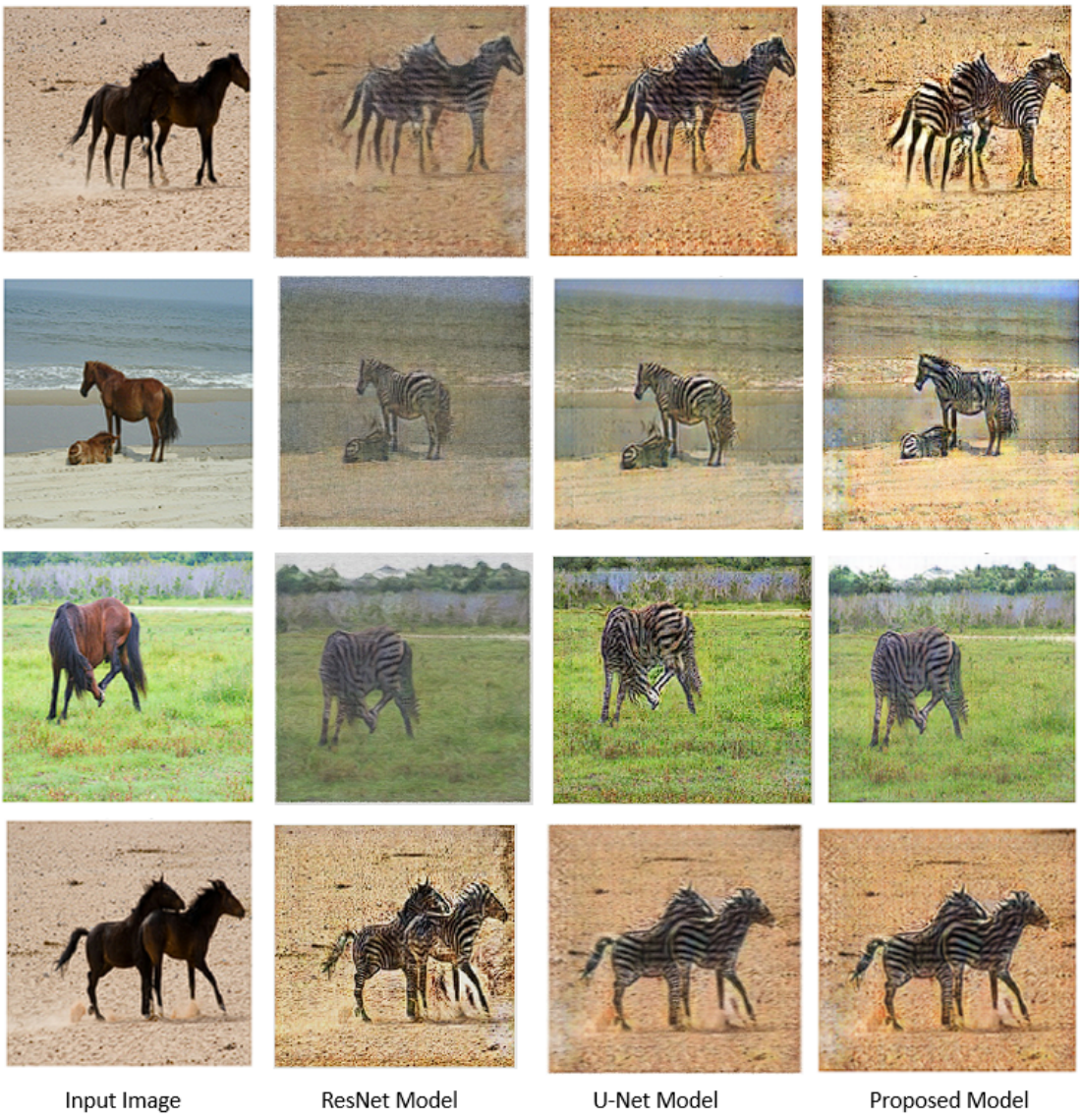
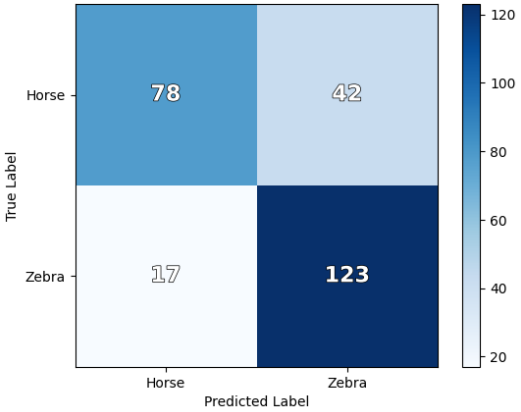


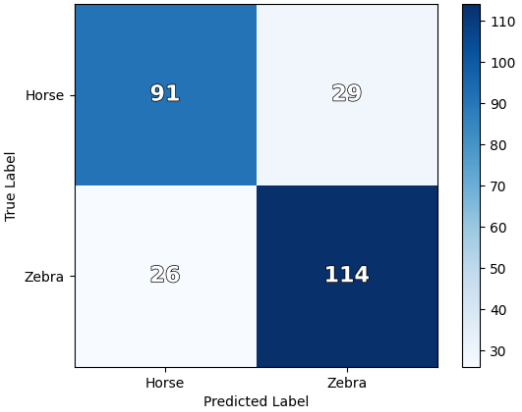
FIGURE 5.7: Image Translation of Horse to Zebra using ResNet, U-Net and Proposed Model

The Figure 5.8 shows the confusion matrix obtained from the test data for the three different models.

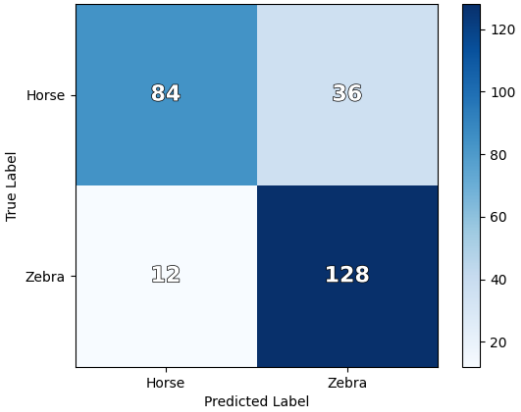
The table 5.1 shows the accuracy values of the ResNet, U-Net and Proposed Model. The outcomes demonstrate that the proposed model surpasses the individual performance of both the ResNet and U-Net models in terms of image translation.



(a) Confusion Matrix of ResNet



(b) Confusion Matrix of U-Net



(c) Confusion Matrix of Proposed Model

FIGURE 5.8: Confusion Matrix of respective ResNet, U-Net and Proposed models

TABLE 5.1: Evaluation Matrix

Model	**Accuracy**
ResNet	0.773
U-Net	0.788
Proposed Model	0.815

CHAPTER 6

Conclusion and Future Work

In this chapter, conclusions and future scopes of this thesis have been discussed.

6.1 Conclusion

In conclusion, this thesis has presented a comprehensive overview of the history and evolution of image-to-image translation techniques, and has focused on the CycleGAN framework for unsupervised image translation without the paired training data. The results obtained in this work have shown that concatenating feature maps obtained from ResNet and U-Net models can result in an enhanced and more comprehensive image representation, resulting in enhanced performance in tasks related to translating of images. A convolutional neural network model for classifying the translated pictures as real and fake was used for evaluating the suggested system, and the experimental results demonstrate that the proposed system outperforms the existing image translation techniques. Overall, concatenating the feature maps helps in getting a better region of interest, a more informative spatial representation of the image. Our work contributes to the ongoing development of deep learning techniques for image-to-image translation, and provides insights into how the combination of different models and techniques can lead to better performance in the filed of computer vision research.

6.2 Future Work

The image-to-image translation capabilities of the model shown in this work seem promising.However,the future scopes of this thesis can be designing an effective system for performing the image translation task that can be more robust to abrupt shape and texture changes of the images. Future work could explore other combinations of models and techniques, and further improve the performance of image-to-image translation systems.

References

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

[2] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CoRR*, vol. abs/1611.07004, 2016.

[3] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," *SIGGRAPH*, 2001.

[4] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," *CoRR*, vol. abs/1603.08155, 2016.

[5] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," *CoRR*, vol. abs/1411.4734, 2014.

[6] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," *CoRR*, vol. abs/1210.5644, 2012.

[7] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *ACM SIGGRAPH 2007 papers*, pp. 10–es, 2007.

[8] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," in *Parallel distributed processing, Vol. 1: Foundations* (D. E. Rumelhart and J. L. McClelland, eds.), pp. 194–281, Cambridge, MA: MIT Press, 1986.

[9] R. Salakhutdinov and H. Larochelle, "Efficient learning of deep boltzmann machines," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 693–700, JMLR Workshop and Conference Proceedings, 2010.

[10] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2022.

[11] M. Liu, T. M. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *CoRR*, vol. abs/1703.00848, 2017.

[12] M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed, "Variational approaches for auto-encoding generative adversarial networks," 2017.

[13] J. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," *CoRR*, vol. abs/1711.11586, 2017.

[14] E. L. Denton, S. Chintala, a. szlam, and R. Fergus, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Advances in Neural Information Processing Systems* (C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds.), vol. 28, Curran Associates, Inc., 2015.

[15] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016.

[16] J. J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," *CoRR*, vol. abs/1609.03126, 2016.

[17] I. J. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," *CoRR*, vol. abs/1701.00160, 2017.

[18] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems* (D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, eds.), vol. 29, Curran Associates, Inc., 2016.

[19] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun, "Disentangling factors of variation in deep representation using adversarial training," in *Advances in Neural Information Processing Systems* (D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, eds.), vol. 29, Curran Associates, Inc., 2016.

[20] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *CoRR*, vol. abs/1411.1784, 2014.

[21] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015.

[22] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," *CoRR*, vol. abs/1604.04382, 2016.

[23] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays, "Scribbler: Controlling deep image synthesis with sketch and color," *CoRR*, vol. abs/1612.00835, 2016.

[24] L. Karacan, Z. Akata, A. Erdem, and E. Erdem, "Learning to generate images of outdoor scenes from attributes and semantic layouts," *CoRR*, vol. abs/1612.00215, 2016.

[25] M. Liu and O. Tuzel, "Coupled generative adversarial networks," *CoRR*, vol. abs/1606.07536, 2016.

[26] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *CoRR*, vol. abs/1703.10593, 2017.

[27] D. Bashkirova, B. Usman, and K. Saenko, "Unsupervised video-to-video translation," *CoRR*, vol. abs/1806.03698, 2018.

[28] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2414–2423, 2016.

[29] X. Huang, M. Liu, S. J. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," *CoRR*, vol. abs/1804.04732, 2018.

[30] T. Park, M. Liu, T. Wang, and J. Zhu, "Semantic image synthesis with spatially-adaptive normalization," *CoRR*, vol. abs/1903.07291, 2019.

[31] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," *CoRR*, vol. abs/1711.09020, 2017.

[32] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," *CoRR*, vol. abs/1704.02510, 2017.

[33] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," *CoRR*, vol. abs/1611.02200, 2016.

[34] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems* (F. Pereira, C. Burges, L. Bottou, and K. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014.

[38] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[39] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *CoRR*, vol. abs/1605.06409, 2016.

[40] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," *CoRR*, vol. abs/1703.06870, 2017.

[41] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *CoRR*, vol. abs/1503.03832, 2015.

[42] P. Kamencay, M. Benco, T. Mizdos, and R. Radil, "A new method for face recognition using convolutional neural network," *Advances in Electrical and Electronic Engineering*, vol. 15, no. 4, 2017.

[43] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *CoRR*, vol. abs/1411.4038, 2014.

[44] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," 2016.

[45] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[47] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016.

[48] "UC Berkeley's official directory of CycleGAN Datasets." https://people.eecs.berkeley.edu/~taesung_park/CycleGAN/datasets/. Accessed: 2022-11-05.