

PEOPLE'S DEMOCTATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

University of Mohamed El-Bachir El-Ibrahimi_Bordj Bou Arreridj
Faculty of Science and Technology

Department of Electronic

Master Thesis



Université Mohamed El Bachir El Ibrahimi - B.B.A -



Université Mohamed El Bachir El Ibrahimi - B.B.A -

Presented to obtained

THE MASTER'S DIPLOMA

SECTOR: ELECTRONICS

Specialty: Telecommunication Systems

By

- BAHRI Chaima
- DERROU Mehdi

Entitled

Application of generative adversarial networks for image super Resolution

Supported on:

First name & family name :

Grade

Quality

Etablissement

Mme.F.FARES

MCA

Chairman

Univ-BBA

M. D.E. BOUDECHICHE

MCA

Supervisor

Univ-BBA

Mme.C.ERRIDIR

MAB

Examiner

Univ-BBA

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

۱۴۳۸



Acknowledgements

Above all, it seems right to thank Allah for providing us with the courage, patience, and knowledge needed to complete this task.

With these brief words of gratitude, I would like to express our appreciation to everyone who helped us accomplish this mission through their presence, support and guidance.

Firstly, I would like to extend my thanks to my supervisors, **Dr. Boudechiche Djameleddine** and **Dr. Messali Zoubeida**.

We should also not forget **Saoudi Rania** for the instructions and guidance she provided us.

We would like to express our special thanks to our supervisor for his invaluable support, help and guidance during the completion of our thesis. It was a pleasure to work with him, and we learned a lot from him. We would like to extend our sincere thanks and express our deep appreciation to him.



Dedications

“Whoever says I am hers will get her.”

The trip was not short, nor should it be. The dream was not close, and the path was not easy, but I did it and achieved it.

To my pure angel, my strength after God, my first and eternal support, my dear mother, to the one whose name I carry with pride and pride, to the one who taught me to give without waiting, my father, may God protect him. And prolong his life.

To the soul of my grandfather, may God have mercy on him, to my grandmother’s precious jewel, to those who believed in my abilities and secure my days, my sisters (Meriem, Fatma , Hadjer and Oumaima).

I dedicate this humble work to you.

Miss Bahri Chaima



Dedications

I dedicate this humble work to...
Who put me first, raised me, and taught me what is right
To my dear father
thanks Dad....
To the one who taught me resilience and hope
The greatest and most compassionate person in existence
To my beloved mother, may God prolong her life
thanks Mom....
To that mountain that when the world leans against me, I
support myself in times of adversity
Thank you, my brother Hicham and my brother Riyad
To the beauty of the universe in my eyes...
You are my smile... You are my blood...
Thank you, my sisters Ghizlene and houda .

Mr. Derrou Mehdi

Abstract

In image super-resolution applications, generative adversarial networks (GANs) are often employed to convert low-resolution images into high-resolution ones, thus improving image quality. These networks are made up of a discriminator network that assesses the validity of the generated samples and a generator network that generates new samples based on the data. Image quality has been significantly improved using GANs, especially in situations where data samples are scarce. In this thesis, we implemented and tested several models of image super resolution based on GANs such as: SRGAN, ESRGAN and Real ESRGAN. We compared the obtained results of the deep learning methods with traditional methods like Bicubic interpolation using the PSNR and the SSIM metrics.

Keywords: Super-resolution, GAN, SRGAN, ESRGAN, Real-ESRGAN

Résumé

Dans les applications d'images à super-résolution, les réseaux antagonistes génératif (GAN) sont souvent utilisés pour convertir des images basse résolution en images à haute résolution, améliorant ainsi la qualité de l'image. Ces réseaux sont constitués d'un réseau discriminatoire qui évalue la validité des échantillons générés et d'un réseau générateur qui génère de nouveaux échantillons à partir des données. La qualité des images a été considérablement améliorée grâce aux GAN, en particulier dans les situations où les échantillons de données sont rares. Dans cette thèse, nous implémentons et testons plusieurs modèles de super-résolution d'images basés sur des GAN, tels que : SRGAN, ESRGAN et Real ESRGAN. Nous avons comparé les résultats obtenus par les méthodes d'apprentissage profond avec les méthodes traditionnelles telles que l'interpolation bicubique en utilisant les métriques PSNR et SSIM.

Mots-clés : la super-résolution, GAN, ESRGAN, Real-ESRGAN.

ملخص

في تطبيقات الصور فائقة الدقة غالبا ما تستخدم شبكات الخصومة التوليدية لتحويل الصور ذات الدقة المنخفضة الى الصور ذات دقة عالية و بالتالي تحسين جودة الصور. تتكون هذه الشبكات من شبكة تمييزية تقوم بتقييم صحة العينات التي تم انشاؤها و شبكة مولدات تقوم بانشاء عينات جديدة بناء على البيانات. يتم تحسين جودة الصور بشكل ملحوظ باستخدام شبكات الخصومة التوليدية خاصة في الحالات التي تكون فيها عينات البيانات نادرة. في هذه الأطروحة، ننفذ ونختبر العديد من نماذج التحليل الفائق للصور القائمة على شبكة GAN مثل: SRGAN و ESRGAN و Real-ESRGAN. قمنا بمقارنة

النتائج التي حصلنا عليها من طرق التعلم العميق مع الطرق التقليدية مثل الاستيفاء التكميبي ثنائي التكعيب باستخدام مقاييس PSNR و SSIM.

كلمات مفتاحية: دقة فائقة GAN, ESRGAN, Real-ESRGAN

List of Figures

Figure 1.1: The graph of a perceptron.

Figure 1.2: Multilayer Perceptron.

Figure 1.3: Types of activation function.

Figure 1.4: Types of deep learning.

Figure 1.5: Convolutional Neural Network.

Figure 1.6: Convolution layer.

Figure 1.7: Pooling layers.

Figure 1.8: VGG19 Architecture

Figure 1.9: variational autoencoder

Figure 1.10: Architecture of GAN.

Figure 1.11: Diffusion models

Figure 2.1: Example of bilinear interpolation

Figure 2.2: Example of bicubic interpolation

Figure 2.3: Nearest Neighbour

Figure 2.4: Super-Resolution Convolutional Neural Networks

Figure 2.5: Generator architecture

Figure 2.6: Discriminator Architecture

Figure 2.7: ESRGAN Architecture

Figure 3.1: Original HR images of the used datasets

Figure 3.2: PSNR and SSIM for dataset5 scale 2

Figure 3.3: PSNR and SSIM for dataset5 scale 4

Figure 3.4: Results of the classical method

Figure 3.5: Learning evolution

Figure 3.6: Loss of trning for 200 epochs:(a) generative Loss, (b)discriminator Loss

Figure 3.7: PSNR and SSIM for dataset5 scale 4 of deep learning super resolution

List of Tables

Table 3.1: datasets for testing and training

Table 3.2: PSNR & SSIM of the classical method

Table 3.3: terms of learning SRGAN

Table 3.4: PSNR & SSIM of SRGAN and Bicubic Interpolation

Table 3.5: PSNR & SSIM of ESRGAN and Bicubic Interpolation

Table 3.6: PSNR & SSIM of Real-ESRGAN and Bicubic Interpolation

List of Abbreviations

AI Artificial intelligence.

DL Deep Learning.

ML Machine Learning.

SR Super Resolution.

NN's Neural Networks.

HR High Resolution.

LR Low Resolution.

MA Mean of these Absolute Errors.

MSE Mean Square Error.

MOS Mean Opinion Score.

ReLU Rectified Linear Unit.

PSNR Peak Signal-to-Noise Ratio.

SSIM Similarity Structural Index Metric.

CNN Convolutional Neural Network.

GAN Generative Adversarial Network.

SRGAN Super Resolution Generative Adversarial Network.

ESRGAN Enhanced Super-Resolution Generative Adversarial Networks.

VAEs variational autoencoders.

Table of Contents

| | |
|---|----|
| Abstract..... | |
| List of Figures..... | |
| List of Abbreviations | |
| CHAPTER I: Principles of deep Learning | |
| I.1 Introduction..... | 17 |
| I.2 Neural Networks | 17 |
| I.2.1 Neuron..... | 17 |
| I.2.2 Perceptron..... | 18 |
| I.2.3 Multilayer Perceptron..... | 19 |
| I.3 Activation functions..... | 19 |
| I.3.1 Linear activation function | 19 |
| I.3.2 Non-linear activation function | 19 |
| A. Sigmoid Function..... | 20 |
| B. Tanh or hyperbolic tangent Activation Function | 20 |
| C. ReLU (Rectified Linear Unit) Activation Function | 20 |
| I.4 Loss Function..... | 21 |
| I.4.1 L1 Loss function..... | 21 |
| I.4.2 L2 Loss Function..... | 21 |
| I.4.3 Mean Absolute Error (MAE)..... | 21 |
| I.5 Optimization | 22 |
| I.5.1 Momentum | 22 |
| I.5.2 RMSProp..... | 22 |
| I.5.3 ADAM optimizer..... | 22 |
| I.6 Deep Learning..... | 23 |
| I.7 Convolutional Neural Network..... | 24 |
| I.7.1 Convolution layer..... | 24 |
| I.7.2 Pooling layers:..... | 25 |
| I.8 Generative Models..... | 26 |
| I.8.1 variational autoencoder | 26 |

| | |
|--|----|
| I.8.2 Generative Adversarial Network | 27 |
| I.8.3 Diffusion models | 28 |
| I.9 Conclusion | 29 |
| CHAPTER II : Basic concepts of super-Resolution | 30 |
| II.1 Introduction | 31 |
| II.2 Super-resolution..... | 31 |
| II.3 Convolutional algorithms for super-resolution | 32 |
| II.3.1 Bilinear interpolation | 32 |
| II.3.2 Bicubic interpolation..... | 33 |
| II.3.3 Nearest Neighbour | 33 |
| II.4 Deep learning for super-resolution | 34 |
| II.4.1 Convolutional Neural Networks for super-resolution..... | 34 |
| II.4.2 Super-Resolution Generative Adversarial Networks (SRGAN)..... | 34 |
| A. Generator architecture..... | 35 |
| B. Discriminator Architecture | 36 |
| II.4.3 Enhanced Super-Resolution Generative Adversarial Network | 37 |
| II.4.4 Real-ESRGAN | 38 |
| II.5 Metric evaluation..... | 38 |
| II.5.1 Peak Signal-to-Noise Ratio (PSNR) | 38 |
| II.5.2 Structural Similarity Index (SSIM)..... | 39 |
| II.6 Conclusion..... | 40 |
| CHAPTER III :Implementation of image Super-Resolution Algorithms based on Generative Adversarial Network and Deep Learning SR Algorithms | 41 |
| III.1 Introduction | 42 |
| III.2 Used material | 42 |
| III.2.1 Used datasets | 42 |
| III.2.2 Original Images | 43 |
| III.3 The classical methods of super resolution..... | 44 |
| III.3.1 Results of classical methods | 44 |
| III.3.2 Comparison of classical method..... | 47 |

| | |
|---|----|
| III.4 Implementation of deep learning super resolution..... | 48 |
| III.4.1 Super resolution of generative adversarial networks (SRGAN)..... | 48 |
| III.4.1.1 Explanation of training conditions | 48 |
| III.4.1.2 Training process..... | 48 |
| III.4.1.3 Algorithm of SRGAN | 49 |
| III.4.2 Results of SRGAN..... | 49 |
| III.4.2.1 Learning evolution..... | 49 |
| III.4.3 Results PSNR and SSIM of deep learning super resolution..... | 51 |
| Fig.3.7. PSNR and SSIM for dataset5 scale 4 of deep learning super resolution | 53 |
| III.5 Comparison for Deep learning for super-resolution and Bicubic..... | 54 |
| III.6 Conclusion..... | 54 |
| General conclusion..... | 56 |

General introduction

Deep learning has seen significant advancements in recent years, largely due to the availability of large datasets, powerful computational resources, and improved algorithms. It has achieved remarkable success in various applications, including image and speech recognition, natural language processing, autonomous vehicles, healthcare, and many others.

In this thesis, we will focus on the generative adversarial network, where generative adversarial networks (GANs) have become a cornerstone in the field of image super-resolution (SR), offering a sophisticated approach to generate high-resolution images (HR) from low-resolution (LR) inputs. The fundamental principle behind GANs involves a generator network learning to produce realistic HR images while being adversarial trained against a discriminator network that aims to differentiate between real and generated images. Through the interplay between these networks, GANs can effectively capture intricate details and textures, enhancing the visual quality of images beyond what traditional interpolation methods can achieve. Despite challenges such as mode collapse and training instability, advancements in GAN architectures and training strategies continue to push the boundaries of SR image, enabling applications in diverse domains such as surveillance, medical imaging, and multimedia content creation.

The manuscript is organized into a general introduction and three chapters, in addition to a general conclusion.

In the first chapter: We presented the basic concepts related to deep learning and high-resolution images.

In the second chapter: we present the most commonly used terms in the field of deep learning and ultra-resolution networks.

In Chapter Three: We explain our implementation of three super-resolution image networks for deep learning, namely: SRGAN, ESRGAN and Real-ESRGAN, and traditional SR algorithms: bicubic, bilinear, and nearest neighbor. We give global observations in the general conclusion.

CHAPTER I

Deep learning

I.1 Introduction

Artificial intelligence (AI) represents a significant stride in comprehending human cognition and problem-solving through machine demonstrations. Within AI, various subdomains encapsulate its multifaceted applications (Machine learning and Deep Learning).

These applications span diverse fields such as cyber security, healthcare, commerce, and scientific research.

Neural networks (NN) and deep learning (DL), integral components of AI, stand out as pivotal solutions across domains like image and audio processing, as well as natural language understanding. These methodologies offer state-of-the-art solutions by leveraging intricate network architectures and sophisticated algorithms.

The on-going research and development in the field ensure that deep learning will play a pivotal role in shaping the future of intelligent systems [1].

I.2 Neural Networks

Neural networks (NN) are a type of machine learning algorithm inspired by the structure and functioning of the human brain. They are composed of interconnected nodes, called neurons, organized in layers. Each neuron receives input, performs a mathematical operation on that input, and then passes the result to the next layer of neurons through connections called weights.

Neural networks are capable of learning complex patterns and relationships in data through a process called training. During training, the network adjusts the weights of connections between neurons based on example data, typically using an optimization algorithm such as gradient descent. This process allows neural networks to generalize from the training data and make predictions or decisions on new, unseen data.

Neural networks have shown remarkable performance in various tasks, including image and speech recognition, natural language processing, and predictive modeling. They are a fundamental building block of artificial intelligence and have become increasingly prevalent in a wide range of applications across industries [2].

I.2.1 Neuron

Neurons, also known as nerve cells, serve as the fundamental components of the brain and nervous system. They play a pivotal role in receiving sensory input from the external environment,

transmitting motor commands to our muscles, and facilitating the transformation and relay of electrical signals at various stages in this intricate process. Importantly, the interactions among neurons not only govern basic physiological functions but also define our individual identities.

It's worth noting that the approximately 100 billion neurons in our nervous system engage closely with other cell types, broadly categorized as glia. Surprisingly, the number of glial cells may even surpass that of neurons, although this remains an area of on-going investigation.

The on-going generation of new neurons in the brain is referred to as neurogenesis, and remarkably, this phenomenon persists into adulthood [3].

1.2.2 Perceptron

A formal neuron, or perceptron, is a fundamental component of an artificial neural network. It operates as a mathematical function, receiving one or multiple inputs, applying weights to them, and generating a single output. The output is calculated by summing the weighted inputs, and then it passes through an activation function. This activation function introduces nonlinearity to the output, enabling the neural network to learn more complex functions [4].

Weight The main function of weight is to emphasize characteristics that have a more significant impact on learning. This is achieved by performing a scalar multiplication between the input value and the weight matrix [4].

Bias The bias is responsible for adjusting the output of the activation function. It plays a role analogous to a constant term in a linear equation (akin to a negative threshold) [4].

As shown in the following figure 1.1

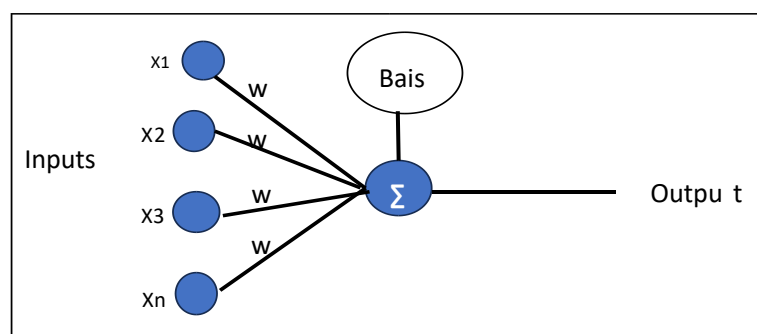


Fig 1.1. The perceptron artificial.

I.2.3 Multilayer Perceptron

This neural network has more than three layers and is particularly suitable for classifying non-linear data. Each node in these networks is connected to all the others. They find their use primarily in applications such as speech recognition and other areas of machine learning [5].

As shown in the following figure 1.2

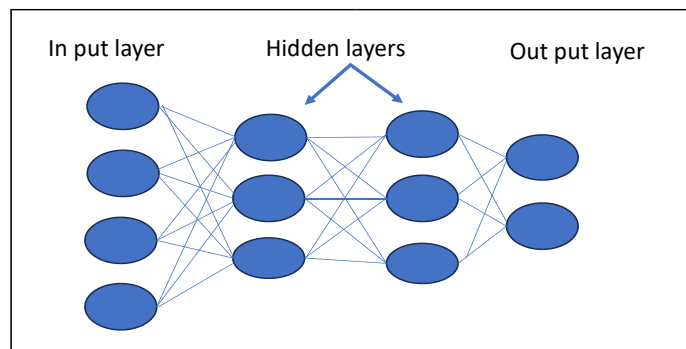


Fig.1.2: Multilayer Perceptron.

I.3 Activation functions

It is used to determine the output of neural network like yes or no. It maps the resulting values in between 0 to 1 or -1 to 1 etc. (depending upon the function) [6] [7] [8] [9].

The Activation Functions can be basically divided into 2 types:

- Linear Activation Function.
- Non-linear Activation Function.

I.3.1 Linear activation function

As evident in figure 1.3.a from the linear nature of the function, its output is not constrained within any specific range.

It doesn't help with the complexity or various parameters of usual data that is fed to the neural networks.

I.3.2 Non-linear activation function

The Nonlinear Activation Functions are widely favoured in neural networks. Their inherent nonlinearity ensures that the graph exhibits curves rather than straight lines, resulting in a more flexible and expressive mode. It makes it easy for the model to generalize or adapt with variety of data and to differentiate between the output.

A. Sigmoid Function

The sigmoid function is widely used in neural networks because it outputs values between 0 and 1, making it suitable for predicting probabilities. It is differentiable, allowing the calculation of slopes at any point, but its derivative is not monotonic. However, its application in neural networks can lead to training issues.

B. Tanh or hyperbolic tangent Activation Function

The tanh function offers the advantage of mapping negative inputs strongly negative and zero inputs near zero on its graph. It is differentiable and monotonic, though its derivative lacks monotonicity. Primarily utilized for classification between two classes, both tanh and logistic sigmoid activation functions find applications in feed-forward neural networks.

C. ReLU (Rectified Linear Unit) Activation Function

The ReLU is the most used activation function in the world right now. Since, it is used in almost all the convolutional neural networks or deep learning.

It seems that the activation functions you're using (ReLU) is experiencing an issue with negative values. This leads to the loss of important information regarding negative values and affects the model's ability to learn properly. To address this issue, you can use another activation function such as Leaky ReLU, Parametric ReLU, or ELU that allows some negative values to pass through without converting them all to zero. This can help in better data representation and improve the model's ability to learn from the data.

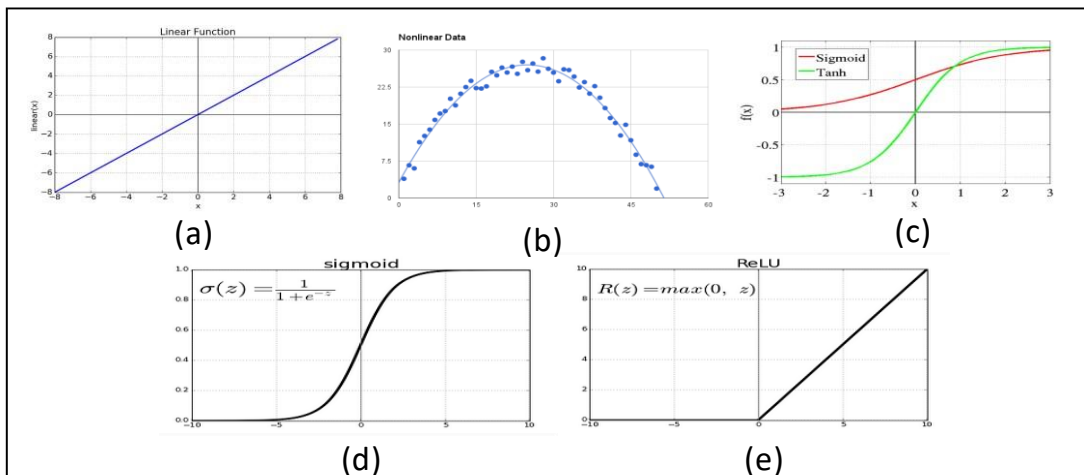


Fig.1.3. Types of activation function: (a). Linear Activation Function. (b). Non-linear Activation Function. (c). Tanh function. (d). Logistic Sigmoid. (e). ReLU [6] [7] [8] [9].

I.4 Loss Function

The loss function is a key element in many deep learning algorithms and is important for both training and optimizing models [10].

I.4.1 L1 Loss function

It is used to minimize the error which is the sum of all the absolute differences in between the true value and the predicted value.

$$\text{L1LossFunction} = \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (1.1)$$

L1 loss is also known as the **absolute error** and the cost is the Mean of these Absolute Errors (MAE).

I.4.2 L2 Loss Function

It is also used to minimize the error which is the sum of all the squared differences in between the true value and the predicted value [10].

$$\text{L2LossFunction} = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (1.2)$$

I.4.3 Mean Absolute Error (MAE)

From a mathematical perspective, MAE is calculated by summing the absolute differences between predicted values (\hat{y}) and actual values (y) [10], and then dividing by the total number of samples (n):

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (1.3)$$

I.4.4 Mean Square Error (MSE)

Mean Square Error (MSE) is a widely used statistical metric to evaluate the quality of predictions or estimations by measuring the average squared difference between predicted values and actual value [10].

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

(1.4)

I.5 Optimization

Within the realm of deep learning, an optimizer holds pivotal significance as it meticulously adjusts the parameters of a neural network throughout its training phase. Its fundamental objective revolves around the reduction of the model's error or loss function, thereby amplifying its overall performance. Diverse optimization algorithms, commonly referred to as optimizers, deploy unique methodologies to steadily approach optimal parameter values, thus facilitating enhanced predictive capabilities in an efficient manner [11].

I.5.1 Momentum

In the realm of neural network optimization, momentum serves as a technique utilized to accelerate the training process by incorporating past parameter updates. This methodology helps mitigate some of the shortcomings observed in traditional gradient descent approaches, such as slow convergence and oscillations around local minima. Essentially, employing momentum provides the model with added strength to overcome obstacles and improve its trajectory towards optimal solutions more rapidly and with greater stability [11].

I.5.2 Root Mean Squared Propagation (RMSProp)

RMSProp, short for Root Mean Square Propagation, emerges as an adaptive optimization algorithm tailored to tackle challenges inherent in training deep neural networks, a domain where stochastic gradient descent (SGD) may fall short. Introduced by Geoffrey Hinton during his coursera neural networks course, RMSProp hasn't been formally published but has garnered widespread adoption owing to its efficacy across diverse applications [11].

I.5.3 Adaptive Moment Estimation (ADAM) optimizer

In our architecture, we used Adam as an optimization algorithm. Adam is a widely used optimization algorithm that combines the advantages of both adaptive learning rate methods and momentum-based methods [11].

I.6 Deep Learning

Deep learning is a subset of machine learning that use a learning algorithm on multiple levels of distributed representations to allow us extract the useful patterns from data automatically with less and less as possible of human intervention.

Deep learning has emerged as a dominant paradigm in artificial intelligence due to its unparalleled ability to automatically learn hierarchical representations of data. Unlike traditional machine learning approaches that rely on manual feature engineering, deep learning algorithms can extract relevant features directly from raw data, enabling them to capture complex patterns and relationships. This flexibility, coupled with scalability and state-of-the-art performance, has made deep learning indispensable across diverse domains such as computer vision, natural language processing, and speech recognition. By enabling end-to-end learning and adaptation to different datasets, deep learning offers a powerful and versatile approach to solving complex real-world problems efficiently and effectively [12].

Connectionist architectures boast a rich history spanning over seven decades, yet their recent resurgence at the forefront of artificial intelligence owes much to the advent of innovative architectural designs and the harnessing of graphical processing units (GPUs). Deep learning isn't confined to a single methodology; rather, it embodies a diverse range of algorithms and network structures adaptable to a myriad of problems.

Although deep learning isn't a novel concept, its current meteoric rise can be credited to the convergence of deeply layered neural networks and the exploitation of GPU acceleration. Moreover, the proliferation of vast datasets, often referred to as big data, has been instrumental in driving this growth. Deep learning heavily relies on training neural networks with extensive example data, thus rewarding them based on their performance. Consequently, the availability of more data equips us better to construct resilient deep learning architectures.

The landscape of deep learning is characterized by a plethora of architectures and algorithms developed over the past two decades. Notably, long short-term memory (LSTM) networks and convolutional neural networks (CNNs) emerge as stalwarts among the earliest approaches in this repertoire, yet they continue to enjoy widespread adoption across diverse applications. This article classifies deep learning architectures into supervised and unsupervised learning and introduces several deep learning architectures [13].

As shown in the following figure

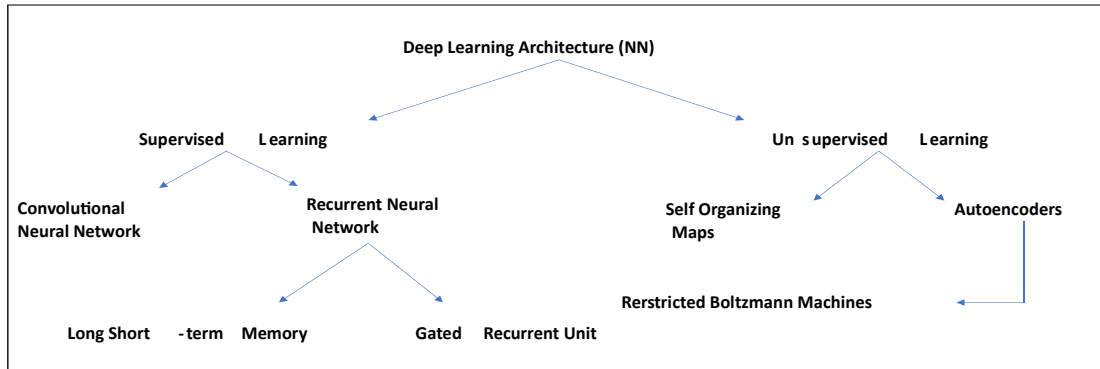


Fig.1.4. Types of deep learning.

I.7 Convolutional Neural Network

The Convolutional Neural Network (CNN) is a powerful deep learning architecture primarily employed for tasks related to image and video processing. It's characterized by its distinct composition of convolutional layers, pooling layers, and fully connected layers, as illustrated in Figure 1.5. These components work collaboratively to extract features from input data, enabling CNNs to achieve remarkable performance in tasks like object detection, image classification, and semantic segmentation [14].

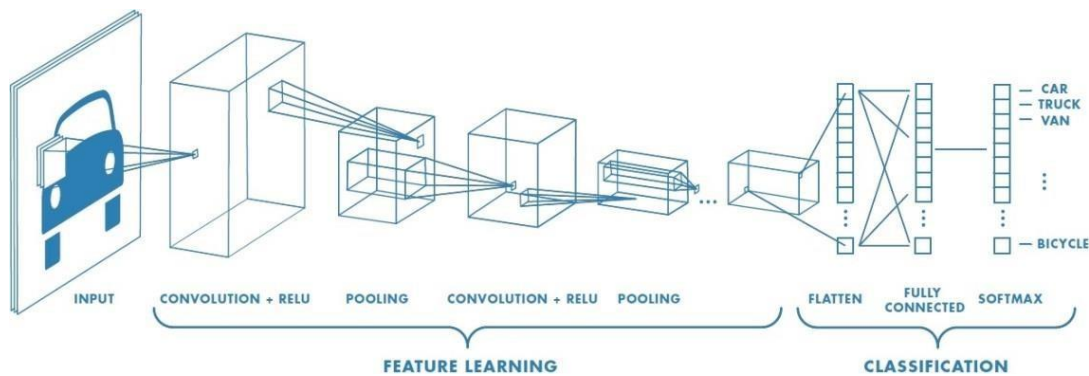


Fig1.5: Convolutional Neural Network [14].

I.7.1 Convolution layer

Convolutional layers play a crucial role as fundamental building blocks within CNNs. These layers are pivotal for feature extraction, employing a set of trainable filters known as convolutional filters or kernels. These filters are applied to input images using convolution operations, which differ from simple matrix multiplications. In convolution, each pixel value is multiplied by the corresponding weight in the filter, and the resulting products are summed to

generate a feature map output matrix. This matrix, representing extracted features, undergoes further processing, typically passing through an activation function layer to introduce nonlinearity into the network [14]. As shown in the following figure 1.6.

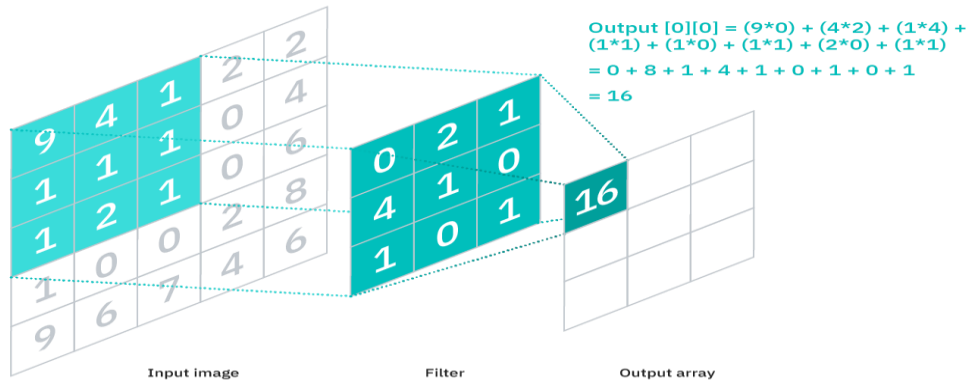


Fig1.6: Convolution layer [14].

I.7.2 Pooling layers:

Pooling layers serve the purpose of dimensionality reduction within convolutional neural networks. As shown in the following figure 1.7

By diminishing the spatial dimensions of feature maps, they contribute to a more efficient processing of data, which is particularly advantageous in terms of computational

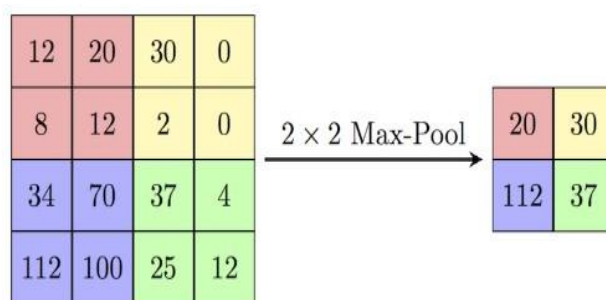


Fig1.7: Pooling layers [15].

resources. The two primary types of pooling, namely Max Pooling and Average Pooling, offer distinct methods of aggregating information while retaining the most salient features for

I.7.3 VGG 19

VGG19 is a convolutional neural network introduced by Simonyan et al [16], is a convolutional neural network of 19 layers, 16 convolution layers, and 3 fully connected layers. VGG19 was trained using the ImageNet collection, which consists of a million images divided into 1000 categories. Using several 3x3 filters in every convolutional layer is a common method for classifying images.

The next three convolutional layers are utilized to classify the features that were extracted using the first sixteen convolutional layers. The five groups of feature extraction layers are followed by a max-pooling layer. This model creates the item's label in the photos after receiving a 224 by 224 pixel image. While many machine learning techniques are employed for classification, the study extracts features using a pre-trained VGG19 model [15]. As shown in the following figure 1.8

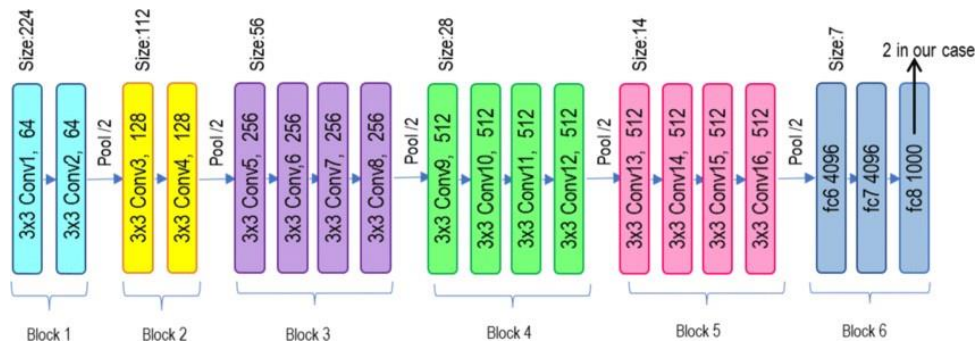


Fig1.8: VGG19 Architecture [16].

I.8 Generative Models

I.8.1 variational autoencoder

Strong deep generative models called variational autoencoders (VAEs) are frequently used to represent high-dimensional complicated data using a low-dimensional latent space that is trained supervised. The input data vectors are handled separately in the original VAE model. Many extensions of the VAE to process sequential data have recently been presented in a number of papers. These extensions rely on recurrent neural networks or state-space models and model both the latent space and the temporal dependencies within a sequence of data vectors and corresponding latent vectors. We provide a review of the literature on these models

in this study. A broad subset of these temporal VAE extensions is referred to as dynamical variational autoencoders (DVAEs) [16] [17]. As shown in the following figure1.9

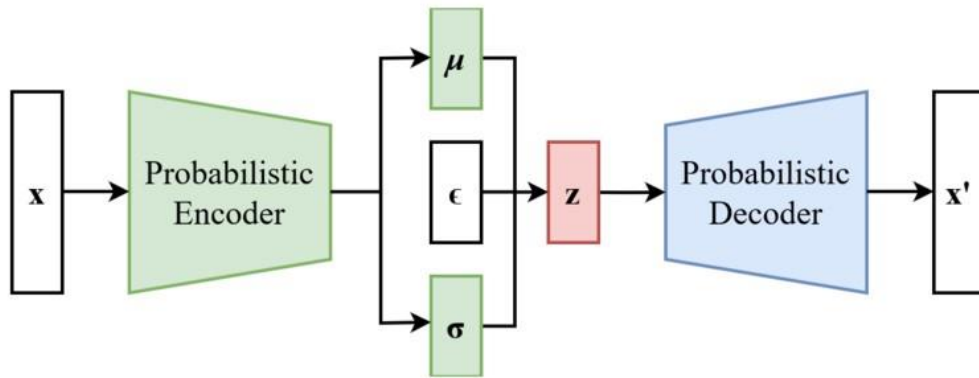


Fig.1.9. variational autoencoder [16].

I.8.2 Generative Adversarial Network

A GAN, or Generative Adversarial Network, is a type of deep learning system where two neural networks are trained to generate new data from an existing dataset. They work in competition: one network generates new data samples while the other tries to distinguish between the generated and real data. This adversarial process leads to the generation of increasingly authentic data until it becomes difficult to differentiate between the generated and original data.

Represent a cutting-edge advancement in deep learning. These networks are designed to generate data, such as images, with remarkable efficiency, often surpassing the capabilities of traditional detectors. Once trained, a GAN can produce synthetic data, like realistic human faces, with impressive fidelity.

The significance of GANs in scientific research and fields like medical image analysis is growing rapidly due to their ability to create lifelike images. This is why we incorporated GAN technology into our study [18].

A GAN consists of two models working in tandem to achieve its objectives.

The initial component, known as the generator, is tasked with generating new data that closely resembles the expected data. It operates much like a human art forger, crafting counterfeit works of art.

On the other hand, the second model is referred to as the Discriminator. Its objective is to differentiate whether input data is authentic – belonging to the original dataset – or counterfeit – fabricated by the forger. In this context, the discriminator functions akin to an art expert, striving to discern truthful artworks from fraudulent ones. As shown in the following figure 1.10

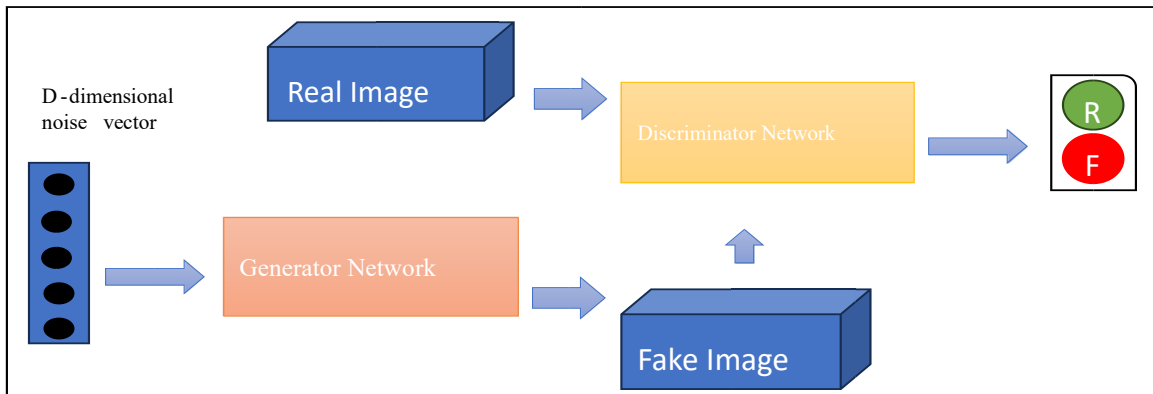


Fig.1.10. Architecture of GAN.

The procedural steps of a GAN are as follows:

1. The generator receives random numbers and produces an image.
2. This newly generated image is presented to the discriminator, alongside a continuous stream of images drawn from the genuine dataset.
3. The discriminator assesses both genuine and counterfeit images, delivering probabilities ranging from 0 to 1, where a value of 1 signifies an authenticity prediction and 0 denotes a fraudulent one.

I.8.3 Diffusion models

Diffusion models are mathematical frameworks used to describe the spread or dissemination of various phenomena, such as innovations, ideas, behaviors, diseases, or information, through a population, network, or spatial domain. These models aim to capture the underlying mechanisms governing the process of how these phenomena propagate over time, often characterized by an initial slow uptake, followed by rapid adoption, and eventually reaching

saturation [18]. As shown in the following figure 1.11

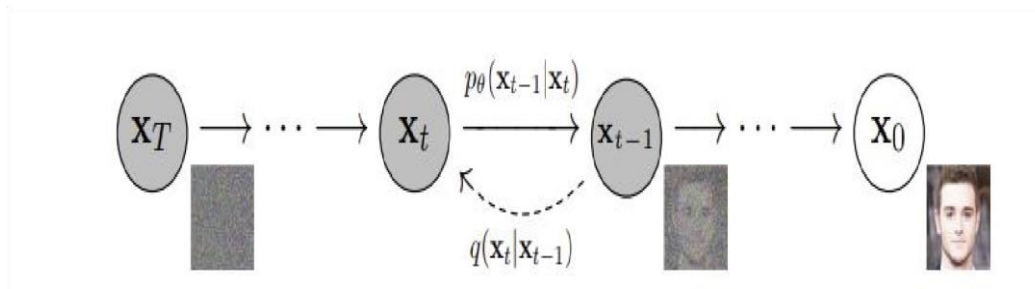


Fig.1.11. Diffusion models [20].

I.9 Conclusion

Deep learning represents a powerful paradigm in the field of artificial intelligence, inspired by the complex structure and functioning of the human brain. Through neural networks and architectures like GANs, deep learning models have demonstrated remarkable capabilities in tasks such as image recognition, natural language processing, and generative modeling. With ongoing advancements in algorithms, hardware, and data availability, deep learning continues to push the boundaries of what is possible in machine learning and AI, promising further breakthroughs and innovations in various domains and industries.

CHAPTER II

Super-resolution

II.1 Introduction

In this chapter, we will talk about the generative adversarial network and its three applications (Super-Resolution Generative Adversarial Networks, Enhanced Super-Resolution Generative Adversarial Network and Real-ESRGAN).

Where super-resolution techniques have become increasingly indispensable in various fields where high-quality image data is paramount. In medical imaging, for instance, SR enables the enhancement of diagnostic images, allowing healthcare professionals to discern finer details crucial for accurate diagnosis and treatment planning. Similarly, in surveillance systems, super resolution can sharpen images captured by security cameras, aiding in the identification of objects or individuals even in challenging conditions such as low light or long distances. Moreover, in satellite imagery, SR facilitates the extraction of precise geographical information and monitoring of environmental changes with enhanced clarity and detail. Digital photography also benefits significantly from SR, enabling photographers to produce sharper, more detailed images even in situations where the available hardware imposes limitations on resolution. Through sophisticated computational algorithms, SR techniques bridge the gap between low resolution input data and the high-quality visual output demanded by various applications, thereby unlocking a myriad of possibilities across industries.

II.2 Super-resolution

Super-resolution refers to a collection of imaging techniques designed to improve the resolution of low-resolution images or videos, resulting in higher-resolution outputs. Although initially mentioned in the mid-1980s, the term "super-resolution" gained traction around 1990 and has since become increasingly valuable [18]. Despite the diverse range of approaches, they generally involve reconstructing missing pixels in low-resolution images using specific algorithms.

The primary objective of super-resolution (SR) is to generate higher-resolution images from lower-resolution inputs. These higher-resolution images boast increased pixel density, thereby capturing finer details of the original scene. Computer vision applications, which seek improved performance in tasks like pattern recognition and image analysis, consistently demand high resolution. Additionally, high resolution is crucial in medical imaging for accurate diagnosis.

Many practical applications necessitate zooming into specific areas of interest within an image, making high resolution indispensable in fields such as surveillance, forensics, and satellite imaging [19].

II.3 Convolutional algorithms for super-resolution

Image interpolation, also known as image upscaling, serves as a prevalent technique in numerous image-related applications. This task frequently employs traditional interpolation methods such as nearest neighbor interpolation, linear interpolation, and cubic interpolation. Among these, bicubic interpolation stands out as a form of cubic interpolation that considers a 4×4 -pixel grid on both axes. This approach yields smoother outcomes with reduced artifacts compared to bilinear interpolation. Nevertheless, entails higher computational complexity, resulting in slower processing speeds [20].

Various techniques have developed in the classical methods of SR to enhance the resolution of images without resorting to deep learning approaches. These methods typically involve image processing and mathematical algorithms to reconstruct HR details from LR images. Here are some of the classical methods commonly used in SR:

II.3.1 Bilinear interpolation

Bilinear interpolation is a simple and fast method that uses a weighted average of the four nearest pixel values to estimate the value of a new pixel. It is a linear interpolation in two dimensions and is commonly used for resizing images. The algorithm consists of three steps: first, it calculates the influence of the two nearest pixels in each direction; next, it calculates the influence of the four nearest pixels; and finally, it calculates the value of the new pixel as the weighted average of the influences of the four nearest pixels [23]. As shown in the following figure2.1

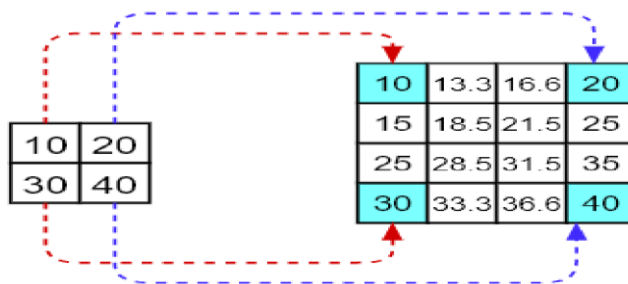


Fig 2.1: Example of bilinear interpolation [23].

II.3.2 Bicubic interpolation

Bicubic interpolation is a widely used method in image processing for estimating the value of a new pixel. It improves upon bilinear interpolation by considering a larger 4x4 neighborhood of known pixels, totaling 16 pixels. This expanded neighborhood allows for a more accurate estimation of the new pixel value.

In bicubic interpolation, each of the 16 neighboring pixels contributes to the calculation of the new pixel value. We give higher weights to pixels closest to the estimated pixel, and less influence to those further away. This weighted approach ensures that nearby pixels have a greater impact on the interpolated result, leading to smoother transitions and more accurate details in the output image.

Due to its ability to consider a larger number of known pixel values during the estimation process, bicubic interpolation produces superior results compared to simpler interpolation methods like nearest neighbor or bilinear interpolation. It is widely regarded as one of the most effective interpolation techniques in image processing, frequently employed for tasks like image resizing, rotation, and enhancement [21]. As shown in the following figure 2.2

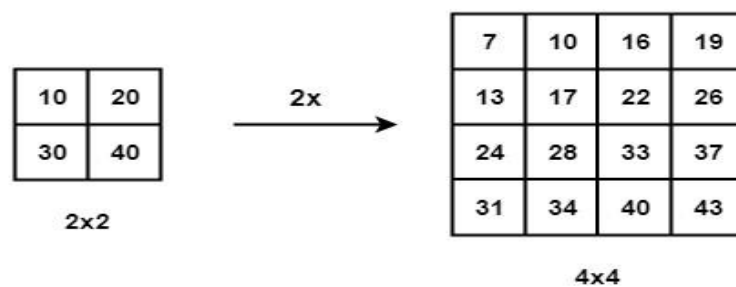


Fig.2.2. Example of bicubic interpolation [24].

II.3.3 Nearest Neighbour

This is the simplest and requires the least processing time of all the interpolation algorithms. The nearest neighbor selects the value of the nearest pixel by rounding the coordinates of the desired interpolation point. Using this method, one finds the closest corresponding pixel in the source image for each pixel in the destination image. We create new pixels identical to those nearby. The pixels, or dots of color, are duplicated to create new pixels as the image grows. It creates pixilation or edges that break up curves into steps or jagged edges. This form of

interpolation suffers from normally unacceptable effects for both enlarging and reducing images [22]. As shown in the following figure2.3

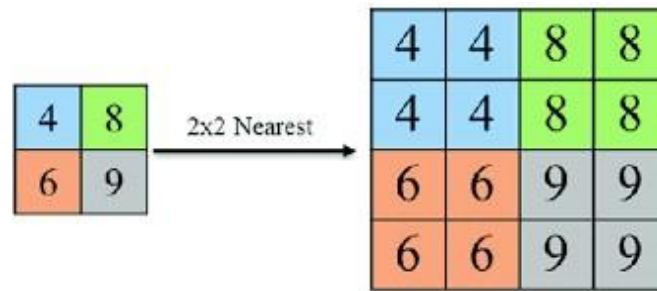


Fig.2.3. Nearest Neighbour [22].

II.4 Deep learning for super-resolution

II.4.1 Convolutional Neural Networks for super-resolution

SRCNNs, which stand for SR convolutional neural networks, are a type of deep convolutional neural network that is capable of converting low-resolution images into high resolution images in an end-to-end manner. Training the network to immediately learn the mapping between HR and LR images is done in order to achieve rapid speed and strong restoration quality for real-world online application. This is done in order to get the desired results [23]. As shown in the following figure2.4

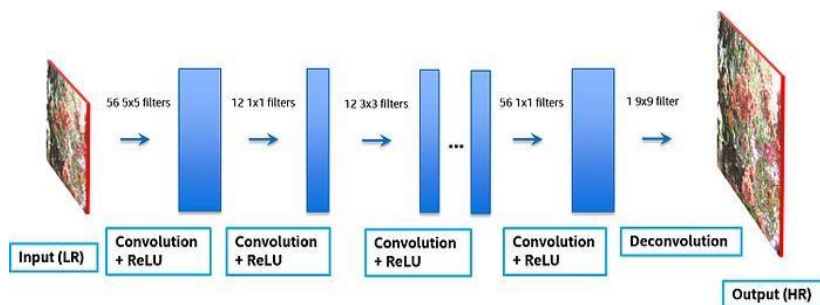


Fig.2.4. Super-Resolution Convolutional Neural Networks [23].

II.4.2 Super-Resolution Generative Adversarial Networks (SRGAN)

The essential concept of GANs is preserved by SRGANs, specifically the min-max function. This function compels the discriminator and generator to learn together by competing with one another [24].

A. Generator architecture

A number of residual blocks, which were model ed after ResNet, are incorporated into the architecture of the generator. It is essential to have these blocks in order to provide efficient training and to enable the network to acquire higher depth, which will ultimately result in superior outcomes. The presence of skip connections inside the residual blocks contributes to an even greater improvement in the effectiveness of training.

The residual blocks are composed of two convolutional layers, each of which utilizes 3×3 kernels and generates 64 feature maps. Following the incorporation of each convolutional layer, batch normalization layers are added, and the parametric ReLU activation function is utilized.

Additionally, the resolution of the input image is enhanced by making use of two sub-pixel convolution layers that have been trained. An improvement in the image's total resolution can be achieved through the use of this augmentation procedure.

Within the framework of this architecture, the activation function is fulfilled by the Parametric ReLU. In contrast to Leaky ReLU, which uses a fixed parameter (alpha) for the rectifier, parametric ReLU learns the rectifier parameters in an adaptive manner. This feature of adaptive learning improves accuracy while requiring only a little amount of additional processing effort [27].

An initial down-sampling of a high-resolution image (HR) is performed during the training process in order to create a low-resolution image (LR). The generator then makes an effort to create a super-resolution version of the LR image by up-sampling it. The discriminator then receives the image that was generated and uses it to differentiate between high-resolution and super-resolution images, which ultimately results in an adversarial loss. This loss is then backpropagated by the generator design, which enables the super-resolution process to be refined even more [27]. As shown in the following figure2.5

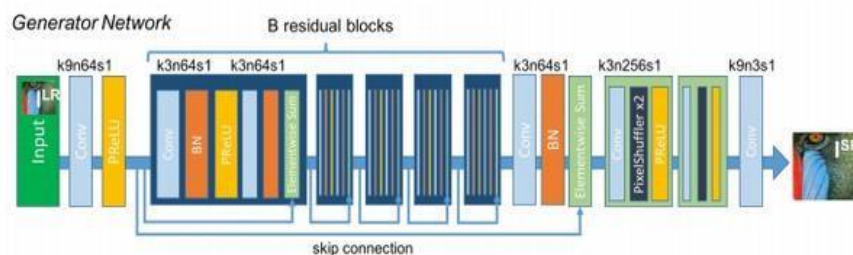


Fig.2.5. Generator architecture [27].

B. Discriminator Architecture

It is the responsibility of the discriminator to discriminate between high-resolution (HR) images that are actually captured and those that are created using super-resolution (SR). Similar to the DC-GAN (Deep Convolutional Generative Adversarial Network) architecture, which is characterized by Leaky ReLU activation functions, this article makes use of a discriminator architecture.

In the discriminator network, there are eight convolutional layers that utilize 3×3 filter kernels. The number of kernels increases by a factor of two, from 64 to 512 with each additional layer.

Every time the number of feature maps is increased by a factor of two, the picture resolution is down sampled using wavelet convolutions. For the purpose of gradually collecting and consolidating visual features, this down-sampling procedure is quite beneficial. Next, the 512 feature maps that were generated are subjected to two intensive layers of processing. With the purpose of introducing non-linearity and improving the network's ability to discriminate, leaky ReLU activation functions are implemented between these layers.

Last but not least, in order to acquire a probability score for sample categorization, a sigmoid activation function is utilized at the output layer. That the input image is a genuine HR image rather than a produced SR image is represented by this probability score, which indicates the likelihood that the image is the former [27].

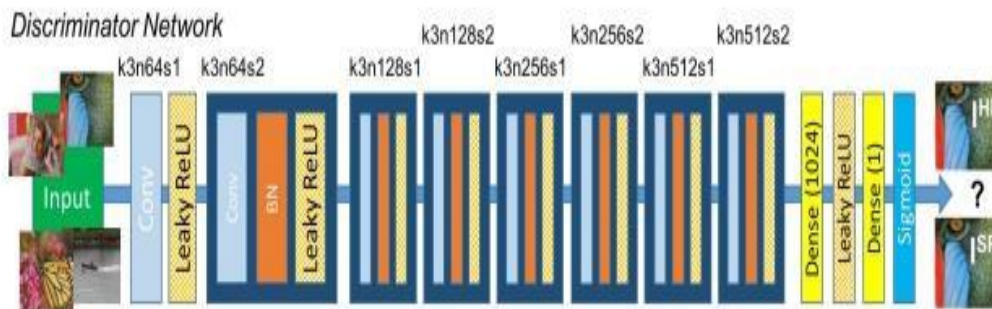


Fig.2.6. Discriminator Architecture [27]

II.4.3 Enhanced Super-Resolution Generative Adversarial Network

A cutting-edge deep convolutional neural network architecture that was built specifically for the challenging task of picture super-resolution is referred to as ESRGAN, which is an abbreviation for Enhanced Super-Resolution Generative Adversarial Network. Xintao Wang and his colleagues in the field of adversarial networks proposed this design in their foundational publication titled "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks"[25]. This architecture represents a significant leap in the field. ESRGAN adds residual-in-residual blocks in order to further improve its performance. This is accomplished by building upon the foundation of SRResNet [26], which is a well-established super-resolution network.

Through the utilization of these specialized blocks, the network is able to record intricate features during the process of upscaling, which ultimately results in a significant improvement in the output quality. ESRGAN makes use of a wide variety of loss functions, such as context, perceptual, and adversarial losses, to guarantee that the images that are generated are spatially consistent, that significant perceptual elements are preserved, and that they closely match genuine high-resolution images. As a consequence of this, ESRGAN has shown that it is superior to other approaches, such as SRGAN, in terms of performance, notably in terms of sharpness and the preservation of details.

ESRGAN has shown practical utility in real-world picture restoration tasks, such as the removal of JPEG compression artifacts, demonstrating its versatility and robustness in tackling a variety of image improvement issues. Its efficacy extends beyond academic benchmarks, as seen by the fact that it has found practical application in these tasks. As shown in the following figure2.7

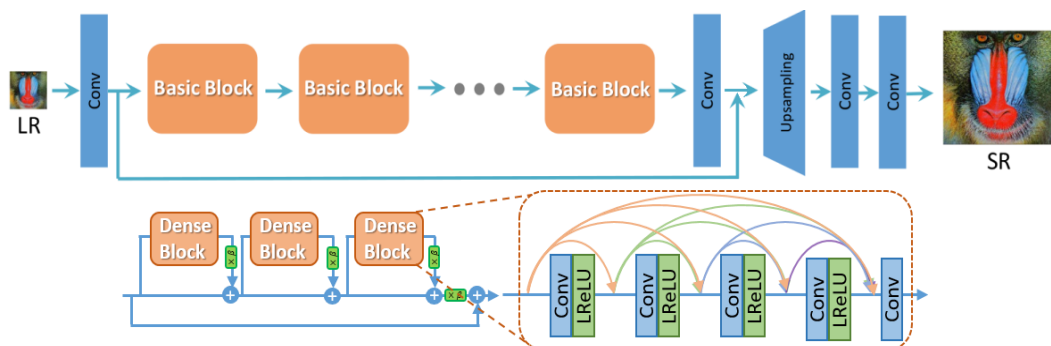


Fig.2.7. ESRGAN Architecture [28]

II.4.4 Real-ESRGAN

The components that make up a real-ESRGAN are a generator and a discriminator network, also known as a GAN [27]. A low-quality image is used as input by the generator network, which then produces a high-quality image. Meanwhile, the discriminator network makes an effort to distinguish between the fake images generated by the generator and the actual photographs.

In its most basic form, the discriminator that is being utilized is a U-net [28] that integrates skip connections. Additionally, spectral normalization is utilized in order to improve stability while training. As an alternative to employing a global realness score, the U-net makes predictions on the authenticity of each individual pixel in the input image. Pixel-level discrimination is improved as a result of this, which in turn improves the suppression of artifacts and the improvement of local detail.

II.5 Metric evaluation

elucidate metrics and their importance in evaluating models and algorithms.

II.5.1 Peak Signal-to-Noise Ratio (PSNR)

Peak Signal-to-The ratio, expressed in decibels, between the reference signal and the distortion signal in a picture is called the noise ratio [26]. The distorted image is more similar to the original the higher the PSNR. Stated differently, it can be expressed as the reciprocal of the MSE decibel scale.

Thus, we may state that given a greater image quality, the PSNR of the image will approach infinity if the MSE approaches 0. Tests have revealed that this isn't always the case, despite the expectation that a higher PSNR number would translate into a higher quality image.

Let us assume that $X = \{x_i | i = 1, 2 \dots N\}$ and $Y = \{y_i | i = 1, 2 \dots N\}$ are two infinite length, discrete signals (this discrete signal is considered as a visual signal), where N is the number of pixels in digital image and x_i and y_i are the values of i th pixel of the digital image X and digital image Y respectively. Mathematically, the PSNR for the full reference Image quality metrics is given by:

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right) \quad (2.1)$$

Where, MPP is Maximum Possible Pixel in an image, i.e. if the image of 8 bits, then the MPP= 255 pixels. MSE (X, Y) is the Mean Square error of the image X and Image Y.

II.5.2 Structural Similarity Index (SSIM)

The average of the SSIM values across the image (also called mean SSIM or MSSIM) gives the final quality measure. The key idea behind the SSIM index is to acknowledge the fact that natural images are highly structured, and that the measure of structural correlation (between the reference and the distorted image) is very important in deciding the overall visual quality. and This statistic measures the structural similarity between the reference and distorted image. Given that it takes into account the brightness, contrast, and structural elements of the image, it is believed to be a more accurate indication of image quality than MSE [26].

$$SSIM = \frac{(2\mu_x\mu_y + c_1) (2\delta_{xy} + c_2)}{(\mu^2_x + \mu^2_y + c_1) (\delta^2_x + \delta^2_y + c_2)} \quad (2.2)$$

With :

- μ_x the pixel sample mean of x
- μ_y the pixel sample mean of y
- δ^2_x the variance of x
- δ^2_y the variance of y
- δ_{xy} the covariance of x and y
- C_1, C_2 : Constant Correlation coefficient

II.6 Conclusion

The sources suggest that the conclusion of Super-Resolution technology underscores the notable progress achieved in image enhancement via deep learning methods such as Generative Adversarial Networks (GANs). These technologies have significantly transformed the procedure of converting low-resolution images to high-resolution, resulting in visual outputs that are more distinct and intricate. Super-Resolution has demonstrated its indispensability across diverse domains, including scientific research, medical imaging, and satellite remote sensing, by providing images of exceptional quality that were hitherto unattainable through conventional means. Ongoing research is dedicated to enhancing the quality of reconstructed images and decreasing the reliance on high-resolution ground truth images for neural network training, which bodes well for the future of Super-Resolution.

CHAPTER III

Results and Discussion

III.1 Introduction

In this chapter, we primarily focus on providing all of the simulation findings we were able to obtain for this study. A variety of super-resolution (SR) techniques, including bicubic, bilinear, and nearest neighbour conventional SR techniques, were employed in our study. SRGAN, ESRGAN, and Real-ESRGAN were also examined. In addition to evaluating the visual quality of the recovered images, the quantitative comparative study compares the outcomes using PSNR and SSIM, two commonly used measures. Note that these two metrics are computed using each gathered test image. The mean PSNR and SSIM values are computed at the end of each simulation experiment. To enable an unbiased comparison, we use the same datasets for each SR technique that is being considered.

III.2 Used material

On a HP PC with an Intel® Core™ i5 processor running Windows 10 Professional, the code sources were generated and run in Python 3.9.13 using Spyder Navigator in the Anaconda environment. Python primary programming package was PyTorch, was made especially for deep learning applications. The main code sources were executed using MATLAB R2022a and Google Colab. The utilization of GPU online in Google Colab allows for the processing of large datasets.

III.2.1 Used datasets

The developed SR algorithms are tested on the datasets used in the recent literature; namely: **Set5** and **Set14** as summarized in Table 3.1

Table 3.1: datasets for testing and training

| Dataset | Quantity (number of images) | Format | Used in |
|----------------|------------------------------------|---------------|----------------|
| Set5[29] | 5 | png | Testing |
| Set14[30] | 14 | png | Testing |
| Div2k [31] | 800 | png | Training |

- A. **Dataset5:** The five images in the Set5 dataset—"baby," "bird," "butterfly," "head," and "woman"—are frequently used to assess how well Image Super-Resolution models perform [29].
- B. **Dataset14:** The 14 images that make up the Set14 dataset are frequently used to assess how well Image Super-Resolution models function [30].
- C. **Div2K:** A well-known single-image super-resolution dataset called DIV2K has 1,000 photos in total, divided into 800 training, 100 validation, and 100 testing images. The images depict a variety of situations. To promote research on image super-resolution with more realistic deterioration, it was gathered for the NTIRE2017 and NTIRE2018 Super-Resolution Challenges. This collection includes many sorts of degradations in low resolution photos. When creating low resolution photos for various tasks, various degradations are taken into account in addition to the conventional bicubic down sampling. NTIRE 2017's Track 2 features low-resolution photos that haven't been downsampled by x4. NTIRE 2018's tracks 2 and 4 correspond to realistically harsh $\times 4$ moderate and realistically harsh $\times 4$ wild circumstances, respectively. When realistically mild x4 settings are used, low-resolution photos suffer from motion blur, pixel shifting, and Poisson noise. Degradations in a true wild x4 scene are expanded to vary in intensity between images [31].

III.2.2 Original Images

In our study, the original images are considered as High resolution (HR). In Figure 3.1, the relevant original photos are displayed. Numerous simulation studies have been carried out. However, we can only display a small portion of the results owing to space constraints.

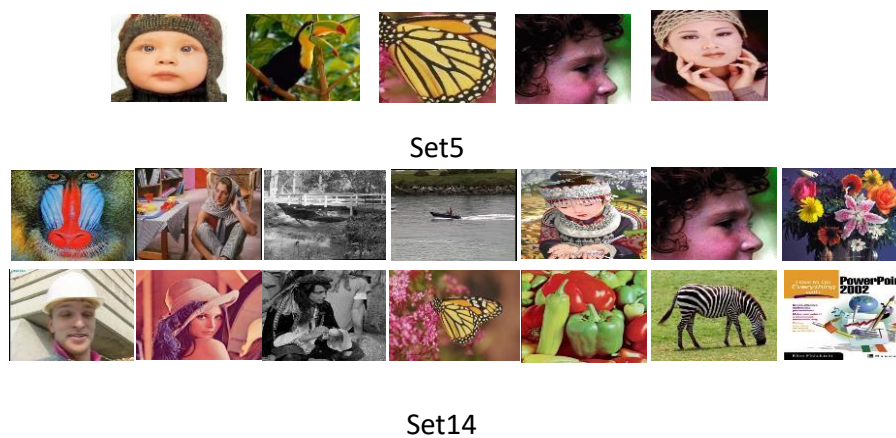


Fig.3.1. Original HR images of the used datasets

III.3 The classical methods of super resolution

III.3.1 Results of classical methods

The obtained PSNR and SSIM are displayed in Table 3.2. The test photos are from data set5 and data set14.

Tables 3.2: PSNR & SSIM of the classical method

| Data | Scale | The classical method | | | | | |
|-------|-------|------------------------|--------|------------------|--------|-----------------------|--------|
| | | Bilinear Interpolation | | Nearest Neighbor | | Bicubic Interpolation | |
| | | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| Set5 | 2 | 30.342 | 0.8028 | 29.214 | 0.6609 | 34.502 | 0.9191 |
| | 4 | 25.695 | 0.6088 | 24.066 | 0.6088 | 32.078 | 0.7952 |
| Set14 | 2 | 28.974 | 0.7533 | 27.736 | 0.6303 | 32.704 | 0.8682 |
| | 4 | 24.530 | 0.5878 | 22.462 | 0.5918 | 31.044 | 0.6047 |

Through the results obtained after calculation in Table 3.2, which represent the PSNR and SSIM values from the classical methods through which dataset5 and dataset14 were tested for scales 2 and 4, we notice the presence of high values in dataset5 and low values in dataset14 and from these results obtained, it is correct to say after comparison that Data Set5 contains complex images, which contain many details or complex structures that require high accuracy to distinguish between them.

The results were as shown in Figure 3.2-3.3:

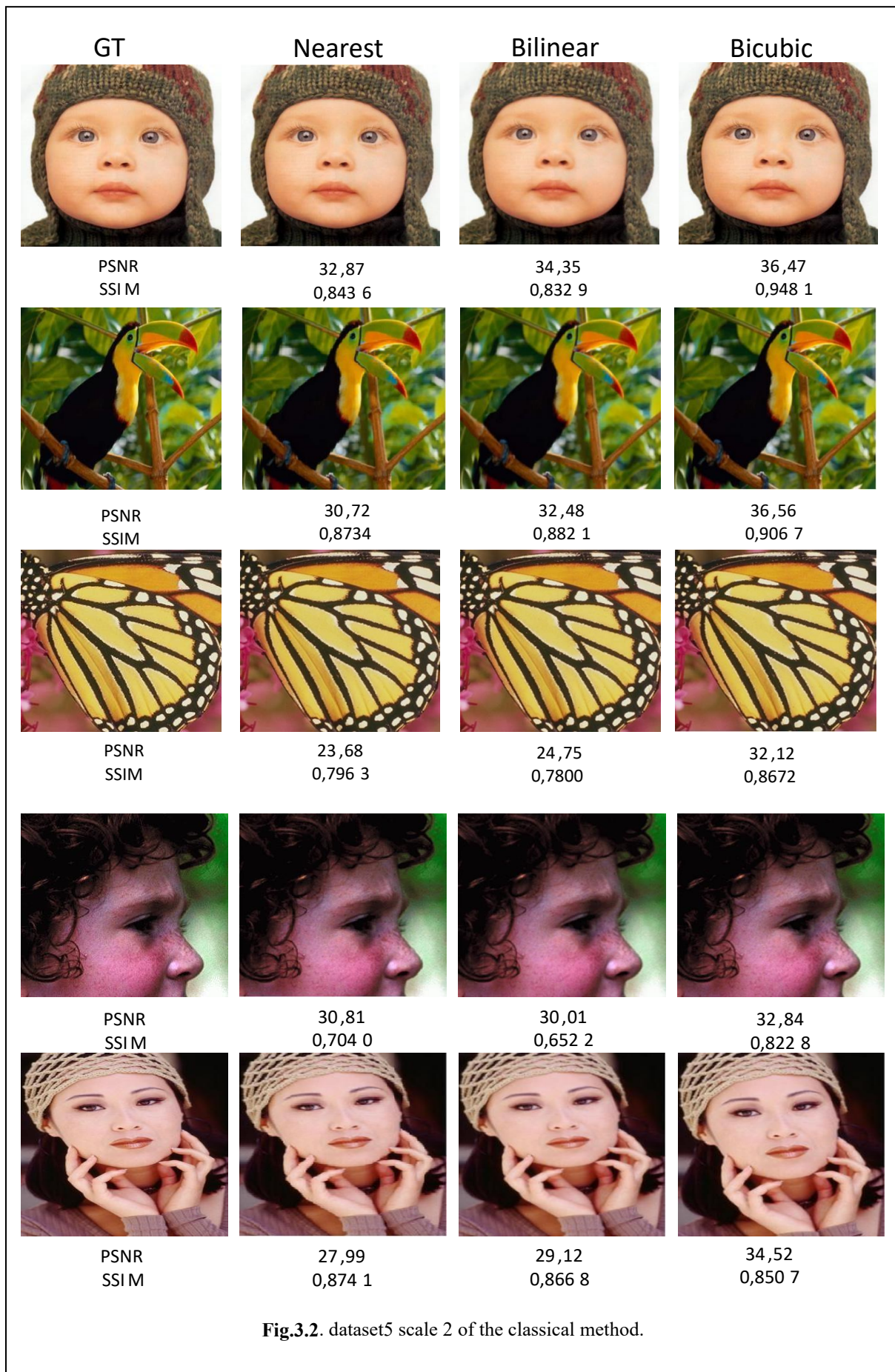
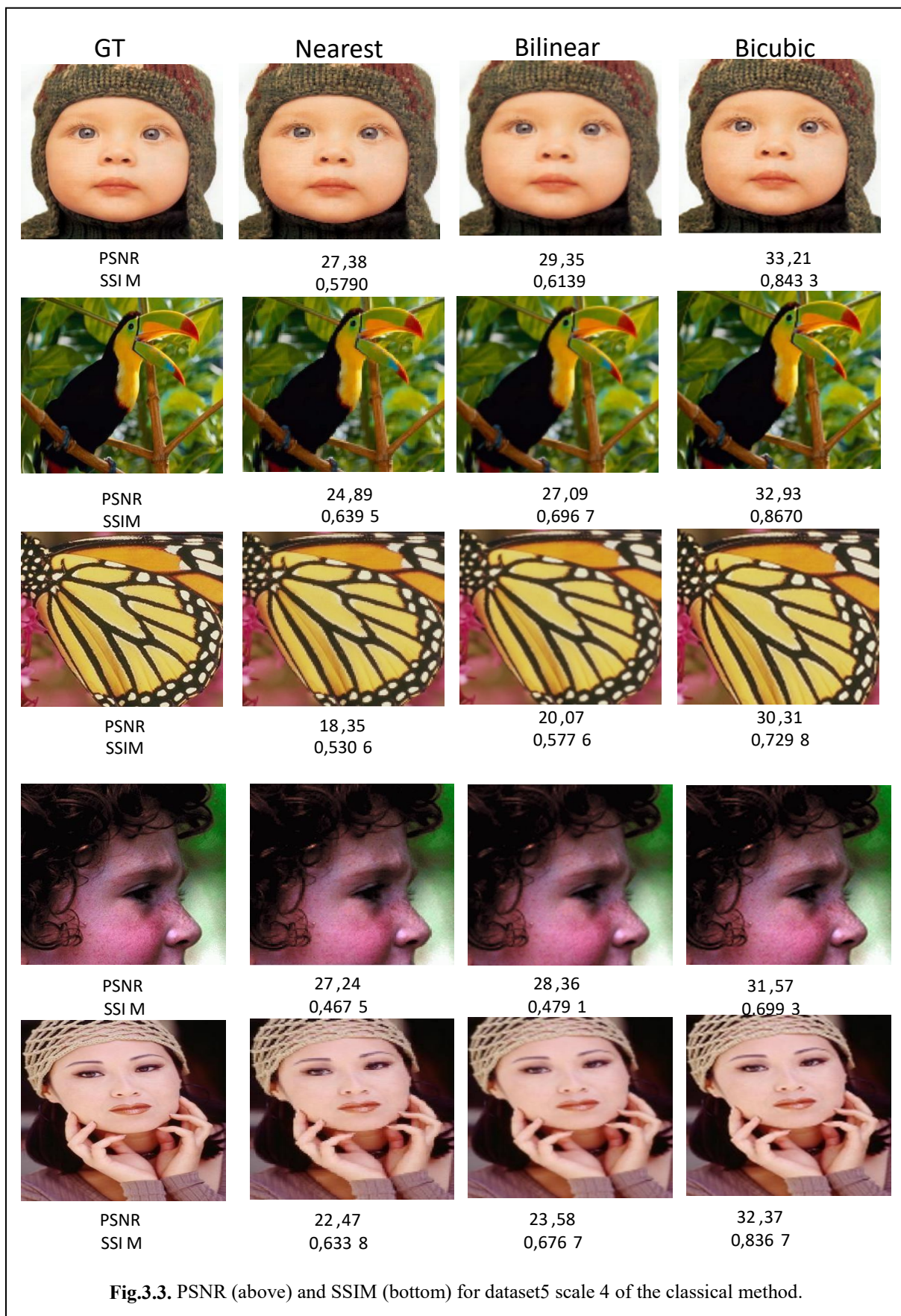


Fig.3.2. dataset5 scale 2 of the classical method.



III.3.2 Comparison of classical method

Nearest neighbor, bilinear, and bicubic interpolation are three common methods used for resizing images. Nearest neighbor interpolation, the simplest among them, duplicates each pixel from the original image to fill in the new, larger image. While it's fast, it tends to produce lower-quality results, especially for significant upscaling. Bilinear interpolation, on the other hand, calculates the weighted average of the four nearest pixels surrounding each target point, resulting in smoother transitions between pixels and better overall quality compared to nearest neighbor interpolation. Bicubic interpolation takes this a step further by considering sixteen surrounding pixels and applying a more sophisticated weighted average calculation. This method yields even smoother results and is particularly effective for preserving fine details and sharp edges, albeit at the expense of increased computational complexity. Overall, while nearest neighbor is the fastest but least accurate, bicubic interpolation stands out for its ability to produce high-quality results, making it a preferred choice for applications where image quality is paramount.

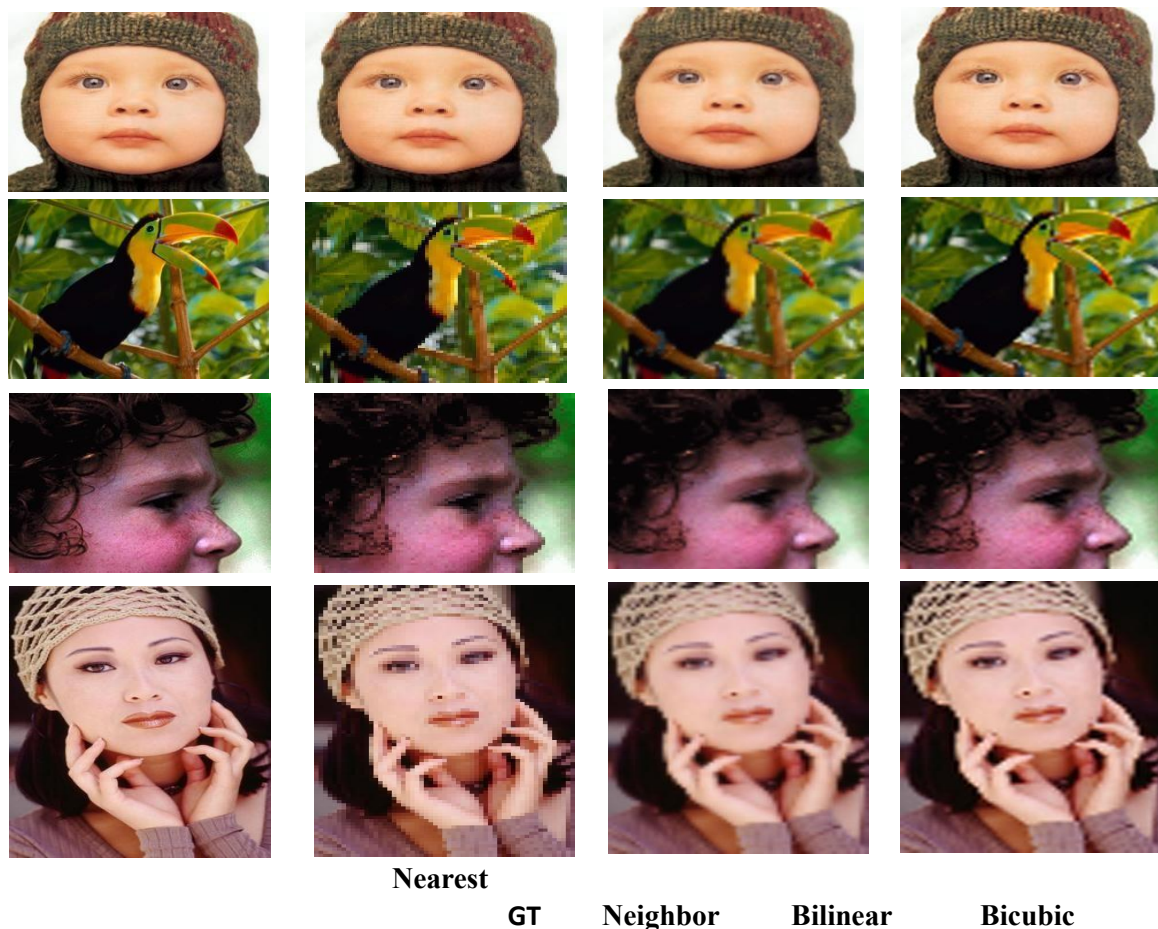


Fig.3.4. Results of the classical method.

III.4 Implementation of deep learning super resolution

III.4.1 Super resolution of generative adversarial networks (SRGAN)

III.4.1.1 Explanation of training conditions

This explanation aims to clarify the set of conditions used in the training program for the neural network model, as shown in Table 3.3

Table 3.3: parameters and hyper-parameters of training the SRGAN model.

| Number of epochs | Batch size | Adam optimization | | Learning rate | Decay epoch | Number of CPU | Hrwidth and Hrheight | Sample interval |
|------------------|------------|-------------------|-------|---------------|-------------|---------------|----------------------|-----------------|
| | | b1 | b2 | | | | | |
| 200 | 4 | 0.5 | 0.999 | 0.0001 | 100 | 3 | 256 | 100 |

These conditions are used to control several different aspects of the training process. We select settings such as start of epoch 0 and total number of epochs 200 for training.

The optimization parameters of the adam optimizer are also configured, such as learning rate 0.0001, b1=0.5, and b2=0.999. Other settings include batch size 4, number of CPU threads to process data 3, time intervals for saving image samples and sample checkpoints. These parameters collectively control different aspects of the training process and can be modified to improve model performance.

III.4.1.2 Training process

Training a super-resolution generative adversarial network (SRGAN) is a complex process that involves several key steps. Initially, a dataset of HR images is collected and preprocessed, which typically involves resizing and normalization. The model architecture consists of a generator network, responsible for upscaling LR images, and a discriminator network, which distinguishes between real and generated high-resolution images. Loss functions play a crucial role in training, with adversarial loss driving the generator to produce realistic images and perceptual loss ensuring visual similarity to real images. Training involves simultaneous optimization of the generator and discriminator through adversarial learning, often using optimization techniques like mini-batch stochastic gradient descent (SGD). Evaluation metrics

such as PSNR and SSIM are used to assess the model's performance, along with qualitative assessment on a separate validation set. Fine-tuning and hyperparameter tuning are conducted to optimize the model further, followed by deployment for real-world SR tasks. Throughout the process, careful monitoring and adjustment are essential for successful training.

III.4.1.3 Algorithm of SRGAN

This algorithm trains a GAN for image super-resolution using adversarial and content losses. Additionally, it saves intermediate results and logs the training progress.

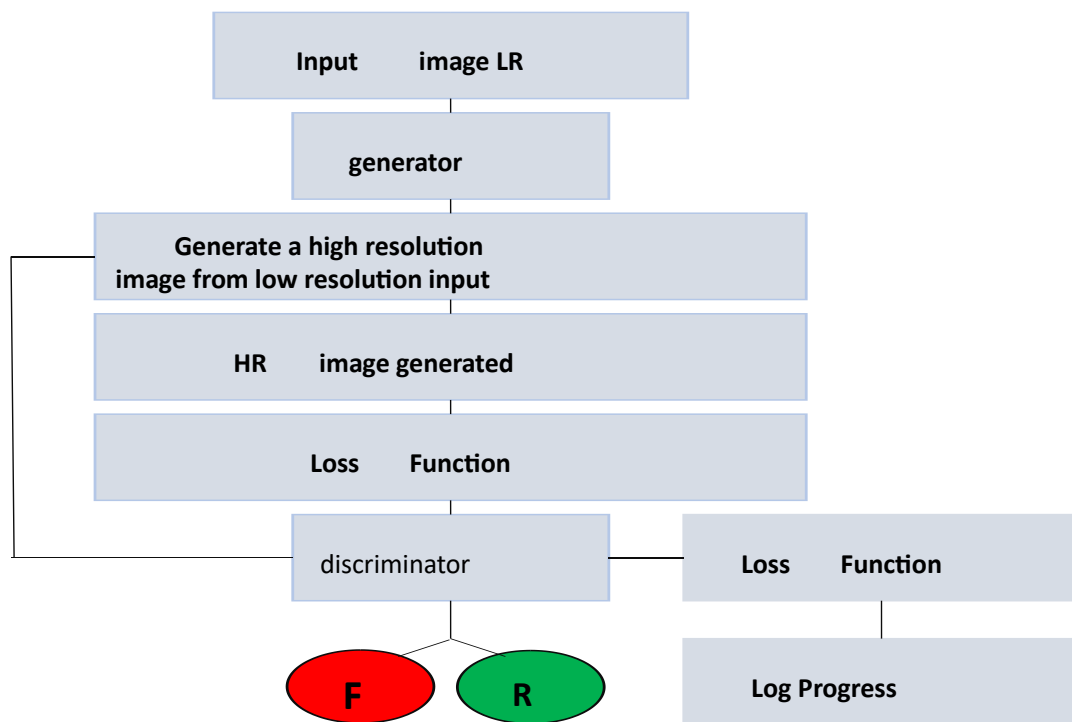


Fig.3.5. Schemes of SRGAN training

III.4.2 Results of SRGAN

III.4.2.1 Learning evolution

We notice in Figure 3.5 the learning stage for both the generator and the discriminator, where this figure contains four images, each image having a meaning. Image (a) represents for us the beginning of learning because a low-resolution image was entered. The result was unclear or distorted and had a blur that we can only see with the galactic eye. It is correct to say that He has not yet learned, and the second picture (b), we see that it has some blurring, but compared to the first picture, he is on the path to learning, and picture (c) confirms that he has begun to learn. As for the last picture, here we can be confident that learning has been completed, because

the picture that was produced is a high-resolution picture. The figure shows the validity of the statement and analysis.

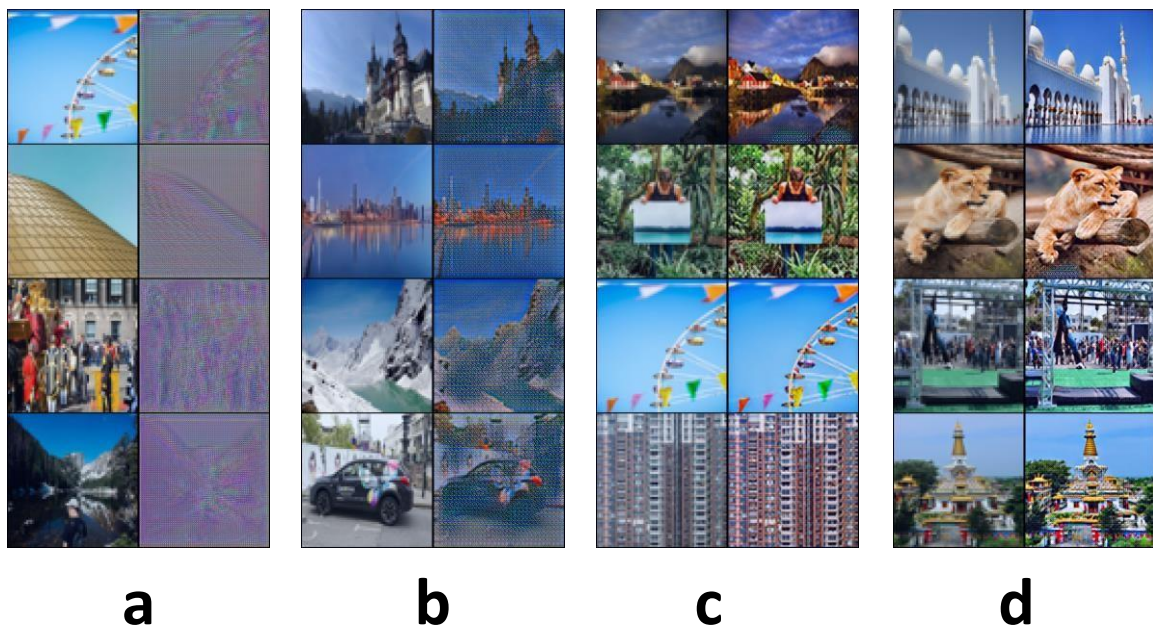


Fig.3.6. evolution of SRGAN training, (a) after 1 epoch, (b) after 10 epochs, (c) after 100 epochs, (d) 200 epochs

After the end of training, we notice the appearance of curves as shown in Figure 3.6:

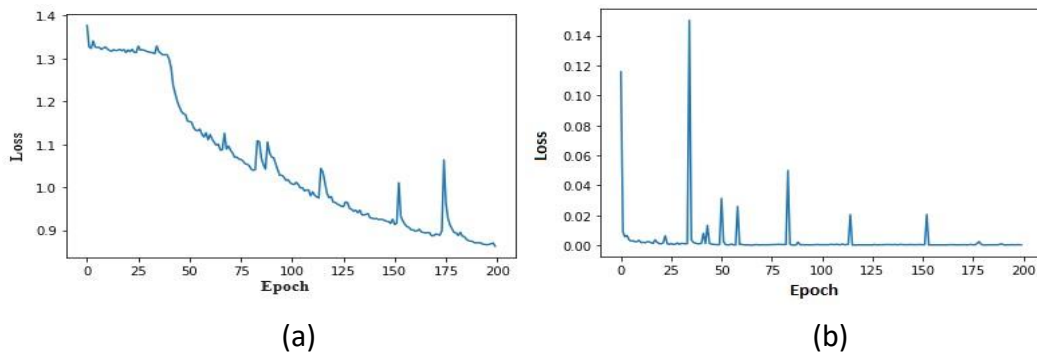


Fig.3.7. the loss of training for 200 epochs:(a) generative Loss, (b) discriminator Loss

It is shown in curve A that as the number epoch increases, the value of the generating loss decreases by approaching approximately 0. This is evidence that the generator is trying to learn and practice so that it can deceive the discriminator. This optimization is done to achieve the goal of fooling the discriminator. As for the discrimination loss, it tries to improve in order to detect

the generator, and the loss value fluctuates between increasing and decreasing until it reaches 0.5, which is the lowest value that discrimination can reach. When he reaches this result, he is unable to distinguish between the correct image and the incorrect image, and it is correct to say that it is a game of deception between the criminal. And the criminal. policeman. One wants to hide the evidence and the other wants to reveal it. Both the born and the privileged want to develop by not giving in to revelation and deception.

III.4.3 Results PSNR and SSIM of deep learning super resolution

The PSNR and SSIM obtained are shown in Tables 3.4-3.5-3.6 For SRGAN, ESRGAN, and Real-ESRGAN and Bicubic, the test images are from Dataset 5 and Dataset 14.

Tables 3.4: PSNR & SSIM of SRGAN and Bicubic Interpolation

| Dataset | Scale | SRGAN | | Bicubic Interpolation | |
|---------|-------|--------|--------|-----------------------|--------|
| | | PSNR | SSIM | PSNR | SSIM |
| Set5 | 4 | 33.920 | 0.8942 | 32.078 | 0.7952 |
| Set14 | 4 | 30.554 | 0.8409 | 31.044 | 0.6047 |

SRGAN tends to outperform bicubic interpolation in terms of both PSNR and SSIM. SRGAN generates more visually pleasing HR images by capturing finer details and textures, whereas bicubic interpolation often produces blurry results, especially when upscaling factors are high.

Table 3.5: PSNR & SSIM of ESRGAN and Bicubic Interpolation

| Dataset | Scale | ESRGAN | | Bicubic Interpolation | |
|---------|-------|--------|--------|-----------------------|--------|
| | | PSNR | SSIM | PSNR | SSIM |
| Set5 | 4 | 31.879 | 0.8656 | 32.078 | 0.7952 |
| Set14 | 4 | 23.298 | 0.6438 | 31.044 | 0.6047 |

ESRGAN produces sharper and more realistic HR images compared to the blurry results often obtained with bicubic interpolation.

Table 3.6: PSNR & SSIM of Real-ESRGAN and Bicubic Interpolation

| Dataset | Scale | Real-ESRGAN | | Bicubic Interpolation | |
|---------|-------|-------------|--------|-----------------------|--------|
| | | PSNR | SSIM | PSNR | SSIM |
| Set5 | 2 | 28.201 | 0.8973 | 34.502 | 0.9191 |
| | 4 | 23.730 | 0.7756 | 32.078 | 0.7952 |
| Set14 | 2 | 26.460 | 0.8947 | 32.704 | 0.8682 |
| | 4 | 24.087 | 0.7785 | 31.044 | 0.6047 |

Real ESRGAN typically demonstrates superior performance. Real ESRGAN generates high-resolution images with finer details and textures, resulting in higher PSNR and SSIM scores compared to bicubic interpolation.

We tested SRGAN and ESRGAN, Real-ESGAN. The sample data used was Dataset 5 and Dataset 14, and the results were as shown in Figure 3.7:



Fig.3.8. PSNR and SSIM for dataset5 scale 4 of deep learning super resolution.

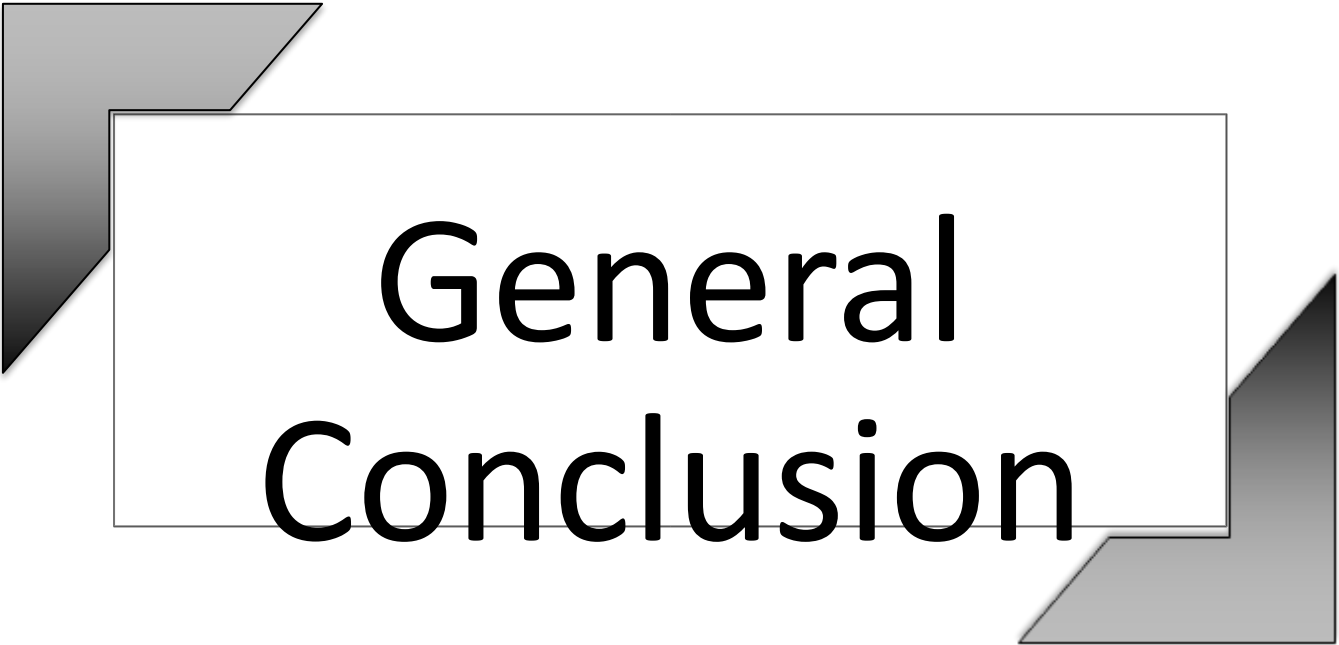
III.5 Comparison for Deep learning for super-resolution and Bicubic

We notice from the results obtained in each of Tables 3.4, 3.5 and 3.6 which carry the values of PSNR and SSIM that this value, according to mathematical observations, was high for bicubic, SRGAN, and ESRGAN, while its decrease was observed in Real-ESRGAN. Figure 3.7 reflects all of these. The results are because the images are blurry, except for Real ESRGAN, which is a very precise image and free of blur in general. It can be said that it provides amazing results in improving image quality, although the measurements of this quality show a decline, but the difference is clear when seeing the images with the eye.

III.6 Conclusion

Three deep learning-based super-resolution (SR) image algorithms—SRGAN, ESRGAN, and Real-ESRGAN—were investigated in this work. Additionally, we used bicubic, bilinear, and nearest-neighbor interpolation—traditional interpolation algorithms. We have made advantage of Set5 and Set14, two well-known datasets in the SR issue space. In addition to the visual quality of the reconstructed SR pictures, we evaluated two metrics: PSNR and SSIM.

We initially converted the original photos into low-resolution (LR) images before carrying out each experiment. The photos produced by this stage were X2 or X4 in size. It is evident from the results that the Real-ESRGAN method performs better in terms of computed metrics and visual quality than other SR techniques. It should be mentioned, nevertheless, that SR deep learning methods enable processing massive amounts of data a large number of photos in a brief period of time, made possible by network architecture and GPU usage. Ultimately, we were able to accomplish our objectives.



General Conclusion

General conclusion

The thesis titled "Application of generative adversarial networks for image super resolution" focuses on comparing two approaches for enhancing the resolution of images: conventional interpolation algorithms and deep learning-based interpolation algorithms. After conducting a comprehensive study, several key conclusions can be drawn from this thesis.

Firstly, the thesis highlights that conventional interpolation algorithms, such as bicubic interpolation, have been widely used for image super resolution tasks. These algorithms are based on mathematical principles and are relatively straightforward to implement. However, the results obtained from conventional interpolation algorithms may lack fine details and fail to produce high-quality super-resolved images.

Secondly, the thesis emphasizes the emergence of deep learning-based interpolation algorithms, which utilize neural networks to learn complex mappings between low-resolution and high-resolution image pairs. These algorithms have shown remarkable performance in generating visually appealing and highly detailed super-resolved images. They can capture intricate patterns and textures that conventional algorithms struggle to reproduce.

Furthermore, the thesis concludes that deep learning-based interpolation algorithms generally outperform conventional interpolation algorithms in terms of perceptual quality and objective metrics. The ability of deep learning models to learn from large-scale datasets enables them to capture intricate image features, leading to superior super-resolution results.

However, the thesis also acknowledges that deep learning-based approaches may have some limitations. They often require significant computational resources and extensive training data to achieve optimal performance. Additionally, the selection of appropriate network architectures, loss functions, and training strategies significantly affects the final results. In conclusion, the thesis provides valuable insights into the comparative study of conventional and deep learning interpolation image super-resolution algorithms. It highlights the advantages of deep learning approaches in generating high-quality super-resolved images, while acknowledging the challenges and considerations involved in their implementation.



References

References

References

- [1] Sandhya Pati/2019/Artificial Intelligence Course in Bangalore - AI Training & Certification <https://morioh.com/p/7ccb1d0d114d>
 - [2] Copeland, B.J.. "Artificial intelligence". Encyclopedia Britannica, 18 Mar. 2022, <https://www.britannica.com/technology/artificial-intelligence>. Accessed 27 April 2022.
 - [3] I. Dabbura . Coding Neural Network — Forward Propagation and Back-propagation. Published in Towards Data Science . 2018.
 - [4] Derek Kwok/May 25, 2018/Machine Learning for My Grandma.
 - [5] I. Goodfellow , Y. Bengio , A. Courville . Deep Learning. MIT Press. 2016.
 - [6] <https://www.analyticsvidhya.com/blog/2017/10/fundamentals-deep-learning>.
 - [7] Christopher M Bishop. Pattern recognition and machine learning. springer, 2006.
 - [8] Simon S Haykin, Simon S Haykin, Simon S Haykin, Kanada Elektroingenieur, and Simon S Haykin. "Neural networks and learning machines", volume 3. Pearson education Upper Saddle River, 2009. [9] Gilbert strang <https://http://math.mit.edu/ gs/learningfromdata/>.
 - [10] Tianfeng Chai and Roland R Draxler. "Root mean square error (rmse) or mean absolute error (mae)?—arguments against avoiding rmse in the literature Geoscientific model development", 7(3):1247–1250, 2014.
 - [11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.
 - [12] Stefano Ermon and Aditya Grover. <https://deepgenerativemodels.github.io>.
 - [13] Z. Wang, J. Chen and S. C. H. Hoi, "Deep Learning for Image Super-Resolution: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3365-3387, 1 Oct. 2021, doi: 10.1109/TPAMI.2020.2982166.
 - [14] D.H.HubelandT.N.Wiesel, "Receptivefieldsofsingleneuronessinthecat'sstriate cortex," The Journal of physiology, vol.148, no.3, p.574, 1959.
 - [15] Wu, L., Qi, M., Jian, M., & Zhang, H. (2020). "Visual sentiment analysis by combining global and local information. Neural Processing", Letters, 51(3), 2063-2075.
 - [16] Girin, Laurent, et al. "Dynamical variational autoencoders: A comprehensive review." arXiv preprint arXiv:2008.12595 (2020).
 - [17] You, Sungmin, et al. « Semi-Supervised Automatic Seizure Detection Using Personalized Anomaly Detecting Variational Autoencoder with behind-the-Ear EEG ». Computer Methods and Programs in Biomedicine, vol. 213, janvier 2022, p. 106542. DOI.org (Crossref), <https://doi.org/10.1016/j.cmpb.2021.106542>.
-

References

- [18] « NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study ». 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017, p. 1122-31. DOI.org (Crossref), <https://doi.org/10.1109/CVPRW.2017.150>.
- [19] Agustsson, Eirikur, et Radu Timofte. « NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study ». 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017, p. 1122-31. DOI.org (Crossref), <https://doi.org/10.1109/CVPRW.2017.150>.
- [20] K. Nakanishi, S. Maeda, T. Miyato, and D. Okanohara, "Neural Multi-scale Image Compression," 2018: <https://doi.org/10.48550/arXiv.1805.06386>.
- [21] P. S. Parsania and P. V. Virparia, "A Review: Image Interpolation Techniques for Image Scaling", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 12, pp. 7409-7413, Dec. 2014.
- [22] Olivier Rukundo and Hanqiang Cao, "Nearest Neighbor Value Interpolation" International Journal of Advanced Computer Science and Applications(IJACSA), 3(4), 2012. <http://dx.doi.org/10.14569/IJACSA.2012.030405>
- [23] Ward, Chris M., et al. "Image quality assessment for determining efficacy and limitations of SuperResolution Convolutional Neural Network (SRCNN)." Applications of Digital Image Processing XL. Vol. 10396. SPIE, 2017.
- [24] Xiong, Yingfei, et al. "Improved SRGAN for remote sensing image super-resolution across locations and sensors." Remote Sensing 12.8 (2020): 1263.
- [25] Wang, Xintao, et al. "Esrgan: Enhanced super-resolution generative adversarial networks." Proceedings of the European conference on computer vision (ECCV) workshops. 2018.
- [26] Ullah, Safi, and Seong-Ho Song. "SRResNet Performance Enhancement Using Patch Inputs and Partial Convolution-Based Padding." Computers, Materials & Continua 74.2 (2023).
- [27] Wang, Ching-Hsiang. "Using Super-Resolution Imaging for Recognition of Low-Resolution Blurred License Plates: A Comparative Study of Real-ESRGAN, A-ESRGAN, and StarSRGAN." arXiv preprint arXiv:2403.15466 (2024).
- [28] Jain, Pankaj K., et al. "Attention-based UNet Deep Learning model for Plaque segmentation in carotid ultrasound for stroke risk stratification: An artificial Intelligence paradigm." Journal of Cardiovascular Development and Disease 9.10 (2022): 326.
- [29] Bevilacqua, Marco, et al. « Low-Complexity Single-Image Super-Resolution Based on Nonnegative Neighbor Embedding ». Proceedings of the British Machine Vision Conference 2012, British Machine Vision Association, 2012, p. 135.1-135.10. DOI.org (Crossref), <https://doi.org/10.5244/C.26.135>.
-

References

- [30] Bevilacqua, Marco, et al. « Low-Complexity Single-Image Super-Resolution Based on Nonnegative Neighbor Embedding ». Proceedings of the British Machine Vision Conference 2012, British Machine Vision Association, 2012, p. 135.1-135.10. DOI.org (Crossref), <https://doi.org/10.5244/C.26.135>.
- [31] Agustsson, Eirikur, et Radu Timofte. « NTIRE 2017 Challenge on Single Image Super-Resolution: Dataset and Study ». 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017, p. 1122-31. DOI.org (Crossref), <https://doi.org/10.1109/CVPRW.2017.150>.
-