الجمهورية الجزائرية الديمقراطية الشعبية

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التعليم العالي والبحث العلمي

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

جامعة سعيدة – د. الطاهر مولاي –

University of Saida - Dr. MOULAY TAHAR

Faculty of Technology

A Dissertation Submitted to the Department of Telecommunications in Partial Fulfilment of the Requirements for Degree of Master of Systems of Telecommunications

Presented by:

Miss. RAHMANI Lamia .

Miss. HAMOU Fatima.

Title

Visual content compression using Deep Learning techniques

Defended on June, ...2024 in front of the jury composed of:

Mr. CHETIOUI Mohamed	MCA	President
Mr. MOKADEM Djelloul	MCB	Supervisor
Mr.MANSOURI Boualem	MCA	Examiner

2023 / 2024

Dedication

To My Beloved Family and Friends,

I want to express my deepest gratitude to each one of you. Your unwavering support, encouragement, and love have been my guiding light throughout this endeavor.

*To My **grandparents**: Your sacrifices, late-night conversations, and belief in my abilities have fueled my determination. Thank you for being my pillars of strength.*

*My dear father Professor **Reda Mohamed,** you who have guided me, through winds and tides, your wisdom and strength have inspired me, In this journey of learning and freedom.*

*The greatest **Mom**, your unconditional love, your endless patience, have shaped in me a resilient spirit, with every challenge, every trial, I remember your soothing words.*

*To my dear brother**Nadir**, who always believed in me even when I doubted myself. Your trust gives me the strength to surpass my limits.*

*To My Siblings **Rana, Rajaa&Ayah**: Your laughter, shared secrets, and occasional teasing have made this journey memorable. You've been my confidantes and cheerleaders.*

*To My Friends specially my teammate**Lamia**: From late-night study sessions to celebrating small victories, you've been my companions on this rollercoaster ride. Your camaraderie has made the challenges worthwhile.*

To My Professors and Mentors: Your wisdom, patience, and guidance have shaped my intellect. I am indebted to you for imparting knowledge and nurturing my growth.

This project bears a piece of each of you—the late nights, the coffee-fueled brainstorming, and the collective belief that we can achieve greatness. As I step into the next chapter of my life, I carry your love and encouragement with me.

Thank you for being my rock, my motivation, and my joy. Here's tous—our shared memories, our dreams, and our unbreakable bonds.

With heartfelt appreciation,

HAMOU FATIMA NIHED

I dedicate this work:

As a sign of gratitude for the love, efforts and sacrifices made every day, as well as the support and understanding in permanence:

To my father ***Sidi-Mohamed****, who encouraged and supported me throughout my studies and for his moral support.*

To my mother ***Abla,*** *many thanks to her, she has always been so loving,devoted and caring towards me.*

Without you, I wouldn't be who I am today. I pray that this workallows me to bring you honour.

To the loving and caring person who I miss day and night, she is my life and breath, my sister ***Nouzha****, may God protect and bless her little family.*

To my sweetest siblings ever ***Amani, Taha.****May god grant them success and guidance in their studies*

To that one person who supported me all along and helped me throughout my bad days, ***Dounia.***

To ***My Grandparents,*** *whose unconditional love and invaluable life lessons have shaped who I am today, forever grateful for their lasting impact on my life's path.*

To all my family that I appreciate their support.

To my lovely friends; ***Ilham, Ikram, Malak****and my teammate* ***Nihed****, May God bless you and help you achieve your dreams.*

To all the people who have contributed directly or indirectly to the accomplishment of this work.

LAMIA RAHMANI

Acknowledgements

First and foremost, we would like to praise ***Allah****theAlmighty, the MostGracious, and the Most Merciful for Hisblessing given to us during our studyand in completing thisthesis. May Allah's blessing goes to His final ProphetMuhammad (peace be up on him), his family and his companions.*

We would like to express our sincere gratitude to our thesis supervisor, ***Dr. MOKADEM DJELLOUL****. His guidance, unwavering support,advicesand especially his patience with us have been invaluable throughout this master's degree final project.*

We also extend our thanks to all the professors, jury members, and individuals who, through their words, writings, advice, and reviews, will guide our reflections and graciously will meet with us to answer our questions during our research.

Thank you.

Abstract

In recent years, the quantity of visual content continues to increase day after day, which is mainly linked to the rise of social networks and video streaming platforms. Even as storage and transmission capacities improve, this growing number of transmitted images and videos requires more efficient compression methods.

Lossy image compression achieves high compression ratios by eliminating information that does not contribute to human perception of images, or that contributes as little as possible. Due to the limitations of the human visual system, such information loss may be acceptable in many scenarios, but the visual artifacts introduced become unacceptable at higher compression ratios. The project focuses on the JPEG compression method and the degradations it causes.

It is a challenging task since there are highly complex unknown correlations between the pixels, as a result, it is hard to find and recover them.

We want to find a well-compressed representation for images and, design and test networks that are able to recover it successfully in a lossless or lossy way.

we will try to useconvolutional neural networks (CNN) deep learning based to achieve the goal we have fixed

Keywords: visual content, compression, CNN, deep learning.

ملخص

في السنوات الأخيرة، استمرت كمية المحتوى المرئي في الزيادة يومًا بعد يوم، وهو ما يرتبط بشكل أساسي بظهور الشبكات الاجتماعية ومنصات بث الفيديو. وحتى مع تحسن قدرات التخزين والنقل، فإن هذا العدد المتزايد من الصور ومقاطع الفيديو المنقولة يتطلب أساليب ضغط أكثر كفاءة.

يؤدي ضغط الصور مع فقدان البيانات إلى تحقيق نسب ضغط عالية من خلال إزالة المعلومات التي لا تساهم في الإدراك البشري للصور، أو التي تساهم بأقل قدر ممكن. ونظرًا للقيود المفروضة على النظام البصري البشري، قد يكون فقدان المعلومات هذا مقبولاً في العديد من السيناريوهات، ولكن التحف البصرية المقدمة تصبح غير مقبولة عند نسب الضغط الأعلى. يركز المشروع على طريقة ضغط JPEG والتدهور الذي يسببه.

إنها مهمة صعبة نظرًا لوجود ارتباطات غير معروفة ومعقدة للغاية بين وحدات البكسل، ونتيجة لذلك، من الصعب العثور عليها واستعادتها.

نريد العثور على تمثيل مضغوط جيدًا للصور وتصميم واختبار الشبكات القادرة على استعادتها بنجاح بطريقة غير قابلة للفقدان أو الضياع.

سنحاول استخدام التعلم العميق للشبكات العصبية التلافيفية (CNN) لتحقيق الهدف الذي حددناه

الكلمات المفتاحية: المحتوى المرئي، الضغط، شبكات العصبية التلافيفية، التعلم العميق.

Résumé

Ces dernières années, la quantité de contenus visuels ne cesse d'augmenter de jour en jour, ce qui est principalement lié à l'essor des réseaux sociaux et des plateformes de streaming vidéo. Même si les capacités de stockage et de transmission s'améliorent, ce nombre croissant d'images et de vidéos transmises nécessite des méthodes de compression plus efficaces.

La compression d'images avec perte permet d'obtenir des taux de compression élevés en éliminant les informations qui ne contribuent pas à la perception humaine des images, ou qui y contribuent le moins possible. En raison des limites du système visuel humain, une telle perte d'informations peut être acceptable dans de nombreux scénarios, mais les artefacts visuels introduits deviennent inacceptables à des taux de compression plus élevés. Le projet se concentre sur la méthode de compression JPEG et les dégradations qu'elle provoque.

Il s'agit d'une tâche difficile car il existe des corrélations inconnues très complexes entre les pixels, ce qui rend difficile leur recherche et leur récupération.

Nous voulons trouver une représentation bien compressée pour les images et concevoir et tester des réseaux capables de la récupérer avec succès, sans perte ou avec perte.

Nous essaierons d'utiliser l'apprentissage profond des réseaux de neurones convolutionnels (CNN) pour atteindre l'objectif que nous nous sommes fixés

Mots clés:contenu visuel, compression, CNN, apprentissage profond.

Content list

Content list

Abbreviationslist

ANN: Artificial Neural Network

AEs: Autoencoders

AI: Artificial Intelligence

CNN: Convolutional Neural Networks

CT: ComputedTomography

CR: Compression Ratio

CV: Computer Vision

DCT: Discrete Cosine Transform

DPCM: Differential pulse-code modulation

DFT: Discrete Fourier Transform

DCT: Discrete Cosine Transform

DL: Deep Learning

DNN: Deep Neural Network

FAX: Facsimile

FLOPs: Floating-Point Operations per second

GIF: Graphical Interchange Format

GAN: Generative Adversarial Networks

GPU: Graphics Processing Unit

HEVC: High Efficiency Video Coding

IFS: Iterated Function System

ILSVRC: ImageNet Large Scale Visual Recognition Contest

JPEG: Joint Photographic Experts Group

JBIG: Joint Bi-level Image Group

LZW: Lempel–Ziv–Welch

MSE: Mean Square Error

ML: Machine Learning

MLP: Multilayer Perceptron

PET: Positron Emission Tomography

PSNR: Peak Signal to Noise Ratio

PNG: Portable Network Graphics

QoE: Quality of Experience

RGB: Red, Green & Blue

ResNet: Residual Network

RNN: Recurrent Neural Networks

SPECT: Single-Photon Emission Computed Tomography

SEO: Search Engine Optimization

SSIM: Structural Similarity Index Measure

TCP: Transmission Control Protocol

TIFF: Tagged Image File Format

UDP: User Datagram Protocol

VGG: Visual Geometry Group

VAE: Variational AutoEncoders

VQ: VectorQuantization

YUV: Luminance (Y), Blue luminance (U) and red luminance (V)

Figureslist

Tables list

General Introduction

General introduction

The growing demand for efficient and high-quality compression solutions for visual content using deep learning techniques is driving the development of visual content compression using deep learning techniques. Deep learning offers a promising paradigm for the development of advanced compression algorithms that can adapt to the complexities of visual content and deliver superior compression performance while maintaining perceptual quality.

The objectives of this project revolve around achieving a balance between compression efficiency, perceptual quality, adaptability, scalability, real-time performance, and energy efficiency, in order to cater to the diverse requirements of contemporary applications and facilitate the widespread adoption of visual content compression technologies.

Image compression is a technique used to reduce the size of digital images while preserving the visual quality to an acceptable level. It is primarily employed to optimize storage space and transmission bandwidth, making it more efficient to store, transmit, and process images.

When compression is lossless, the original image can be perfectly reconstructed from the compressed version. Removal of redundant data and exploiting patterns within the image are used to achieve this. Compression without loss is used for images where every little thing counts, like medical images, technical drawings, and text documents.

Lossy compression reduces the image's dimensions by permanently removing some of the information. Quality can be balanced against compression ratio by controlling the degree of loss. Web images, digital photography, and multimedia applications often employ lossy compression.

Image compression offers numerous benefits, including reduced storage requirements, faster transmission, bandwidth efficiency, improved loading times, enhanced user experience, cost savings, and environmental sustainability. These advantages render image compression a crucial instrument for optimizing digital workflows and augmenting the efficacy and performance of diverse applications and platforms.

This thesis establishes three chapters, in the first chapter we present the principles of data transmission in telecommunications networks and

Principles of digital image.

The second chapter is devoted to Image Compression Techniques.

In the third chapter, we present Approaches based on CNN (deep learning) for image compression, we use the Python programming language and we explain the different steps adopted.

Finally, we will end this thesis with a general conclusion which summarizes our work in its theoretical part and simulation of the results.

Chapter I.

Data transmission and digital image principles in telecommunications networks

I.1. Introduction

This chapter aims to present the principles of data transmission in telecommunications networks and also presents the principles of digital images to better understand the work carried out in this dissertation. Data transmission in telecommunication networks is a key area of telecommunications, involving the communication of data from one point to another or from one time to another. Coding techniques are used to ensure efficient data transmission, particularly in digital communications, where data rates can vary in either direction [1].

Telecommunications networks are generally electromagnetic and are concerned with information, rather than its physical medium. The main elements of a telecommunications network include the channel, transmitter and receiver [2]. The channel is the physical medium of communication, such as a cable, an optical fiber, a microwave line or infrared waves. The transmitter provides a signal matched to the channel, while the receiver receives and processes the signal.

Computer and information networks can be the vector and target of crimes, such as hacking and the distribution of viruses[2]. Network security is therefore essential, involving protocols and standards to ensure confidentiality, integrity and availability of data. Figure 1.1 illustrates a computer system.

Data transmission in telecommunication networks relies on techniques such as circuit switching, packet switching, and frame switching[3]. Circuit switching establishes a dedicated connection between two nodes for the duration of the communication, while packet switching divides data into packets and transmits them separately. Frame switching transmits data in blocks, with headers and error-correcting codes.

In summary, telecommunication networks use protocols such as TCP, UDP and IP to provide communication between nodes, and data compression techniques to reduce the size of transmitted data[3].

Figure I.1: a computer system.

I.2. Principles of data transmission in telecommunications networks

I.2.1 Introduction

The way we live today and how business must be made quickly and the requirements of an immediate access to accurate information, we rely on computer networking and data communication [4].

Personal Computer developments and the Revolution. Of data communication and networking has brought about Tremendous. Changes for business. Industry, science and education [3]. The advance of technology is making it possible for communication links to carry more and faster signals. As a result, services are evolving to allow use of this expanded capacity. For example, established telephone services such as conference calling, call waiting, voicemail and caller ID have been extended [5].

Our main goal is to be able to exchange data such as text, audio and video from all points in the world. We want to access the Internet to download and upload information quickly and accurately and at any time [6].

I.2.2. Data Communication

Communicating is the act of sharing information, whether it's local (face to face) or remote (over a distance). The term "telecommunication" includes Telephony, Telegraphy, and Television, means communication at a distance (tele is Greek for "far") [7].

The word "data" refers to information. It can be any text, image, audio, video and multimedia files. Thus, data communication is the exchange of data between two devices via some form of transmission medium such as a wire cable. For data communications to occur, the communicating devices must be a part of a communication system made-up of a combination of hardware (physical equipment) and software (programs)[6].

- Data communications system depends on four fundamental characteristics[7]:

  a. **Delivery:** Data must be delivered by the system to the right destination and received only by the destined receiver (device or user)
  b. **Accuracy:** The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.
  c. **Timeliness:** The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of deliveryiscalled real-time transmission.

d. **Jitter:** Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets. For example, let us assume that video packets are sent every 3D-ms. If some of the packets arrive with 3D-ms delay and others with 4D-ms delay, an uneven quality in the video is the result[8],[10].

I.2.3. Components of Data Communication

A communication system is made up of the following components [9]

a. **Message:** A message is the data (information) that is to be transmitted. It can be a text, audio, video, etc.
b. **Sender:** It is simply the device that sends the message, it can be a computer, video camera, etc.
c. **Receiver:** It is the device that receives the transmitted data. It can be a telephone handset, television, etc.
d. **Transmission medium:** It is the path that take it the data transmitted by the sander to get to the receiver, it connects two or more workstations and can be either wired/wireless media.
e. **Protocol:** It is a set of rules that defines how the data will be transmitted. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating[9].

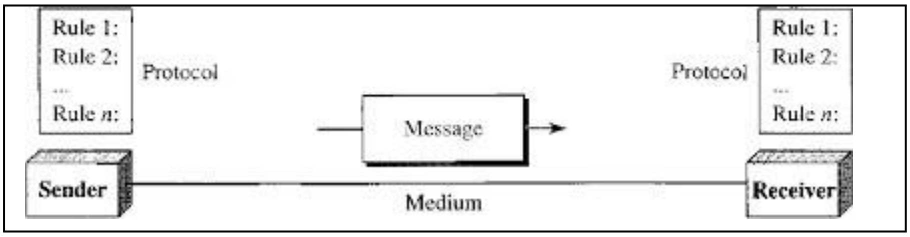


figure I.2: components of data communication system

I.2.4. Data representation:

Information today comes in different forms such as text, numbers, images, audio, and video.

- **Text:** Text is represented as a bit pattern. A sequence of bits (0s or 1s). Every text symbol can be represented by a designed sets of bit patterns aka code. The process of representing symbols is coding.
- **Numbers:** Numbers are also represented by bit patterns. A number is directly converted to a binary number to simplify mathematical operations.
- **Images:** Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the resolution. For example, an image can be divided into 1000 pixels or 10,000 pixels [9]. In the second case, there is a better representation of the image (better resolution), but more memory is needed to store the image. After an image is divided into pixels, each pixel is assigned a bit pattern. The size and the value of the pattern depend on the image. For an image made of only black and white dots, a 1-bit pattern is enough to represent a pixel[10]. If an image is not made of pure white and pure black pixels, we can increase the size of the bit pattern to include gray scale. For example, to show four levels of gray scale, we can use 2-bit patterns. A black pixel can be represented by 00, a dark gray pixel by 01, a light gray pixel by 10, and a white pixel by 11. There are several methods to represent color images. One method is called **RGB**, so called because each color is made of a combination of three primary colors: red, green, and blue. Another method is called **YCM**, in which a color is made of a combination of three other primary colors: yellow, cyan, and magenta [10].
- **Audio:** Audio refers to the recording or broadcasting of sound or music. Audio is by nature different from text, numbers, or images. It is continuous, not discrete. Even when we use a microphone to change voice or music to an electric signal, we create a continuous signal [11].
- **Video:** Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity, or it can

be a combination of images, each a discrete entity, arranged to convey the idea of motion[11].

I.2.5. Data flow:

There are three communication channels simplex, half-duplex, or full-duplex.

- **Simplex:** In simplex mode the communication is unidirectional. Only one of the two connected devices can transmit, the other can only receive. For example, a radio station usually sends a signal to the viewer, but never receives a signal from the viewer, so that the radio station is a simplex channel. It is also common to use simplex channels in fiber optic communications [12].
- **Half duplex:** Half duplex is a simplex channel that can switch the direction of the transmission, which means that data can be transmitted on both sides of the signal carrier at different times [12].
- **Full duplex:** also known as duplex. Full duplex communication channel is capable of transmitting information in both directions simultaneously on the signal carrier. It is also known as a pair of simplex links that allowed two-way simultaneous transmission
- Telephone network is a full duplex communication, because two people can communicate by a telephone line, and both can talk and listen at the same time [12].

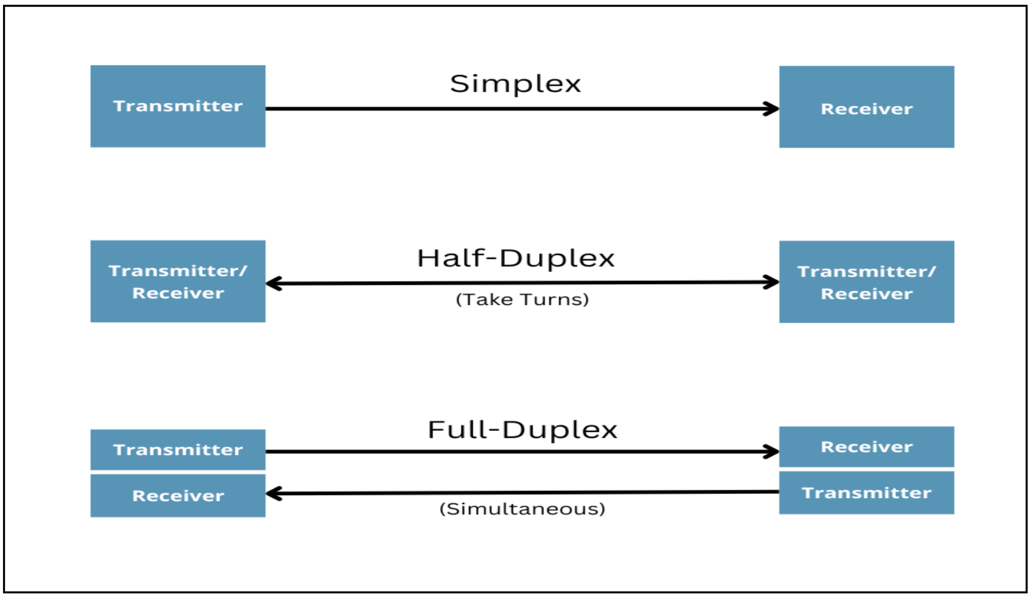


Figure I.3: The three communication channels

I.3. Pixels:

Pixels are the fundamental building blocks of digital images. They are often referred to as "picture elements" or tiny squares, dots that make up an image on a screen or in a photo. Each pixel represents a specific color and intensity, and when combined in large numbers, they create a clear and cohesive image[12].

I.3.1. Key Concepts:

Pixel Count and Density: The number of pixels in an image determines its quality and clarity. A higher pixel count and density result in a more detailed and crisper image, which is particularly important for larger displays or printed materials[13].

a. Megapixels: Megapixels are a unit used to express the total number of pixels in an image. One megapixel is equal to one million pixels. A higher number of megapixels can result in a more detailed and crisper image.

b. Pixel Dimensions: Pixel dimensions refer to the number of pixels that make up an image, typically expressed as a set of two numbers, such as width and height. This is often used to describe the resolution of an image.

c. File Size and Compression: Image file size increases with the number of pixels and color depth. Compression algorithms are used to decrease the size of a file, which is particularly important for high-resolution images.

d. Pixel Representation: Pixels are arranged in a 2-dimensional grid, represented using squares. Each pixel has a specific location and value, which can be represented as a digital image.

e. Image Processing: Digital image processing involves processing digital images using a digital computer. This includes various techniques such as image enhancement, restoration, and compression[13].

I.3.2. Applications:

a. Photography: Understanding pixels and resolution is crucial for achieving high-quality digital images in photography. This includes factors such as camera resolution, image compression, and file size[7].

b. Graphic Design: Pixels and resolution are essential for designing high-quality images for various platforms, such as print and social media.

c. Digital Imaging: Pixels and resolution play a significant role in digital imaging, including image processing, compression, and representation.

Pixels are the fundamental elements of digital images, and understanding their principles is crucial for achieving high-quality images in various applications. Factors such as pixel count, megapixels, file size, and compression all contribute to the quality and clarity of digital images.

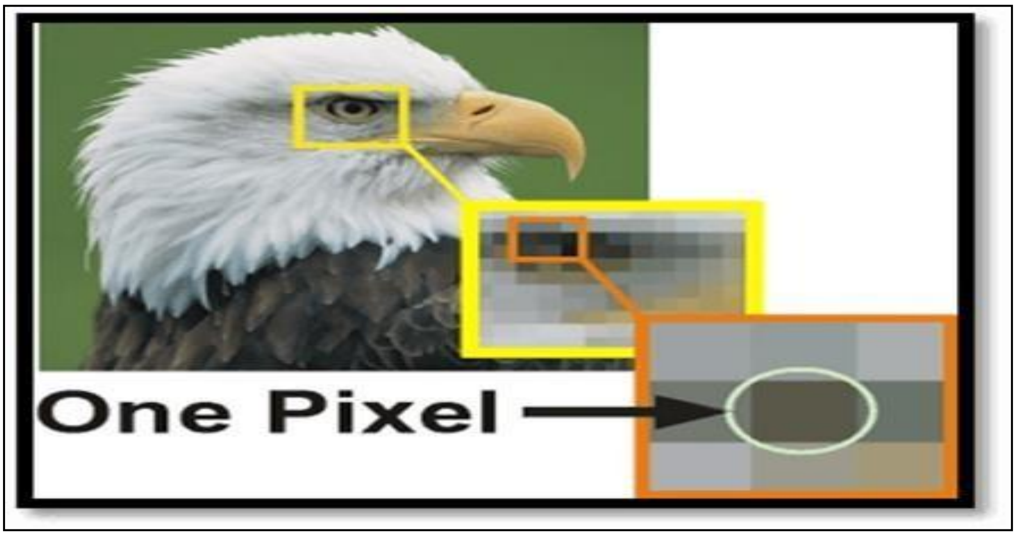


Figure I.4: Pixels

I.4. Digital image principles

I.4.1. Digital Image

A digital image can be obtained by converting an analog image into a digital image using a camera, scanners …etc. This process is called Digital Image Acquisition, it is the first step in turning the physical phenomena (what we see in real life) into a digital representation (what we see on our computers)[8].

I.4.2. Fundamental principles of digital imaging

a. Image Acquisition

Digital images can be generated using various methods, including[7]:

- **Digital cameras:** Capturing scenes using optical sensors.
- **Video recorders:** Recording moving images.
- **Image scanners:** Converting physical prints or film into digital form.
- **Image processing algorithms:** Creating images through computational techniques.

- **Computer graphics and animation:** Generating synthetic images.

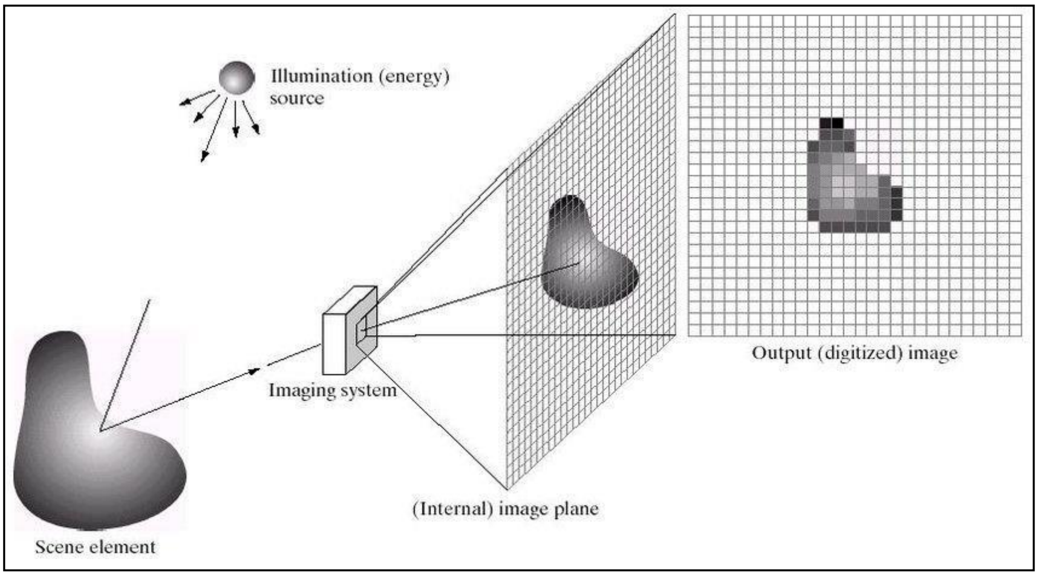


Figure I.5: Image Acquisition model

b. Quantization and Sampling

- **Quantization:** Representing continuous intensity levels (e.g., brightness) as discrete values (e.g., 8-bit grayscale).
- **Spatial sampling:** Capturing pixel values at specific locations to create a grid of discrete samples.

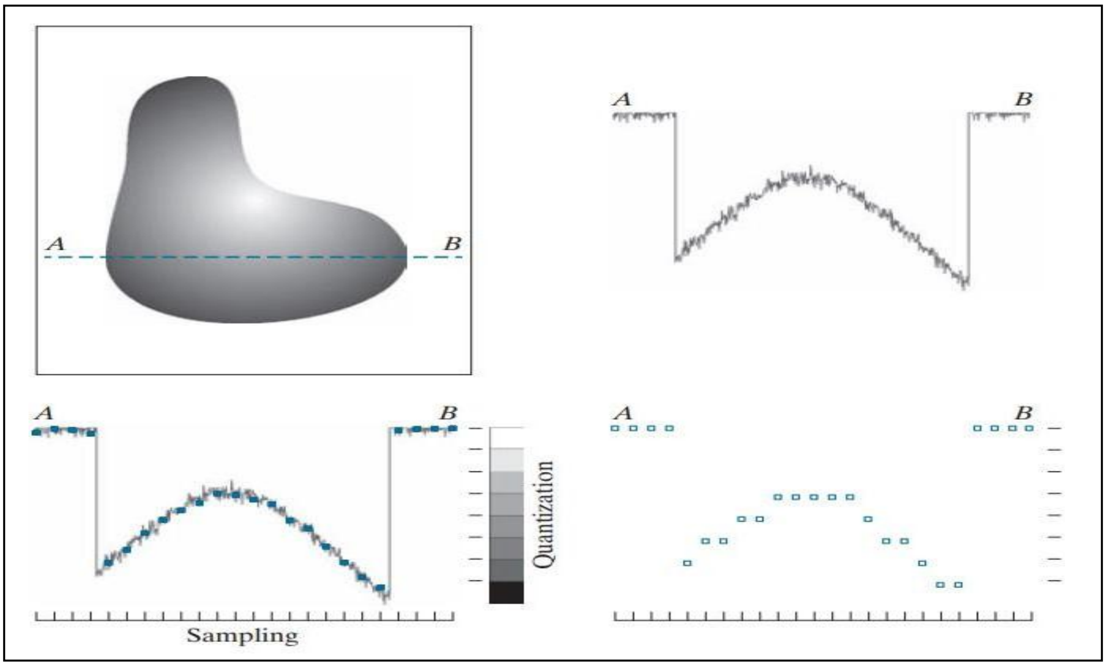


Figure I.6: Image sampling and quantization

c. Photometry and Colorimetry

- **Photometry:** Measuring light intensity (brightness) in an image.
- **Colorimetry:** Quantifying color properties based on human perception.

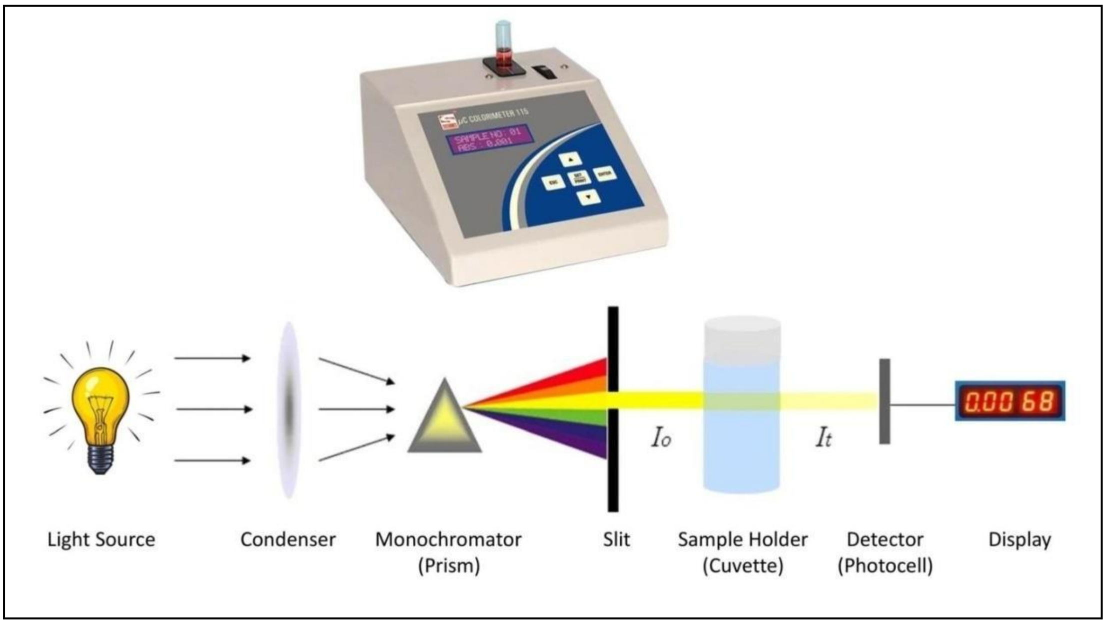


Figure I.7: Colorimeter-principle

d. Image Transformation

Techniques like rotation, scaling, and geometric transformations modify image appearance[9].

Transformations preserve essential features while altering the image.

e. Image Enhancement

Improving visual quality by adjusting contrast, brightness, and sharpness.

Techniques include histogram equalization and adaptive filtering.

f. Image Restoration

Removing degradation caused by noise, blur, or other factors.

Restoration algorithms recover the original image from the degraded version.

g. Image Classification and Segmentation

- **Classification:** Assigning labels or categories to image regions (e.g., identifying objects).
- **Segmentation:** Dividing an image into meaningful regions (e.g., separating foreground and background).

h. Image Compression

Reducing storage space by encoding images efficiently.

Lossless and lossy compression methods balance quality and file size.

I.4.3. Types of Digital Imaging

i. **Visible Light Imaging:** This includes digital photography and videography captured using digital cameras.

ii. **X-ray Imaging:** Used in digital radiography, fluoroscopy, and computed tomography (CT).

iii. **Gamma Ray Imaging:** Utilized in digital scintigraphy, single-photon emission computed tomography (SPECT), and positron emission tomography (PET).

iv. **Sound-Based Imaging:** Examples include medical ultrasonography (ultrasound) and sonar.

v. **Radio Wave Imaging:** Used in radar systems

I.4.4. Advantages of Digital Imaging

- **Infinite Reproduction:** Unlike analog images (such as film photographs), digital images can be copied indefinitely without any loss of quality[10].
- **Processing Flexibility:** Digital images can be easily manipulated, edited, and analyzed using software tools.
- **Efficient Storage and Transmission:** Digital formats allow efficient storage and transmission of images.

- **Immediate Feedback:** Digital cameras provide instant feedback, allowing adjustments during capture.

I.4.5. Practical Applications

Digital image processing encompasses several key steps:

- **Image Correction/Restoration:** Enhancing image quality by removing noise, artifacts, or distortions.
- **Image Enhancement:** Improving visual appearance by adjusting contrast, brightness, and sharpness.
- **Image Transformation:** Changing the spatial domain of an image (e.g., rotation, scaling).
- **Image Classification:** Categorizing images based on specific features or criteria[12].

I.4.6. Mathematical Foundations

Understanding digital image processing requires knowledge of mathematical concepts such as:

- **Quantization:** Representing continuous intensity values as discrete levels.
- **Spatial Sampling:** Discretizing the image grid.
- **Photometry and Colorimetry:** Quantifying light and color.
- **Color Sampling:** Representing colors in digital images.
- **Estimation of Image Model Parameters:** Fitting mathematical models to image data.
- **Image Restoration:** Recovering original images from degraded versions[13].

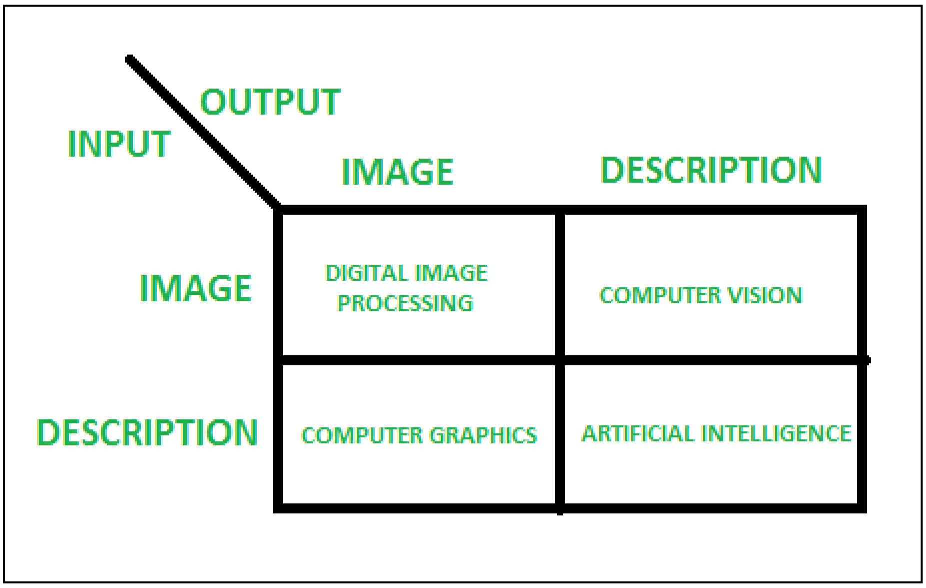


Figure I.8: Digital image processing basics

Chapter II.
Image compression techniques

II.1. Introduction

With the continuing growth of modem communications technology, demand for image transmission and storage is increasing rapidly, but due to the large number of bits required to represent a single digital image, the storage/transmission of even a few images would pose a problem. For example, consider the transmission of the low-resolution 512 × 512 × 8 bits/pixel × 3-color video image over telephone lines. Using a 9600 baud (bit/s) modem, the transmission would take approximately 11 minutes for just a single image which is unacceptable for most applications. This is why we need Image Compression[15].

II.2. Image Compression

Fortunately, digital images in their canonical representation generally contain a significant amount of redundancy[15]. Image compression, which is the art/science of efficient coding of picture data aims at taking advantage of this redundancy to reduce the number of bits required to represent an image. This can result in significant savings in the memory needed for image storage or in the channel capacity required for image transmission.

In general, Image compression involves reducing the size of image data files, while retaining necessary information. The reduced file is called the compressed file and is used to reconstruct the image, resulting in the decompressed image. The original image, before any compression is performed, is called the uncompressed image file[16]. The ratio of original uncompressed image file and the compressed file is referred to as the compression ratio. It is often written as SIZEU: SIZEC. The compression ratio is denoted by:

$$CompressedRatio = \frac{UncompressedFileSize}{CompressedFileSize} = \frac{SIZEu}{SIZEc}$$

The compression system model consists of two parts: (a) Compressor and (b)Decompressor.

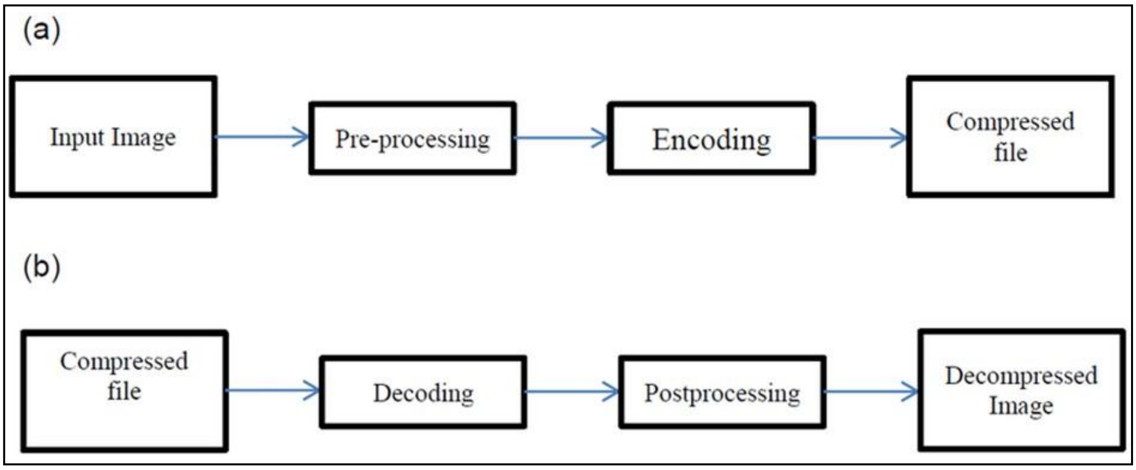


Figure II.1: Compression system model

II.3. Assessment of compressed images:

The compression ratio is the compression calculation of the image. Different factors including mean squared error and peak noise signal are determined by the compression efficiency. There are also several other techniques; PSNR and MSE are mostly used because they are simple to measure[17].

II.3.1. Mean square error (MSE).

This is calculated by the mean difference between compressed and input (original) pixel square intensities. The MSE isindicated by the following equation[18].

$$MSE = \frac{1}{m*n} \sum_{i=0}^{m-1} \sum_{j=1}^{n-1} [g'(p,q) - g(p,q)]^2$$

In these instances, the pixels of g′ (p, q) and g (p, q) respectively are the values of the restored images. And m & n are the spatial domain number of rows and columns

II.3.2. Peak signal to noise ratio (PSNR).

PSNR is the typical way in which fidelity measurement is carried out. The term PSNR is the relation between the highest signal value of deforming noise, which changes its representation value and the signal value of the deforming noise. In terms of logarithmic decibel scale the PSNR is usually represented[15]. In decibels, PSNR is computed (dB). In general, a large difference is said to be 0.5–1 dB.ThePSNR'smathematicalrepresentationis:

$$PSNR = 10\log_{10}\left(\frac{m*n}{SE}\right)$$

where MSE is Mean Square Error, and m & n are image rows & columns in spatial form.

II.3.3. Compression ratio (CR).

Is a further (extra) measurement metric for compression measurement. It is the relative sum between the bits needed by the image input to the bits required by the compressed image[16]. The following, is the compression ratio equation

$$CR = \frac{Uncompressed(original)imagesize}{Compressedimagesize}$$

II.4. Image compression methods

There are two image compression methods, traditional and deep learning compression.

II.4.1. Traditional methods:

JPEG and HEVC are among the most commonly used image and video compression Formats.There is also other techniques such as PNG,TIFF …etc. [19].

a. JPEG: JPEG is able to compress images by taking advantage of the fact that most images are locally redundant. For example, if you were to take a picture of a landscape, the sky does not contain a great amount of detail, and as a result can be dramatically reduced to save space. JPEG compresses images by first converting an image from the RGB colorspace to the YUV colorspace. Then the two chroma components (U and V) are then downsampled because the human eye is not particularly sensitive to these two channels[16]. Next, each channel of the image is then subdivided into 8x8 pixel blocks, also known as macroblocks. This macroblock then undergoes the Discrete Cosine Transform, which decomposes this 8x8 block from a set of pixels into a sum of a set of cosines with different frequencies and amplitudes. A macroblock that is generated from the sky in our landscape photo will only contain low frequency information, so the DCT will be able to represent this macroblock with only a few coefficients, as opposed to the entire 8x8 grid of pixels. Finally, this set of coefficients is quantized by being rounded to the nearest integer, and then fed through a custom lossless entropy coding technique to minimize the representation. To recover the image, this process is then conducted in reverse, with the exception of the quantization as this information has been destroyed through this process[17].

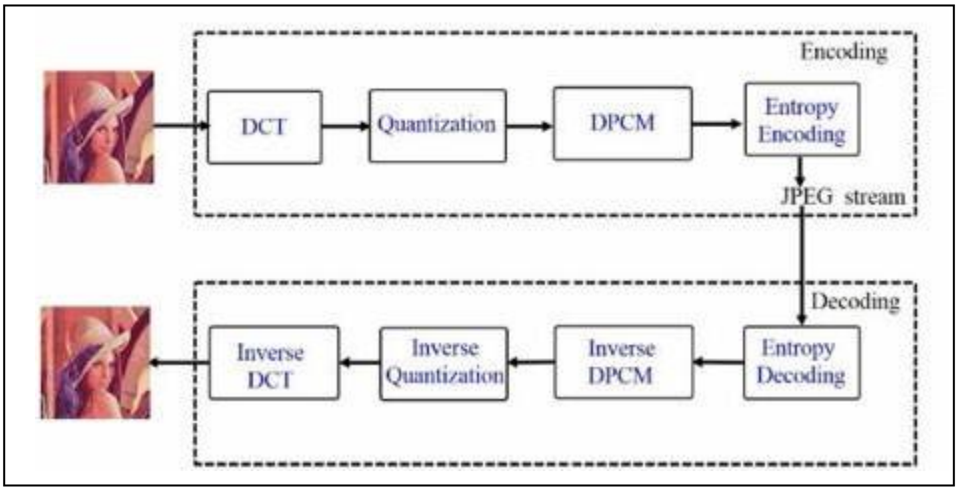


Figure II.2: JPEG Compression

b. HEVC is similar to JPEG in that it is able to take advantage of spatial redundancy, however, because it is a video format it is also able to take advantage of temporal redundancies as well[16]. The process for video compression with HEVC involves intraprediction, in which macoblocks are estimated based on surrounding macoblocks. It also involves interprediction in which macoblocks are estimated based on previously recovered macroblocks from previous frames. The approach HEVC uses for intraprediction is similar to that of JPEG in that the macroblock is decomposed into its DCT and quantized. However, the main difference is that on the encoder side, HEVC calculates a prediction and compares its prediction to the ground truth it has available. This residual between the prediction and the ground truth is then quantized, compressed, and sent to the decoder side. For temporally based intraprediction, HEVC decomposes a set of video frames into a background and the motion on top of that background. This approach allows HEVC to not 4 have to send each frame over and over, and instead, HEVC can send only the parts that are changing[15].

c. PNG. PNG is Portable Network Graphics' short form. This format is used for image compression without loss. The PNG file format replaces the GIF file format ascompressionis 10–30%

higher than the GIF. PNG creates smaller files which allows for more colors.PNG supports partial transparency. There are two variants of PNG-24 (224 = 16777216 supports) and PNG-8 (28 = 256 colors supported)[17].

d. TIFF. It stands for Tagged Image File Format short form. It can be considered alosslessformat. The format is used primarily in photography and desktop publishing,because of its extremely high quality. Saving a total of 8, 16 bits for a total 24 and 48bits respectively per(red, green, blue). Compression of TIFF in web applications is relatively poor and not used. They are particularly used in massive sizes in high-quality prints[15].

II.4.2. Deep Learning Methods

Before we dive into deep learning methods let's make sure we all understand what is deep learning, machine learning.

a. **Machine Learning:** It is a field in artificial intelligence, where computers learn from data and it is more dependent on human interventions to learn and improve its accuracy.Machine learning algorithms leverage structured labelled data to make predictions[20].

b. **Deep Learning:** It is a subset of machine learning; its applications use a layered structure of algorithms called an artificial neural network (ANN). The design of such an ANN is inspired by the biological neural network of the human brain, leading to a process of learning that's far more capable than that of standard machine learning models[24].

Now we can talk about CNNs and GANs: Deep Learning methods

While JPEG and HEVC have and continue to perform well, their main limitation is that they are essentially highly tuned heuristic algorithms. Although such algorithms perform well, in the age of big data, we are capable of training machines to learn efficient representations of images [22].

1. **CNNs** The breakthrough moment for convolutional neural networks (CNN) came from the 2012 ImageNet Large Scale Visual Recognition Contest (ILSVRC). When AlexNet, a CNN, outperformed all other entries by 10% in top-1 accuracy, many researchers took note and began developing their own CNNs. The key contribution of AlexNet was the insight to train deep networks by utilizing GPUs for accelerating training. Years later, many higher performing CNN models have been introduced to ILSVRC. These now famous architectures include GoogLeNet, VGG Net, ResNet and Inception Net. This dramatic increase in top-1 accuracy has sparked a revolution in computer vision related artificial intelligence tasks. While ILSVRC was focused on image classification, many researchers began developing approaches for other tasks such as image compression. Manyresearchers have already demonstrated that CNNs stand to benefit the field of image and video compression [24].
2. **GANs**As the dramatic improvement of CNNs began to slow down, Ian Goodfellow et al introduced a new approach to training a CNN called generative adversarial networks (GANs). This idea was to use one neural network to train another. In theory, this would allow neural networks to achieve even greater performance. In practice, GANs have proven to be difficult to train. However, if trained properly, GANs are capable of achieving impressive results. GANs are particularly interesting for the task of image and video compression because GANs are designed to 'hallucinate 'images into its output. This means that GANs are able to learn what a realistic image should look like a is then capable of producing something that one might actually observe in an image. In image and video compression, because data is thrown away during compression, GANs offer the ability to reintroduce the data that was thrown away by sampling a prior knowledge from images and videos it has trained on [24].

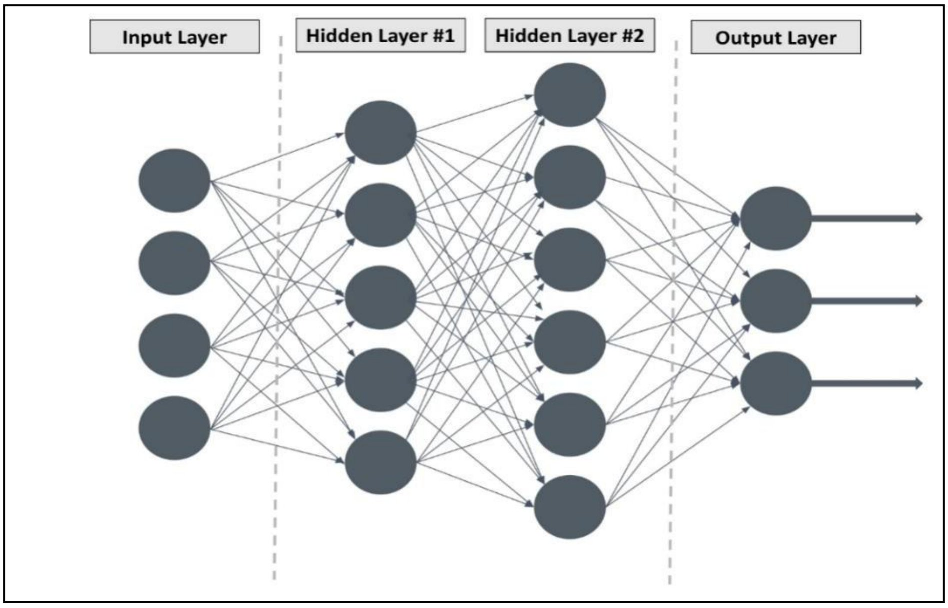


Figure II.3: A simple artificial neural network

II.5. Categories of Image Compression

Generally, compression is split into two categories: Lossy compression and Lossless compression.

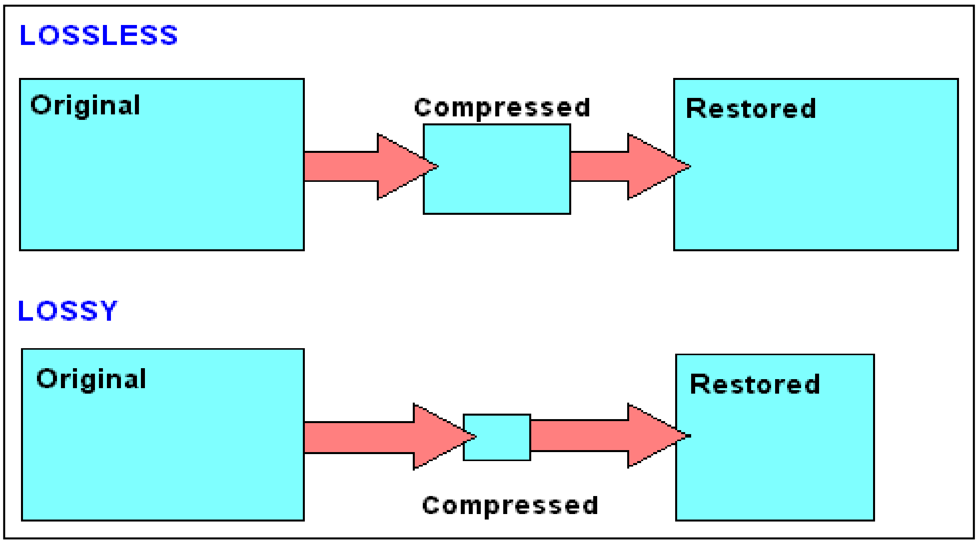


Figure II.4: The difference between lossy and lossless compression

II.5.1. Lossy Compression:

Lossy compression is a way of minimizing data size while maintaining usable or important data. Redundant bits are replaced by lossy compression methods, minimizing the file size. As some of the data is lost in the compression, it is not possible to convert the compressed file back to the original file. The recover file after de-compression is an approximation of the original file which is restored based on the compression program's comprehension. Methods which fall under the technique of lossy compression are Transformation Coding, Vector Quantization (VQ), Fractal Coding[17].

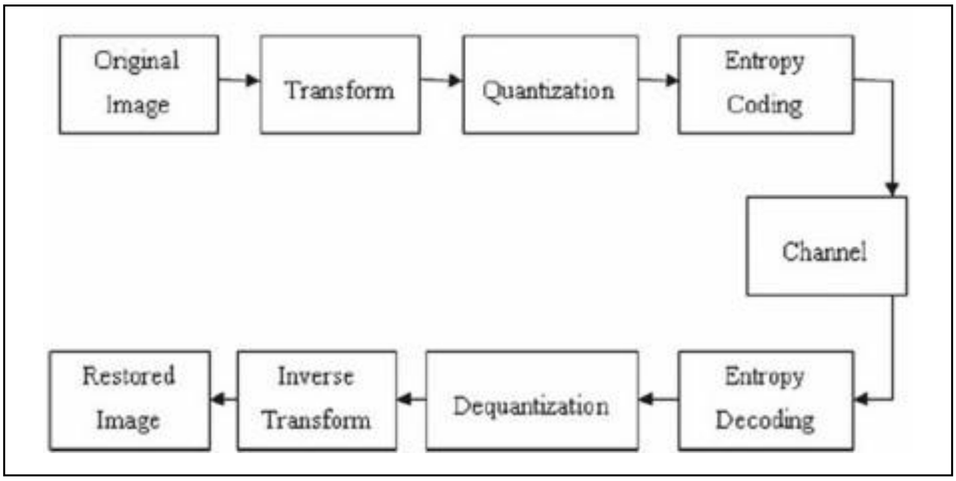


Figure II.5: LOSSY Image Compression

a. Transformation coding:

In this coding scheme, transforms such as DFT (Discrete Fourier Transform) and DCT (Discrete Cosine Transform) are used to change the pixels in the original image into frequency domain coefficients (called transform coefficients). These coefficients have several desirable properties. One is the energy compaction property that results in most of the energy of the original data being concentrated in only a few of the significant transform coefficients. This is the basis of achieving the compression. Only those few significant coefficients are selected and the remaining are discarded. The selected coefficients are considered for further quantization and entropy encoding. DCT coding has been the most common approach to transform coding. It is also adopted in the JPEG image compression standard[21].

b. Vector quantization:

The basic idea in this technique is to develop a dictionary offixed-size vectors, called code vectors. A vector is usually ablock of pixel values. A given image is then partitioned into non- overlapping blocks (vectors) called image vectors. Then for each image vector, the closest matching vector in the dictionary is determined and its index in the dictionary is used as the encoding of the original image vector. Thus, each image is represented by a sequence of indices that can be further entropycoded[19].

c. Fractal coding:

The essential idea here is to decompose the image into segments by using standard image processing techniques such as color separation, edge detection, and spectrum and texture analysis. Then each segment is looked up in a library of fractals. The library actually contains codes called iterated function system (IFS) codes, which are compact sets of numbers. Using a systematic procedure, a set of codes for a given image are determined, such that when the IFS codes are applied to a suitable set of image blocks yield an image that is a very close approximation of the original. This scheme is highly effective for compressing images that have good regularity and self-similarity[17].

II.5.2. Lossless Compression:

In lossless compression, processes do not result in any loss of data or information. The compression is achieved by building a file with fewer bits without any loss of data. This is achieved with the aid of different statistical and mathematical methods like entropy coding which are often used to transform the compressed file back to the original uncompressed file[15]. The following are the methods that come under lossless compression: Huffman Encoding, Lempel, Ziv and Welch (LZW) Coding, Predictive Coding.

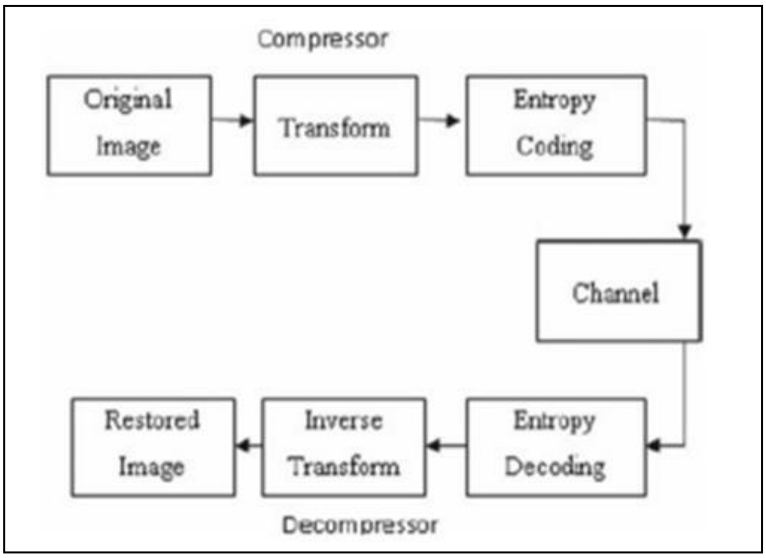


Figure II.6: LOSSLESS Image Compression

a. Huffman coding:

This is a general technique for coding symbols based on their statistical occurrence frequencies (probabilities). The pixels in the image are treated as symbols. The symbols that occur more frequently are assigned a smaller number of bits, while the symbols that occur less frequently are assigned a relatively larger number of bits. Huffman code is a prefix code. This means that the (binary) code of any symbol is not the prefix of the code of any other symbol[16]. Most image coding standards use lossy techniques in the earlier stages of compression and use Huffman coding as the final step.

b. Lempel-Ziv coding:

This is based on storing frequently occurring sequences of symbols (pixels) in a dictionary (table). Such frequently occurring sequences in the original data (image) are represented by just their indices into the dictionary. This has been used in TIFF (Tagged Image File Format) and GIF (Graphical Interchange Format) file formats. This scheme has also been used for compressing half-tone images. (Halftone images are binary images that provide the visual effect of continuous-tone gray images by using variations of the density of black dots in the images)[19].

c. Predictivecoding:

This assumes that the pixels in images conform to the autoregressive model, where each pixel is a linear combination of its immediate neighbors. The lossless differential pulse code modulation (DPCM) technique is the most common type of lossless predictive coding. In the lossless DPCM scheme, each pixel value (except at the boundaries) of the original image is first predicted based on its neighbors to get a predicted image. Then the difference between the actual and the predicted pixel values is computed to get the differential or residual image. The residual image will have a much less dynamic range of pixel values. This image is then efficiently encoded using Huffman coding[14].

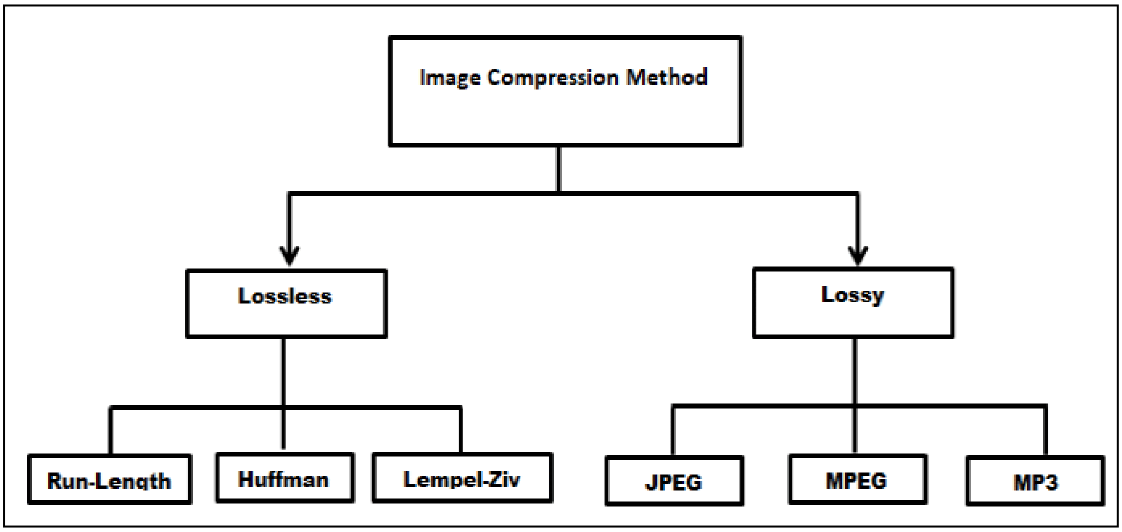


Figure II.7: Image Compression Methods

II.6. Image compression standards:

Image compression standards have been developed to facilitate the interoperability of compression and decompression of schemes across several hardware platforms, operating systems and applications. Most standards are hybrid systems making use of a few of the basic techniques already mentioned. The major image compression standards are Group 3, Group 4, and JBIG (Joint Bi-level Image Group) for bi-tonal images, and JPEG (Joint Photographic Experts Group) for continuous-tone images[15]. The most common application that uses compression of bi-tonal images is digital facsimile (FAX).

Lossy Compression	Lossless Compression
JPG	GIF
JPEG	PNG
JPEG2000	TIFF

Table II.1: Different Image Compression Formats

II.7. Practical Considerations in Compression

The choice of compression technique often depends on the intended use of the image. For instance, lossy methods are suitable for natural images in applications where a slight loss of fidelity is acceptable, such as online media. Conversely, lossless methods are preferred for archival purposes and in fields requiring high precision like medical imaging or technical drawings. The method chosen can significantly impact the load times, which is crucial for applications like SEO, conversion rates, and overall user experience[17].

By understanding and applying the appropriate image compression techniques, it is possible to effectively manage the balance between image quality and file size, optimizing both storage and transmission of visual content[15].

Chapter III.
CNN (Deep learning) based approach for image compression

III.1. Introduction

While the traditional methods (JPEG, etc.) have and continue to perform well, their main limitation is that they are essentially highly tuned heuristic algorithms. Although such algorithms perform well, in the age of big data, we are capable of training machines to learn efficient representations of images.

Neural network is a recent compression tool because data processing is in parallel manner and thus requires less time, and it is general performance is superior to any other technique.

It is therefore important to transform a neural network image information efficiently. The data can be transformed by techniques like PCA based on factorizing techniques developed in linear algebra.

In this chapter we present CNN, the machine learning method for image compression but before we dive into CNN, let’s make sure we all understand the difference between Machine learning and Deep learning.

III.2. Difference between machine learning (ML) and deep learning (DL):

Since deep Learning algorithms are Machine Learning algorithms. it might be better to think about what makes Deep Learning special within the field of Machine Learning [20].

While traditional Machine Learning algorithms have a rather simple structure, such as linear regression or a decision tree, Deep Learning is based on an artificial neural network. This multi-layered ANN is, like a human brain, complex and intertwined [21].

Deep Learning requires large amounts of data for training but learns on its own from environment and past mistakes unlike Machine Learning that requires more human intervention to correct and learn [21].

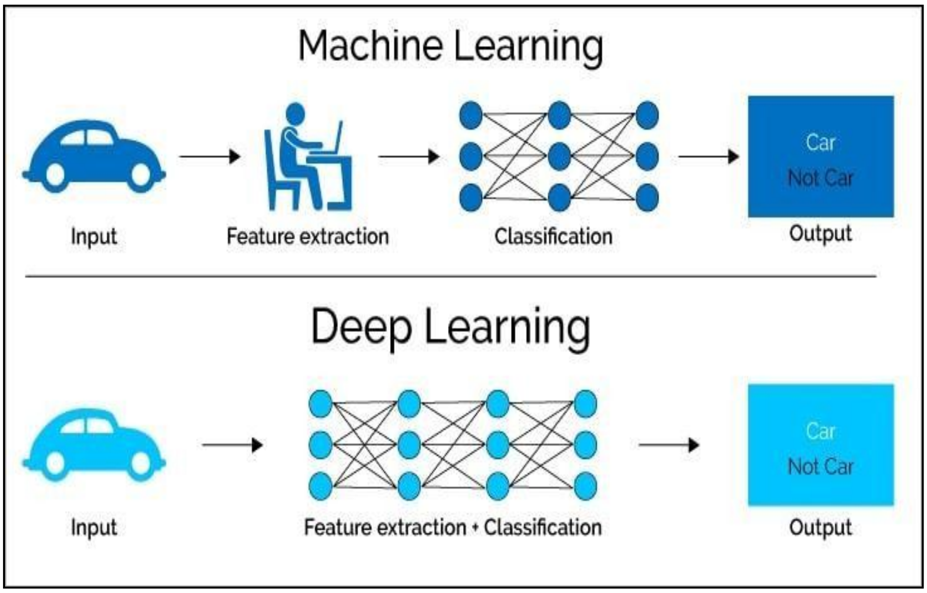


Figure III.1: Difference between ML and DL

III.3. Convolutional Neural Network:

(CNNs) are analogous to traditional ANNs in that they are comprised of neurons that self-optimize through learning. Each neuron will still receive an input and perform an operation (such as a scalar product followed by a non-linear function) - the basis of countless ANNs. From the input raw image vectors to the final output of the class score, the entire of the network will still express a single perceptive score function (the weight). The last layer will contain loss functions associated with the classes, and all of the regular tips and tricks developed for traditional ANNs still apply.

The only notable difference between CNNs and traditional ANNs is that CNNs are primarily used in the field of pattern recognition within images. This allows us to encode image-specific features into the architecture, making the network more suited for image-focused tasks - whilst further reducing the parameters required to set up the model.

One of the largest limitations of traditional forms of ANN is that they tend to struggle with the computational complexity required to compute image data. Common machine learning benchmarking datasets such as the MNIST database of handwritten digits are suitable for most forms of ANN, due to its relatively small image dimensionality of just 28 × 28. With this dataset a single neuron in the first hidden layer will contain 784 weights (28 × 28 × 1 where 1 bare in mind that MNIST is normalized to just black and white values), which is manageable for most forms of ANN.

If you consider a more substantial colored image input of 64 × 64, the number of weights on just a single neuron of the first layer increases substantially to 12, 288. Also take into account that to deal with this scale of input, the network will also need to be a lot larger than one used to classify color-normalized MNIST digits, then you will understand the drawbacks of using such models

III.4. CNN Architecture:

CNNs are comprised of three types of layers. These are convolutional layers, pooling layers and fully-connected layers. When these layers are stacked, a CNN architecture has been formed. A simplified CNN architecture for MNIST classification is illustrated in Figure 9:

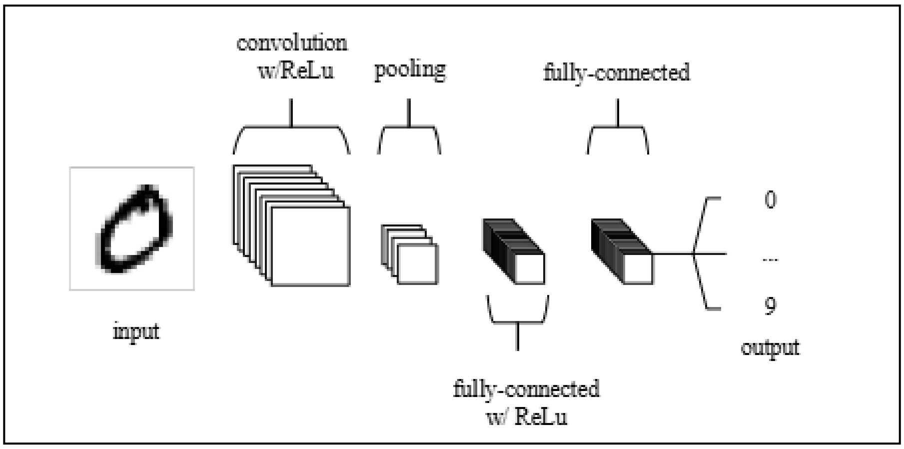


Figure III.2: A simple CNN architecture, comprised of just five layers

The basic functionality of the example CNN above can be broken down into four key areas:

1. As found in other forms of ANN, the input layer will hold the pixel values of the image.

2. The convolutional layer will determine the output of neurons of which are connected to local regions of the input through the calculation of the scalar product between their weights and the region connected to the input volume. The rectified linear unit (commonly shortened to ReLu) aims to apply an 'elementwise' activation function such as sigmoid to the output of the activation produced by the previous layer.

3. The pooling layer will then simply perform down sampling along the spatial dimensionality of the given input, further reducing the number of parameters within that activation.

4. The fully-connected layers will then perform the same duties found in standard ANNs and attempt to produce class scores from the activations, to be used for classification. It is also suggested that ReLu may be used between these layers, as to improve performance.

Through this simple method of transformation, CNNs are able to transform the original input layer by layer using convolutional and down sampling techniques to produce class scores for classification and regression purposes.

III.5. Use of Convolution Neural Networks in Image Compression:

How can CNNs capture image artifacts. They have been used recently in many image compression architectures.

Jiang, et al. [27] has used fully convolutional auto-encoder to obtain a compressed representation of an image. The architecture shown in Figure 10 below, has two distinct parts ComCNN and RecCNN. Series of convolutional layers stacked in this way can capture features of an image. The author claims that because of the use of multi-layer CNNs, this architecture can maintain the structural composition of an image as well.

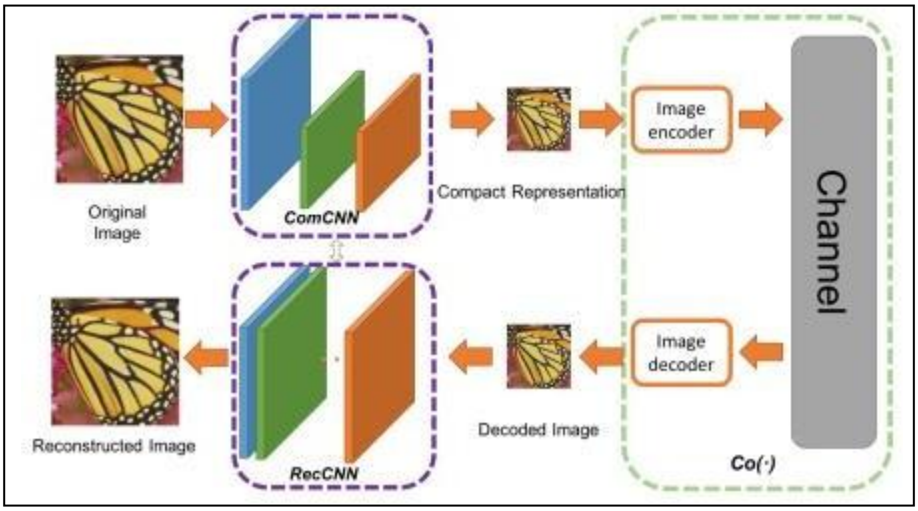


Figure III.3: End to end image compression framework using CNN.

The **ComCNN** is a network responsible in compressing these images in such a way that resultant images can be effectively reconstructed by reconstruction network. This network consists of three convolutional layers with the second layer followed by batch normalization layer. Since the first convolutional layer uses a stride of two, the image size is reduced by half.

RecCNN layer uses twenty Neural network layers. Apart from the first and the last layer, each layer in this formation carries out Convolutional and batch normalization operation. The author trained this network using 400 grayscale images and 50 epochs. SSIM and PSNR metrics of these images are better than JPEG images.

In lossy image compression techniques, artifacts of image compression algorithm are visible in images. An example of such artifacts is visible on images for which tiling was used for quantization. In such images, these tile boundaries continue to remain in the images.

III.6. Filter Pruning Strategy and CNN Model Efficiency

A filter pruning strategy is employed to enhance the efficiency of CNN architectures by assigning importance weights to convolutional filters. This method not only addresses the challenge of oversized CNN models but also significantly reduces the required storage space and computational time. The pruning effectively decreases the number of floating-point operations per second (FLOPs) and parameters, maintaining near-original accuracy of the model[20].

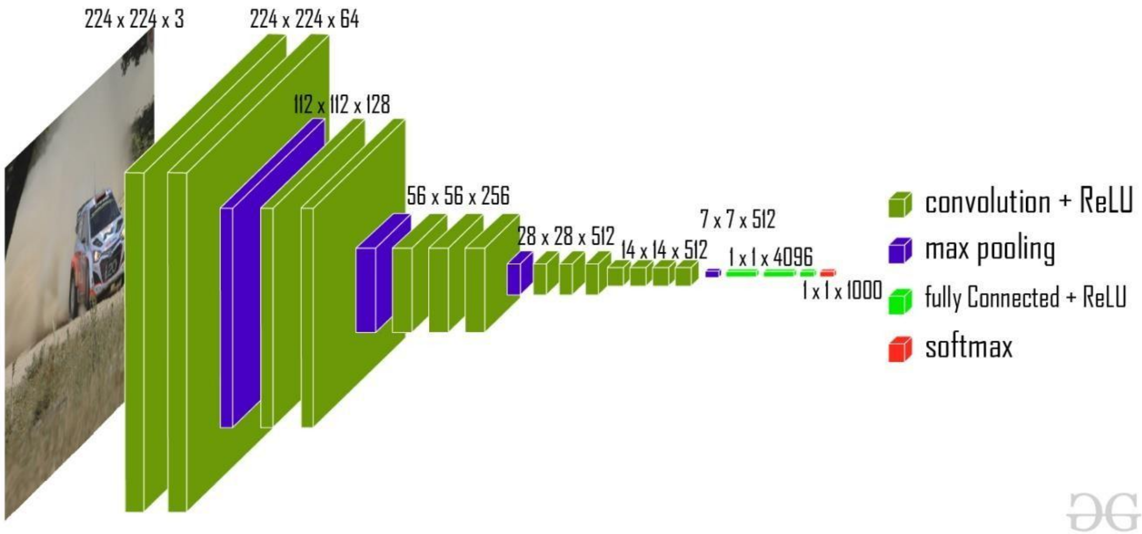


Figure III.4: CNN model

III.6.1. Comparative Analysis of Deep Learning Models for Image Compression

1. Evaluated Models and Techniques

- Factorized Prior Autoencoder
- Nonlinear Transform Coder with Factorized Priors
- Hyperprior Model with Non-Zero-Mean Gaussian Conditionals

These models were rigorously tested against traditional codecs like PNG using the JPEG method to determine their effectiveness in image compression[15].

2. MetricsUsed for Evaluation

The performance of these models was assessed using several key metrics:

- Compression Coefficient
- SSIM (Structural Similarity Index Measure)
- PSNR (Peak Signal-to-Noise Ratio)

The dataset for this evaluation included images of 5 bottles of Italian wines and 1 bottle of sauce, providing a diverse range of visual content for analysis.

3. Results of Machine Learning Models

The analysis revealed that the mbt2018-mean-msssim-5 model exhibited the best compression performance with a compression coefficient of approximately 0.13 while maintaining an SSIM of around 0.97. Following closely was the bmshj2018-factorized-msssim-6 model with a compression coefficient of approximately 0.23[23].

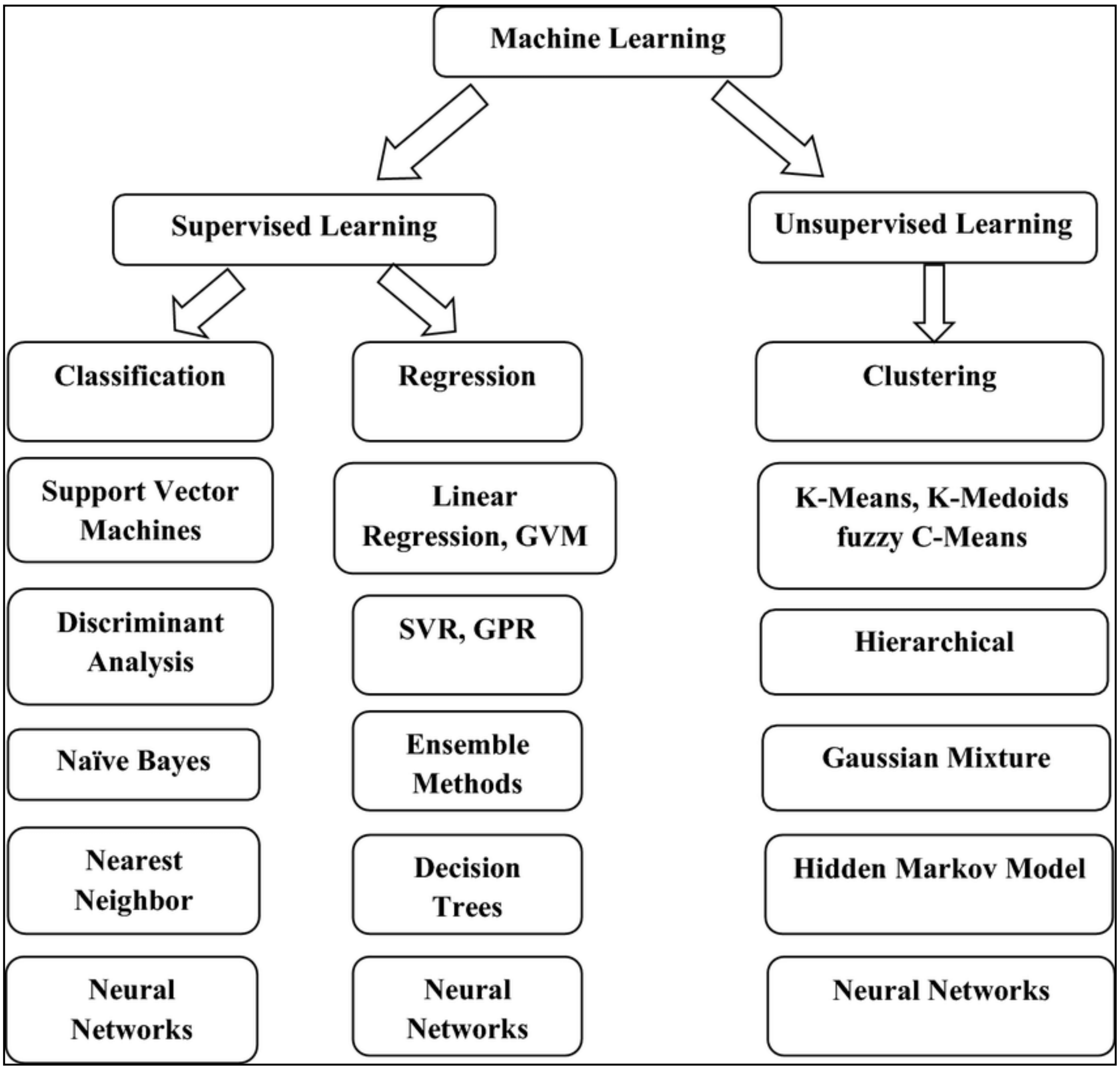


Figure III.5: Machine Learning model overview

III.6.2. Advancements in Video Compression Using Deep Learning

Deep learning has significantly advanced the field of video compression. Techniques such as DNNs, RNNs, GANs, and AEs have been applied to enhance both lossy and lossless compression results, aiming to reduce bitrate while preserving data quality and complexity[22]. The paper titled "Advances in Video Compression System Using Deep Neural Network: A Review and Case Studies" discusses the application of deep learning across three major functional blocks of video compression: pre-processing, coding, and post-processing. These advancements are critical as they aim to maximize the end-user quality of experience (QoE) under limited bit rate budgets[24].

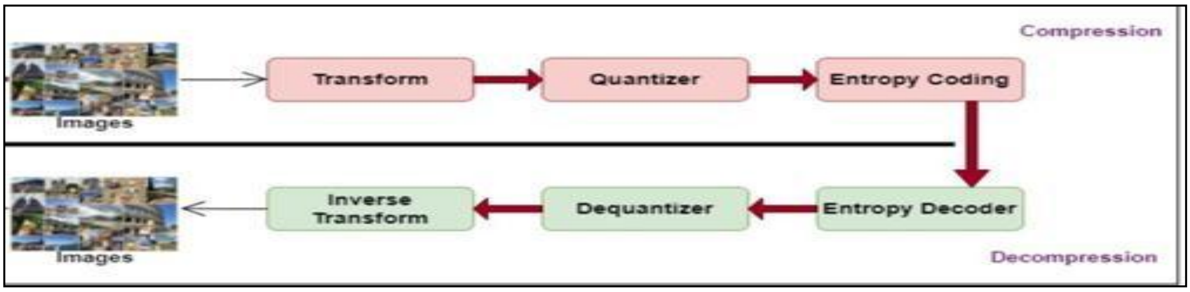


Figure III.6: Deep learning approaches for image compression.

III.7. Challenges and Solutions

III.7.1. Deep Learning Model Limitations

Deep learning models, while powerful, face several challenges that can hinder their effectiveness in visual content compression:

1. **Data Availability:** Obtaining large datasets necessary for training deep learning models is often challenging, particularly in specialized fields where data may be scarce or proprietary
2. **Computational Demands:** Training deep learning models is resource-intensive, requiring significant computational power and time, which can be a barrier for smaller organizations or individuals
3. **Overfitting Risks:** There is a risk of models being too closely fitted to the training data, resulting in poor performance on new, unseen datasets
4. **Interpretation Difficulties:** Deep learning models can be opaque, making it difficult to understand the reasoning behind their decisions or predictions.

III.7.2. Addressing the Challenges

To mitigate these challenges, several strategies can be employed[23]:

1. **Transfer Learning:** Utilizing pre-trained models can reduce the need for large training datasets and shorten training times

2. **Regularization Techniques:** Implementing methods such as dropout and weight decay can help prevent overfitting, ensuring models perform well across different datasets
3. **Data Augmentation:** Enhancing the diversity and quality of training data through techniques like rotation, scaling, and color adjustment can improve model robustness
4. **Interpretability Tools:** Employing tools and techniques to increase the transparency of model decisions can aid in troubleshooting and refining model outputs

III.7.3. Technological and Practical Constraints

Deep learning for visual content compression also encounters practical and technological constraints[24]:

1. **Hardware Requirements:** High-performance GPUs are essential for efficiently training deep learning models, which may not be accessible to all users
2. **Complexity in Deployment:** The complexity of deep learning models can make them difficult to integrate and maintain within existing systems
3. **Bandwidth and Transmission Issues:** Compression techniques must contend with bandwidth limitations and noisy transmission channels, which can degrade user experience.

III.7.4. Innovative Approaches and Solutions

Despite these challenges, innovative approaches continue to emerge:

1. **Improved Compression Techniques:** Leveraging advanced neural network architectures like MLP, CNN, and GAN can lead to superior compression outcomes, enhancing both the efficiency and quality of visual content compression
2. **Hybrid Models:** Integrating traditional compression methods with deep learning approaches can provide a balance between performance and complexity

3. **Quality Metrics Enhancement:** Deep learning models have been shown to improve key quality metrics such as PSNR and SSIM, particularly in comparison to older standards like JPEG2000

These solutions not only address the inherent challenges of deep learning but also harness its potential to revolutionize the field of visual content compression[24].

III.7.5. Comparing Traditional and Deep Learning Techniques

1. **Performance and Accuracy**

Deep learning (DL) models are renowned for their ability to perform complex cognitive tasks with accuracy that often matches or surpasses human capabilities. This is particularly evident in image processing applications such as image classification, semantic segmentation, and object detection, where DL minimizes the error between actual and predicted. Traditional computer vision (CV) techniques, while efficient, typically require more expert analysis and fine-tuning compared to the self-learning capabilities of deep learning models[20].

2. **Efficiency and Resource Management**

Traditional CV techniques are often lauded for their efficiency, requiring fewer lines of code and less computational power, making them suitable for deployment on low-cost microcontrollers or for applications such as image stitching and 3D mesh reconstruction. In contrast, DL models, particularly those based on convolutional neural networks (CNNs), require substantial computational resources, which can be a limiting factor for their use in edge computing scenarios[22],[23].

3. **Flexibility and Adaptability**

DL models offer unparalleled flexibility, as they can be retrained and adapted to new datasets and use cases. This adaptability is crucial in fields where the types of visual content and the tasks required can vary significantly. Traditional techniques, on the other hand, offer predictability and are not class-specific, which can be advantageous in applications requiring consistent performance across varied scenarios[21].

4. Hybrid Approaches

Recognizing the strengths and limitations of both traditional CV and DL, many researchers and practitioners advocate for hybrid approaches. These methods combine the rapid deployment and efficiency of traditional techniques with the high accuracy and adaptability of deep learning. Such approaches are particularly valuable in high-performance systems where quick implementation is necessary[19].

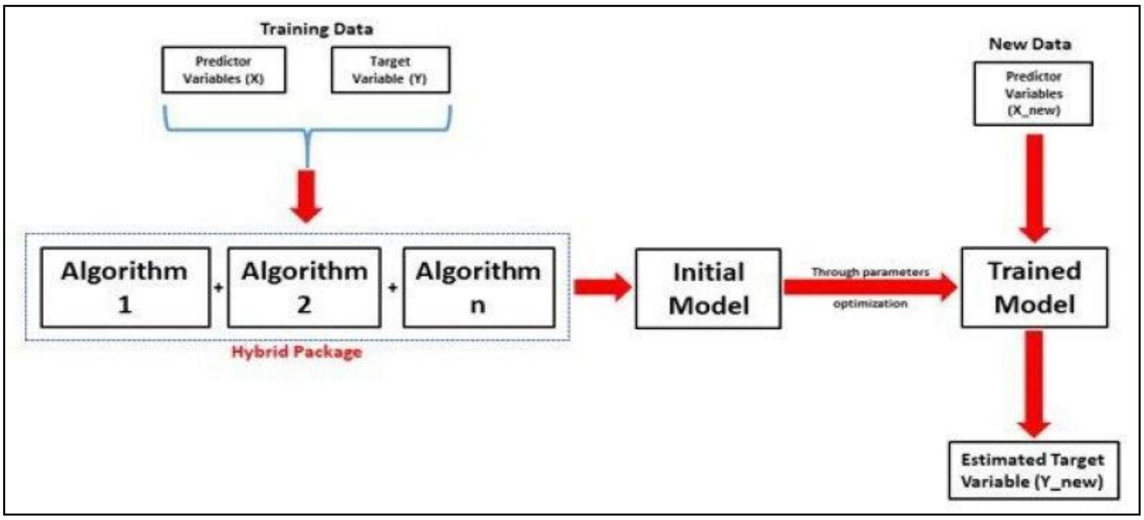


Figure III.7: hybrid Machine learning.

5. Application-Specific Considerations

The choice between traditional CV techniques and DL models often depends on specific application requirements, including the need for real-time processing, the availability of training data, and computational resource constraints. Traditional techniques are preferred in situations where transparency and low resource usage are paramount, while DL models are favored for their superior performance in handling large and diverse datasets.

Overall, while deep learning offers significant advantages in terms of accuracy and flexibility, traditional CV techniques remain relevant due to their efficiency and lower resource requirements. The ongoing development of hybrid models suggests a promising direction for leveraging the strengths of both methodologies to address the diverse needs of modern image processing applications.

III.7.6. Future of Deep Learning in Compression

a. Expanding Horizons in Deep Learning Techniques

Deep learning continues to evolve, pushing the boundaries of data processing and analysis across various modalities. Its application in atomistic simulations, materials imaging, spectral analysis, and natural language processing highlights its versatility and growing importance in scientific research. These advancements are underpinned by the development of high-quality forward models and generative unsupervised deep learning methods, which have significantly enhanced the capabilities of synthetic data generation[25].

b. Optimization and Acceleration of Deep Neural Networks

Recent research has focused on making deep neural networks (DNNs) more efficient, particularly in terms of computation and memory usage. Techniques such as parameter pruning, quantization, low-rank factorization, and knowledge distillation are being actively developed to compact and accelerate DNN models. These methods not only improve the performance of deep learning models but also make them more feasible for use in resource-constrained environments[26].

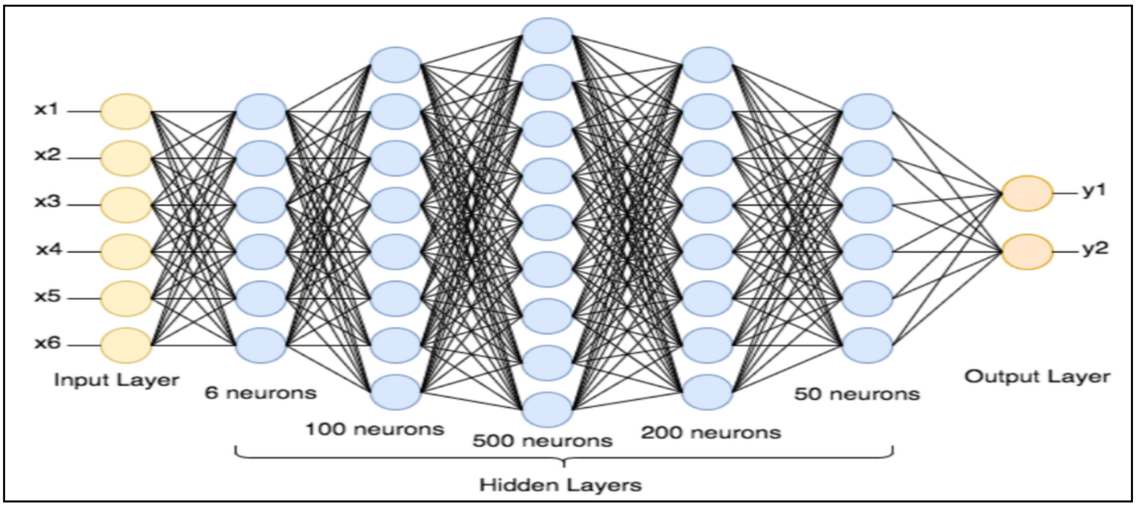


Figure III.8: Deep Neural Network architecture.

c. Integration of Deep Learning with Symbolic AI

The fusion of deep learning with symbolic artificial intelligence (AI) represents a significant trend in the field. This hybrid approach facilitates the learning of visual contents and semantic parsing without the need for explicit supervision, opening new avenues for AI to understand and interact with complex data structures more naturally[26].

d. Self-Supervised Learning and Neuroscience Influences

Self-supervised learning techniques, which train models to self-label data using raw data forms, are gaining traction. This approach reduces the reliance on large labeled datasets, which are often costly and time-consuming to produce. Additionally, the integration of neuroscience principles into deep learning architectures is helping to bridge the gap between artificial and biological neural processes, potentially leading to more intuitive and adaptable AI systems[25].

III.8. Future Challenges and Directions

The ongoing growth of data volumes presents both opportunities and challenges for deep learning. The estimated increase to over 180 zettabytes by 2025 underscores the critical need for advanced data science tools capable of managing and analyzing this vast amount of information[27].

As deep learning continues to be a pivotal technology in the Fourth Industrial Revolution, its integration into various domains like healthcare, cybersecurity, and text analytics will likely increase, necessitating further innovations in model training and AI platform development [26].

By addressing these challenges and continuing to innovate, deep learning is set to remain at the forefront of technology, driving progress in data compression and beyond.

Chapter IV. Implementation and discussion

IV.1. Introduction

The advent of the digital age has led to an explosion in the quantity of images generated, stored and exchanged daily across the world. This abundance of visual data poses major challenges in terms of storage, transmission and processing. Image compression plays a crucial role in solving these challenges by helping to reduce file sizes while preserving acceptable visual quality. However, traditional compression methods, such as JPEG and JPEG2000, often reach their limits in terms of quality and efficiency.

Faced with this observation, Deep Learning is emerging as a promising approach for image compression. Convolutional neural networks (CNNs) have demonstrated the ability to learn efficient representations of visual data, opening new avenues for image compression. By leveraging the machine learning capabilities of CNNs, it is possible to develop more efficient image compression algorithms that can outperform traditional methods while adapting to a wide variety of visual content.

In this context, our Master's thesis aims to explore image compression using Deep Learning. We will be particularly interested in the design and evaluation of image compression models based on convolutional neural networks. Our goal is to develop an approach to image compression that combines efficiency and visual quality, while providing superior performance to traditional methods.

IV.2. Problematic

The problem of our study can be formulated as follows:

Faced with the growth in the use of visual data and the limitations of traditional image compression methods, how can Deep Learning be exploited to develop more efficient and adaptive image compression techniques?

To respond to this problem, our research work will be divided into several axes:

Exploring the theoretical foundations: We will examine in detail the principles of Deep Learning and convolutional neural networks, as well as the basics of image compression. This exploration will allow us to lay the necessary foundations for the design of our image compression models using Deep Learning.

Design of image compression models: We will develop convolutional neural network architectures specifically adapted to image compression. Building on recent advances in the field, we will design models capable of learning efficient representations of images while minimizing quality loss.

Performance Evaluation: We will evaluate the performance of our image compression models using objective and subjective metrics. We will also compare our approaches with traditional image compression methods to demonstrate their superiority in terms of quality and efficiency.

By addressing these different aspects, our Master's thesis aims to contribute to the advancement of research in image compression through Deep Learning, by proposing innovative and effective solutions to meet current and future challenges linked to the management of visual data.

Deep learning algorithms used in the field of image compression

In the field of image compression, several algorithms based on Deep Learning have been developed to improve performance compared to traditional methods such as JPEG and JPEG2000. Here are some of the most commonly used Deep Learning algorithms in this field:

Autoencoders: Autoencoders are neural networks used to learn compact representations of input data by compressing them into a reduced-dimensional latent space and then decompressing them to reconstruct the original data. Autoencoders are often used as the basis for Deep Learning image compression algorithms.

Variational Autoencoders (VAE): VAEs are an extension of autoencoders that introduce a constraint on the distribution of the latent space, allowing more regular and general representations of the data to be learned. VAEs are widely used in image compression to generate high-quality images from a compact latent space.

Convolutional Neural Networks (CNN): CNNs are neural network architectures specifically designed for processing visual data.

They are widely used in image compression to extract relevant features from images and reduce spatial redundancy.

Transform Coding Networks: These networks use transform layers to extract compressed representations of images, which are then decoded to reconstruct the original images. Transform Coding Networks can be trained to learn adaptive transformations to input data, enabling more efficient compression.

Generative Adversarial Networks (GAN): GANs are neural network architectures composed of two adversarial networks, a generator and a discriminator, which are trained concurrently to generate realistic data. In the context of image compression, GANs can be used to generate high-quality compressed images from a latent representation.

These Deep Learning algorithms are often combined with quantification, entropy coding and optimization techniques to form complete Deep Learning image compression systems. They have demonstrated superior performance to traditional methods across a wide range of quality metrics and compression ratios.

IV.3. Explanation of the different steps adopted to design the image compression python code:

1. **Image Preprocessing:** Before compressing images, we perform preprocessing operations such as resizing and normalizing pixel values.
2. **Choose a compression method:** There are several image compression methods, the traditional method and the Deep Learning method, we chose to use both methods.
3. **Implement the compression algorithm:** Transform coding is a lossy image compression algorithm that often uses a technique called Discrete Cosine Transform (DCT).

4. **Evaluate performance:** After implementing the compression algorithm, we evaluated its performance in terms of compression rate and image quality. then we compared the compressed images with the original images.

Figure IV.1: The original image before compression .

Using Python on Google colab and Google drive, here is how the code works:

1. The code imports the images from My drive, first we mount the drive using drive. Mount then we indicate the path of the image in Google drive using image path.
2. It opens the original image using Image.Open('imagepath') and retrieves its dimensions with img.size.
3. It displays the original size of the image in KB using os.path.getsize to get the file size in bytes.
4. It resizes the image for compression using img.resize where newsize is the new desired size.
5. It saves the compressed image with compressed_img.save.Quality sets the compression quality (between 1 and 95).
6. It displays the size of the compressed image in KB using os.path.getsize again.

We can also control the quality of compressed images

1. **High quality:**

```
quality = 95  # La qualité varie de 1 (la plus basse) à 95 (la plus élevée)
image.save(compressed_image_path, quality=quality)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
Taille image originale: 61 Ko
Taille image compréssée: 33 Ko
Taux de compression: 1.8610355834022934
```

Figure IV.2: High quality compressed image

High quality compressed image characteristic

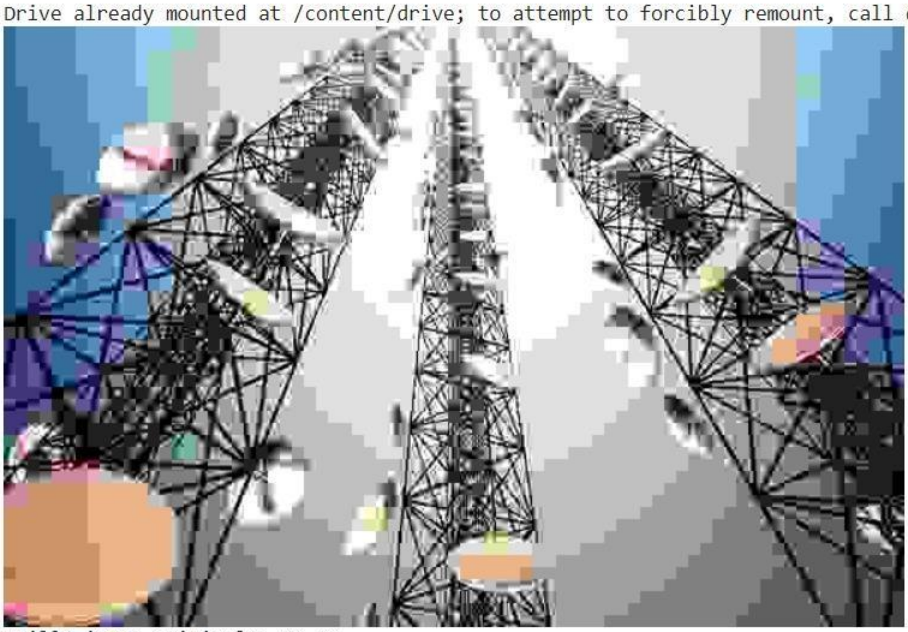
Image compression aims to reduce the size of image files while preserving visual quality as much as possible. A high-quality compressed image has several key characteristics: High Compression Rate, Preserved Visual Quality, Color Fidelity, High Signal-to-Noise Ratio (SNR), Computational Efficiency.

In summary, a high-quality compressed image achieves an optimal balance of reducing file size and preserving visual quality, while being computationally efficient and adaptable to various needs and types of visual content.

2. low quality:

```
quality = 1  # La qualité varie de 1 (la plus basse) à 95 (la plus élevée)
image.save(compressed_image_path, quality=quality)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/d
```



```
Taille image originale: 61 Ko
Taille image compréssée: 9 Ko
Taux de compression: 6.828977334345516
```

Figure IV.3: Low quality compressed image

Disadvantages of low-quality image compression

Low-quality compressed images have several disadvantages that can hinder their use in various applications. Here are the main disadvantages associated with poor quality compression: Loss of Detail, Compression Artifacts, Color Degradation, Signal-to-Noise Ratio (SNR) Reduction, Poor Quality Printing, Impact on Computer Vision Algorithms.

In summary, although low-quality compression can reduce file sizes, it has many disadvantages that compromise the visual quality and usefulness of images in various applications. For many professional and mission-critical uses, maintaining high image quality is essential, despite potentially higher costs in terms of storage and bandwidth.

IV.4. Load the dataset: we loaded the CIFAR-10 image dataset from Kaggle.

Figure IV.4: the classes in the CIFAR-10 dataset

The CIFAR-10 dataset is a widely used image set for machine learning and computer vision algorithms. It was developed by researchers at the CIFAR institute and consists of 60,000 32x32 color images divided into 10 different classes.

Main Features:

• The CIFAR-10 dataset consists of 60,000 images, divided into 10 classes.

• Each class contains 6000 images, divided into 5000 for training and 1000 for testing.

• The images are colored and have a size of 32x32 pixels.

• The 10 different classes represent planes, cars, birds, cats, deer, dogs, frogs, horses, boats and trucks.

Structure of the dataset:

• The dataset is divided into five preparation groups and one test group, each containing 10,000 images.

• Staging groups contain the rest of the images from arbitrary queries, but some staging groups may contain a greater number of images from one class than another.

Use:

• The CIFAR-10 dataset is commonly used for training and testing in machine learning and computer vision.

• It is often used to train convolutional neural networks (CNN) for image classification tasks, as in the sample code provided.

Results:

• The results obtained with the CIFAR-10 dataset vary depending on the algorithm used and the hyperparameters adjusted. Here are some examples of results:

• Test error of 18% without information growth and 11% with.

• 15% test error (without information augmentation) using Bayesian hyperparameter engineering.

In summary, the CIFAR-10 dataset is a widely used image set for machine learning and computer vision algorithms. It is divided into five preparation groups and one test group, and is commonly used to train convolutional neural networks for image classification tasks.

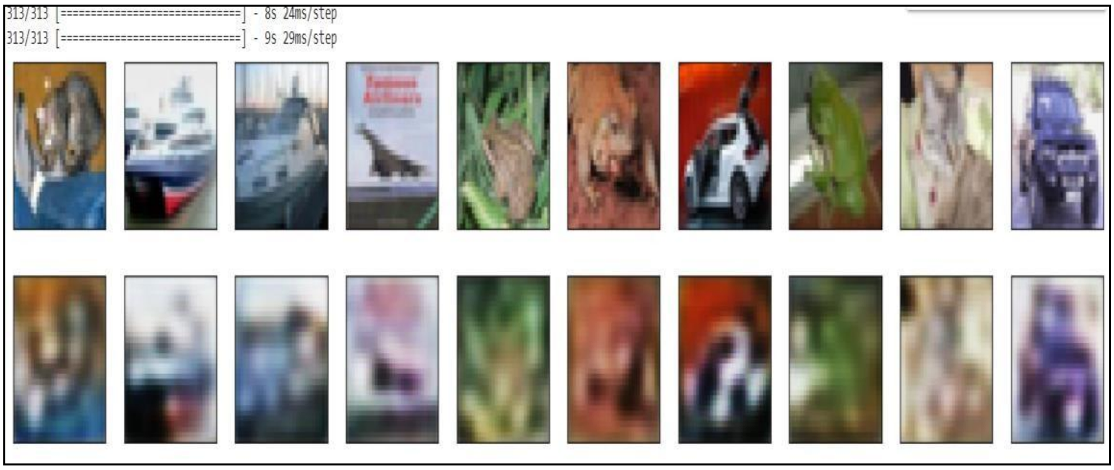


Figure IV.5: results of the second python code using CIFAR-10 dataset

This code defines a CNN model to classify images from the CIFAR-10 dataset, then trains and deepens the model. Then it compresses the images by converting the pixels into binary values (0 or 1) and displaying the original and compressed images.

Our code also displays the entropy in each iteration.

```
Epoch 1/10
196/196 [==============================] - 173s 871ms/step - loss: 0.5902 - val_loss: 0.5704
Epoch 2/10
196/196 [==============================] - 167s 854ms/step - loss: 0.5667 - val_loss: 0.5649
Epoch 3/10
196/196 [==============================] - 167s 851ms/step - loss: 0.5631 - val_loss: 0.5630
Epoch 4/10
196/196 [==============================] - 172s 881ms/step - loss: 0.5615 - val_loss: 0.5617
Epoch 5/10
196/196 [==============================] - 169s 864ms/step - loss: 0.5606 - val_loss: 0.5609
Epoch 6/10
196/196 [==============================] - 167s 854ms/step - loss: 0.5599 - val_loss: 0.5604
Epoch 7/10
196/196 [==============================] - 170s 866ms/step - loss: 0.5593 - val_loss: 0.5608
Epoch 8/10
196/196 [==============================] - 175s 892ms/step - loss: 0.5589 - val_loss: 0.5593
Epoch 9/10
196/196 [==============================] - 165s 842ms/step - loss: 0.5585 - val_loss: 0.5594
Epoch 10/10
196/196 [==============================] - 167s 852ms/step - loss: 0.5582 - val_loss: 0.5590
```

Figure IV.6: entropy display

With these figures we made a histogram and a curve "entropy as a function of iteration".

1. The histogram:

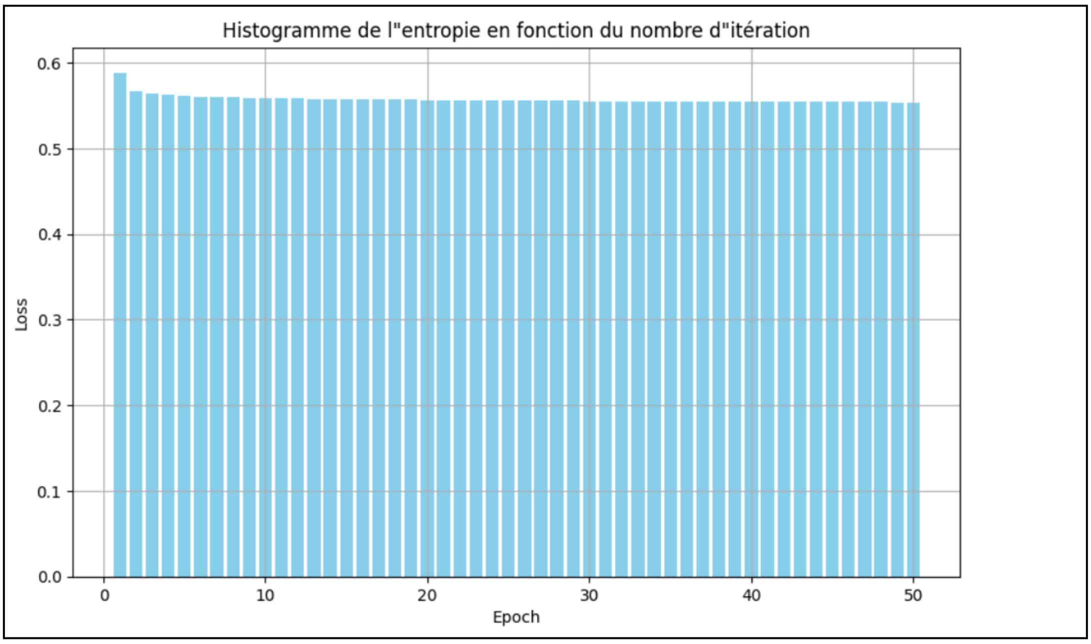


Figure IV.7: The histogram of the entropy as a function of the iteration

2. The Curve:

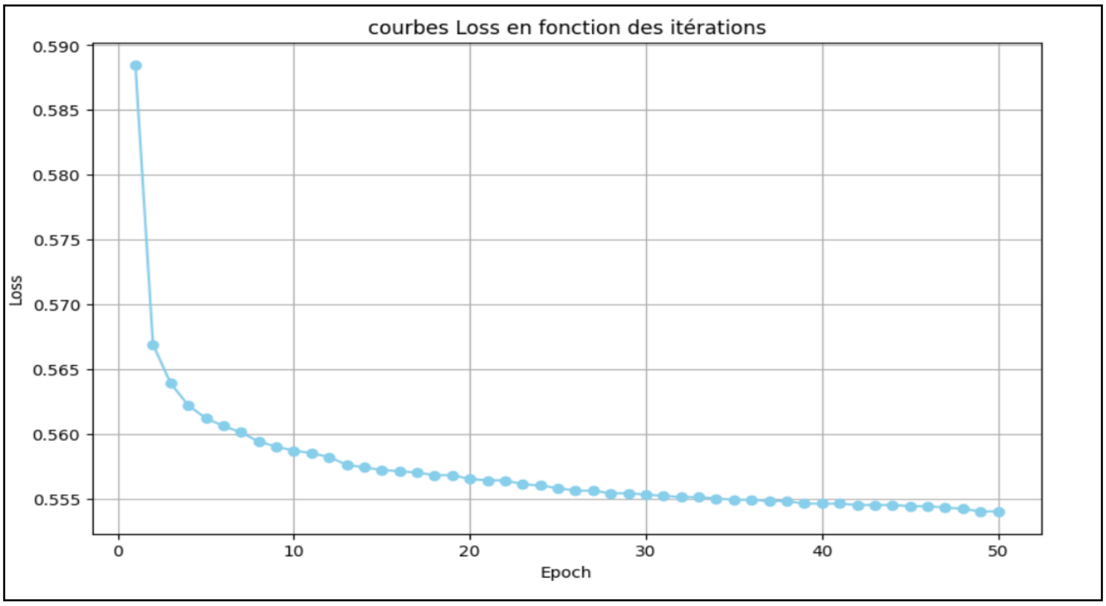


Figure IV.8: The curve of the entropy as a function of the iteration

We notice in the two graphs above that there is a certain stability of the LOSS from a certain number of iterations. We also notice that these values are larger at lower iterations. Our experiments have proven that a number of iterations equal to 50 is the best compromise to have the best results with an acceptable time.

IV.4. Conclusion:

Image compression via deep learning is an effective technique for reducing the size of images while maintaining their quality. Deep neural networks can be used to learn a single transform that is efficient in terms of rate-distortion tradeoff at different bit rates.

General Conclusion

General Conclusion

Visual content compression is an active and evolving area of research, with many applications in areas such as multimedia data transmission and storage. Deep learning techniques have recently shown their potential to improve compression efficiency, by learning compact representations of visual data directly from the data.

This master's thesis explored image compression techniques using deep learning, implementing these techniques with Python. Image compression is a critical area due to the exponential increase in visual data produced and shared daily. Traditional compression methods, although effective, are beginning to show their limits in the face of current requirements in terms of image quality and reduction of storage and transmission costs.

The application of deep learning, particularly convolutional neural networks (CNN) and autoencoders, to image compression has shown promising results. These models are capable of learning compact and efficient representations of images, often outperforming traditional compression techniques. We implemented and evaluated several neural network architectures using Python libraries such as TensorFlow and PyTorch, analyzing their performance in terms of compression ratio, reconstructed image quality, and computational complexity.

The results of our experiments show that deep learning-based compression models provide high compression rates while maintaining acceptable visual quality. Autoencoders, in particular, have been shown to be effective in capturing essential features of images and reproducing them with minimal loss of information. However, these models require significant computing power for training and can be difficult to deploy in resource-constrained environments.

Despite the challenges, our study confirmed the potential of deep learning-based compression techniques.

These techniques represent a notable advancement over traditional methods, providing better performance in various application scenarios. However, further efforts are needed to improve the computational efficiency, robustness and generalizability of the models to make them more practical for widespread use.

Speaking of Entropy in image compression is a fundamental concept in information theory and data compression. Entropy is defined as the measure of uncertainty or randomness in a probability distribution. In the context of image compression, entropy measures the amount of information contained in an image.

In conclusion, this work paves the way for exciting new prospects for deep learning-based visual content compression, with many promising applications in communications, multimedia, and virtual reality.

Bibliographicreferences

Bibliographic references

[1] https://dl.acm.org/doi/10.1145/3474085.3478877 ,15january 2024.

[2] https://www.nowpublishers.com/article/Details/CGV-101,march 2024

[3] https://arxiv.org/pdf/2202.06533
https://link.springer.com/article/10.1007/s42979-021-00815-1,december2023.

[4] https://medium.com/deelvin-machine-learning/image-compression-convolutional-neural-networks-vs-png-4d19729f851c,15january2024.

[5] https://www.researchgate.net/publication/357037859_Image_Compression_Using_Neural_Networks_A_Review ,2023

[6] https://www.computersciencecafe.com/23-data-transmission-principles.html,april 2024.

[7] https://www.microscopyu.com/digital-imaging/fundamentals-of-digital-imaging ,04november2024.

[8] https://www.sciencedirect.com/science/article/pii/S0011853222013003/pdf?md5=7af8fae825b3944d0e3b498da7af83be&pid=1-s2.0-S0011853222013003-main.pdf ,15march 2024.

[9] Un système de compression fractale d'images basé sur les réseaux de neurones profonds par Hana Djeddi (2020) [Université laarbitebessitebessa]

[10] https://www.nature.com/articles/s41598-021-86726-w
https://www.cloudflare.com/learning/performance/glossary/what-is-image-compression/ ,February 2024

[11] https://www.techtarget.com/whatis/definition/image-compression ,2024

[12] https://en.wikipedia.org/wiki/Image_compression ,october 2023

[13] https://www.khanacademy.org/computing/computers-and-internet/xcae6f4a7ff015e7d:digital-information/xcae6f4a7ff015e7d:data-compression/a/simple-image-compression ,1may 2024.

[14] Barouni, Ala Eddine. "Drawing algorithms for survivable telecommunication networks." Thèse, Bibliothèque nationale du Canada, 1997.

[15] Techniques de compression des images par Jean-Pierre Guillois (1996)

[16] Fractal Image Compression par Michael F. Barnsley et Lyman P. Hurd (1993)

[17] https://fritz.ai/how-deep-learning-solves-compression-challenges/ , September 2023

[18] https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8 ,December 2023.

[19] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10607786/ , 5october 2023

[20] https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063, 21 April 2024

[21] Nguyen, Thi Thu Thuy. "A novel approach for practical real-time, machine learning based ip traffic classification." Swinburne Research Bank, 2009.

[22] https://www.quora.com/What-is-the-different-between-using-deep-learning-machine-learning-or-image-processing-to-classify-an-image

[23] Djeddi, Hana. "Un système de compression fractale d'images basé sur les réseaux de neurones profonds." Universitélaarbitebessitebessa, 2020.

[24] https://www.aiacceleratorinstitute.com/top-6-deep-learning-trends-in-2022/ ,June 2024

[25] https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063 , May 2024.

[26] Bousmaha, R., Hamou, R. M., & Amine, A. (2022). Automatic selection of hidden neurons and weights in neural networks for data classification using hybrid particle swarm optimization, multi-verse optimization based on Levy flight. Evolutionary Intelligence, 15(3), 1695-1714.

[27] Bousmaha, R., Hamou, R. M., & Amine, A. (2022). Optimizing Connection Weights in Neural Networks Using Hybrid Met heuristics Algorithms. International Journal of Information Retrieval Research (IJIRR), 12(1), 1-21.