الجمهورية الجزائرية الديمقراطية الشعبية
DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA
وزارة التعليم العالي و البحث العلمي
Ministry of Higher Education and Scientific Research
جامعة أبي بكر بلقايد تلمسان –
Abou Bekr Belkaïd University – Tlemcen –
Faculty of TECHNOLOGY

Final Year Project Thesis

Submitted in Partial Fulfillment of the Requirements for the Degree of MASTER of science in BIOMEDICAL ENGINEERING

Speciality : Medical Imaging

Presented by : Cheriet Malek Amina

Thesis Title

MRI IMAGE RECONSTRUCTION USING DEEP LEARNING

Mr. Djebari Abd el ghani	Pr	Tlemcen university	president
Ms. Belaidi Asma	MCB	Tlemcen university	Examiner
Ms. Iles Amel	MCA	Tlemcen university	Supervisor

Academic year 2024-2025

Dedicate

To my dearest parents, source of life, love, and affection

To my dear sisters Fatima, Nour, Nihad and Hanaa

To all my family, source of hope and motivation

To all my friends, especially Dyma

To you, dear reader

Acknowledgement

First and foremost, I praise and thank Allah, the Almighty, for His countless blessings, guidance, and strength that enabled me to complete this work successfully.

I also take a moment to acknowledge my own journey, the late nights, the challenges overcome, and the unwavering faith that carried me forward. This milestone is not just an academic accomplishment but a testament to resilience, personal growth, and trust in Allah's plan.

My deepest and most heartfelt gratitude goes to my esteemed supervisor **Ms. *Iles Amel,*** whose wisdom, patience, and unwavering support have been the cornerstone of this work.

I extend my sincere appreciation to the members of the examination committee, ***Mr. Djebari Abd El Ghani*** and ***Ms. Belaidi Asma***, for generously dedicating their time and expertise to evaluate my thesis.

My heartfelt thanks go to all members of the research laboratory of Tlemcen university for their warm welcome and excellent treatment during my research period. I am particularly grateful to the laboratory supervisor, ***Ms. Kholkhal Shahra***, for her support and encouragement.

No words can fully express my gratitude to my beloved family my parents and sisters whose unconditional love, endless encouragement, and steadfast belief in me have been my greatest source of strength.

I would like to thank my dear friends for their continuous support and encouragement. Special recognition goes to ***Laneg Dyma***, ***Mekkaoui Takoua***,, ***Dilmi Hadjer*** and ***Briki Nada*** for their friendship and assistance during challenging times.

Finally, to everyone who has touched my life, whether in small or significant ways, I carry your influence with gratitude. May this work be a reflection of the kindness, knowledge, and support I have been blessed to receive.

Abstract

Recent MRI reconstruction methods like SENSE, GRAPPA, and Compressed Sensing (CS) allow for fast acquisition of high-quality images. However, these methods can remain susceptible to folding artifacts that significantly degrade reconstruction quality. To address this problem, we propose a new approach that combines Graph Neural Networks (GNN) with CS to improve reconstruction while preserving anatomical details. Due to initial results of the proposed method and time constraints, we implemented a hybrid post-processing pipeline using CNN, GNN, and CS. Our results show that using CNN for correction cannot fully address the artifacts caused by acquisition. This highlights that significant quality improvements need to come from better initial reconstruction.

Keywords: MRI, MRI reconstruction, Compressed sensing, K_space, Deep Learning, Convolutional Neural Network (CNN), Graph Neural Network (GNN), Graph Convolutional network (GCN).

Résumé

Les méthodes récentes de reconstruction d'images IRM, telles que SENSE, GRAPPA et le Compressed Sensing (CS), permettent d'obtenir des images de bonne qualité en un temps réduit. Cependant, elles souffrent d'artefacts de repliement, qui altèrent significativement la qualité des reconstructions. Pour résoudre ce problème, nous proposons une approche innovante basée sur les Graph Neural Networks (GNN), intégrée au CS, afin d'améliorer la reconstruction tout en préservant les détails anatomiques. En raison des résultats obtenus et des contraintes de temps et pour optimiser davantage notre méthode, nous avons également implémenté un post-traitement hybride (CNN + GNN + CS). Cependant, les résultats montrent que le CNN seul ne peut pas corriger les artefacts induits lors de l'acquisition, confirmant ainsi que l'amélioration doit d'abord passer par une reconstruction initiale de meilleure qualité.

Mots-clés : IRM, reconstruction IRM, détection compressée, K_space, apprentissage profond, réseau neuronal convolutif (CNN), réseau neuronal graphique (GNN), réseau convolutif graphique (GCN).

الملخص

توفر طرق إعادة بناء صور التصوير بالرنين المغناطيسي الحديثة، مثل SENSE و GRAPPA والاستشعار المضغوط (CS)، صوراً ذات جودة عالية في وقت قصير. ومع ذلك، فإنها تعاني من التشويش التعريفي الذي يغير بشكل كبير من جودة عمليات إعادة البناء. ولحل هذه المشكلة، نقترح نهجاً مبتكراً يعتمد على شبكات الرسم البياني العصبية(GNN) ، مدمجاً مع الاستشعار المضغوط(CS) ، لتحسين إعادة البناء مع الحفاظ على التفاصيل التشريحية. ونظرًا للنتائج التي تم الحصول عليها وضيق الوقت، ولتحسين طريقتنا، قمنا أيضًا بتنفيذ المعالجة اللاحقة الهجينة (شبكة سي إن إن + شبكة الشبكات العصبية البيانية .(CS + ومع ذلك، تُظهر النتائج أن شبكة CNN وحدها لا يمكنها تصحيح التشوهات التي تحدث أثناء الاستحواذ، مما يؤكد أن التحسين يجب أن يتضمن أولاً إعادة بناء أولية بجودة أفضل.

الكلمات المفتاحية: التصوير بالرنين المغناطيسي، إعادة بناء التصوير بالرنين المغناطيسي، الاستشعار المضغوط، مساحة K، التعلم العميق، الشبكة العصبية التلافيفية (CNN)، الشبكة العصبية الرسومية (GNN)، الشبكة التلافيفية الرسومية (GCN).

Table of content

Table of Figures

Table of Tables

List of abbreviations and symbols

Symbol	Abbreviation
MRI	Magnetic Resonance Imaging
B0	static magnetic field of intensity B0
B1	radio frequency electromagnetic field
T1	Longitudinal relaxation time
T2	Tranverse relaxation time
TE	Echo time
TR	Repetition time
Grappa	Generalized Auto calibrating Partially Parallel acquisitions
Sense	Sensitivity Encoding
CS	Compressed Sensing
DL	Deep Learning
CNN	Convolutional Neural Network
GNN	Graph Neural Network
Relu	Rectified Linear Unit Layer
LM	Longitudinal magnetization
TM	Transverse Magnetization
IR	Inversion Recovery

PD	Proton Density
IFT	Inverse Fourier Transform
GCN	Graph Convolutional Network
GAT	Graph Attention Network

General Introduction

Magnetic Resonance Imaging (MRI) is one of the most powerful and widely used medical imaging technologies today. Unlike X-rays or CT scans, MRI does not rely on ionizing radiation, making it a safer option for repeated use. Instead, it uses strong magnetic fields and radio waves to produce highly detailed images of soft tissues, organs, and other internal structures. Due to its exceptional clarity and versatility, MRI plays a crucial role in diagnosing diseases, monitoring treatment progress, and advancing medical research.

Despite its advantages, MRI has one significant limitation: it is a slow imaging technique. A single scan sequence can take 30 minutes or longer, depending on the body part being examined. This extended scanning time creates several challenges. For patients, lying still inside the MRI machine for prolonged periods can be uncomfortable, especially for children, elderly individuals, or those with conditions that cause pain or involuntary movements. Even slight motion can introduce artifacts into the images, reducing their diagnostic quality. For healthcare providers, long scan times limit the number of patients that can be examined in a day, increasing wait times and reducing efficiency.

To address this issue, researchers have focused on accelerating MRI data acquisition. The scan duration is primarily determined by the number of measurements taken in k-space (the frequency domain where MRI raw data is collected). Traditional acceleration techniques reduce scan time by acquiring fewer measurements—essentially skipping some of the data points. However, this undersampling can result in aliasing artifacts, which distort the final image and make it harder to interpret

Over the years, several computational methods have been developed to reconstruct high-quality images from undersampled MRI data. Compressed Sensing (CS) is one such approach, which leverages the idea that medical images are often "sparse" and can be accurately reconstructed from limited data. Another method, parallel imaging, uses multiple receiver coils to gather additional spatial information, allowing for faster scans. While these techniques have significantly improved MRI efficiency, they still face challenges in balancing speed and accuracy.

More recently, deep learning has emerged as a groundbreaking tool for MRI reconstruction. Artificial intelligence, particularly Convolutional Neural Networks (CNNs), excels at recognizing patterns in complex data, making it well-suited for processing medical images. By training on vast datasets of MRI scans, deep learning models can learn to predict missing details, suppress noise, and reconstruct high-fidelity images even from heavily under sampled data. This not only speeds up the scanning process but also enhances image quality, leading to more reliable diagnoses.

As MRI technology continues to evolve, the integration of AI-driven reconstruction methods promises to make scans faster, more accessible, and more comfortable for patients—without compromising diagnostic precision. This advancement could revolutionize clinical workflows, reduce healthcare costs, and improve patient outcomes in the years to come.

Our thesis is organized as follow:

Chapter 1: Magnetic Resonance Imaging

This chapter covers the physical principles of MRI, instrumentation details including magnets and RF coils, and reconstruction methods ranging from classical Fourier transforms to advanced compressed sensing techniques.

Chapter 2: Compressed Sensing and Deep Learning

This chapter explores compressed sensing in detail and examines deep learning approaches, specifically CNNs and GNNs, highlighting their differences. The focus is on demonstrating how integrating deep learning with compressed sensing can significantly improve image reconstruction quality.

Chapter 3: CS-GNN Application

This chapter presents the step-by-step integration of compressed sensing into graph neural networks for MRI reconstruction. Due to initial performance limitations, an additional CNN-based post-processing step was implemented to enhance results, with comprehensive materials, methods, results, and discussion provided.

Chapter 01:Magnetic Resonance Imaging MRI

Introduction

Magnetic Resonance Imaging (MRI) is a widely utilized medical imaging technique that produces high-resolution images of both anatomical structures and physiological processes. It operates using strong uniform magnetic fields (B0), and radio waves (B1). Due to its non-invasive nature and exceptional ability to distinguish soft tissues, MRI has become an indispensable tool in contemporary diagnostic medicine [1].

The chapter is organized into three principal parts:

- The Physical Principles of MRI**, covering the foundational physics underlying magnetic resonance imaging.**
- MRI Instrumentation**, detailing the hardware components (e.g., magnets, gradients, RF coils) and their roles.**
- MRI Reconstruction Methods**, including classical and advanced techniques (e.g., Fourier transforms, compressed sensing).**

I. The physical principles of MRI

I.1. MRI fundamentals

The core principle behind Magnetic Resonance Imaging (MRI) relies on *magnetization*. To fully grasp this concept, it is essential to first explore the foundational elements that contribute to it: **magnetism, electromagnetism, nuclear spin, alignment, and precession.** These physical phenomena govern the behavior of atomic nuclei in a magnetic field, forming the basis of MRI signal generation. In the following sections, we will examine each of these key aspects in detail.

• I.1.1. Magnetism and electromagnetism

Magnetism is a natural phenomenon caused by the movement of electric charges, which creates forces that can either pull objects together or push them apart.

Electromagnetism, on the other hand, describes the relationship between electricity and magnetism, when electric charges move, they create magnetic fields (as described by Ampere's Law), and when magnetic fields change, they can produce electric currents (as explained by Faraday's Law)[2].

• I.1.2. nuclear spin

Atoms are composed of three key particles: protons, neutrons and electrons.

Protons and Neutrons: These reside in the atom's nucleus. Protons carry a positive charge, while neutrons are electrically neutral.

Electrons: are negatively charged that orbit the nucleus in distinct energy shells.

In MRI, the behavior of the nucleus is what matters. Both protons and neutrons spin on their own axes, either clockwise or counterclockwise. When a nucleus has an even mass number, these spins balance out, resulting in no overall spin. But if the mass number is odd, the spins don't cancel, leaving the nucleus with a net spin [3]. MRI specifically targets atoms with odd mass numbers, such as Hydrogen-1, Carbon-13, and Fluorine-19.

Hydrogen is especially useful in MRI because it's abundant in the human body (making up about 70%) and has a strong magnetic moment.

As protons spin, their movement generates a tiny magnetic field, essentially turning each one into a microscopic bar magnet. This magnetic field can be described as a vector known as the magnetic moment which has both strength and a direction [3].

- **I.1.3. Alignment and precession**

In a typical environment, the magnetic moments of protons are oriented randomly (see Figure 1.1). However, when an external magnetic field such as that produced by an MRI magnet is applied, the protons align and spin in the direction of the field. This alignment can occur in two ways: parallel, which aligns with the field, or antiparallel, which opposes it, this effect is known as Zeeman splitting. The parallel alignment represents a lower energy state, making it the favored option, leading to a greater number of protons aligning in that direction. Once aligned, the protons engage in precession, a wobbling motion that occurs at a specific frequency referred to as the precession frequency. The speed of this precessional motion is directly influenced by the strength of the magnetic field [3, 4].

The stronger the field, the faster the protons precess. This relationship is encapsulated by the Larmor equation,

$$\omega = \gamma\beta \qquad (1)$$

Where : ω : the precession frequency (in Hz or MHz), β : the strength of the external magnetic field, which is given in Tesla (T), γ : the so-called gyromagnetic ratio (This gyro-magnetic ratio is different for different materials (e.g. the value for protons is 42.5 MHz/T).).

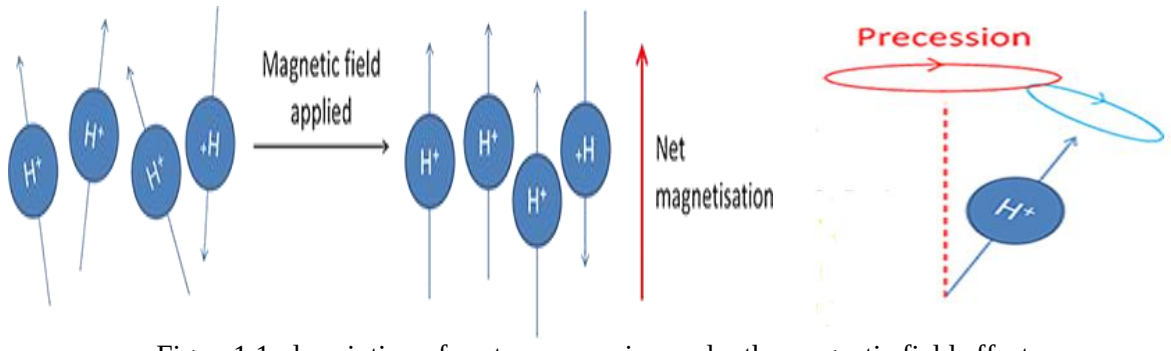


Figure1.1: description of proton precession under the magnetic field effects

I.2. Longitudinal magnetization

When an external magnetic field (B0) is applied, protons align either parallel (lower energy) or antiparallel (higher energy) to this field. This orientation can be described using a three-

dimensional coordinate system with X, Y, and Z axes. The external magnetic field is directed along the Z-axis, which is conventionally considered the long axis of the patient [5].

Since lower energy states are preferred, there is a slight predominance of protons that align parallel to the field compared to those that align antiparallel. Although the forces exerted by the parallel and antiparallel protons counterbalance each other, the surplus of parallel protons generates a net magnetic force along the Z-axis (see Figure 1.2), known as longitudinal magnetization[4, 5]. However, this longitudinal magnetization cannot be measured directly. To obtain a measurable signal, the magnetization must be shifted into the transverse plane (X-Y plane), where it can be detected.

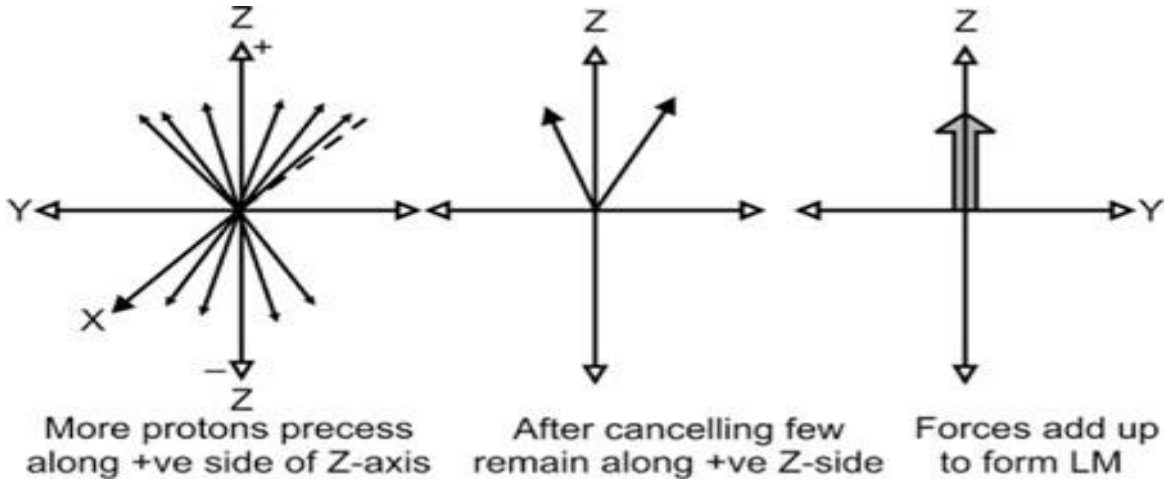


Figure 1.2: Longitudinal Magnetization

I.3. Transverse magnetization

Since direct measurement of magnetization along the Z-axis is not possible, we use radiofrequency (RF) pulses to change its direction. When the RF pulse is applied, two simultaneous events happen [5]:

1. Energy Absorption: Precessing protons absorb energy from the RF pulses and move to a higher energy state, which causes some protons to transition to the antiparallel configuration (aligned along the negative Z-axis). This leads in a decrease in longitudinal magnetization along the Z-axis and the development of a new magnetization in the transverse (X-Y) plane, known as transverse magnetization(see Figure 1.3).

2. Phase Coherence: The RF pulse creates phase coherence in the transverse plane by synchronizing protons to move in step with each other.

To allow this energy exchange to occur, the RF pulse frequency must match the protons' precession frequency, a phenomenon called as resonance (the "R" in MRI), which is the fundamental premise of MRI [5].

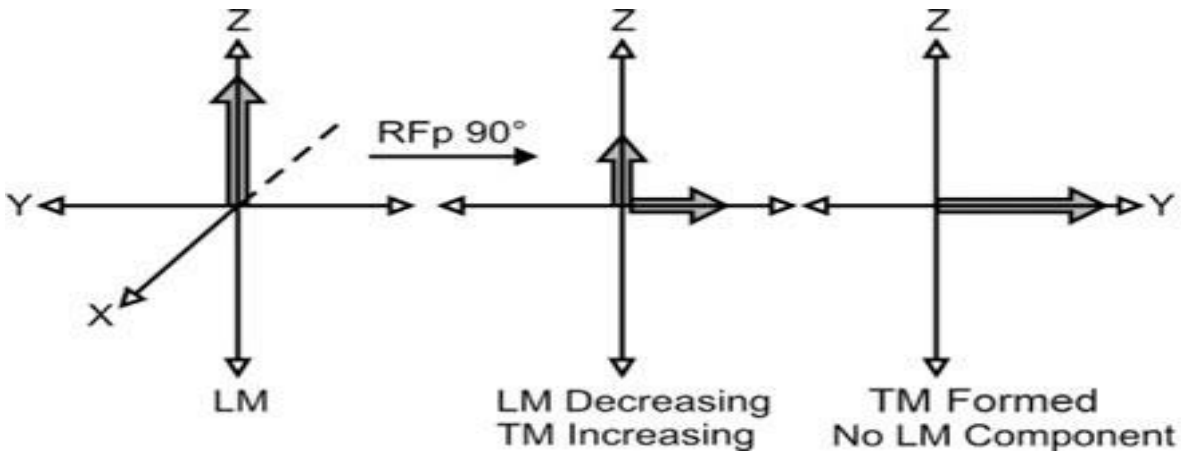


Figure 1.3: Transverse Magnetization

Once magnetization is established, the return of nuclear spins to their equilibrium state known as ***relaxation*** plays a crucial role in image formation. The following sections will explore these relaxation mechanisms in detail.

I.4. Relaxation phenomenon

Relaxation refers to the process through which protons return to their original state after being disturbed by radiofrequency stimulation. When the RF pulse is turned off, longitudinal magnetization (LM) begins to increase along the Z-axis, while transverse magnetization (TM) diminishes in the transverse plane. The recovery of LM is referred to as longitudinal relaxation, whereas the reduction in the amplitude of TM is known as transverse relaxation[5]

- **I.4.1. Longitudinal Relaxation**

When the RF pulse is turned off, protons release the absorbed energy and return to a lower energy state, aligning upwards again. This process increases longitudinal magnetization, which gradually returns to its original value. The released energy is transferred as thermal energy to the surrounding lattice, leading to the terms: Longitudinal relaxation or spin-lattice relaxation [2, 5].

Plotting longitudinal magnetization against time after the RF pulse results in a T1 recovery curve. The time it takes for longitudinal magnetization to recover to 63% of its original value is known as the longitudinal relaxation time, or T1. [5]

- **I.4.2. Transverse relaxation**

We previously noted that protons precess at a frequency determined by the strength of the magnetic field in which they are in. However, this statement requires some refinement. The magnetic field produced by the MR magnet, which envelops the patient (B0), is not entirely uniform; it exhibits slight variations that result in differing precession frequencies.

Furthermore, each proton is influenced by the small magnetic fields generated by neighboring nuclei, which also contributes to these variations. When the RF pulse is turned off, the protons are no longer forced to remain in sync. As each proton precesses at its own frequency, they quickly become out of phase, a phenomenon known as dephasing. As a result, the transverse magnetization begins to decrease gradually until it eventually disappears. Since this dephasing is related to the static or slowly fluctuating magnetic fields caused by adjacent spins (protons), transverse relaxation is also referred to as spin-spin relaxation. [2, 5]

Plotting transverse magnetization as a function of time after an RF pulse yields a T2 decay curve. The time required for the transverse magnetization to decay to 37% of its original value is known as the transverse relaxation time, or T2 (see Figure 1.4). [5]

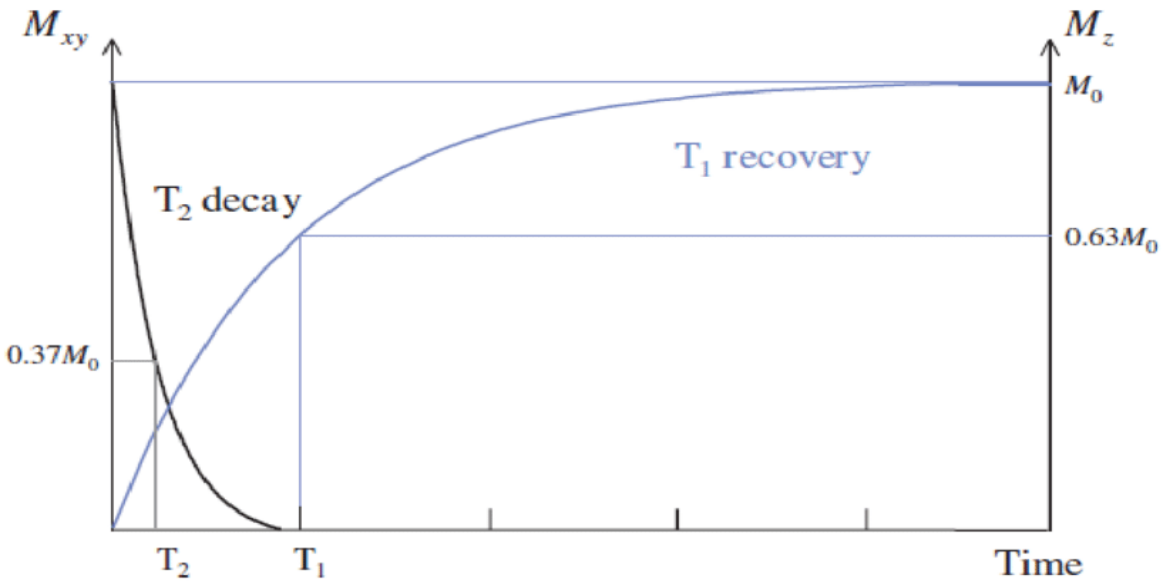


Figure 1.4: T1 recovery and T2 decay curves

The formation of an MRI image requires the conversion of this relaxing magnetization into measurable signals. The following sections detail how these signals are captured and processed for medical imaging.

I.5. MR signal generation

Once the RF pulse is turned off, the transverse magnetization decreases as the protons lose phase coherence, while the longitudinal magnetization gradually recovers. The net magnetization vector (NMV), formed by combining the longitudinal and transverse magnetization vectors gradually changes from the transverse X-Y plane to the vertical Z-axis in a spiral motion. According to Faraday's law, as the NMV rotates within the transverse plane, it induces an electric current in the receiver coil. This coil detects the current as a magnetic resonance signal (see Figure 1.5). The MR signal is then processed using advanced mathematical techniques, such as Fourier transformation, to create an image. [6]

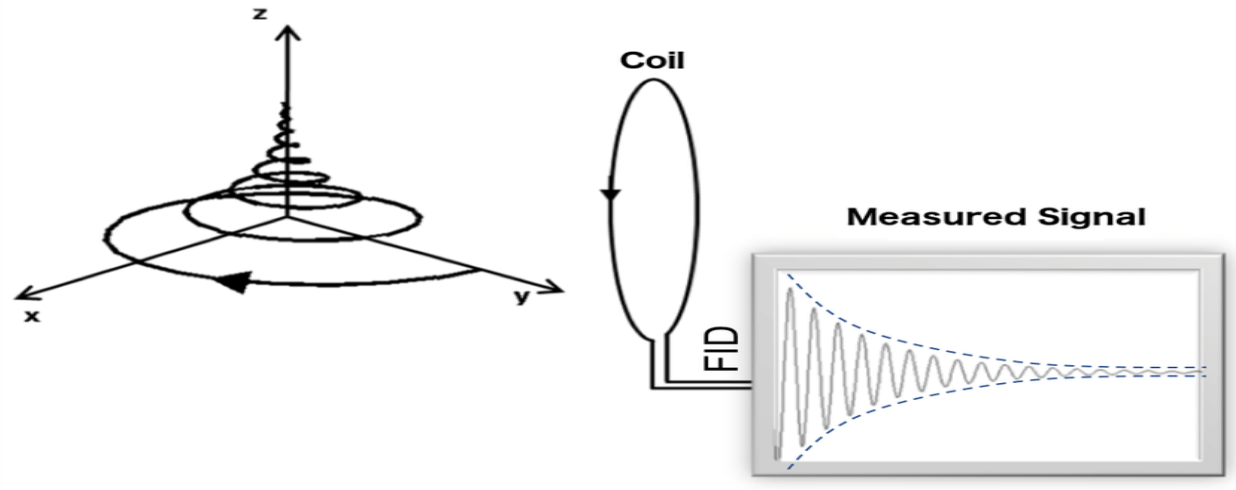


Figure 1.5: MRI signal detection

I.6. Image production

Gradient coils (see instrumentation section) are crucial for this process. First, the desired slice is selected. Next, spatial encoding comprising frequency encoding and phase encoding is applied to generate the image [5].

Gradients are cylindrical coils used to create linear variations in the main magnetic field. There are three sets of gradients: the Z gradient, the Y gradient, and the X gradient, which alter the magnetic field strength along the Z, Y, and X axes respectively. When a gradient is applied, nuclei within a specific slice location experience a unique magnetic field strength and thus a distinct precessional frequency; by transmitting an RF pulse matching this frequency, only spins in that slice resonate, enabling selective excitation. The Z-gradient selects axial slices (head vs. feet), the Y-gradient selects coronal slices (back vs. front), and the X-gradient selects sagittal slices (left vs. right), while combinations of two gradients produce oblique slices. This spatial encoding combined with frequency and phase encoding forms the basis of MRI image formation [2].

- ### I.6.1. Slice thickness

Slice thickness in MRI determines how 'thick' or 'thin' the image slices are (see Figure 1.6). Thinner slices show finer details but have weaker signal , while thicker slices provide stronger signal but can blur nearby tissues together (partial volume averaging). The thickness is controlled by two settings: the gradient strength (stronger gradients = thinner slices) and the RF pulse's bandwidth (narrower bandwidth = thinner slices). It’s essential to find a balance between resolution and signal quality based on the specific requirements of your scan. [2].

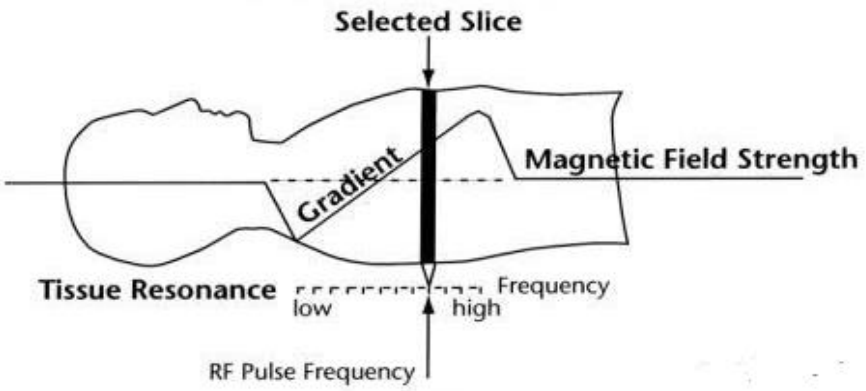


Figure 1.6: Slice selection

- **I.6.2. Phase encoding**

After a slice has been selected, the system has to use gradients to gain two-dimensional information representing the spatial location of the spins within the slice. The gradient changes the phase of the magnetic moment of each nucleus. Gradient along one axis of the image causes a phase shift of nuclei along the length of the gradient; this phase shift is used to locate the nuclei spatially along one dimension of the image. [2]

- **I.6.3. Frequency encoding**

The gradient causes a change in the frequency of nucleus; it produces a frequency change or frequency shift.

These phase and frequency informations are stored in a matrix called K_space.

- **I.6.4. K_space characteristics**

K-space is a spatial frequency domain (see Figure 1.7) used to store raw data acquired from the MR signal. . It represents a raw data matrix that contains information about the frequency and phase of the nuclei. K-space features two axes: the horizontal axis denotes the frequency, and the vertical axis represents phase. The number of lines filled in K-space is determined by the various phase encoding steps employed. K-space itself does not directly mirror the final image. Instead, it holds a Fourier representation of the image, where the central regions contain contrast information and the peripheral regions contribute to fine details. To reconstruct the final image, a two-dimensional Fourier transform (2D-FT) is applied to the K-space data, converting the frequency and phase information into a spatial domain image [5].

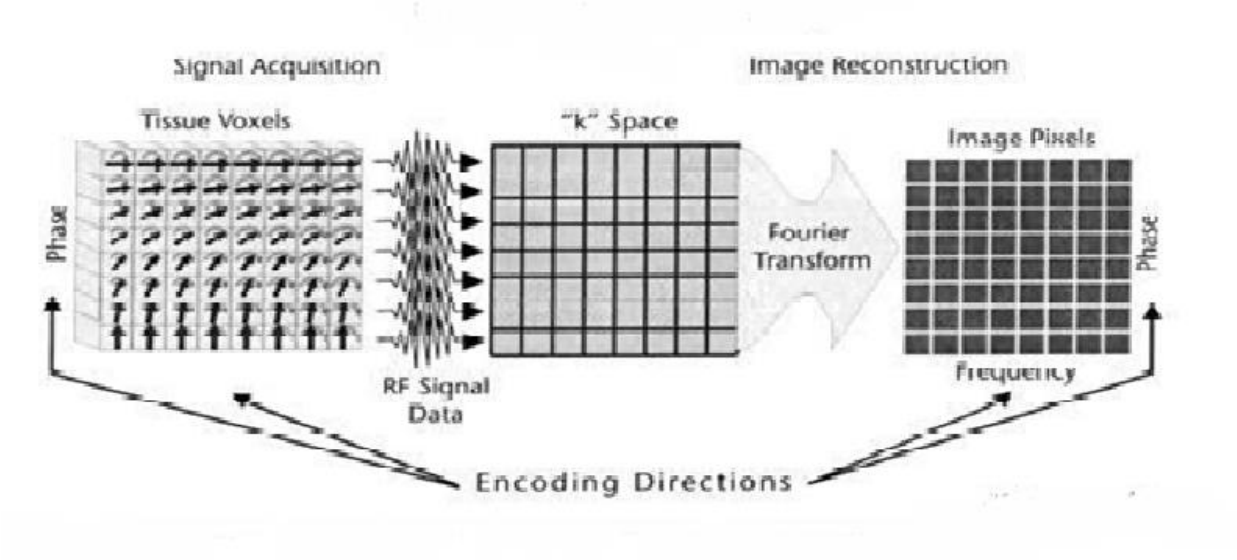


Figure 1.7: MRI image production

The successful generation of MRI images requires more than just signal acquisition, it demands careful protocol selection based on the specific clinical needs of each examination. These protocols are implemented through specialized pulse sequences that control image contrast and quality.

I.7. Sequences

In order to understand sequences, it's important to discuss two essential parameters: TR and TE.

TR, or Time to Repeat, refers to the interval between the start of one RF pulse and the beginning of the next.

TE, or Time to Echo, is the duration from the start of the RF pulse to the reception of the signal (echo).

A sequence is a synchronized series of radiofrequency (RF) pulses and magnetic field gradient used to recover the attenuated MR signal caused by spin dephasing after RF excitation, these sequences are characterized by two fundamental timing parameters: repetition time (TR) and echo time (TE). Pulse sequences are essentially divided into two categories: Spin Echo and Gradient Echo, with Inversion Recovery and Echo Planar Imaging representing their respective specialized versions. [2].

- **I.7.1. Spin Echo**

This process employs 90° and 180° radiofrequency (RF) pulses (see Figure 1.8) to compensate static field inhomogeneities (T2 effects). The initial 90° pulse shifts the net magnetization from the longitudinal (z-axis) into the transverse (XY) plane, generating a weak signal unsuitable for imaging (T2*). To amplify this signal, a 180° rephasing pulse is applied, which reverses proton dephasing and produces a coherent echo [9]. By repeating the two pulses (90°and 180°), a train of spin echoes can be produced, substantially improving

signal recovery. This approach is widely used in clinical MRI and serves as the foundation for T1_ , T2_ , and proton density-weighted imaging (see next section for more details).

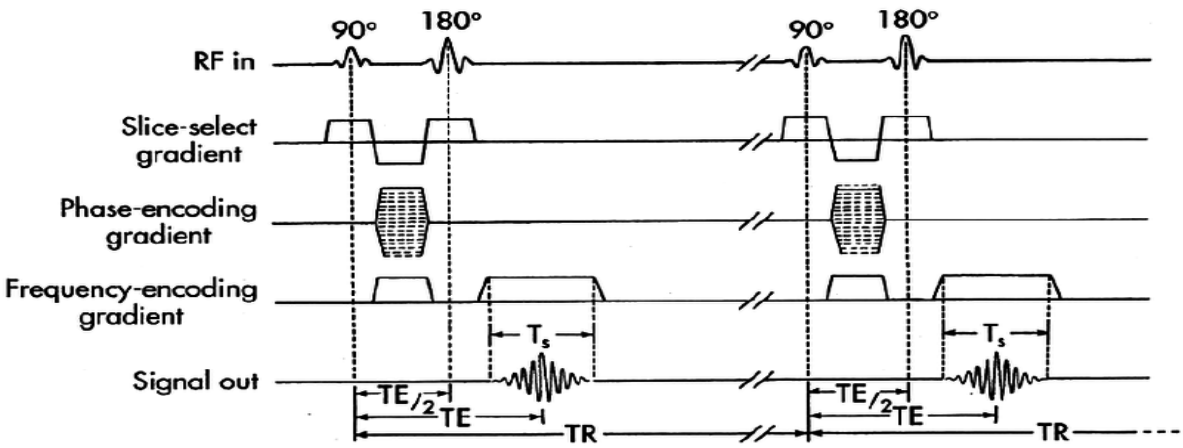


Figure 1.8: Spin Echo pulse sequence

Inversion recovery is a spin echo sequence preceded by a 180 degree pulse to invert the magnetization before the standard spin echo process (see Figure 1.9). The inversion 180 degree pulse flips LM along negative side of the Z-axis. This saturates fat and water completely at the beginning When 90 degree excitatory pulse is applied after LM has relaxed through the transverse plane, contrast in the image depends on the amount of longitudinal recovery of the tissues with different T1. IR sequence used to suppress particular tissue using different time inversion (TI) the waiting period between the initial 180° pulse and the 90° pulse [5].

This makes IR useful for two main purposes:

- **Creating high-contrast T1-weighted images to show anatomy clearly.**
- **Selectively removing unwanted signals (e.g., fat in STIR or fluid in FLAIR)**

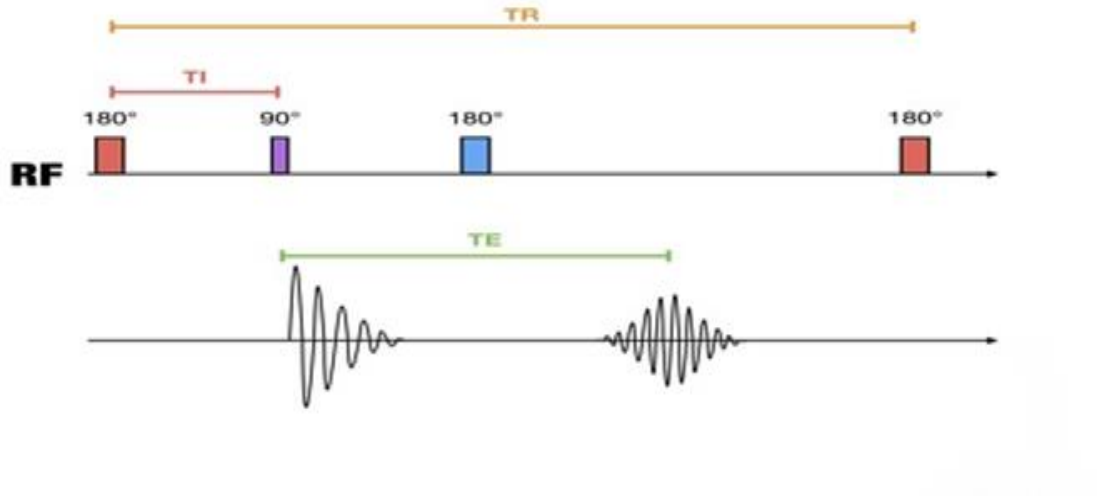


Figure 1.9: Inversion Recovery pulse sequence

- **I.7.2. Gradient Echo**

In gradient echo (GRE) sequences, spatial encoding gradients replace the 180° RF rephasing pulse used in spin echo techniques (see Figure 1.10). Unlike spin echo methods, GRE does not compensate for magnetic field inhomogeneity effects because the gradient-based rephasing cannot fully reverse T2* dephasing. Consequently, nuclei affected by static field imperfections remain partially dephased, making GRE images more susceptible to T2* artifacts from both intrinsic tissue properties and external magnetic field inhomogeneities [6]. However, gradient rephasing enables significantly faster signal recovery than RF-based methods, permitting much shorter TE and TR times. This temporal efficiency allows rapid image acquisition, making GRE sequences ideal for dynamic or time-sensitive clinical applications.

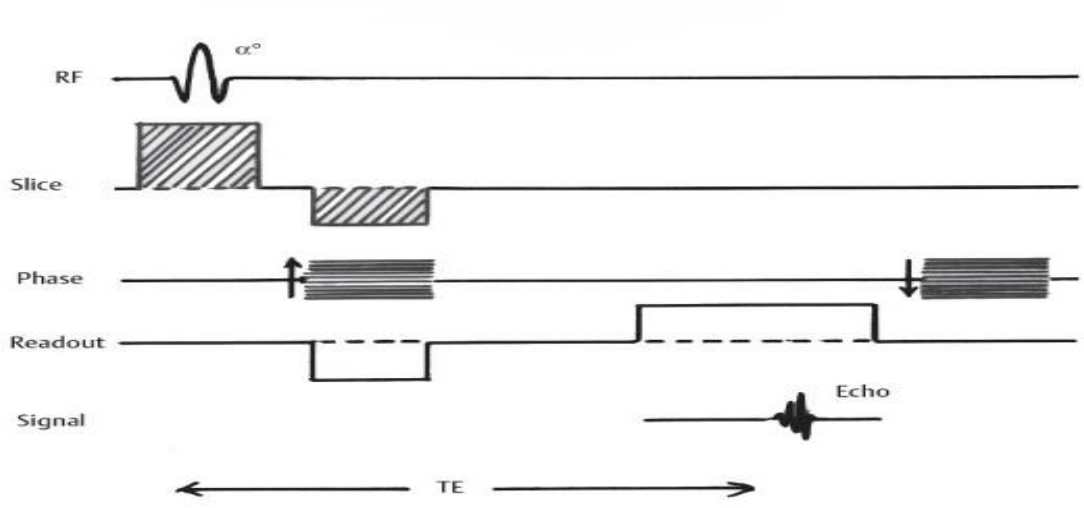


Figure 1.10: Gradient Echo pulse sequence

Echo planar imaging sequence: Multiple echoes generated in single TR in EPI (see Figure 1.11) , These multiple echoes are either generated by 180 degree rephasing pulses or by gradient , Therefore, EPI can be Spin Echo EPI (SE-EPI) or Gradient echo EPI (GRE-EPI)[6].

SE-EPI is not routinely used. GRE-EPI is very sensitive to susceptibility artifacts because T2* decay is not compensated in GRE sequences. Signal-to-noise ratio (SNR) is poor in single shot EPI [2]. These problems can be overcome to certain extent by the hybrid sequence. Hybrid sequence combines advantages of gradient (speed) and RF pulses (compensation of T2* effects). SNR can be improved by use of multi-shot EPI.

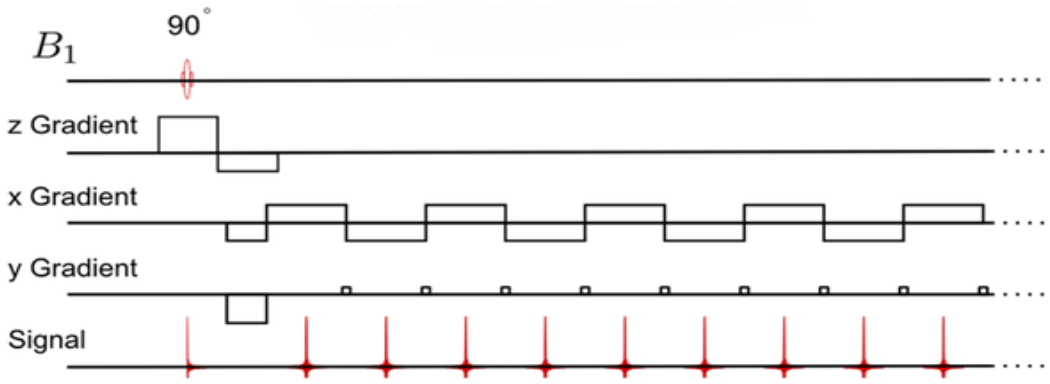


Figure 1.11: Echo Planar Imaging sequence

By strategically adjusting sequence parameters (TR, TE), we can generate different types of weighted images. Each weighting highlights distinct tissue properties, enabling radiologists to differentiate anatomical structures and identify pathological changes with greater precision.

Weighted images are characterized by two fundamental timing parameters: repetition time (TR) and echo time (TE) which have been previously discussed in the context of sequences.

There are three types of weighting images: T1 weighted images, T2 weighted images and proton density images [2].

- **T1 weighted image**

T1 relaxation is influenced by tissue composition, structure, and surrounding environment. The transfer of energy from protons to their surroundings occurs rapidly when the fluctuations of the magnetic fields in the lattice (surrounding tissues) match the Larmor frequency. Due to the rapid motion of water molecules, the protons in water take longer to transfer their energy, resulting in a long T1 time. In contrast, fatty acids have a frequency that aligns closely with the Larmor frequency, allowing for quicker energy transfer from fat protons to their surroundings. Consequently, fatty tissues exhibit a short T1 time. In T1-weighted imaging, a short repetition time (TR) is used; allowing tissues with short T1 values to recover their maximum longitudinal magnetization (LM) quickly after the radiofrequency (RF) pulse is

turned off. If the TR is extended, tissues with longer T1 values will also regain maximum LM, leading to minimal differences in the signal intensity of tissues with varying T1 times [5].

- **T2 weighted image**

T2 relaxation is affected by the inhomogeneity of local magnetic fields in tissues. Water molecules move swiftly, causing rapid oscillations in their local magnetic fields, which tend to cancel each other out. As a result, magnetic field intensity varies minimally within such tissues, allowing protons to stay in phase for an extended period of time, resulting in a longer T2. In contrast, when the liquid is impure or the tissue includes bigger molecules, their mobility decreases. This increases the inhomogeneity of the intrinsic magnetic field within the tissue, forcing protons to shift out of phase more quickly. Therefore, fat has a shorter T2 [5].

In T2-weighted imaging, the echo time (TE) is extended. At short TE, there is usually no significant difference in signal intensity between tissues with short and long T2 values. However, at longer TE, only those tissues with long T2 will produce a sufficiently strong signal.

- **Proton density (PD) image**

Proton density (PD) imaging generates contrast based on the concentration of protons in tissue. To minimize T1 and T2 effects, a long repetition time (TR) and a short echo time (TE) are used. The combination of a long TR and short TE yields a PD-weighted image, which primarily reflects the distribution and density of hydrogen nuclei in different anatomical structures (see Figure 1.12) [6].

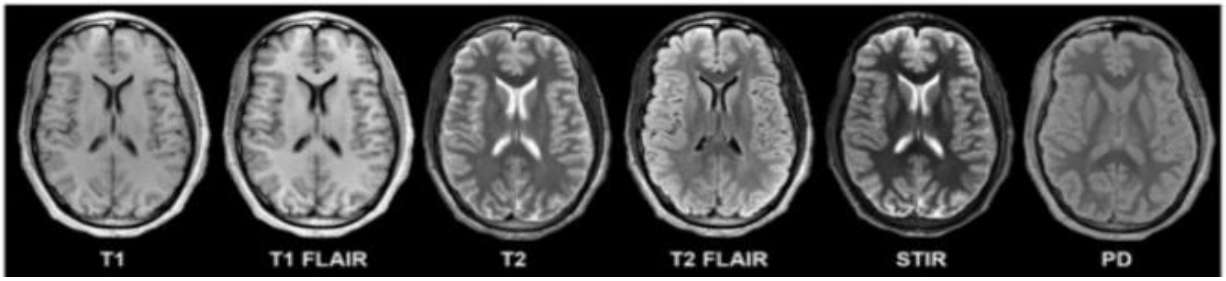


Figure 1.12: examples of weighted images

While T1, T2, and PD weightings provide essential contrast for clinical diagnosis, their effectiveness depends on accurate signal representation. In practice, imperfections in signal acquisition and processing can distort these contrast mechanisms, leading to image artifacts.

I.8. Artifacts

Artifacts are defined as undesirable signals that can occur during imaging processes. Nearly all imaging modalities are prone to these artifacts, which can lead to deterioration in image quality. In this section, we will explore several types of artifacts that are commonly observed in Magnetic Resonance Imaging (MRI) [2].

- **I.8.1. Ghosts / Motion artifacts**

Ghosting artifacts appear in MRI images as blurred, duplicated structures caused by patient motion during acquisition. These artifacts manifest as smeared repetitions along the phase-encoding direction, Motion artifacts can be categorized as either periodic (e.g., cardiac pulsation, respiration, blood flow) or random (e.g., voluntary patient movement) [6].

- **I.8.2. Aliasing / wraparound**

Aliasing (wraparound artifact) occurs when the field of view (FOV) is smaller than the anatomical region being imaged, causing signal from outside the FOV to falsely appear on the opposite side of the image. This artifact can manifest along any imaging axis: frequency-encoding, phase-encoding, or slice-selection directions [2].

- **I .8.3. Shading Artifacts**

Shading artifacts refer to the loss of signal intensity in specific areas of an image, which occurs due to the non-uniformity of the RF field during image acquisition. On MRI images, this artifact manifests as regions of high signal intensity at the periphery and areas of low signal intensity at the center [6]. There are several strategies to minimize or avoid RF inhomogeneity artifacts, one of which is shimming using shim coils (see section instrumentation). This technique can be employed to correct magnetic field inhomogeneities within the region of interest (see examples in Figure 1.13).

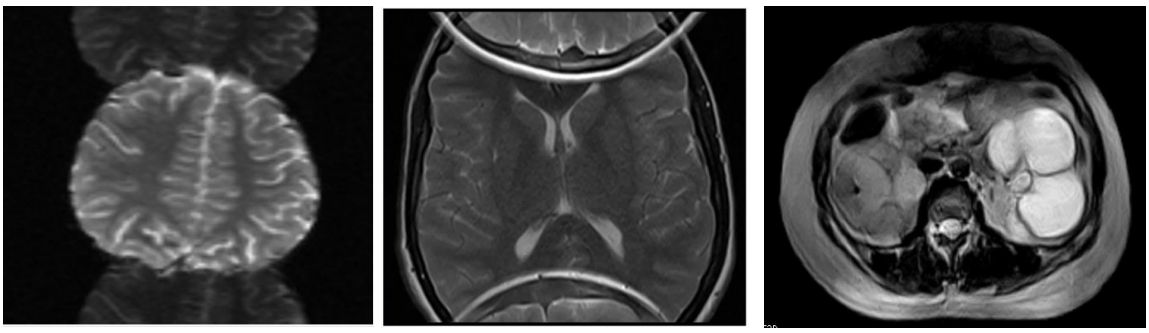


Figure 1.13: examples of common MRI artifacts

(Left) **Ghosting**; (middle) **Aliasing**; (right) **Shading**

II. Instrumentation

The most essential components in magnetic resonance imaging (MRI) systems include (Figure 1.14): **Main magnets, Radiofrequency (RF) coils, Gradient coils, Shim coils, and Computer systems.**

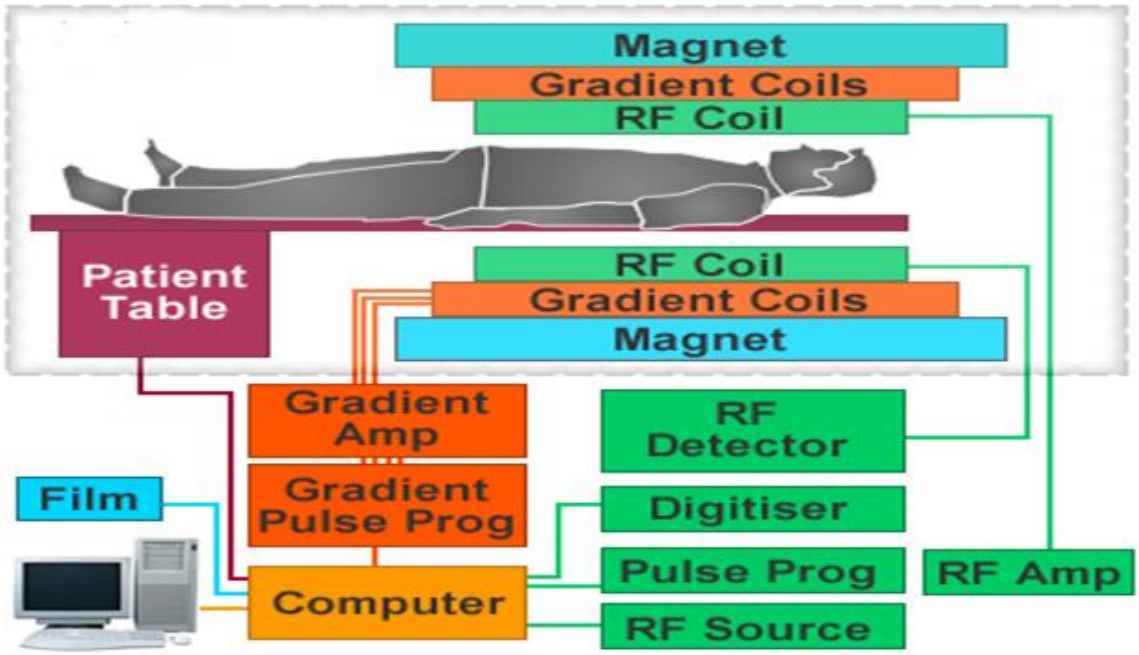


Figure 1.14: key elements in an MRI scanner

II.1. Magnets

In general, there are three types of magnets: permanent magnets, resistive magnets, and superconducting magnets.

- **II .1.1. Permanent Magnets**

A permanent magnet is composed of ferromagnetic materials that can be easily magnetized. The most common material used is alnico, an alloy made from aluminum, nickel, and cobalt. These magnets do not require a power supply, making them low-cost options. However, they can only achieve relatively low magnetic field strengths, typically ranging from 0.2 to 0.3 Tesla. Consequently, they produce images with lower signal-to-noise ratio (SNR) and reduced resolution, often necessitating longer scan times to compensate for their weaker magnetic fields [6].

- **II .2.1. Resistive magnets**

A resistive magnet is an electromagnet designed to carry high electrical currents, generating a static magnetic field stronger than permanent magnets. Unlike permanent magnets, it can be instantly activated/deactivated by controlling power supply. Resistive systems require continuous electricity input and active water cooling to manage Joule heating from coil resistance. Though more cost-effective to manufacture, their operational efficiency is lower due to substantial power demands for both field generation and cooling systems [3].

- **II .2.3. Superconducting magnets**

Superconducting magnets create powerful magnetic fields by exploiting zero-resistance current flow in cooled coils. This phenomenon occurs at absolute zero temperature (−273°C or 4 K). Initially, current flows through loops of wire to create and strengthen the magnetic field, a process known as ramping. Subsequently, the wires are super cooled using cryogenic substances, typically liquid helium (He) or liquid nitrogen (N), to eliminate resistance [2]. This results in high magnetic field strengths ranging from 0.5 to 3 T, enabling advanced applications and optimal image quality. However, it is important to note that there is a high capital cost associated with this technology [5].

Superconducting magnets are the most commonly used type in clinical MRIs.

II.2. Radiofrequency coils

RF coils, are composed of wire loops that generate a magnetic field when an electric current flows through them. They serve the dual purpose of transmitting RF pulses into the patient and receiving signals from the patient. RF coils can function as transmitters, receivers, or both. Depending on their design, RF coils can be categorized into three types: volume coils, surface coils, and phased-array coils [6].

- **II .2.1. Volume coils**

Serve as both transmitters and receivers of radiofrequency (RF) signals in MRI systems. Standard scanners feature a built-in body coil—a type of volume coil integrated into the magnet bore for general imaging, while specialized smaller volume coils (e.g., head or knee coils) are used for targeted examinations [5]. Although volume coils provide uniform RF excitation across large volumes, ideal for consistent transmission, their wide reception profile also captures excess noise, making them less efficient for small regions of interest compared to surface or phased-array coils

- **II .2.2. Surface coils**

Surface coil is a specialized class of radiofrequency (RF) coils designed for placement in close proximity to the anatomical region of interest. This configuration provides three key advantages: enhanced signal reception efficiency, improved spatial resolution, and superior signal-to-noise ratio (SNR). However, these benefits are counterbalanced by inherent limitations, including a restricted field of view (FOV) and reduced B_1 field homogeneity compared to volume coils. Consequently, surface coils are particularly advantageous for high-resolution imaging of superficial structures, where their SNR gains outweigh spatial coverage constraints[5, 6].

- **II .2.3. Phased array coils**

Phased-array (PA) coils (see Figure 1.15) represent a sophisticated RF coil configuration that consists of multiple geometrically aligned surface coil elements working in concert. This innovative design effectively merges the enhanced signal-to-noise ratio (SNR) of surface coils with the broader field-of-view (FOV) typically associated with volume coils. During signal acquisition, each individual coil element independently captures MR signals from its localized

sensitive region. The outputs from these elements are **then** processed and computationally combined to reconstruct a composite image that boasts both high SNR and extensive anatomical coverage.

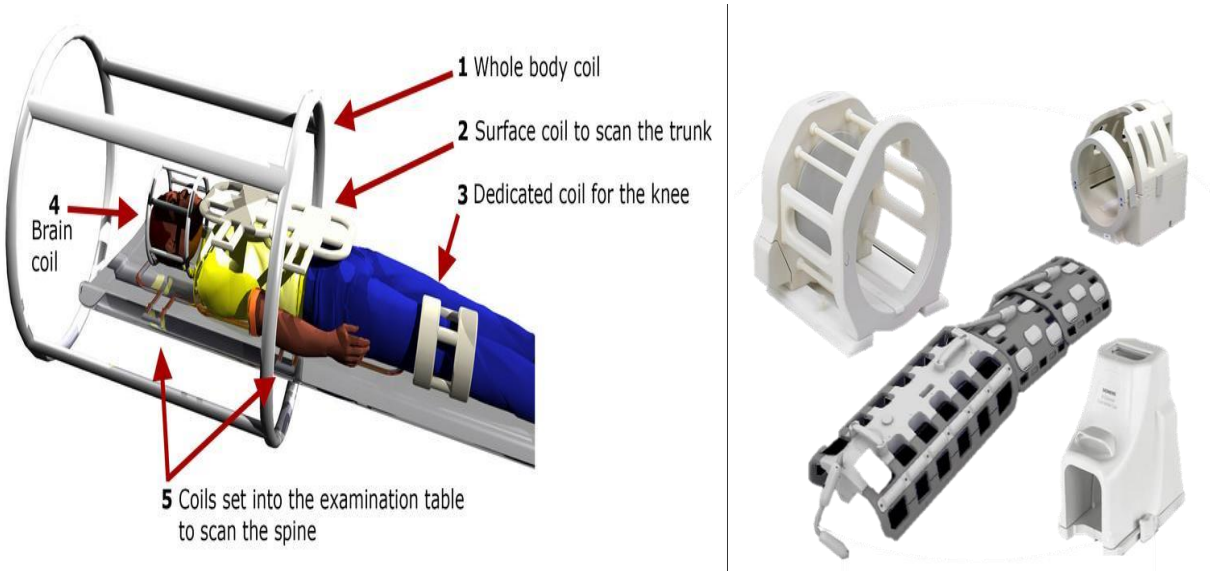


Figure 1.15: Radiofrequency coils (left) volume and surface coils; (rights) phased array coils

II.3. Gradient coils

Gradient coils (see Figure 1.16) are used to create intentional variations in the main magnetic field (B0). Each of the three sets of gradient coils corresponds to one of the axes: x, y, and z. these variations in the magnetic field enable the localization of image slices, as well as phase and frequency encoding. Stronger gradients enable both high-speed and high-resolution imaging by improving spatial encoding efficiency [2].

They play a crucial role in gradient echo sequences as well. Gradients eliminate the need for 180° pulses by rephasing the protons that become out of phase after the RF pulse is turned off, resulting in significantly faster gradient echo sequences.

II.4. Shim coils

Although the magnetic field is generally uniform, minor inhomogeneities can still occur. The process of achieving a homogeneous magnetic field is referred to as "shimming." To rectify these discrepancies, an additional loop of current-carrying wire, known as a shim coil, is strategically placed in the area of inhomogeneity. This effectively compensates for the irregularities in the main magnetic field.

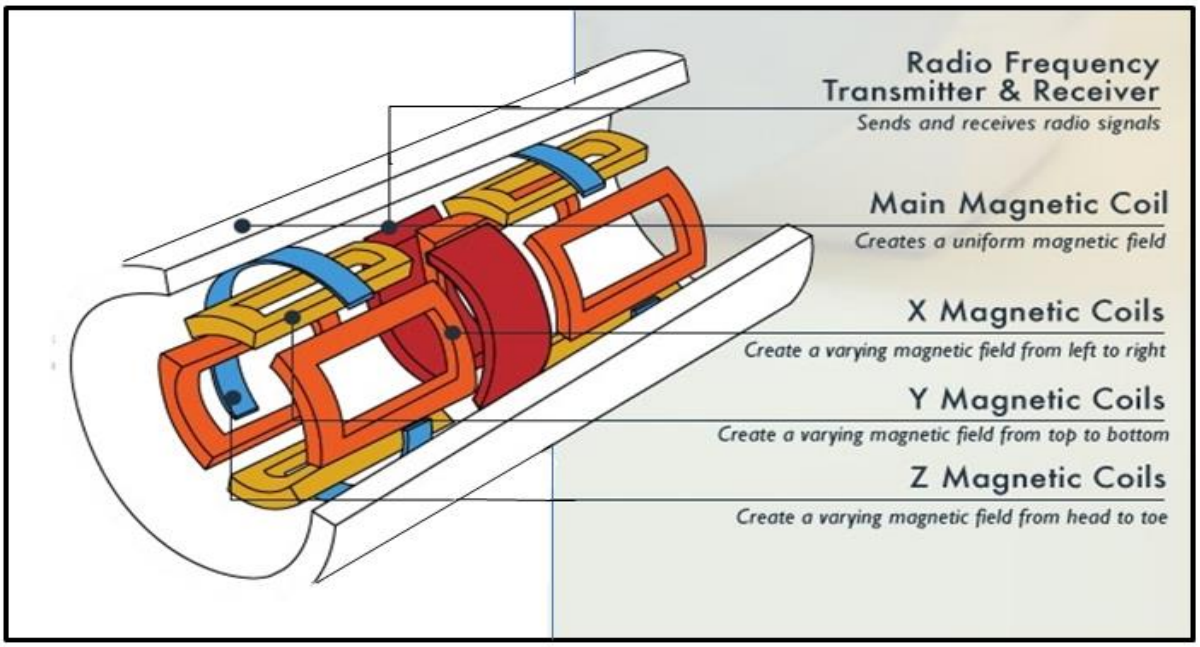


Figure 1.16: core MRI components: main magnet, Radiofrequency coils and gradient coils(X , Y , Z)

II.5. Computers and accessories

Computer system controls every application. They perform functions of data collection and manipulation, image viewing, storage, retrieval and documentation.

Image reconstruction plays a vital role in MRI, as it transforms raw frequency-domain (k-space) data into clinically meaningful spatial images. However, the process of acquiring fully sampled k-space data can be time-intensive, which may result in patient discomfort and motion artifacts. To address these challenges, MRI reconstruction techniques are specifically designed to recover high-quality images from undersampled or noisy k-space data, allowing for faster scans without compromising diagnostic integrity.

III. MRI reconstruction methods

Since MRI scanners do not produce direct anatomical images, reconstruction algorithms ranging from the fundamental Fast Fourier Transform (FFT), **Zero-Filling Reconstruction** to more sophisticated approaches like parallel imaging and compressed sensing (CS) and deep learning are essential to facilitate quicker scans thereby enhancing efficiency and reducing artifacts [2, 3].

III.1. Inverse Fourier Transform (IFT)

The relationship between k-space and the image domain is established through a Fourier transform, the simplest reconstruction method. It involves calculating the 2D or 3D inverse Fourier transform of k-space data using the formula

$$I(x, y) = F^{-1}\{S(k_x, k_y)\} \qquad (2)$$

The image $I(x, y)$ is obtained by taking the **inverse Fourier transform (IFT)** of the k-space data$S(k_x, k_y)$.

This approach is quick and mathematically exact when the data is fully sampled. However, it struggles with undersampled data and can produce aliasing artifacts [7].

III.2 Zero-Filling Reconstruction

In MRI, data acquisition is frequently undersampled to minimize scan duration. When using zero-filling for reconstruction, the missing k-space data is filled with zeros, which can introduce significant artifacts into the resulting images [8]. These zero-filling artifacts in MRI can manifest as blurring, smoothing, grid-like patterns, and ghosting or replica artifacts. (See artifacts section)

III.3. **Parallel imaging**

Parallel Imaging (PI) techniques, introduced in the late 1990s, have significantly enhanced MRI by reducing scan times while maintaining image quality. The core principle of PI lies in leveraging the varying spatial sensitivities of phased-array receiver coils to reconstruct the missing k-space data. This approach minimizes the number of required phase-encoding steps; thereby shortening overall scan duration. Two dominant PI methods emerged:

- **SENSE (Sensitivity Encoding) operates in image-space, effectively unfolding aliased signals by utilizing pre-calculated coil sensitivity maps. It reconstructs the combined image directly by solving a linear system of equations, taking into account the spatial response of each coil.**
- **GRAPPA (Generalized Auto calibrating Partially Parallel Acquisition) functions in k-space by synthesizing missing phase-encoding lines through local interpolation. It derives reconstruction weights from fully sampled auto calibration signals (ACS) embedded at the center of k-space and applies these weights to populate the unsampled regions.**

In both methods, the final image is synthesized by combining coil-specific reconstructions using the root sum of squares (RSS) [9].

While PI is a valuable advancement in MRI, it does have limitations. A major concern is noise amplification, which tends to increase as acceleration factors rise. Additionally, PI is vulnerable to various artifacts, including aliasing artifacts that arise from under-sampling. Inaccurate estimation of coil sensitivities or insufficient k-space data can lead to distorted images, potentially obscuring critical anatomical details and negatively impacting diagnostic confidence.

III.4. Compressed sensing

Compressed sensing (CS) has effectively overcome the limitations associated with previous reconstruction methods. Notably, it addresses the aliasing problem inherent in parallel imaging by utilizing sparse signal recovery techniques, which significantly enhances image

fidelity. Furthermore, CS improves contrast by employing a strategically designed mask that emphasizes the most relevant information from the center of k-space, where the highest signal energy is concentrated. By exploiting the inherent sparsity of MR images, this innovative approach achieves superior image quality with high contrast while requiring considerably fewer data points, making it both computationally efficient and highly effective. Therefore, CS represents a groundbreaking advancement in medical imaging, seamlessly integrating robustness, efficiency, and enhanced diagnostic precision[10].

CS can be combined with parallel imaging to achieve higher acceleration factors.

III.5. Deep Learning-Based MRI Reconstruction

Latest-generation MRI systems now incorporate advanced image reconstruction techniques to enhance image quality while significantly reducing acquisition times**.** Among these, deep learning (DL)-based methods represent a transformative leap forward, outperforming conventional algorithms by directly extracting the most diagnostically relevant features from large datasets. Unlike traditional approaches, DL drastically shortens scan durations by requiring fewer sampled data points, thereby improving patient throughput and comfort**.** Among the most advanced solutions currently deployed in clinical settings are General Electric's Recon DL, Siemens Healthineers' Deep Resolve, and Canon Medical's AICE. These groundbreaking platforms exemplify how artificial intelligence is transforming radiology practice. Unlike conventional reconstruction methods, these AI-powered systems leverage neural networks trained on vast datasets to deliver superior image quality from significantly undersampled data. These innovations demonstrate how deep learning can simultaneously address multiple challenges in MRI - reducing scan times by up to 50%, improving patient comfort, and providing radiologists with clearer, more detailed images for accurate diagnosis. The clinical implementation of these technologies marks a pivotal advancement in medical imaging, offering tangible benefits to patients, clinicians, and healthcare systems alike [11] .

Conclusion

This chapter covers the core concepts of MRI, from k-space signal acquisition to image reconstruction. While Fourier transform (FFT) reconstruction itself is computationally efficient, the requirement for fully sampled k-space data leads to long scan times. This limitation has driven the development of accelerated imaging techniques. Parallel imaging (SENSE/GRAPPA) and compressed sensing. More recently, deep learning has emerged as a powerful approach, learning to reconstruct high-quality images from undersampled data. Since each method has distinct advantages and limitations, current research focuses on hybrid approaches that combine the strengths of DL and traditional methods. In the next chapter, we'll explore how integrating compressed sensing with deep learning offers both rapid acquisition and reliable reconstruction.

Chapter 02:Compressed Sensing and Deep Learning

Introduction

As discussed in Chapter one, deep learning has become an effective technique for generating high-quality images from data that is not fully sampled. Ongoing studies are concentrating on hybrid approaches that merge the advantages of deep learning (DL) with conventional techniques.

MRI reconstruction has evolved from classical techniques such as parallel imaging (PI) and compressed sensing (CS) to modern deep learning (DL)-based methods. Early PI techniques, including **GRAPPA** and **SENSE.** accelerated scans by leveraging multi-coil sensitivity information but suffered from noise amplification at high acceleration factors [12]. CS methods improved reconstruction by exploiting signal sparsity in transform domains (e.g., wavelet or total variation), enabling higher undersampling rates [13]. However, CS faced challenges in computational complexity and manual parameter tuning, limiting its adaptability across diverse clinical scenarios [14].

DL has revolutionized MRI reconstruction by automating feature extraction and improving reconstruction speed and quality. Key advancements include:

- **End-to-End Learning:** Convolutional neural networks (CNNs, U-Nets) and transformers [15] directly map undersampled k-space or aliased images to high-fidelity reconstructions, outperforming traditional CS methods in benchmarks like fastMRI [12].
- **Unrolled Optimization:** Hybrid approaches such as MoDL, L+S-Net, and neural proximal gradient descent [16] integrate iterative CS solvers with learned regularization, combining physics-based models with data-driven priors.
- **Generative Models:** GANs (RefineGAN, SARAGAN) and diffusion models [17] enhance perceptual quality by generating realistic tissue textures, while variational autoencoders (VAEs) provide probabilistic reconstructions [12].
- **Federated Learning (FL):** Addresses data scarcity and privacy concerns by enabling collaborative training across institutions without sharing raw patient data [14].

These methods have demonstrated superior performance in PSNR, SSIM, and artifact suppression at high acceleration factors (up to 8×) [15]. However they have significant limitations: their inability to capture long-range dependencies in complex anatomical structures and an excessive reliance on local patterns.

It is in this context that Graph Neural Networks (GNNs) emerge as an innovative solution: unlike CNNs, they naturally capture spatial relationships between pixels via graphs while avoiding the excessive complexity of transformers, making them ideal candidates for precise and scalable reconstruction of undersampled MRI data.

Senior et al. [18] provides a comprehensive overview of the application of Graph Neural Networks (GNNs) in vision-language tasks, specifically focusing on image captioning, visual question answering (VQA), and image retrieval. The authors present a taxonomy of graph types used in these tasks, such as semantic, spatial, and knowledge graphs, and review various GNN architectures, including Graph Convolutional Networks (GCNs), Gated Graph Neural Networks (GGNNs), and Graph Attention Networks (GATs). The paper highlights the advantages of GNNs in capturing relational and structural information within images, which is crucial for tasks requiring scene understanding. It also discusses the limitations of current datasets, including biases, and explores future directions, such as integrating GNNs with emerging technologies like latent diffusion models. The survey emphasizes that GNNs offer strong inductive biases for structured data, making them particularly effective for vision-language tasks compared to more general architectures like Transformers.

Pradhyumna and al. [19] explore the application of Graph Neural Networks (GNNs) in image and video understanding for computer vision tasks. It highlights the limitations of traditional Convolutional Neural Networks (CNNs) in handling non-Euclidean data and emphasizes the advantages of GNN variants like Graph Attention Networks (GAT) and Graph Convolutional Networks (GCN) in modeling relationships between objects, actors, and their environments. The paper discusses key challenges such as node classification, link prediction, and graph classification, and presents GNN architectures for tasks like action recognition, human-object interaction, and zero-shot learning. It also reviews practical applications in traffic control, molecular studies, and video surveillance, while addressing implementation challenges like hyperparameter tuning and scalability. The authors conclude that GNNs offer a powerful framework for structured data analysis in computer vision but require further research to optimize performance and adaptability across diverse datasets.

Recent advancements in Graph Neural Networks (GNNs) have spurred significant interest in their application to healthcare, particularly in medical imaging and diagnostics. Researchers are now exploring GNNs for MRI reconstruction, with emerging methodologies

demonstrating promising results. Two key studies highlight the potential of GNNs in this domain: one offering a broad systematic review of GNN applications in healthcare, and the other proposing a novel GNN-based approach for accelerated MRI reconstruction. (GNN_healthcare). The study underscores GNNs' superiority over traditional machine learning methods in processing heterogeneous, graph-structured data such as electronic health records and medical imaging. Key applications include disease prediction, drug discovery, patient monitoring, and clinical decision support. Despite their potential, challenges remain in interpretability, scalability, and data integration. The review outlines future directions, such as personalized medicine, federated learning for privacy preservation, and real-time decision support, positioning GNNs as a critical technology for improving diagnostic accuracy and therapeutic outcomes [20]. In the context of medical imaging, the GNN_self_similarity study introduces the Graph Convolutional Enhanced Self-Similarity (GCESS) network, a novel framework for MRI reconstruction [21]. This method combines local spatial convolution with non-local graph convolution, leveraging both fine-grained details and broader structural self-similarities within images. By constructing a patch graph to model non-local relationships, the GCESS network enhances reconstruction quality, achieving superior artifact suppression and structural integrity. Experimental results on knee and brain datasets demonstrate its effectiveness, outperforming state-of-the-art methods with a PSNR of 34.19 dB and SSIM of 0.8994 under 4× acceleration. these studies illustrate the expanding role of GNNs in healthcare, from broad diagnostic and therapeutic applications to specialized tasks like MRI reconstruction. Future research should focus on overcoming existing limitations while further exploring real-world clinical implementations.

In this chapter, we will specifically explore compressed sensing, as well as deep learning in general, with a focus on convolutional neural networks (CNNs) and graph neural networks (GNNs) and how integrating Deep learning and Compressed sensing can improve the image reconstruction quality.

2.1 Compressed Sensing

Compressed Sensing (CS) MRI is an innovative technique designed to accelerate the acquisition of MRI data while preserving image quality. Traditional MRI methods require the collection of vast amounts of raw data to reconstruct high-resolution images. CS takes advantage of the fact that MRI scans can be effectively represented using fewer data points in specific domains, all without a significant loss of information [22].

It is based on three key principles:

- **Transform Sparsity**

Many MRI images are not sparse in their original spatial domain but show sparsity when transformed into wavelet, Fourier, or total variation domains. This means most signal energy is concentrated in a few significant coefficients, with others being negligible. This transform sparsity is essential for enabling compressed sensing.

- **Pseudo-Random Undersampling**

In traditional MRI, k-space is sampled uniformly, whereas compressed sensing (CS) MRI uses pseudo-random or variable-density sampling. This method undersamples the outer high-frequency regions of k-space while preserving samples near the center (low-frequency).This strategy creates noise-like artifacts instead of the structured aliasing seen in uniform undersampling. Since these artifacts are incoherent, they can be effectively removed during image reconstruction

- **Nonlinear Image Reconstruction**

CS-MRI reconstruction solves an optimization problem that balances L1-norm sparsity regularization (to enforce transform-domain sparsity) with L2-norm data fidelity constraints (to match acquired k-space measurements), typically using iterative algorithms like nonlinear conjugate gradient or soft thresholding.

2.2 Deep learning

Extensive research has been conducted to advance machine intelligence. Learning, a fundamental human ability, has become a crucial attribute of machines, achieved through various evolving techniques. Traditional machine learning algorithms have been widely used across numerous sectors, with researchers continuously working to enhance their accuracy. This ongoing effort has led to the development of deep learning, a sophisticated subset of machine learning. While the applications of deep learning are still being explored, it holds significant potential for addressing challenges in a range of emerging fields and subdomains [23].

One unique aspect of deep learning is representation learning—the ability of neural networks to automatically extract features from data. Unlike traditional machine learning, which relies

on manual feature engineering, deep learning models learn to identify relevant patterns independently

Deep learning is making significant progress in solving problems that have been challenging for artificial intelligence. It excels at finding complex patterns in large datasets, making it valuable in many areas, such as science, business, and government. Additionally, it has outperformed other machine learning methods in various tasks related to natural language understanding, including topic classification, sentiment analysis, question answering, and language translation (see Figure 2.1).

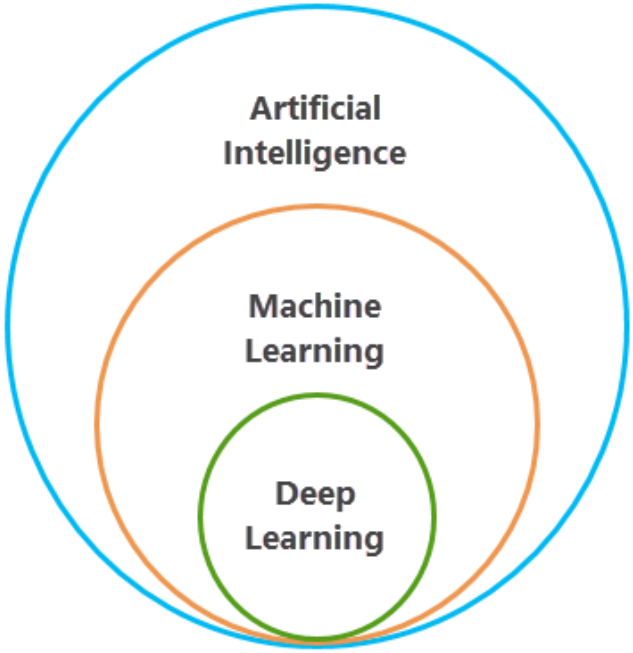


Figure2.1 : Relationship between Deep Learning , Machine Learning and Artificial intelligence.

A deep learning model typically comprises three core layers (see Figure 2.2):

Input Layer: Receives raw data (e.g., images, text, or numerical values) and forwards it for processing.

Hidden Layers: Perform the bulk of computations, transforming input data into increasingly abstract representations. Each hidden layer applies weights to inputs, adds a bias term, and processes the result through a non-linear activation function (e.g., ReLU, sigmoid, or tanh). This non-linearity allows the network to learn complex patterns. Nodes within these layers

perform specific operations—such as feature detection in images—and pass processed data sequentially to subsequent layers.

Output Layer: Generates the model's final predictions or classifications.

A model qualifies as "deep" when it contains more than three layers (input, hidden, and output). This depth enables the network to capture intricate data relationships, excelling in tasks like image recognition, natural language processing, and autonomous decision-making.

Deep learning models operate through two key processes:

- **Forward Propagation: Data moves from the input layer through hidden layers, undergoing transformations via weights and activation functions to produce predictions.**
- **Back propagation: After prediction, the model calculates errors using a loss function and adjusts weights/biases via gradient descent, minimizing errors over iterations.**

Training deep neural networks demands substantial computational power, often requiring GPUs and cloud platforms. Frameworks like TensorFlow, PyTorch, and JAX streamline model development and training.

A major challenge is overfitting**,** where models memorize training data instead of generalizing. Techniques like L1/L2 regularization, dropout, and early stopping help prevent this, improving model robustness.

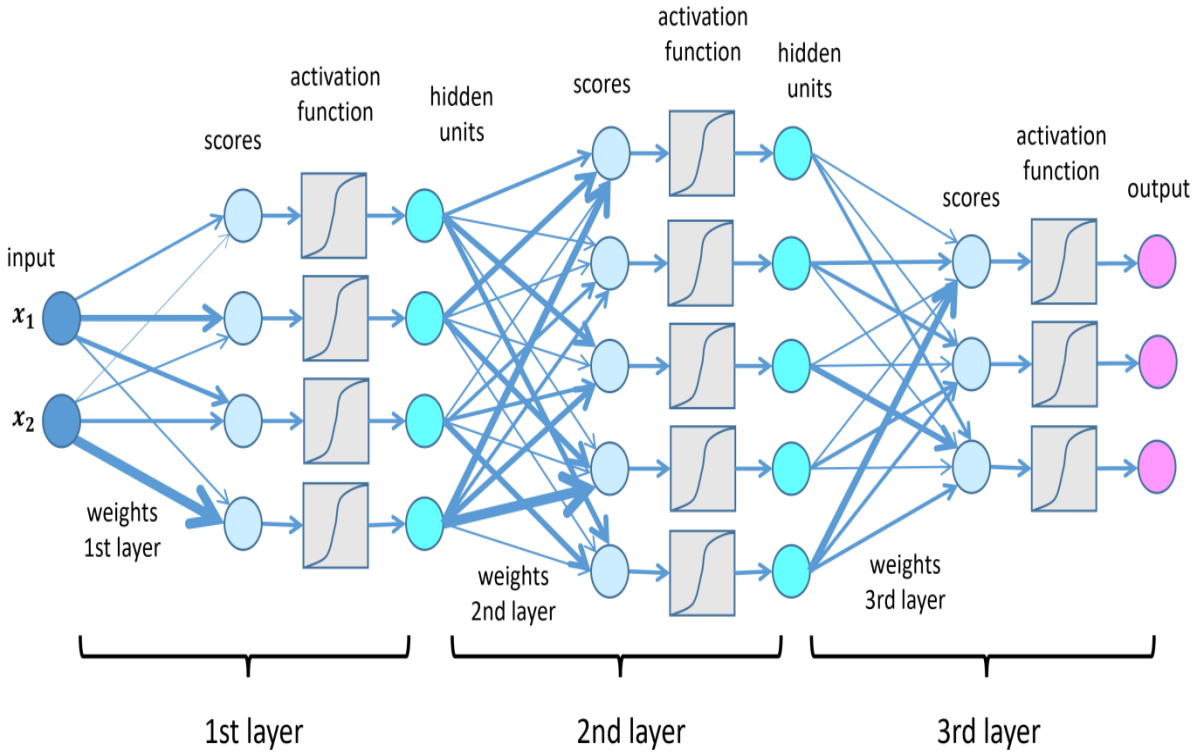


Figure2.2 : Deep Neural Network architecture

- **2.2.1. Convolutional Neural Networks (CNN)**

Convolutional Neural Networks (CNN) represents a specialized class of deep learning models particularly effective for processing visual data. Their unique architecture makes them exceptionally suited for image classification and object recognition tasks. A prominent application of CNNs is in medical imaging analysis, where they demonstrate superior performance in detecting anomalies in X-rays and MRIs compared to traditional methods, offering both increased speed and diagnostic accuracy.

The CNN architecture comprises three fundamental layer types that work in concert to process visual information:

- **convolutional layer**
- **pooling layer**
- **fully connected layer**

Convolutional Layer

The convolutional layer is the main part of a Convolutional Neural Network (CNN), where most of the calculations happen. It performs a convolution operation on the input and sends the result to the next layer. This operation combines all the pixels in a certain area into a single value. Depending on the problem you want to solve and the features you're learning, you can use different types of convolutions. The most common is the 2D convolution, often called conv2D. In a conv2D layer, a filter or kernel slides over the 2D input data (image as an example). The filter typically is a 3x3 matrix and performs element-wise multiplication. This process sums the results into a single output pixel. The kernel does this for every location it covers, turning a 2D matrix of features into another 2D matrix called a feature map.

Three important settings affect the output size that you must decide before training the neural network:

- **Number of Filters: this affects the output depth. For example, using three different filters creates three different feature maps, leading to a depth of three.**
- **Stride: this is the distance the kernel moves over the input matrix. Larger stride values can make the output smaller, but values of two or greater are usually rare.**
- **Zero-Padding: this is used when filters do not fit the input image. It sets all elements outside the input matrix to zero, creating a larger or equally sized output.**

After each convolution, a CNN applies a Rectified Linear Unit (ReLU) transformation to the feature map. This step adds nonlinearity, helping the model learn complex patterns. Without this, the CNN would work as a simple linear model, which is limited.

Pooling layer

Pooling layers, also known as downsampling, perform dimensionality reduction while retaining the most critical information: https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/ Similar to the convolutional layer, the pooling operation sweeps a filter over the entire input; however, the key difference lies in the fact that this filter does not contain any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array:

There are two primary types of pooling:

- **Max Pooling: As the filter traverses the input, it selects the pixel with the maximum value to send to the output array. This method is generally more commonly used than average pooling.**
- **Average Pooling: As the filter moves across the input, it calculates the average value within the receptive field to populate the output array.**

While a lot of information is lost in the pooling layer,they provide several advantages to convolutional neural networks (CNNs). These include reducing complexity, enhancing efficiency, and minimizing the risk of overfitting.

Fully connected layer

A Fully Connected Layer (also known as a Dense Layer), as the name implies, is a layer where each neuron is connected to every neuron in the preceding layer. This layer plays a crucial role in classification by utilizing the features extracted through the previous layers and their various filters. While convolutional and pooling layers typically employ ReLU activation functions, Fully Connected Layers often utilize the softmax activation function to accurately classify inputs, (see Figure 2.3).

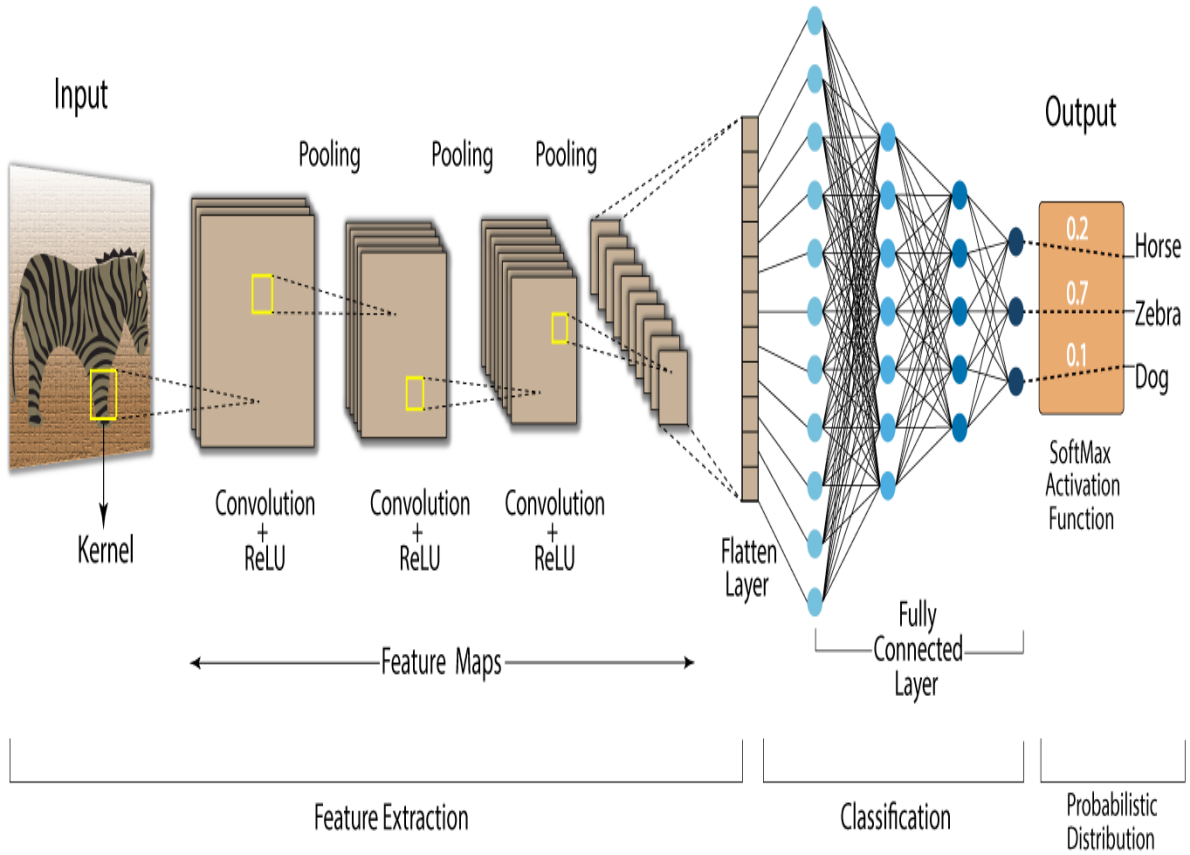


Figure2.3 : Convolutional Neural Network architecture

- **2.2.2 Graph Neural Networks**

The foundational concept of Graph Neural Networks (GNNs) draws inspiration from Convolutional Neural Networks (CNNs), but adapts their principles to handle irregular, relational data structures (see Figure 2.4). Where CNNs excel at processing Euclidean data like images (where pixels form a regular grid with uniform connectivity), GNNs address the challenges of non-Euclidean domains where relationships are arbitrary and node connectivity varies—such as in social networks, traffic systems, or molecular structures.

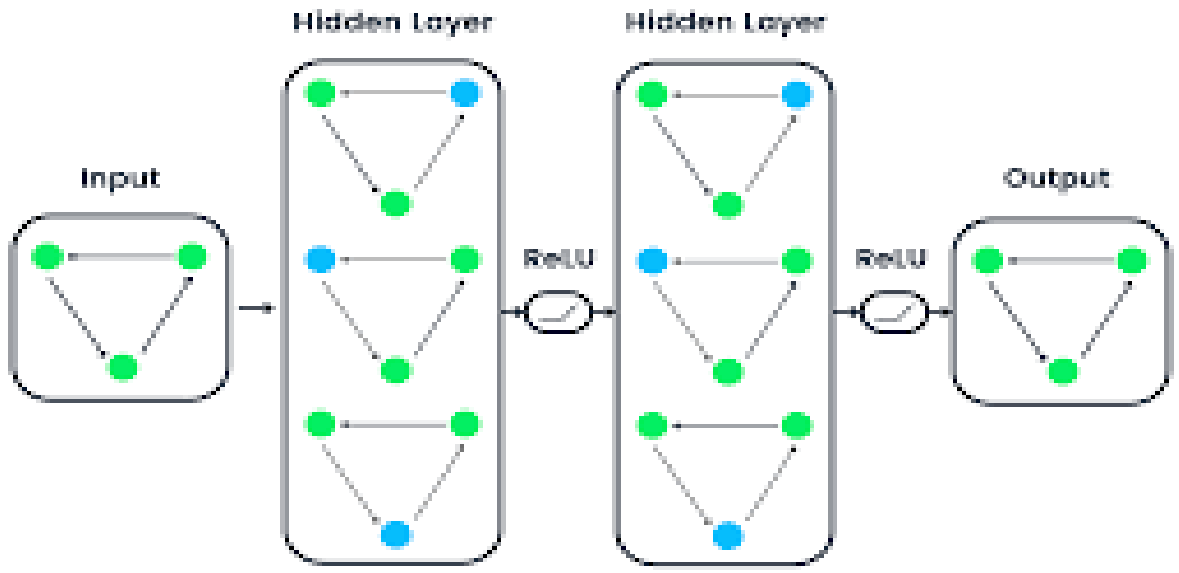


Figure 2.4: Graph Neural Network architecture

In graph theory, a graph comprises nodes (representing entities) and edges (capturing relationships between them). While an image can be viewed as a specialized graph with fixed, grid-like connections, most real-world graph data lacks this regularity. Traditional CNNs, designed for Euclidean space, cannot natively process these irregular structures. GNNs

overcome this limitation by introducing flexible operations that adapt to variable connectivity patterns, enabling deep learning to model complex relational systems (see Figure 2.5).

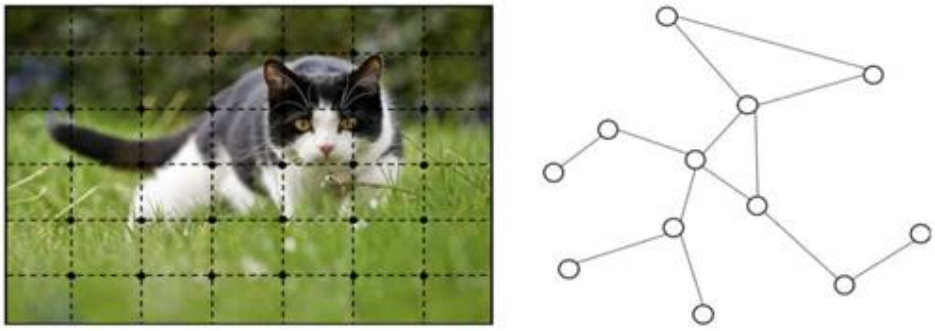


Figure 2.5: Left : image in Euclidean space and Right: graph in non-euclidean space

GNNs operate through an iterative *message passing* framework, where nodes aggregate and process information from their local neighborhoods (see Figure 2.6). This process unfolds in three key steps:

- **Message Computation:** Each node generates a message based on its current state (embedding) and the properties of its connecting edges.
- **Message Aggregation:** The node collects messages from its neighbors, combining them through permutation-invariant operations (e.g., summation, averaging, or attention-weighted fusion).
- **State Update:** The node updates its own embedding using a nonlinear transformation of its previous state and the aggregated messages.

Through repeated iterations, each node progressively incorporates information from increasingly distant neighbors, building a rich representation that reflects both local and global graph structure. Crucially, this process is *order-invariant*—the results are consistent regardless of how nodes are sequenced, ensuring robustness to input permutations.

Like other neural networks, GNNs are trained via gradient descent: Initial random weights are assigned to message computation and update functions. A task-specific loss function (e.g., node classification or link prediction error) guides weight adjustments across layers.

Backpropagation through message-passing steps enables end-to-end learning of both node features and graph topology.

For hierarchical representation learning or graph-level tasks, GNNs employ *pooling* to reduce dimensionality:

- **Node embeddings within a subgraph are aggregated (via summation, attention, or other operators) into a coarser representation.**
- **This allows the model to capture higher-level patterns while maintaining computational efficiency, particularly critical for large-scale graphs.**

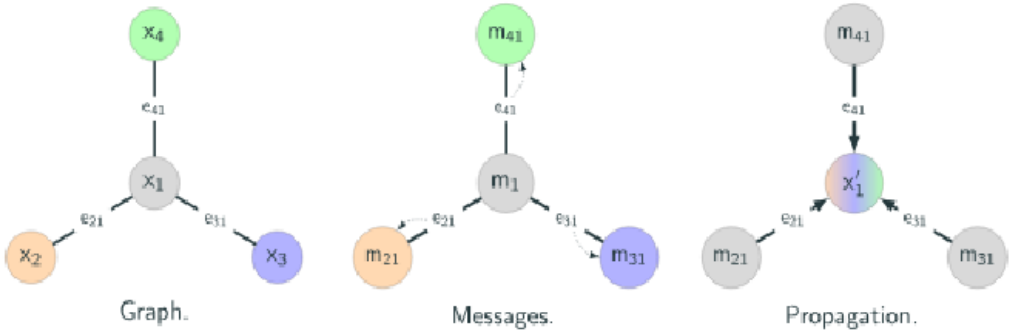


Figure 2.6 : message passing

Graph Neural Networks (GNNs) can be categorized into several key architectures recurrent GNNs, convolutional GNNs (GCNs), graph autoencoders, and spatio-temporal GNNs each designed to address distinct computational challenges.

- **Message Passing Neural Networks (MPNNs)**

The foundational framework for most GNNs.Operates via iterative neighborhood aggregation, where nodes update their representations by exchanging and combining information with connected neighbors.

- **Graph Convolutional Networks (GCNs)**

GCNs extend convolutional operations to **non-Euclidean graph data.** Unlike traditional CNNs (which process grid-structured images via fixed sliding filters), GCNs apply **weight-sharing convolutions** over irregular node neighborhoods.

Key innovation: Adapts convolution for **variable graph topologies**, enabling feature learning without assuming a rigid data structure.

- **Graph Attention Networks (GATs)**

GATs enhance message passing with dynamic attention mechanisms. computes learned attention weights between nodes, allowing the model to prioritize relevant neighbors during aggregation.

- **2.2.3 Graph Convolutional Networks (GCNs) vs. Convolutional Neural Networks (CNNs)**

While CNNs and GCNs both leverage localized feature aggregation, they differ fundamentally in their data assumptions and operational mechanics. Below is a systematic comparison:

Table 01: key differences between CNN and GNN

Feature	Convolutional Neural Networks (CNNs)	Graph Convolutional Networks (GCNs)
Data Structure	Euclidean (grid-structured: e.g., images, text sequences).	Non-Euclidean (irregular: e.g., social networks, molecules).
Connectivity	Fixed/local (e.g., 3×3 pixel neighborhoods).	Arbitrary (node degrees vary; no spatial regularity).dynamic
Weight Sharing	Filters slide uniformly across input (translation invariance).	Filters adapt to node neighborhoods (no fixed sliding window).
Spatial	Preserved (pooling reduces grid	No natural ordering; pooling requires

Hierarchy	dimensions predictably).	graph coarsening methods.
Key Operation	Discrete convolution (sum of element-wise products).	Message passing (aggregate neighbor features + transformation).
Invariance Properties	Translation invariance (useful for object detection).	Permutation invariance (order of nodes doesn't affect output).
Example Applications	Image classification, object detection, NLP (text as sequences).	Node classification, link prediction, recommendation systems.
Challenges	Struggles with irregular data (e.g., graphs with varying sizes).	Scalability to large graphs; over-smoothing in deep layers.

Both CNNs and GNNs are powerful neural network architectures tailored for specific types of data:

- **CNNs are ideal for capturing local patterns in regular, grid-like data structures, making them indispensable in computer vision and related fields.**
- **GNNs are designed to handle complex, relational data represented as graphs, allowing for sophisticated modeling of interactions and dependencies in various domains like social networks, molecular chemistry, and recommendation systems.**

2.3. Towards deep learning-supported compressive sensing

Machine learning methods can effectively enhance compressive sensing. The introduction of GPUs and TPUs, along with support for deep learning frameworks like TensorFlow Lite and Pytorch, have made deep learning widely accessible, even on mobile and embedded devices. Over the past ten years, compressive sensing has progressed from theory to practical application. Initially, its effectiveness was limited by the time it took for reconstruction algorithms to process. However, deep learning has significantly sped up this process, allowing for real-time reconstruction. Deep learning not only improves speed but also enhances the quality of reconstruction. DL-based methods outperform traditional algorithms by avoiding strict sparsity assumptions and better capturing relevant data features. By speeding up signal reconstruction, adjusting the sampling matrix, and enhancing high-level inference, the CS–DL integration can lower energy, storage, and processing demands. As a result, certain applications that previously required excessive resources can now be managed more easily.

- **2.3.1 Deep learning for iterative reconstruction**

The first category of compressive sensing (CS) reconstruction methods employs neural networks specifically designed **to** mimic iterative CS algorithms through a technique known as **algorithm unrolling** (or unfolding). This approach transforms each iteration of a classical optimization algorithm (e.g., ISTA, ADMM) into a dedicated neural network layer, stacking multiple such layers to form a deep architecture

Key Mechanism:

- **Unrolling Process:** Each algorithmic iteration (e.g., proximal gradient step in ISTA) is mapped to a trainable network layer, Traditional parameters (e.g., step sizes, thresholds) become learnable weights.
- **End-to-End Training:** The unrolled network is trained using stochastic gradient descent (SGD) on CS measurement data. Optimizes both linear measurements (e.g., sensing matrices) and non-linear features (e.g., sparsity constraints).

Examples:

ISTA-Net: Unrolls Iterative Shrinkage-Thresholding Algorithm (ISTA) with learned thresholds and transforms (see Figure 2.7).

ADMM-Net: Embeds Alternating Direction Method of Multipliers (ADMM) steps into a network with trainable proximal operators.

TV-Net : (unrolling total variation minimization).

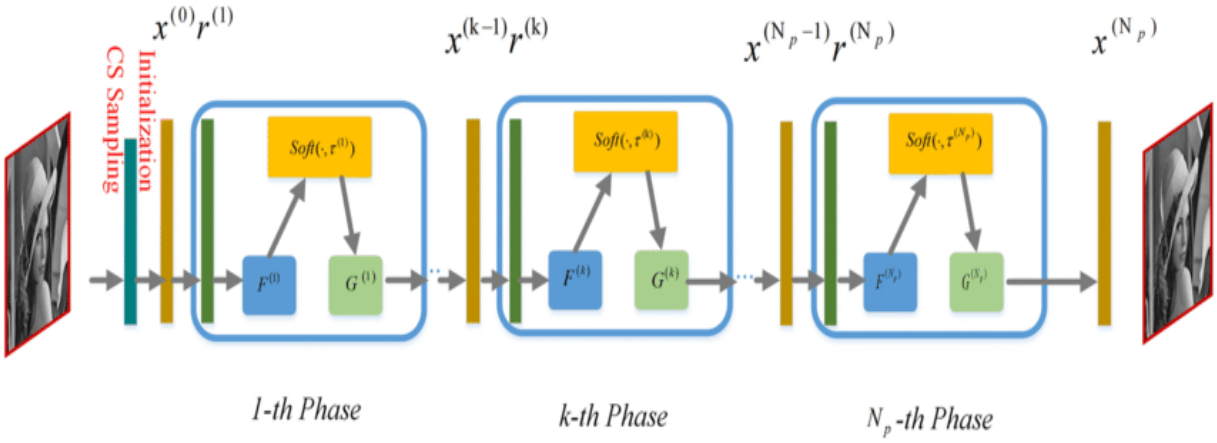


Figure 2.7 : Example of ISTA_Net

Moving beyond iterative approaches, researchers have developed an even more radical solution: *deep learning for direct reconstruction.* This paradigm completely bypasses traditional reconstruction algorithms by training deep neural networks to directly transform compressed measurements into high-quality reconstructions in a single feedforward pass. This approach achieves unprecedented reconstruction speeds but requires careful network design and extensive training to maintain reconstruction accuracy.

- **2.3.2 Deep learning for direct reconstruction**

Direct reconstruction using Deep Learning (DL) transforms compressed measurements (such as undersampled MRI data) into complete images in a single step, eliminating the complex iterative processes of traditional methods. To illustrate : while conventional approaches progressively reconstruct images like assembling a puzzle piece by piece (a slow, iterative method), direct DL learns to predict the final image instantly from a critical subset of data, functioning like an expert who can guess the complete puzzle after seeing just a few key pieces. This capability is achieved through intensive training where the network analyzes thousands of examples pairing partial measurements (input) with reference images (output)—for instance, in MRI, incomplete k-space data and their full reconstructions. Once trained, the system operates in real time (within milliseconds), offering decisive advantages: unmatched speed (versus minutes/hours with traditional compressed sensing), quality optimized through learning, though with challenges including substantial training data requirements and some decision-making opacity. This approach is revolutionizing fields where speed and precision are critical, such as accelerated MRI and satellite imaging (see Figure 2.8).

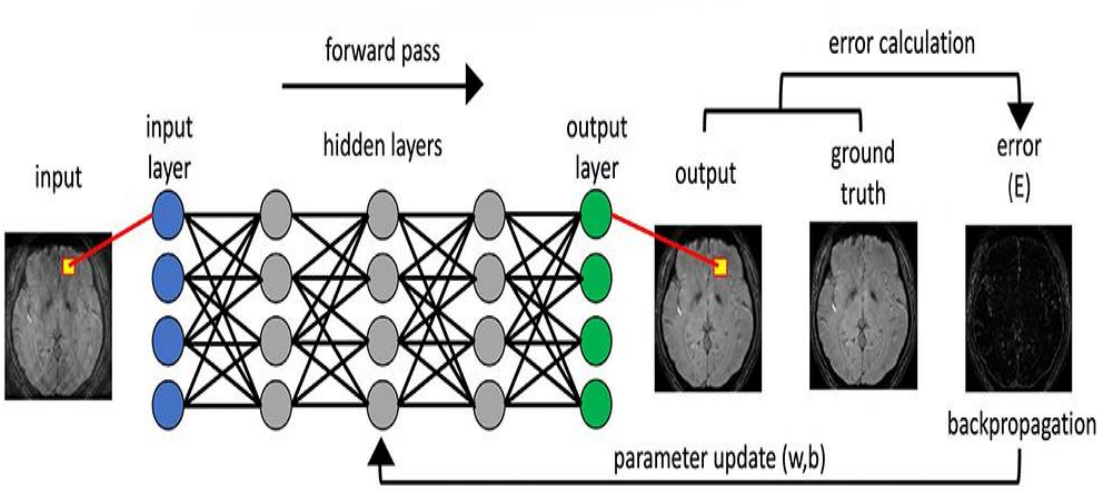


Figure 2.8 : Overall process of the learning process for the simplified version of the neural network ,

Conclusion

Compressed Sensing (CS) and Deep Learning (DL) are pivotal in advancing MRI applications, offering significant improvements in reconstruction accuracy and computational efficiency. This chapter outlined the fundamental principles of CS and DL, with particular emphasis on Convolutional Neural Networks (CNNs) and Graph Neural Networks (GNNs). Building on these foundations, our work introduces a novel hybrid approach that integrates CS with GNNs to optimize MRI reconstruction. The following chapter will delve into the detailed implementation of the proposed methodology, including its experimental setup and performance evaluation.

Chapter 03:CS_GNN application

Introduction

In this chapter, we explain how to integrate Compressed Sensing (CS) into Graph Neural Networks (GNNs) step by step. Because of time limits, the first version of our method did not produce fully satisfactory results. This issue led us to look for an extra post-processing step using a basic Convolutional Neural Network (CNN) architecture to improve performance. The chapter is organized as follows: First, we describe the materials and methods we used. Next, we present and discuss the results. Finally, we wrap up the chapter with a summary of key findings and their implications.

3.1 Materials

This section details the experimental framework of our study, including the dataset specifications and computational resources employed for implementation. We first describe the MRI database used for training and validation, followed by the hardware and software configurations that supported our computational experiments.

- **3.1.1 Dataset**

Our initial work utilized the FastMRI dataset, a widely recognized open-source resource for MRI reconstruction research developed through a collaboration between Facebook AI Research (FAIR) and NYU Langone Health. This comprehensive dataset provides valuable k-space raw data alongside reconstructed DICOM images, encompassing both knee and brain MRI scans.

However, several practical limitations emerged during implementation. Most notably, the dataset lacks fully sampled test scans for quantitative evaluation, and its considerable size (exceeding 1TB for the complete collection) presented significant downloading and processing challenges. These constraints motivated us investigation of alternative benchmarks with comparable clinical relevance but more manageable scale.

After careful evaluation, we selected the Calgary-Campinas Public Dataset as our primary benchmark. This resource offers several advantages: (1) focused specifically on brain MRI reconstruction, aligning with my research objectives; (2) provides a more tractable data volume without sacrificing clinical utility; and (3) includes appropriate reference standards for meaningful performance evaluation.

- **Calgary Campinas description**

The Calgary-Campinas dataset comprises 3D T1-weighted brain MR images accompanied by segmentation masks and corresponding k-space raw data. The dataset is organized into two distinct subsets: (1) the cc359 collection containing reconstructed images for segmentation tasks, and (2) the raw data repository featuring both multi-coil and single-coil k-space measurements. For our compressed sensing MRI reconstruction experiments, we specifically employed the single-coil k-space data from the raw dataset. This choice was motivated by two key considerations: first, the single-coil configuration provides a more straightforward implementation framework for our proof-of-concept study; second, it eliminates the need for coil sensitivity estimation and other complexities inherent to multi-coil reconstruction pipelines.

- **Single coil k_space data**

The single-coil k-space dataset comprises 35 fully-sampled 3D T1-weighted brain MRI scans acquired on a 3T GE Discovery MR750 scanner using a 12-channel head coil, processed into single-channel format through GE's Orchestra Toolbox reconstruction pipeline. The original multi-coil data was converted by combining normalized coil sensitivity maps into complex-valued images, then transformed back to single-channel k-space at 256×256 resolution. Organized into three training files (25 subjects, 4,524 slices) and one validation file (10 subjects, 1,700 slices), this structure provides an ideal benchmark for compressed sensing research while avoiding multi-coil complexities. For our implementation, we consolidated all training partitions before applying an 80-20 training-validation split (3,619 vs. 905 slices). Due to hardware constraints, we worked with a representative 1,200-slice subset (26.5% of total data) while maintaining proportional distribution (960 training/240 validation slices), preserving data diversity through stratified sampling.

- **3.1.2 Computational Resources**

For the development and testing of our CS-GNN hybrid method, we utilized the Kaggle cloud computing environment. This platform provided the following technical specifications.

- **Cloud Platform: Kaggle**

Kaggle is a cloud-based collaborative platform that provides an integrated environment for machine learning research, featuring GPU-accelerated computing (NVIDIA T4/P100), 2-core Intel Xeon CPUs, 13GB RAM, and 100GB SSD storage. The platform combines Jupyter Notebooks with pre-installed deep learning frameworks (PyTorch, TensorFlow) and managed Python environments, enabling efficient prototyping while ensuring reproducibility. Its built-in support for public datasets, collaborative tools, and benchmarking metrics proved particularly advantageous for our MRI reconstruction work, as it eliminated local hardware limitations, facilitated rapid experimentation, and provided direct access to medical imaging data and community-shared resources.Local Hardware Specifications

- **System Configuration**

- **OS**: Windows 10 Professional
- **CPU**: Intel® Core™ i5-6200U @ 2.30GHz (2.40 GHz boost)
- **RAM**: 8.00 GB
- **Storage**: 238 GB SSD (SanDisk model SD8SBAT256G1002)
- **Graphics**: Intel® HD Graphics 520 (128 MB dedicated memory)
- **System Type**: 64-bit operating system, x64-based processor

- **Software Stack**

The implementation utilizes Python 3.11 as the primary programming language with PyTorch 2.5.1 (CUDA 12.4 backend) serving as the core deep learning framework. Key computational libraries include NumPy for efficient array operations and mathematical computations, alongside Matplotlib for comprehensive result visualization. Quantitative evaluation employs scikit-image for standard image quality metrics (SSIM, PSNR, NRMSE). Graph neural network operations are implemented through PyTorch Geometric (v2.5), which provides optimized sparse matrix operations and specialized graph convolution layers. The Calgary-Campinas dataset interface was customized using PyTorch's native DataLoader for efficient batch processing and memory management during training. This combination of established scientific computing libraries ensures both computational efficiency and methodological

reproducibility while maintaining compatibility with standard medical imaging evaluation protocols.

3.2 Method

Building upon the theoretical foundations of compressed sensing (CS) and deep learning established in Chapter 2, this work specifically investigates the integration of CS-based undersampling patterns with graph neural networks (GNNs) - currently the most advanced architecture for MRI reconstruction. Our hybrid approach combines the computational efficiency of CS undersampling with the representational power of GNNs to optimize the image reconstruction pipeline

- **3.2.1 Compressed Sensing**

Compressed Sensing (CS) is an advanced signal acquisition framework that allows accurate signal reconstruction from significantly fewer samples than dictated by the Nyquist-Shannon criterion. The sampling pattern plays a decisive role, as it governs reconstruction quality (reducing artifacts such as aliasing and noise), computational efficiency (enabling faster reconstruction algorithms), and hardware compatibility (e.g., MRI scanner constraints). Consequently, optimizing the sampling strategy is crucial to achieving a balance between reconstruction accuracy, processing speed, and practical implementation in CS-based systems. Two widely adopted sampling patterns in CS include:

Random Sampling (Incoherent Sampling)

In compressed sensing MRI, we randomly select a subset of Fourier measurements instead of acquiring full data. This randomness helps because it scatters any missing information as random noise across the image rather than creating structured artifacts.

Variable-Density Sampling (VDS)

VDS strategically concentrates measurements in the central k-space region where most signal energy resides, while progressively undersampling higher frequencies. By sampling the k-space center more densely, VDS achieves superior signal-to-noise ratio (SNR) for clinically relevant image features while still enabling acceleration through peripheral undersampling. The resulting acquisition combines the benefits of robust central contrast with efficient high-frequency detail capture.

- **3.2.2 Graph Neural Network Integration**

The proposed method transforms raw k-space data into a graph representation, where nodes correspond to individual k-space points and edges encode spatial relationships between them. Prior to input into the GNN architecture, the graph undergoes normalization to ensure numerical stability during training. Through iterative message passing between connected nodes, the network learns to identify critical structural patterns in the undersampled data and accurately predict missing k-space samples. This graph-based approach effectively captures both local and global frequency relationships, enabling high-fidelity MRI reconstruction while preserving anatomical details

For the graph representation of k-space data, we implemented the following approach:

Node Creation:

In our graph representation of k-space data, we treated each pixel in the 256×256 matrix as a distinct node, resulting in 65,536 nodes per slice. To fully capture the physical information contained in the MRI measurements, we decomposed each node's complex-valued k-space data into two fundamental features: (1) the real component encoding magnitude information, and (2) the imaginary component representing phase information. This decomposition preserves the complete electromagnetic characteristics of the original signal while creating an input structure suitable for graph neural processing. The resulting graph maintains all spatial relationships between k-space points while transforming the reconstruction problem into a node feature prediction task that the GNN can effectively learn.

- **Edge Formation:**

We established node connections using k-nearest neighbors (KNN) with k=8, creating undirected edges between each k-space point and its eight closest spatial neighbors. This specific neighborhood size was carefully selected to: (1) preserve local spatial relationships in frequency space, (2) enable effective information propagation across adjacent frequencies, while (3) maintaining an optimal balance between computational efficiency and topological representation accuracy. The resulting connectivity pattern generated a total of 524,288 edges across the complete k-space graph, forming a dense but structured network that captures both local and regional frequency correlations essential for accurate MRI reconstruction.

This graph representation effectively transforms the traditional 2D k-space array into a network structure suitable for graph neural network processing while preserving all the original scan information.

- **3.2.3 Normalization**

Normalization is a post-processing step that scales output values to match the original data's range/distribution for proper metric evaluation and visualization.

Our implementation incorporated four normalization strategies:

(1) z-score standardization (μ=0, σ=1).

(2) logarithmic transformation for dynamic range compression.

(3) variance-based normalization.

(4) max-value normalization to unit range [0, 1].

Z-Score standardization: Centers data around zero mean (μ=0) with unit variance (σ=1) to standardize feature scales.

$$\text{Xnorm} = \frac{\text{X} - \mu}{\sigma} \qquad (3)$$

Where: X: data, μ : mean of the data , σ: standard deviation

Logarithmic transformation: Compresses dynamic range using log transformation, ideal for high-variance data.

$$Xnorm = \log(1 + X) \qquad (4)$$

Variance-based normalization : this function ensures that each feature is centered by subtracting its mean and scaled by dividing by its variance . A small constant (1e-8) is added to the denominator for numerical stability.

$$Xnorm = \frac{X - \mu}{\sigma^2 + \epsilon} \qquad (5)$$

Where X: data, μ : mean of the data, σ^2: variance and ε= 1e-8.

Maximum value normalization: also known as min-max normalization when the minimum is zero , it divides by maximum absolute value to constrain data to [-1,1] or [0,1] range.

$$Xnorm = \frac{X}{\max(X) + \epsilon} \qquad (6)$$

Where X: data and ϵ = 1e-8

- **3.2.4 Training**

From the GNN variants analyzed in Chapter 2 (GCN, GAT, etc.), Our implementation employs Graph Convolutional Networks (GCNs), selected over other GNN architectures for their spectral-domain efficiency in processing k-space data and streamlined implementation compared to attention-based variants like GATs.

The proposed Graph Convolutional Network (GCN) architecture is designed to process graph-structured data for tasks such as node regression or signal reconstruction on graphs. The model consists of two graph convolutional layers with batch normalization, non-linear activation, and dropout regularization to improve learning stability and generalization.

The network begins by inheriting from PyTorch's torch.nn.Module, enabling full integration with PyTorch's training and optimization tools. The model consists of the following components:

Graph Convolution Layer 1 (conv1)

A GCNConv layer maps the input features from a dimensionality of 2 (e.g., real and imaginary parts of k-space values) to 16 hidden features.

This layer performs message passing and aggregation over the graph structure defined by the adjacency information (edge_index).

Batch Normalization Layer

A BatchNorm layer normalizes the output of the first convolution across feature channels, Accelerates training and improves convergence by reducing internal covariate shift.

- **Non-linear Activation**

A LeakyReLU activation function introduces non-linearity into the model.

- **Dropout Regularization (dropout)**

Randomly zeroes out 50% of neuron outputs during training, Prevents overfitting and encourages robust feature learning.

- **Graph Convolution Layer 2 (conv2)**

The second GCNConv layer projects the 16-dimensional hidden representations back to a 2-dimensional output space.

- **Forward Pass and Optimization**

In the forward method:

• Input node features (x) and graph connectivity (edge_index) are extracted from the input data object.

• The first convolution is applied, followed by batch normalization, LeakyReLU activation, and dropout.

• Finally, the second convolution produces the predicted node features.

The optimization process of the proposed Graph Convolutional Network (GCN) model is based on the Adam optimizer, which is widely used for training deep learning models due to its adaptive learning rate mechanism and robustness across various tasks.

The Adam optimizer is used with an initial learning rate of 10−4. This value was chosen empirically as a good trade-off between fast convergence and stable learning dynamics.

• The L1 loss (Mean Absolute Error) is selected as the objective function:

$$L1(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \qquad (7)$$

Where y_i is the true value , and $\hat{y}_i$ is the predicted value.

After training the GCN model, the reconstructed image is evaluated using these parameters:

- **3.2.5 Evaluation metrics:**

Signal to noise ratio

The signal-to-noise ratio (SNR) is a fundamental factor in determining image quality, representing the ratio of useful signal to unwanted noise. It is mathematically defined as the mean signal intensity divided by the standard deviation of background noise. Higher PSNR indicates better reconstruction quality.

$$\text{PSNR} = 10 \,.\log_{10}\left(\frac{\text{L}^2}{\text{MSE}}\right) \qquad (8)$$

Where : L: Maximum possible pixel value, MSE = Mean Squared Error

The structural similarity index measure (SSIM)

SSIM (Structural Similarity Index Measure) is a widely used image quality metric that quantifies the similarity between a reference image (x) and a processed image (y) through three comparative components: luminance (l), contrast (c), and structure (s).

- ### 3.2.6 Methodological Pipeline

This section presents our hybrid approach, integrating graph neural networks (GNNs) with classical compressed sensing methods to enhance MRI reconstruction, combining their complementary advantages while mitigating inherent limitations

Figure 3.1 provides a synthesis of all the steps outlined above, offering a clear and organized representation of the overall methodology.

The study began with a subset of 1,200 k-space samples from the dataset. To address storage constraints associated with raw k-space data, we adopted an optimized processing strategy whereby each k-space was first transformed into image space through inverse Fourier operations, then subsequently reconverted to k-space format via Fourier coefficient extraction. This approach significantly reduced memory demands while preserving essential spectral information. Following data preparation, we applied undersampling using a variable-density mask designed to retain critical center-weighted frequencies (adhering to compressed sensing principles). Both fully sampled and undersampled datasets were decomposed into their real and imaginary components, which were then transformed into graph representations. These graphs underwent normalization to ensure numerical stability before being processed by our Graph Convolutional Network (GCN). The Graph Convolutional Network architecture processed the data across 60 training epochs, with mini-batch optimization driven by the Adam optimizer and L1 loss minimization between predicted and target node values. Post-training, the reconstructed outputs were demoralized, transformed back to k-space coordinates, and finally converted to image space through inverse Fourier transformation. Quantitative evaluation of reconstruction fidelity was performed using established metrics: Peak Signal-to-Noise Ratio for pixel-level accuracy and Structural Similarity Index for perceptual quality assessment.

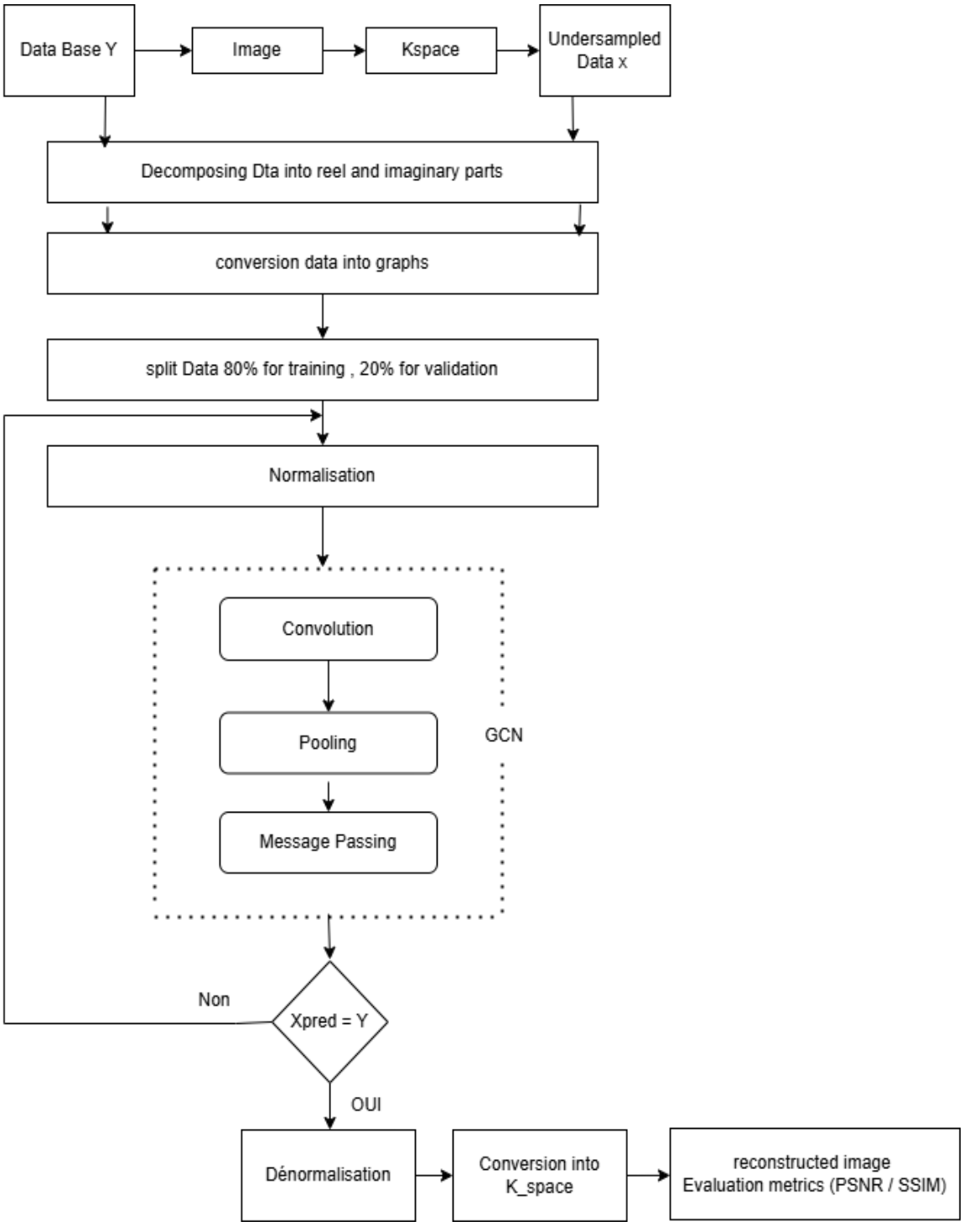


Figure 3.1: Organigram of the overall methodology

3.3 Results and discussion

To evaluate our method's performance, we present validation results. During our algorithm's execution, we observed that several factors affected image quality, including dataset composition, optimization parameters (particularly loss function selection), and normalization techniques.

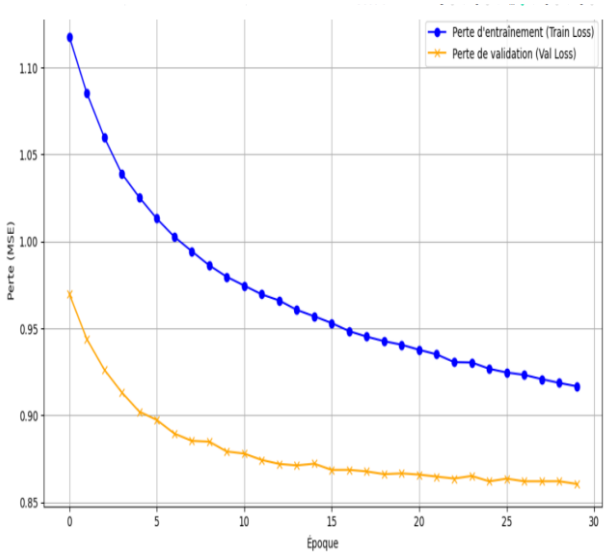
- **3.3.1 Dataset**

Due to RAM limitations on Kaggle, we initially worked with only 212 k-space samples, which resulted in degraded image quality. This showed that the model needs a lot more training data for better performance. To fix this, we converted the original k-space dataset into images. This reduced RAM usage and allowed us to use 1,200 samples. This adjustment led to noticeable improvements in the outcomes.

- **3.3.2 Loss function analysis**

We compared the performance of L1 loss versus mean squared error (MSE) while maintaining a fixed training period of 30 epochs. This experimental setup allowed us to systematically evaluate the impact of each loss function on model performance.

Figures 3.2(a) and 3.1(b) present the training and validation loss curves for MSE and L1 loss, respectively, demonstrating their comparative convergence behaviors.



a

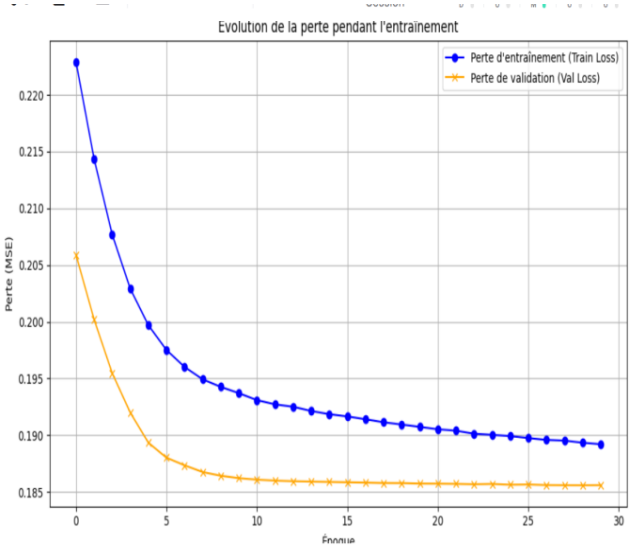


Figure3.2(a),(b): training and validation loss curves for MSE and L1 loss respectively

MSE Loss: The training loss starts above 1.10 and decreases to 0.9, while the validation loss begins above 0.95 and ends at 0.85. The persistent gap between the two curves suggests potential overfitting or greater sensitivity to outliers.

L1 Loss: in contrast, the training loss declines from 0.22 to 0.19, and the validation loss moves from 0.20 to 0.18. The tight alignment between these curves indicates better generalization and robustness.

A key observation is that the gap between the training and validation curves is significantly wider for MSE compared to L1 loss. Based on these results, we conclude that L1 loss is more efficient than MSE.

- **3.3.3 Epoch Comparison Analysis (L1 Loss)**

After *demonstrating* L1 loss's effectiveness, We tested the model with different training lengths: 30 epochs, 60 epochs, and extended training beyond 100 epochs.

Figure 3.3(a),(b),(c) presents the model performance at 30,60 and 100 training epochs respectively

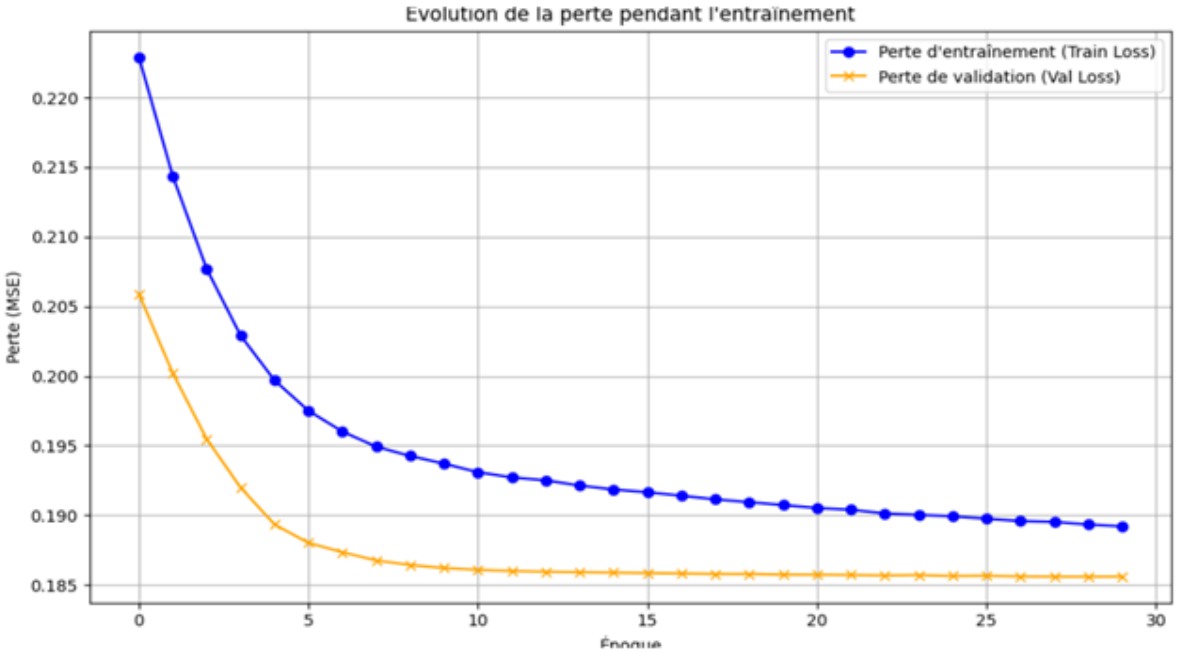


Figure. 3.3.a: Model performance at 30 training epochs

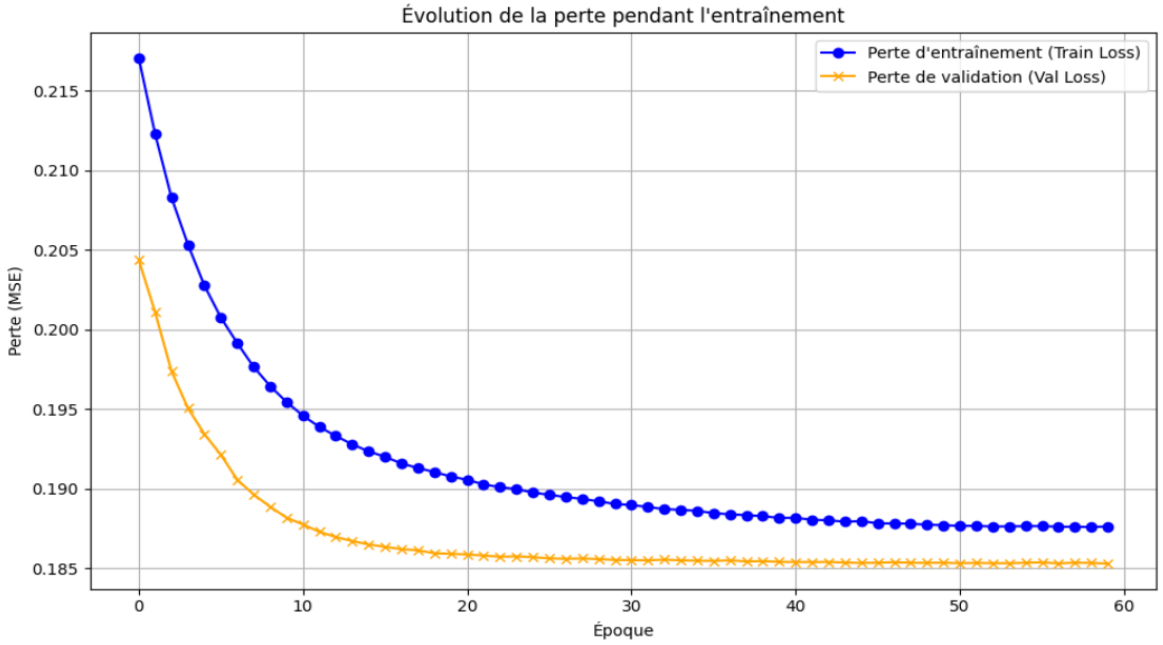


Figure 3.3.b: Model performance at 60 training epochs,

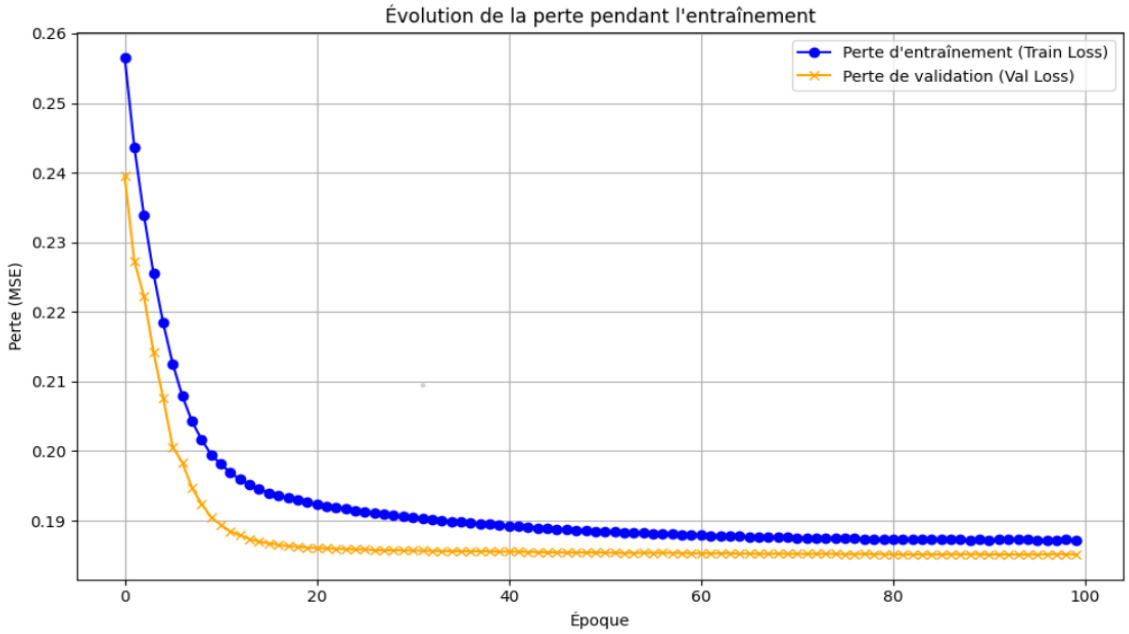


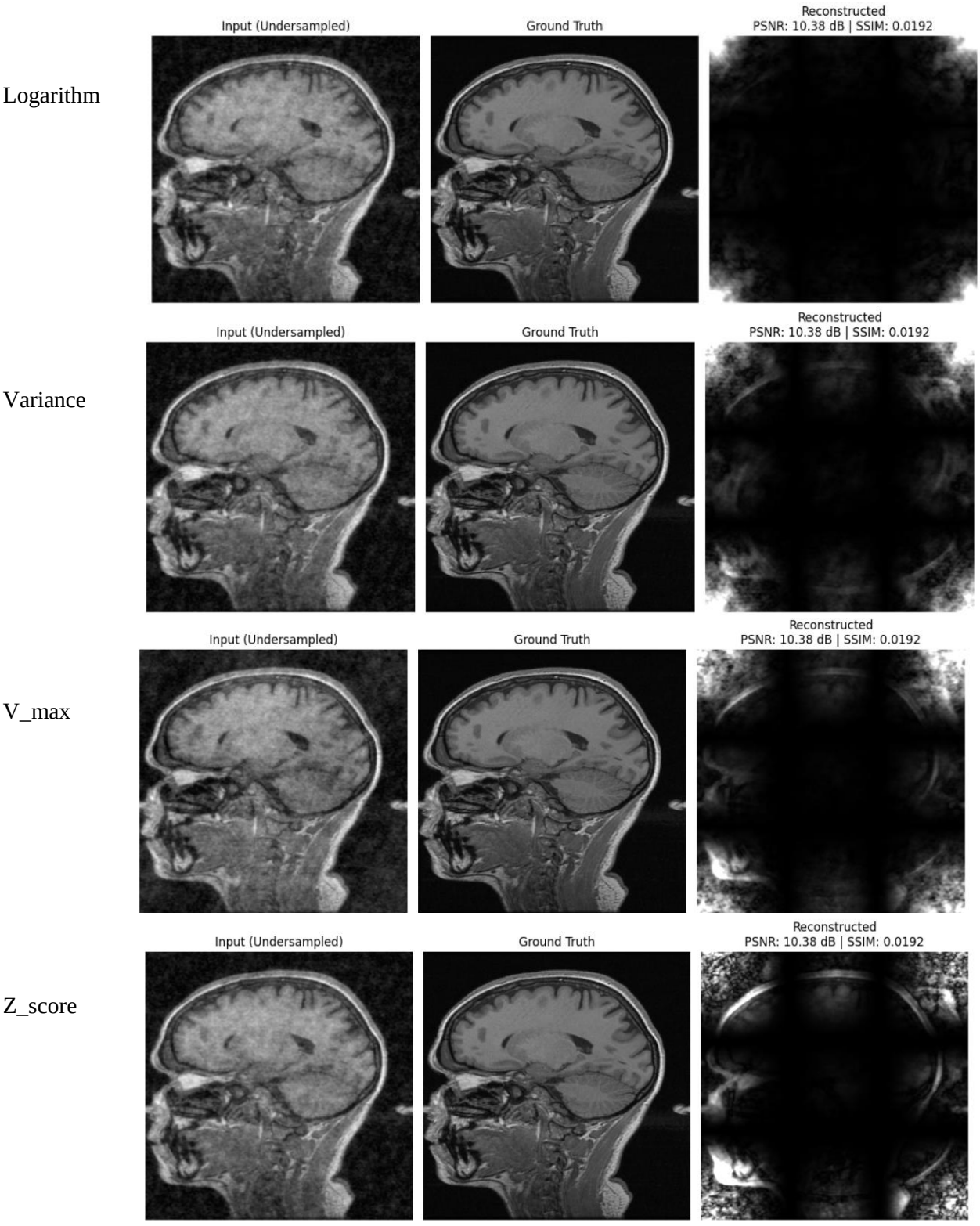
Figure 3.3.c**:** Model performance at 100 training epochs,

Training progresses up to 30 epochs, but model performance stabilizes after 60 epochs. Thus the Optimal choice is 60 epochs.

- ### 3.3.4 Normalization methods Evaluation

In our work, we examined and compared four distinct normalization approaches: Logarithmic Transformation, Vmax Normalization, Variance Normalization and Z-Score Normalization,

Figure 3.4 presents: (a) the VD_undersampled image, (b) the ground truth reference, and (c) a line-by-line comparison of CS-GNN reconstructions using four normalization methods (logarithmic, variance , Vmax, , and Z-score).



a b c

Figure 3.4 results of different normalization methods

We observe that image quality improves with each change in normalization method. The z-score normalization yields superior results compared to other methods. However, the PSNR and SSIM metrics remain unchanged despite the visual quality improvement. This occurs because normalization methods such as z-score, min-max, and variance normalization are linear transformations that preserve the underlying image structure. Specifically, these methods maintain the relative relationships between pixels after denormalization.

The optimal configuration L1 loss + Z-score normalization + 60 epochs delivered superior reconstruction quality, outperforming other parameter combinations.

Figure 3.5 presents: some examples of this optimal configuration. (a) the VD_undersampled image, (b) the ground truth reference, and (c) the CS_GNN predicted image.

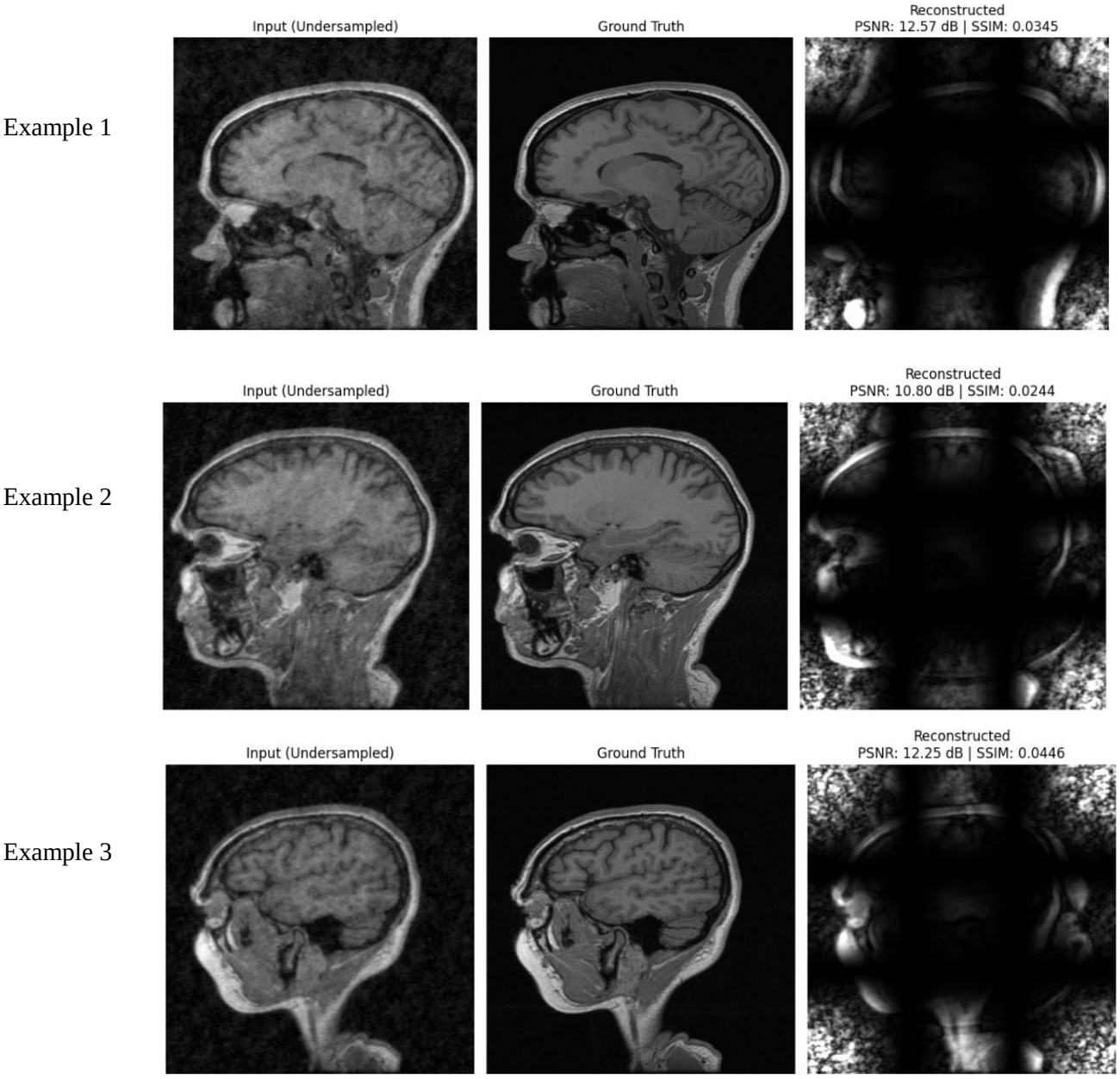


Figure 3.5 examples of CS_GNN reconstruction uunder optimal parameter configuration

(Z_score, L1 loss, and 60 training epochs)

To improve results, we thought to add CNN processing after GNN training for better outcomes capitalizing on CNNs' strong compressed sensing performance.

First, we conducted a preliminary evaluation of the CNN's performance in compressed sensing (CS) for each sampling mask configuration.

- **3.3.5 CNN performance evaluation**

We evaluated the CNN model (simple architecture) using standard compressed sensing undersampling patterns: random and variable density (VD) masks *(previously defined in Compressed sensing section).*

The figure 3.6 Compares CNN reconstruction performance using: (a) Random undersampling mask , (b) Variable density (VD) undersampling mask

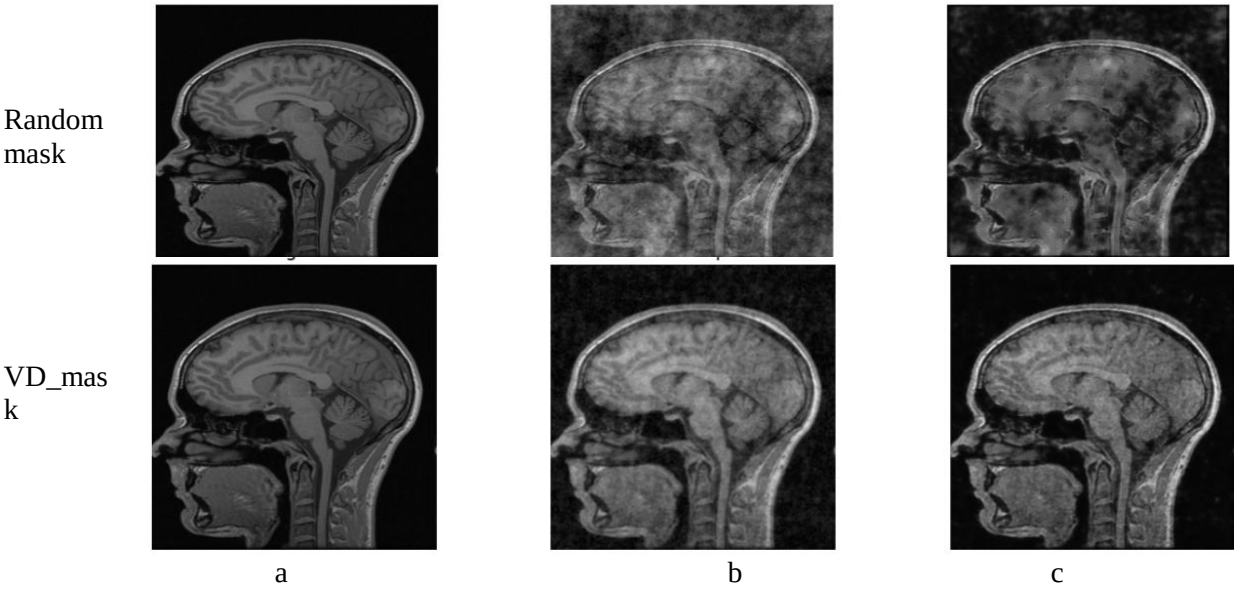


Figure 3.6: evaluation of CNN performance using random undersampling and VD undersampling masks.

The variable density (VD) mask demonstrates significantly better performance than random sampling.

CNNs show strong compatibility with compressed sensing (CS) frameworks

Based on these promising results, we proposed enhancing the CS-GNN outputs through CNN-based post-processing.

The figure 3.7 presents some examples after the post_processing.

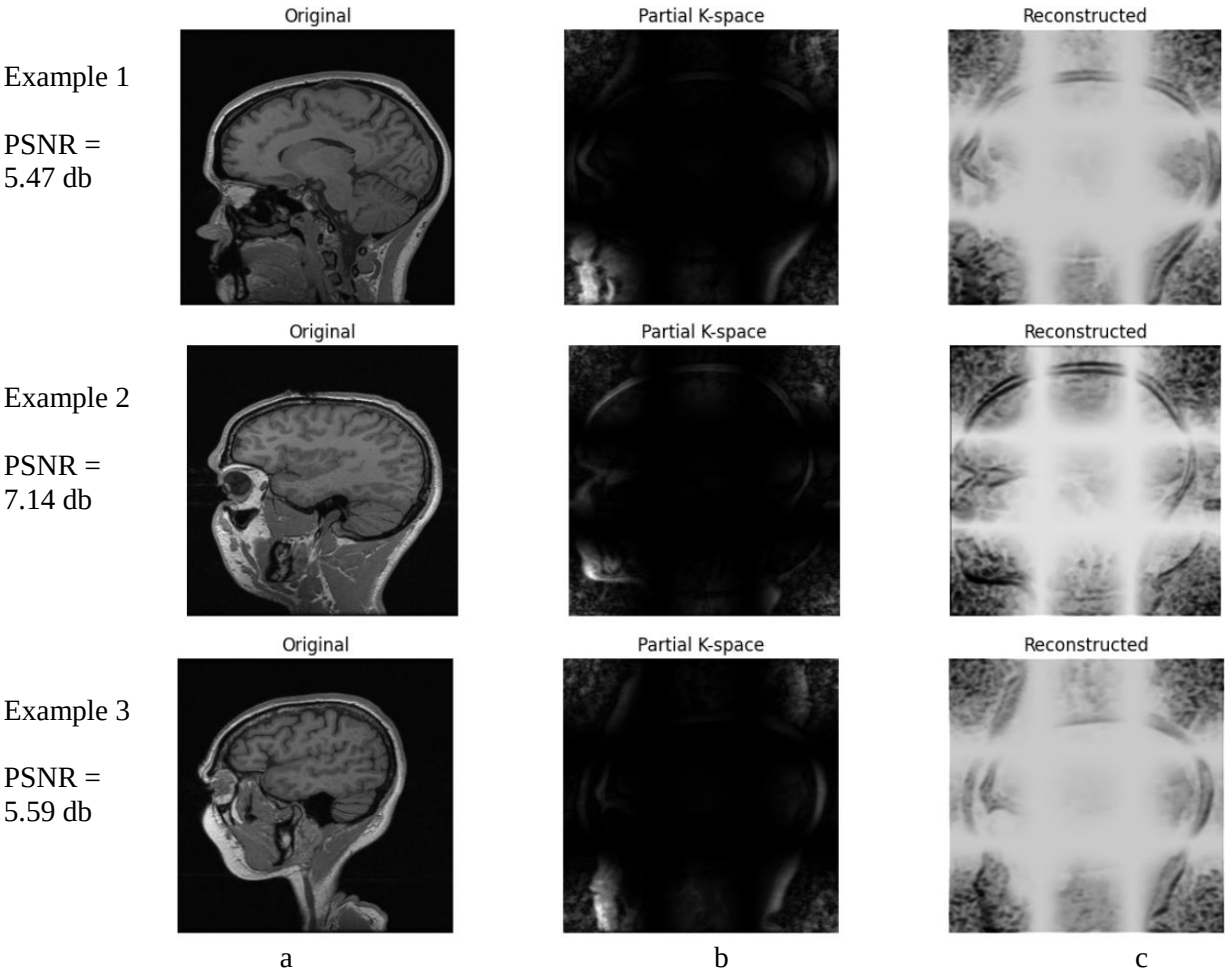


Figure 3.7: some examples of CNN_CS_GNN reconstruction (post processing)

While the CNN enhanced visual contrast in reconstructed images, overall quality remained degraded. Quantitative metrics confirmed the post-processing's limited efficacy.

- **3.3.6 Discussion**

Through comprehensive testing, we identified key factors influencing reconstruction quality:

- **Dataset Impact** : Expanding the database size improved both qualitative and quantitative image reconstruction quality - as the number of training images increased, finer image details became more apparent. However, the GNN's computationally intensive architecture requires substantial storage capacity, which limited our practical dataset size to 1,200 images.
- **Optimization Results**: Our results showed that the choice of the loss function is critical, we noticed that the image quality improves much more with the choice of L1

loss. Thus by increasing the number of epochs the quality improves but always by monitoring the over-learning. This aspect still remains to be studied as a proposal in order to achieve more relevant results.

- **Normalization Results**: Comparative evaluation of normalization techniques demonstrated that z-score normalization achieved superior performance, enhancing image quality both quantitatively and qualitatively (improved structural preservation).
- Following comprehensive testing, our results highlight the need for efficient quality improvements motivating our implementation of streamlined CNN post-processing. The hybrid CNN_CS_GNN approach demonstrated that while post-processing enhances outputs, it cannot fully compensate for acquisition-level limitations.

Our analysis identified L1 loss, 60 epochs, and z-score normalization as the optimal configuration for maximizing image quality within current system constraints.

Our MRI reconstruction study using the CS-GNN approach identified multiple factors significantly impacting output image quality.

The findings discussed above indicate that our proposed method requires further algorithmic refinements, enhanced **computational resources** (particularly high-performance workstations), and **additional development time** to address current quality limitations**.**

Conclusion

Our study proposed a novel integration of compressed sensing (CS) with graph neural networks (GNN) to enhance MRI reconstruction quality. Experimental results demonstrated that three key factors improved outcomes: Expansion of the training dataset, Implementation of L1 loss function, Application of z-score normalization

Despite these improvements, reconstructed images still exhibited quality limitations. We subsequently incorporated a convolutional neural network (CNN) as a post-processing stage. The persistent quality issues led us to two critical conclusions:

- Post-processing techniques are ineffective when applied to fundamentally flawed reconstructions
- Meaningful quality enhancement requires optimization of the core reconstruction model architecture.

General Conclusion

The Graph Neural Network is a widely used method that has proven its effectiveness in different domains such as: social networks, molecules... However, GNN is underdeveloped in the field of MRI image reconstruction.

Various reconstruction methods have been developed recently such as SENSE, GRAPPA, and Compressed Sensing to reduce acquisition time while maintaining good image quality.

Through this final year project, we proposed an approach that combines the most recent CS method with GNN to study its performance in the acquisition domain.

Our study showed that different factors influence the image quality predicted by our approach such as: the limited database size, the choice of loss function, and normalization. By adjusting these different factors, the results show that the optimal parameters could not guarantee good quality.

Our CS_GNN approach can be improved under better conditions, these conditions require more time with powerful resources like using a high-performance computing station.

With these limited conditions, we thought to add an effective and fast post-processing using a simple CNN architecture. However, this post-processing did not achieve the desired (expected) results.

We can conclude that post-processing is a very important phenomenon in image processing, but it does not work effectively on the quality of corrupted images from reconstruction methods (acquisition domain).

Future Perspectives

Future work will focus on:

- Full dataset utilization,
- Advanced loss function combinations.
- GNN architectural improvements including GAT implementations, requiring upgraded computational resources for optimal results.

References

[1] Johns Hopkins Medicine, "Magnetic Resonance Imaging (MRI)," *Johns Hopkins Medicine*, [Online]. Available: https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/magnetic-resonance-imaging-mri. [Accessed: Jun. 26, 2025].

[2] **Westbrook C,** ***MRI at a glance,*** **4th Edition**, Wiley-Blackwell, 2021.

[3] Brown, M. A, Semelka, R. C, MRI: Basic principles and applications, 5th edition, Wiley-Blackwell, 2019.

[4] *Introduction to MRI,* Radiology Cafe. Available at: https://www.radiologycafe.com/frcr-physics-notes/mr-imaging/introduction-to-mri/ (Accessed: 22/02/2025).

[5] Schild, H. H, *MRI made easy,* 5th edition, 2020.

[6] Chavhan, G. B, *MRI made easy (for beginners)* 2nd edition, Jaypee Brothers Medical Publishers, 2017.

[7] Physics, Available at: https://physics.stackexchange.com/questions/820213/why-the-t-1-recovery-t-2-decay-are-independent (Accessed: 22/03/2025)

[8] Bian, W, "*A brief overview of optimization-based algorithms for MRI reconstruction using deep learning",* arXiv:2406.02626, 2024. https://doi.org/10.48550/arXiv.2406.02626

[9] Govind B. C, MRI made easy, 2nd edition, chapter 1, 2013.

DOI: 10.5005/jp/books/11814.

[10] Bustin A, *Advanced reconstruction techniques in free-breathing multi-contrast high-resolution cardiac magnetic resonance imaging* [Conference abstract]. *Journal of Cardiovascular Magnetic Resonance,* 2017.

[11] Perry S, *Magnetic resonance imaging : principles, methods, and techniques,* Medical Physics Publishing, Madison, Wisconsin, WI 53705-4964, 2000,

[12] Arghya P, Yogesh R, A review and experimental evaluation of deep learning methods for MRI reconstruction, Mach Learn Biomed Imaging, 2022

[13] Samuel Cahyawijaya, Biomedical Image Reconstruction: A Survey, Electrical Engineering and Systems Science, Image and Video Processing, 2023

[14] Md Biddut Hossain , Rupali Kiran Shinde , Sukhoon Oh, Ki-Chul Kwon, Nam Kim, A Systematic Review and Identification of the Challenges of Deep Learning Techniques for

Undersampled Magnetic Resonance Image Reconstruction, Sensors. 24(3):753, , 2024. doi: 10.3390/s24030753

[15] Dilbag Singh et al., Emerging Trends in Fast MRI Using Deep-Learning Reconstruction on Undersampled k-Space Data: A Systematic Review, Bioengineering;10(9):1012, 2023. doi: 10.3390/bioengineering10091012

[16] Julio A Oscanoa et al, Deep Learning-Based Reconstruction for Cardiac MRI: A Review, Bioengineering, 10(3):334, 2023. doi: 0.3390/bioengineering10030334

[17] Mojtaba Safari, Zach Eidex , Chih-Wei Chang , Richard L J Qiu , Xiaofeng Yang, Advancing MRI Reconstruction: A Systematic Review of Deep Learning and Compressed Sensing Integration, ArXiv [Preprint].arXiv:2501.14158v2, 2025

[18] Henry Senior · Gregory Slabaugh · Shanxin Yuan · Luca Rossi , "Graph neural networks in vision-language image understanding: a survey " The Visual Computer 2025

[19] Pradhyumna P, Shreya G P, Mohana, Graph Neural Network (GNN) in Image and Video Understanding Using Deep Learning for Computer Vision Applications, Second International Conference on Electronics and Sustainable Communication Systems (ICESC), IEEE Xplore Part Number: CFP21V66-ART; ISBN: 978-1-6654-2867-52021, 2021. DOI:10.1109/ICESC51422.2021.9532631

[20] SHOWMICK GUHA PAUL et al., A Systematic Review of Graph Neural Network in Healthcare-Based Applications: Recent Advances, Trends, and Future Directions, IEEE Access 12: 15145-15170, 2024

[21] Qiaoyu Ma , Zongying Lai, Zi Wang, Yiran Qiu , Haotian Zhang , Xiaobo Qu, MRI reconstruction with enhanced self-similarity using graph convolutional network, BMC Med Imaging, 17;24(1):113, 2024. doi: 10.1186/s12880-024-01297-2.

[22] D. L. DONOHO, "Compressed sensing". IEEE Transactions on Information Theory, 52(4):1289–1306, 2006.

[23] Pramila P. Shinde; Seema Shah, A Review of Machine Learning and Deep Learning Applications, Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018. **DOI:** 10.1109/ICCUBEA.2018.8697857

[24] Roberto Souza, Oeslle Lucena, Julia Garrafa, David Gobbi, Marina Saluzzi, Simone Appenzeller, Letícia Rittner, Richard Frayne, and Roberto Lotufo. "An open, multi-vendor,

multi-field-strength brain MR dataset and analysis of publicly available skull stripping methods agreement."*NeuroImage* 170 (2018): 482-494.