University of Brasilia at Gama – FGA/UnB
Software Engineering

Generative Adversarial Network Prior Information for Improved Compressed Sensing Magnetic Resonance Image Reconstruction

Gabriel Gomes Ziegler

Advisor:Cristiano Jacques Miosso, PhD
Co-advisor:Davi Benevides Gusmão, MSc

UnB – University of Brasilia

FGA – Gama Faculty

Software Engineering

Gabriel Gomes Ziegler

Advisor: Cristiano Jacques Miosso, PhD

Co-advisor: Davi Benevides Gusmão, MSc

Bachelor Thesis

Software Engineering

Brasilia/DF, September 2020

UnB – University of Brasilia
FGA – Gama Faculty
Software Engineering

Gabriel Gomes Ziegler

Submitted in partial fulfillment for the Bachelor Degree in Software Engineering (Honor's Degree) to the department of Engineering of University of Brasilia

Approved by:

Cristiano Jacques Miosso, PhD

(Advisor)

Prof. José da Silva, PhD

Prof. João da Silva, PhD

Brasilia/DF, September 2020

Ficha Catalográfica

Ziegler, Gabriel
Título para a ficha,
[Distrito Federal], 2018.
xxxvp., 210 × 297 mm (FGA/UnB Gama, Bachelor of Engineering in Software Engineering, 2020).
Bachelor Thesis, UnB Gama Faculty, Software Engineering

1. Deep Learning
2. Magnetic Resonance Image Reconstruction
3. *Compressive Sensing*
4. Generative Adversarial Network

I. FGA UnB/UnB.
II. Título (série)

Reference

Ziegler, Gabriel (2020). Título para a ficha (parte 2). Trabalho de Conclusão de Curso, Engenharia de Software, Faculdade UnB Gama, Universidade de Brasília, Brasília, DF, xxxvp.

Cessão de Direitos

---

gabrielziegler3@gmail.com

Brasília, DF – Brasil

Resumo

A versão final do documento incluirá um resumo de todo o trabalho, incluindo metodologia, resultados e conclusão.

Abstract

The final version of this document will include an abstract. This will summarize the introduction (contextualization, objectives, justification), the methodology, the results, and the conclusion.

Contents

List of Tables

List of Figures

Nomenclature and Abbreviations

MRI Magnetic Resonance Imaging xiv

MR Magnetic Resonance xiv

ANN Artificial Neural Networks x

DL Deep Learning xv

ML Machine Learning xiv

GAN Generative Adversarial Network xiv

CNN Convolutional Neural Network xv

GPU Graphics Processing Units xiv

CV Computer Vision xv

NLP Natural Language Processing xv

CS Compressed Sensing xiv

MLP Multilayer Perceptron xxi

DFN Deep Feedforward Networks xxi

SNR Signal-to-Noise Ratio x

1 Introduction

In this thesis, we propose a Generative Adversarial Network (GAN) approach for prior information extraction to feed a Compressed Sensing (CS) algorithm, aiming to reconstruct images with both reduced signal-to-noise error and less acquisition time compared to conventional CS. Achieving higher quality with reduced number of samples allows faster exam procedures, making Magnetic Resonance Imaging (MRI) cheaper, faster and more convenient for both patients and clinics.

1.1 Context

MRI is a widely used imaging modality in medical practice because of its great tissue contrast capabilities, it has evolved into the richest and most versatile biomedical imaging technique today[6], making MRI the best option for medical imaging whenever it is possible to use.

However, like everything in life, there is a trade-off to consider when using MRI. Typically, reconstructing an MRI is an ill-posed linear inverse task (a problem that has either none or infinite solutions in the desired class). Problems of this nature impose a trade-off between *accuracy* and *speed*[7]. The information obtained from Magnetic Resonance (MR) is commonly represented by individual samples in the k-space, which translates to the Fourier transform of the image to be reconstructed[8]. This MR sampling sparse nature makes CS a liable technique to use when reconstructing MRI, hence we here propose a novel CS prior information approach for better results.

CS has been for years the state-of-art technique in MRI reconstruction and has been improved later by the use of prior information[8]. CS uses the premise that given a signal with a sparse representation in some known domain, it is possible to reconstruct the signal using limited linear measurements taken from a non-sparse representation.

Machine Learning (ML) methods have been utterly developed and improved recently with the use of higher computing power derived from the invention of Graphics Processing

Units (GPU) and other hardware improvements, allowing ANN to come to practicality. These ANN models, often referenced as Deep Learning (DL), have become the state-of-art in various areas, such as Computer Vision (CV), Natural Language Processing (NLP), Recommendation Systems, amongst other fields[9, 10, 11]. These fast-paced developments led to improvements in medical data processing using DL as well. ML techniques can be used in several different manners to improve medical analysis, here we focus on applying GAN in the process of attaining improved prior information to feed the CS algorithm obtaining higher signal-to-noise ratios and faster computation procedures.

1.2 Scientific Problem Definition and Proposal

MRI is great for high-quality tissue images, but there are some drawbacks: MRI exams are often very long and require the patient to be in a static position throughout the whole process, this makes the exam challenging for patients that have difficulties in keeping a still position for several minutes. Another intrinsic complication in MRI procedures is that it is nearly impossible to get images from moving tissues like a beating heart or flowing blood veins as that would require an enormous amount of samples, which with current technologies used in clinics is not viable. Algorithms that reconstruct MRI try to tackle this sampling issue by producing the best possible quality images for the least amount of samples collected, making the exams faster and less sample-dependent.

CS algorithms have been the state-of-art in MRI reconstruction for the past few years, and now with the advances of DL, new techniques are being produced taking advantages of how ANN are powerful in imaging processing, especially Convolutional Neural Network (CNN) and, more recently, GAN networks are becoming the new state-of-art techniques in several computer vision areas. A problem with CS applications is that the reconstruction process can be very slow. Newer CS algorithms try to tackle this issue by adding prior information to make the algorithm abstract static information in the region analysed.

Prior information for CS can go from previous MRI frames and exams to even medical records. Prior information is normally generated by simplistic mathematical approaches like filtering and thresholding on the images. Besides the simpler technique applied, these information extraction procedures oftentimes is restricted to few frames and does not take into account the nature of organs and tissues structures, a feature that DL should be able to identify and use in order to generate better quality information. This means that there is a lot of room for improvement towards prior information engineering techniques, as DL models have been proven superior in tasks of this nature.

Within this context, we propose a modern prior information engineering system with the usage of GAN, aiming for higher quality prior information to feed the CS and reducing the number of samples dependability. This will reduce the number of samples needed, making the MRI exams faster and, consequently, cheaper.

1.3 Objectives

1.3.1 General Objective

This thesis' goal is to develop an MR prior information system retriever based on GAN architecture to analyse if the quality of the prior information fed to CS algorithms can be improved, hence improving quality in reconstructed MRI and decreased necessity for larger sampling.

1.3.2 Specific Objective

In order to achieve the general objective described above, we have set the following specific goals:

- Implement direct and indirect CS MRI reconstruction algorithm and apply to *k-space* measurements.
- Evaluate CS MRI reconstructions with real image data.
- Implement a GAN for regular image generation with a known CV dataset.
- Implement a GAN architecture for prior information retrieval and train it against *k-space* measurements.
- Evaluate the use of GAN architecture for prior information retrieval against state-of-art prior information techniques.

2 Theory Foundation and State-of-Art

2.1 Magnetic Resonance Imagery

MRI is an indirect process that produces cross-sectional images with high spatial resolution from nuclear magnetic resonances, gradient fields and hydrogen atoms of the subject's anatomy[12]. The acquisition of these signals is performed by a measuring intrument called *receiver coil* and it can be done by using one receiver coil or in some cases with multiple coils[1]. These receiver coils are placed in proximity to a specific region in the subject to be imaged. During the imaging process, the MRI machine generates a sequence of spatially and temporally-varying magnetic fields which indulce the body to emit resonant electromagnetic response fields which are then measured by the receiver coil[1].

2.1.1 K-space

The k-space is the output generated by the MRI machine scan after extracting measurements from a given subject tissue. The k-space is represented in the spatial frequency in two or three dimensions of a subject and may also be referred as the Fourier space. This k-space representation contains an implicit sparsity that is exploited when performing undersampling[13] and reinforce the usage of algorithms like CS for MRI reconstruction as CS depend on signals that have a sparse representation in an orthonormal basis.

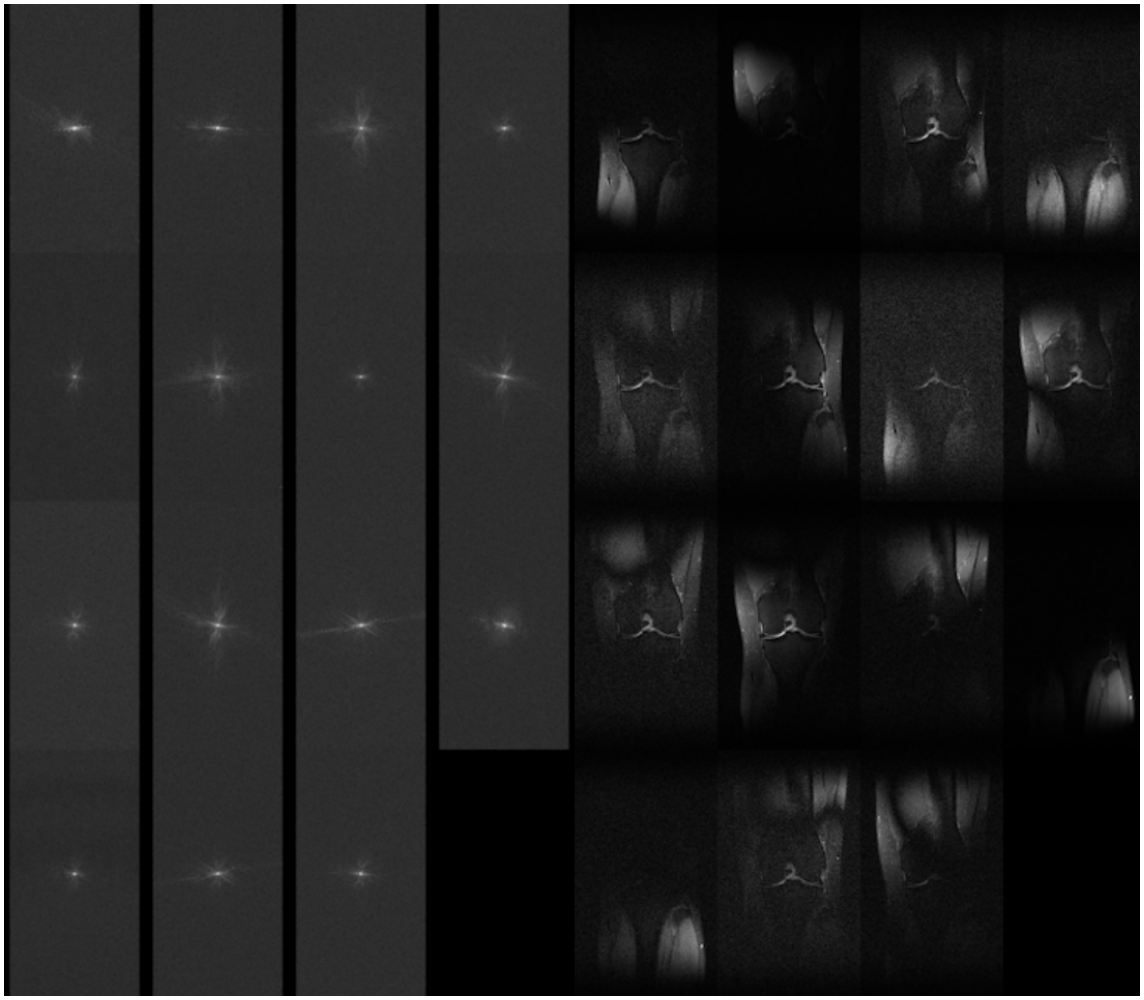

Figure 2.1. (a) FastMRI K-space data from 15 coils (b) FastMRI individual fully sampled coil spatial images[1].

2.1.2 Sampling

The time required to acquire all the measurements responses from every single atom in a subject would be extremely high and problematic to every one involved (pacients, doctors and clinics). The way machines can do faster MRI is by performing *undersampling*, also referred as subsampling and sampling, when scanning the subject.

Undersampling is performed by giving the machine a known prescribed path in which it will extract measurements from the multidimensional k-space representation. This allows machines to collect only a fraction of data measurements needed for image reconstruction hence speeding up the data acquisition process without critical quality loss.

There are some undersampling patterns to use and each has its benefits depending

on several parameters, such as: the subject's region extraction, algorithm used for reconstruction, acquisition time.

In the figure below we can see some of the most used patterns. In this research, we will focus mostly on the cartesian undersampling method, as that is the one used in the FastMRI dataset[1], which we will use for our experiments.

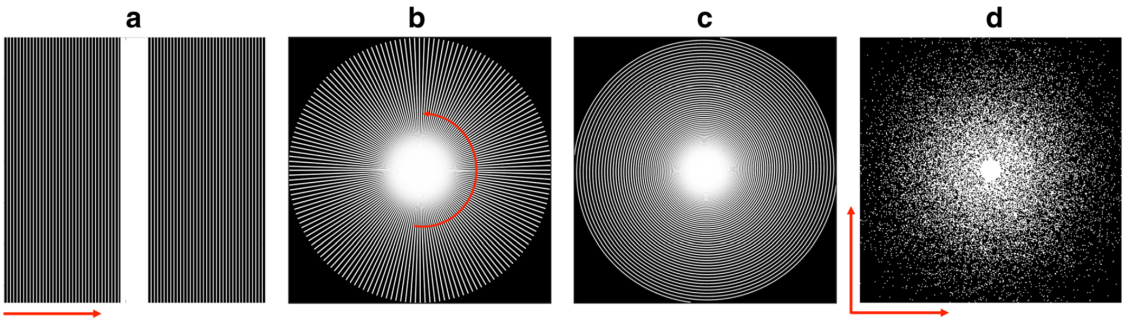

Figure 2.2. Under-sampling patterns. **(a) Cartesian undersampling**, **(b) radial undersampling**, **(c) spiral undersampling**, **(d) isolated samples in the k-space, according to the realisation of a random process**[2].

2.2 Compressed Sensing

CS is an extremely powerful algorithm that was introduced in 2004 proposing a novel technique for the acquisition of signals of sparse or compressible nature. CS has disrupted the signal processing field as it has broken the *Shannon's theorem*: the sampling signal rate must be at least twice the maximum frequency present in the signal (Nyquist rate). CS has been proven to sample the signal at a much lower rate than the Nyquist sampling rate. In MRI, when *k-space* is undersampled, the Nyquist criterion is violated.[13]

The idea was inpired from questioning the necessity of extracting large portions of samples when much of these samples are discarded, exposing the inefficiency of trying to gather all signal.

?Why go to so much effort to acquire all the data when most of what we get will be thrown away? Can we not just directly measure the part that will not end up being thrown away?[14]**?**

CS tackles the necessity to reconstruct signals with

CS parts from the principle that if given x, a digital image or signal has a sparse representation in an orthonormal basis (e.g.wavelet, Fourier), then the N most important coeffiecients in that expasion allow reconstruction with l_2 error $O(N^{1/2-1/p})$[14].

2.3 Artificial Neural Networks

2.3.1 Biological Inspirations

ANNs, as the name suggests, have been (loosely) inpired by biological neural networks (brains) from animals. The concept of using many layers of vector-valued representation is drawn from neuroscience. The choice of the functions $f^{(i)}(x)$ used to compute these representations is also loosely guided by neuroscientific observations about the functions that biological neurons compute[15]. Another trait they share is that just like the human brain can be trained to pass forward only meaningful signals to achieve larger goals of the brain, the neurons on a neural network can be trained to pass along only useful signal[3].

2.3.2 Multilayer Perceptron

The most basic unit in ANNs is the *artificial neuron*. These artificial neurons that are modeled mirroring the biological neurons behaviour as both of them are stimulated by inputs. Each artificial neuron play an analogous role to a neuron carrying some information they receive to other artificial neurons in the ANN. Artificial neurons take in inputs $x_1, x_2, \ldots, x_n$, each and multiply them by their respective weights $w_1, w_2, \ldots, w_n$. Then these weighted inputs are summed together producing the *logit* of the artificial neuron, $z = \sum_{i=0}^{n} w_i x_i + b$, with b being a constant number added called *bias*. After this, the logit is passed to a function f in order to generate the value $y = f(z)$.

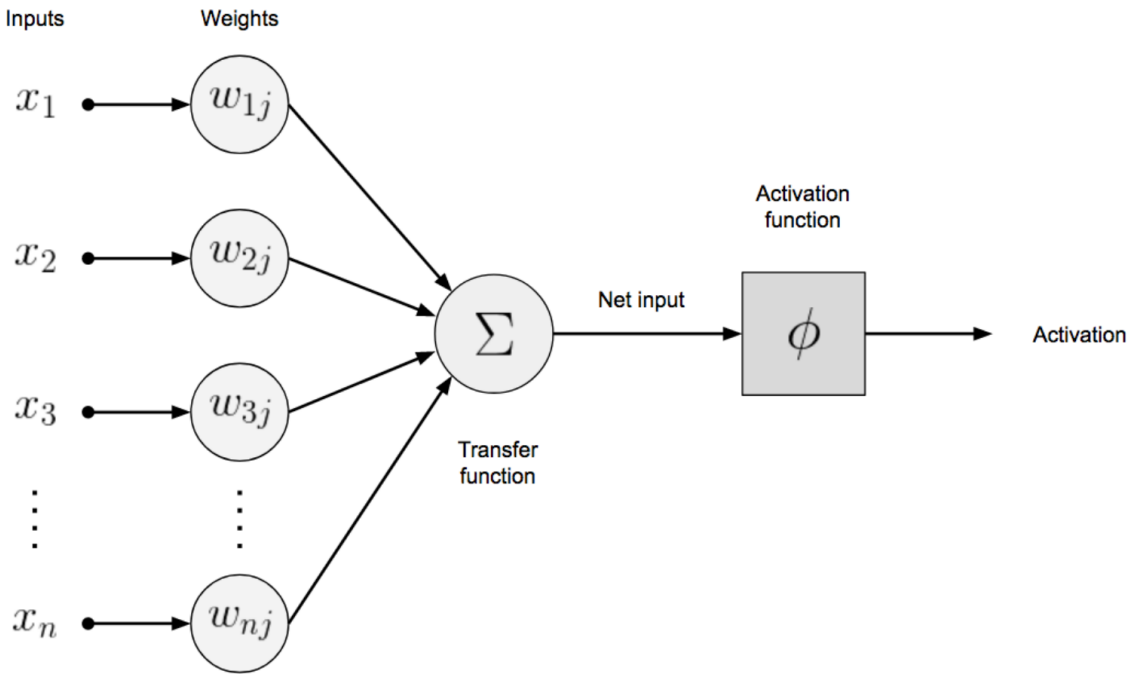

Figure 2.3. Schematic of an Artificial Neuron.[3]

Deep Feedforward Networks (DFN) or Multilayer Perceptron (MLP)s are a type of ANN very commonly used. It is the foundation to many famous architectures like CNNs. DFNs have an input layer followed by one or many hidden layers and a single output layer. Each layer is fully connected to the adjacent layer.

MLPs are computational models that flow information throuth the function being evaluated from $\mathbf{x}$. The goal is to approximate some function $f*$, for instance, for a classifier, $y = f*(x)$ maps an input x to a category y. The feedforward defines a mapping $y = f(x;\theta)$ and learns the value of the parameters θ that result in the best function approximation[15].

The behaviour of an ANN is shaped by its architecture, which describes the number of units it should have and how these units connect to each other.

Most ANNs are organized into rows of neurons called layers. These layers are arranged in a chain-like structure, with each layer being a function of the layer before it. These layers' goal is to extract *representations* out of the data fed and generalize what is meaningful towards minimizing the error rate. This architecture scheme is represented by the following equation:

$$h^{(i)} = g^{(i)}(W^{(i)T}x + b^{(i)})$$

Where i is the layer index

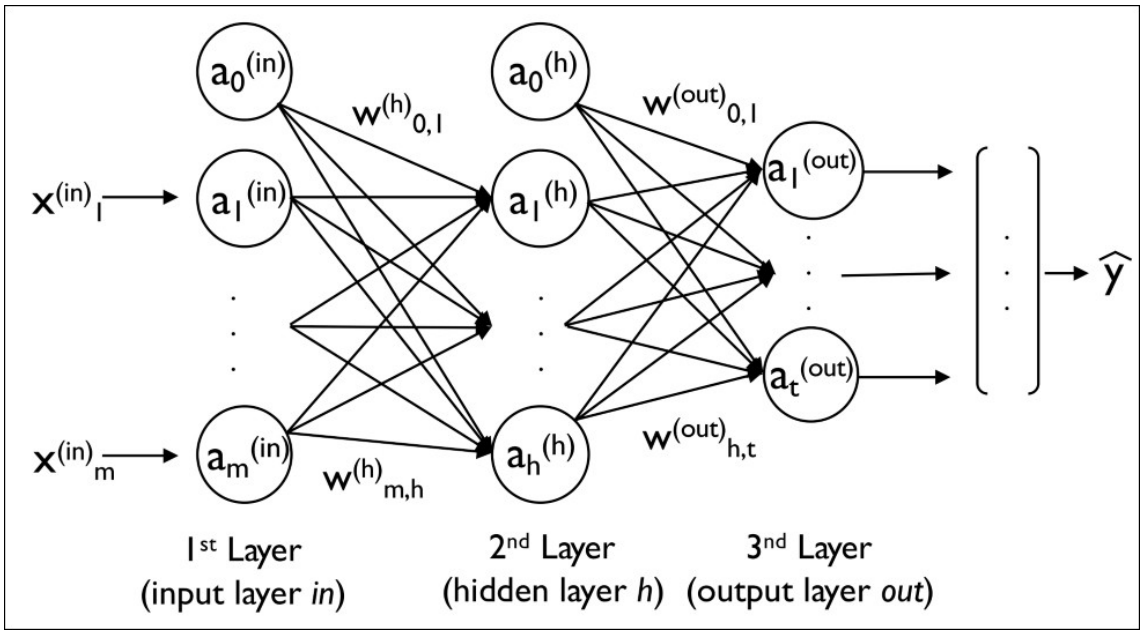

Figure 2.4. ANN Architecture Sample.

2.3.3 Activation Functions

2.3.4 Loss Functions

2.3.5 Backpropagation

2.3.6 Gradient Descent

2.4 Generative Adversarial Networks

2.4.1 MRI Application

3 Methodology

3.1 1-D Direct vs Indirect L1-Minimization

L1-minimization admits both direct and indirect approaches, in which there is the accuracy x resources trade-off. The direct method often produces a higher quality reconstruction but is very memory consuming, whilst the indirect method loses a little bit of quality, but requires much less memory to compute the equations system.

To visualize this trade-off, I have created 200 random 1-D arrays ranging values from the standard normal distribution and have taken 10% of data points randomly to reconstruct the whole signal using different L sizes. The L variable denotes the The figure below displays the first 10 signals created.

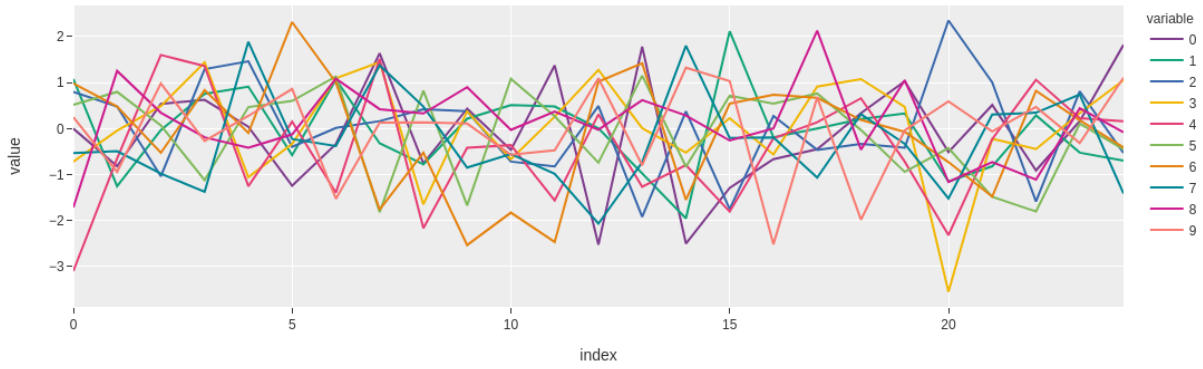

Figure 3.1. First 10 random 1-D signals

3.2 Phantom Compressed Sensing Reconstruction with Pre Filtered Signal

To test the method of applying pre-filtering to the input signals in the k-space, I have conducted an experiment with the well-known Shepp-Logan phantom[16] to evaluate how the usage of sparsifying filters impact the total-variation minimization[8].

3.2.1 Subsampling

I then created an phantom image with dimension of 256×256, hence 65536 data points, using the phantominator python module. Then, I simulated an undersampled phantom image by applying the spiral undersampling pattern achieving approximately 30.95% of data points from the fourier space which resulted in a matrix with 20285 non zero elements.

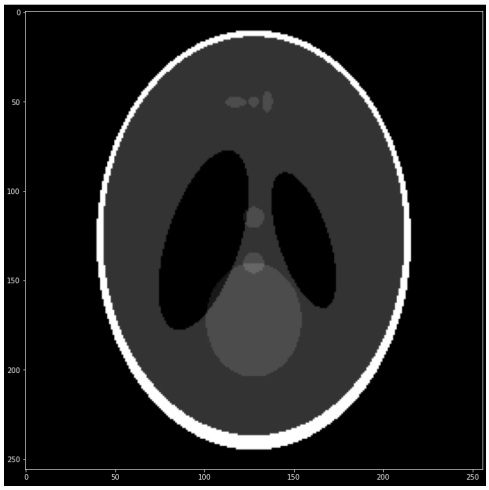

Figure 3.2. 256×256 Shepp-Logan phantom.

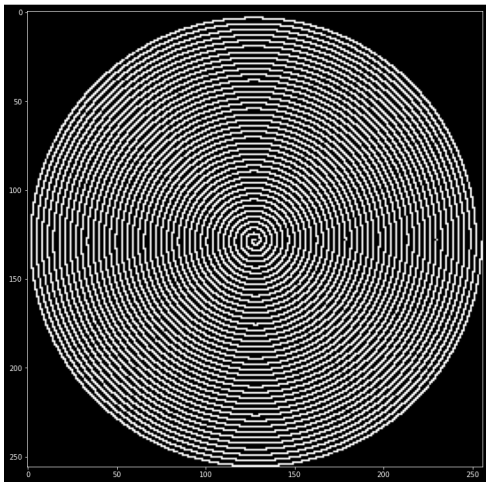

Figure 3.3. Spiral undersampling method.

3.2.2 Pre-filtering sparsifying transform

For the pre-filtering step, three filters are used to increase sparsity in the signal to be reconstructed. The filters are all 2×2 matrices and increase the sparsity in the signal from different perspectives, using more filters leverages the ability to sparsify the signal. The filtered images are then composed to one single image containing the highest gain each filter could provide given a single pixel in the image[8]. The different filters used can be better seeing in the figure below.

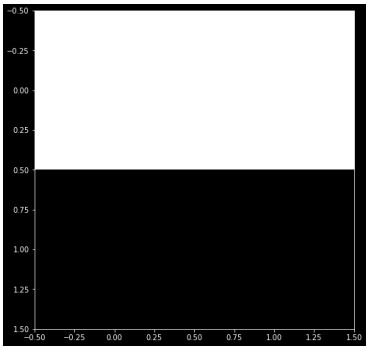

Figure 3.4. High pass horizontal filter.

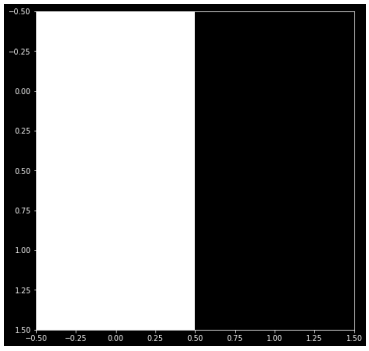

Figure 3.5. High pass vertical filter.

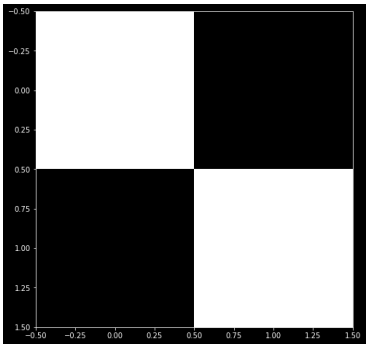

Figure 3.6. High pass diagonal filter.

The pre-filtering method is evaluated against the zero-fill reconstruction method (used as a dummy baseline) and a L1-minimization method without pre-filtering with the very same parameters used in the pre-filtering L1-minimization.

3.3 Preliminary Tests with Generative Adversarial Networks

In order to test the usage of GANs for data generation and in the future use it along with *prior information* for CS systems, I have developed a GAN capable of generating handwritten digits from 0 to 9 using the notable MNIST dataset. The MNIST dataset contains 60,000 examples for training and 10,000 examples for testing. The digits have been size-normalized and centered in a fixed-size image (28×28 pixels) with values from 0 to 9. For simplicity, each image has been flattened and converted into a 1-dimensional numpy array of 784 features (28×28).

The idea is to test if the neural network can output liable digits that look both readable (to the extent in which the MNIST dataset is) and also like it has been made by a human, just like the dataset itself.

Each MNIST image contains a 28×28 black and white image, like the following:

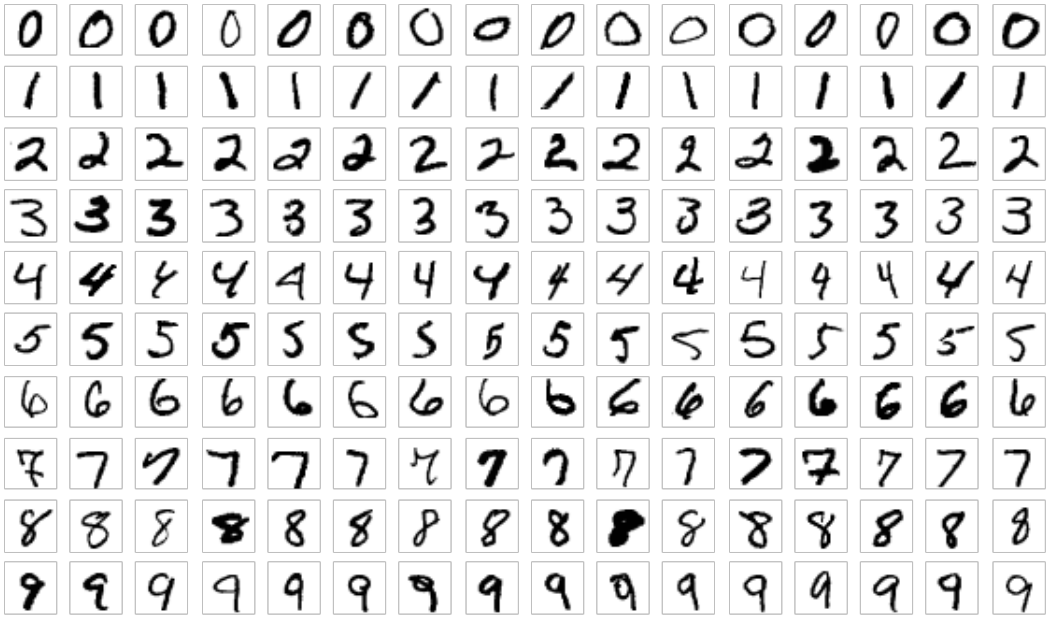

Figure 3.7. Sample of digits from MNIST

A Deep Convolutional GAN (DCGAN) was used for the experiment. A DCGAN is an extension of the GAN, except that it explicitly uses convolutional and convolutional-transpose layers in the discriminator and generator networks, respectively.[4]

3.3.1 Data Transformation

Each input image used by the *dataloader* went through a computer-vision pre-processing step that include:

- Grayscale transform: convert the image to greyscale. When loaded, the MNIST digits are in RGB format with three channels. Greyscale reduces these three to one.
- ToTensor: convert the image to a PyTorch Tensor, with dimensions (channels, height, width). This also rescales the pixel values, from integers between 0 and 255 to floats between 0.0 and 1.0.
- Normalize: scale and translate the pixel values from the range 0.0, 1.0 to -1.0, 1.0. The first argument is μ and the second argument is σ, and the function applied to each pixel is:

$$\rho \leftarrow \frac{(\rho - \mu)}{\sigma} \tag{3.1}$$

3.3.2 Generator Network Architecture

The generator network architecture is implemented using PyTorch as:

- A linear *fully-conected* module (or layer) to map the latent space to a $7 * 7 * 256 = 12544$ dimensional space that will later be undersampled several times until we reach $1 \times 28 \times 28$.
- An optional 1-dimensional batch normalization module
- A leaky ReLU module.
- A 2-dimensional convolutional layer with $padding = 2$, $stride = 1$ and 5×5 kernel (or filter).
- Two 2-dimensional transposed convolutional layers with $padding = 1$, $stride = 2$ and 4×4 kernel.
- Two optional 2-dimensional batch normalization modules after each 2-dimensional transposed convolutional layer.
- A *Tanh* activation function, rescaling the images to a $[-1, 1]$ range.

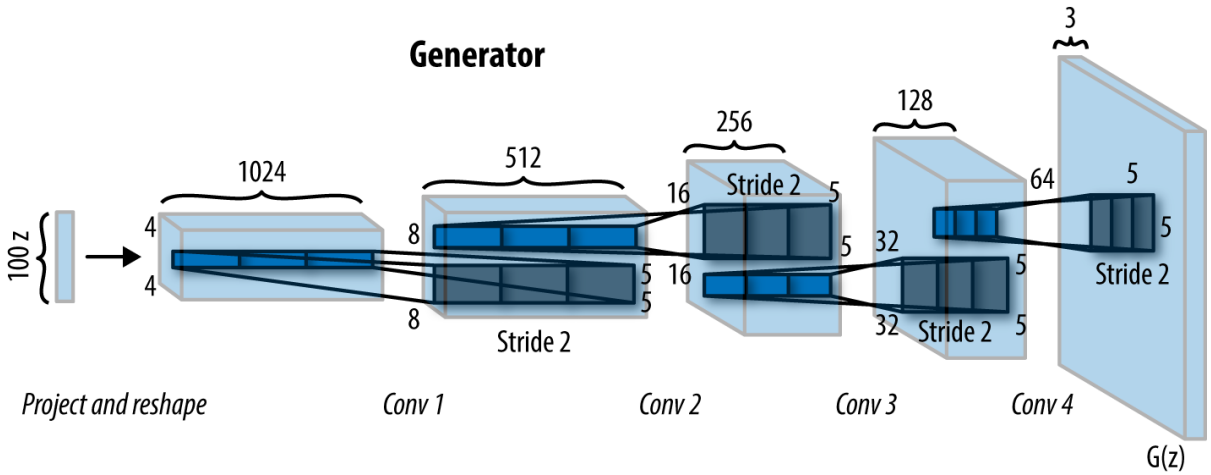

Figure 3.8. Generator network architecture of a 3-D image. Source: [4, 5]

The latent space (random signal) input goes through each layer being upscaled until it reaches the target image dimension 28×28 and then fed into the discriminator network.

3.3.3 Discriminator Network Architecture

The discriminator is a CNN-based image binary classifier network that takes an image as input and outputs a scalar probability that the given image is real or generated. The architecture is quite similar to the Generator network, except backwards. Here, the discriminator takes a $1 \times 28 \times 28$ input image, processes it through a series of convolutions, batch normalizations, and LeakyReLU layers, and outputs the final probability through a Sigmoid activation function.

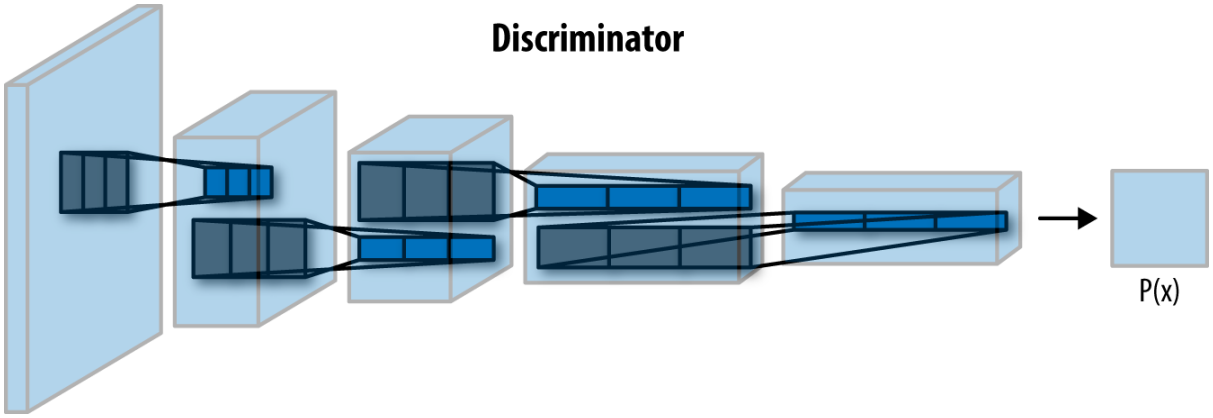

Figure 3.9. Discriminator network architecture of a 3-D image. Source:[4, 5]

4 Preliminary Results

4.1 1-D Compressed Sensing Reconstruction

The quality over resources demanded trade-off really starts making a big difference with L size around 200 samples, which is close to 80% of the data present in the signal to be reconstructed. This demonstrates that there is no big prejudice in using indirect reconstrution method for very undersampled signals (close to 10% of the data) such as the ones used in MRI

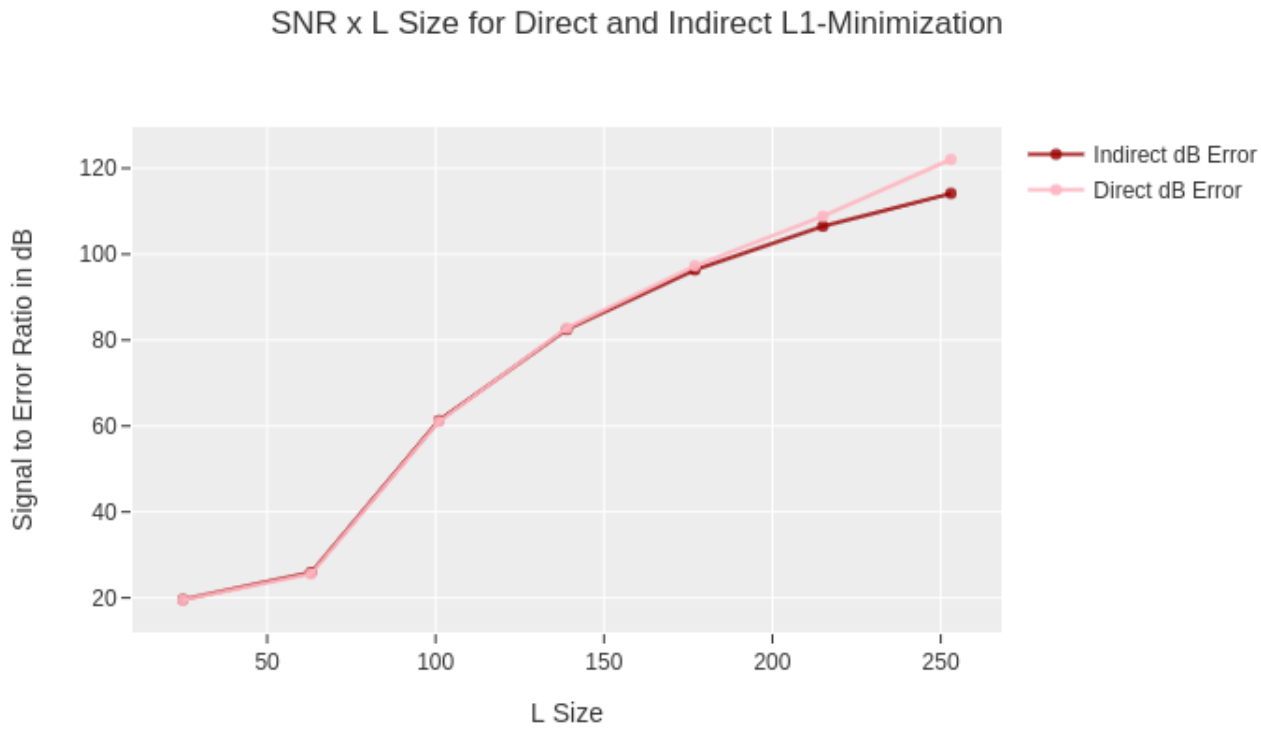

Figure 4.1. SNR x L size for direct and indirect L1-minimization for 200 random 1d signals

4.2 Compressed Sensing Reconstruction with Pre Filtered Signal

The results in the experiment shows a huge gain of resolution in the pre-filtering method reconstructed image compared to the L1-minimization alone.

The zero-filled reconstruction (dummy baseline) was unable to reconstruct a high fidelity image and performed very poorly in the PSNR and SNR metrics. The L1-minimization compressed sensing approach reconstructed the image with some noticeable noise artifacts, yet much better than the zero-filling approach. Finally, the L1-minimization along the usage of sparsifying pre-filtering delivered a great looking image without eye-catching artifacts and also increased the metrics hugely.

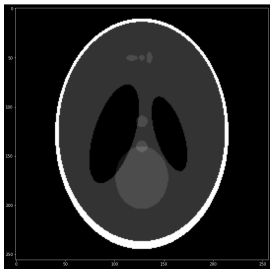

Figure 4.2. Reference Image

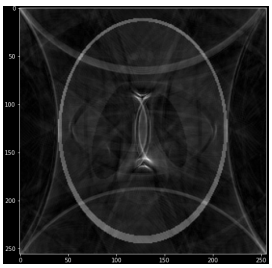

Figure 4.3. Zero-fill

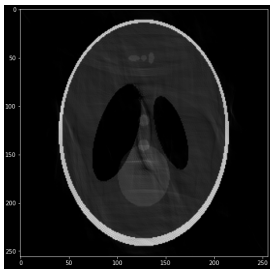

Figure 4.4. L1-minimization

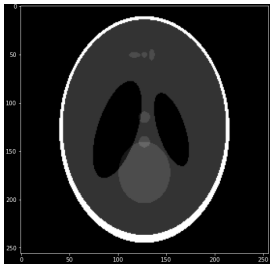

Figure 4.5. L1-minimization with pre-filtering

The usage of L1-minimization for certainly improves the MRI reconstruction, but the metrics reinforce how adding the pre-filtering step to preprocess the image achieves incredibly higher scores in Peak Signal-to-Noise Ratio (PSNR) and SNR.

	PSNR	**SSIM**	**SNR**
Zero-fill	64.8096	0.9976	3.4805
L1-minimization	79.4408	0.9999	19.1227
Pre-filtering L1-minimization	183.9644	0.9999	122.0507

Figure 4.6. Reference Image

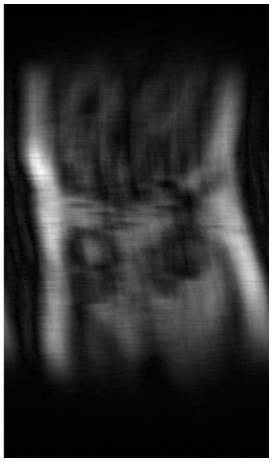

Figure 4.7. Zero-fill

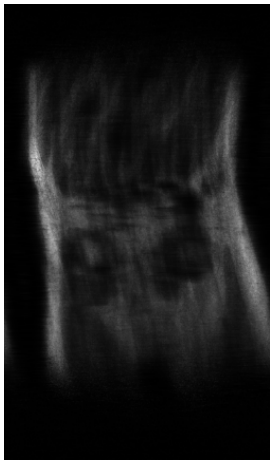

Figure 4.8. L1-minimization

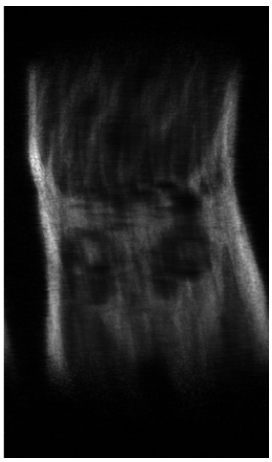

Figure 4.9. L1-minimization with pre-filtering

4.3 MRI Compressed Sensing Reconstruction

4.4 MRI BART Compressed Sensing Reconstruction

4.5 Preliminary Tests with Generative Adversarial Networks

GANs have a particular loss curve behaviour as both discriminator losses start very high and quickly decreases as the generator loss curve goes up in an invertly correlated manner. This happens because the generator starts tricking the discriminator network that is very naïve to determine if an image is real or generated, but quickly the discriminator manages to interpret that the generated images are very different from the training examples it is seeing. This phenoma reveals how bad the generator is in the first epochs and how easily the discriminator que distinguish between created and real. Then as the epochs go by, the generator starts to get better at creating the desired signal style and makes the discriminator's loss get higher again as it is observed in the loss curves below.

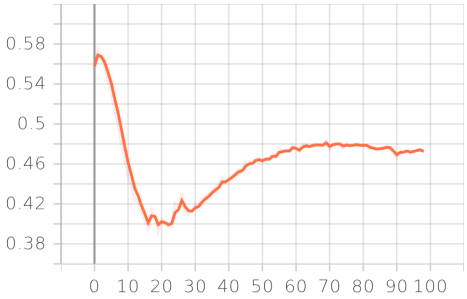

Figure 4.10. Discriminator fake loss over epochs

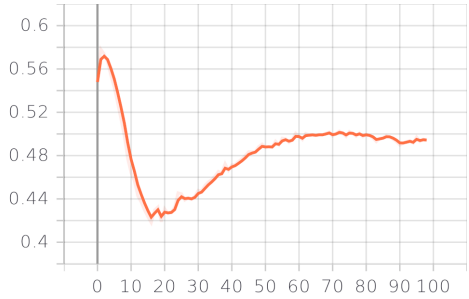

Figure 4.11. Discriminator real loss over epochs

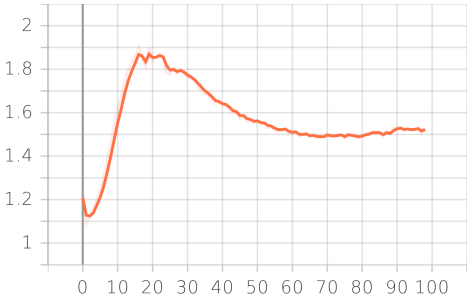

Figure 4.12. Generator loss over epochs

After 100 epochs, the DCGAN for MNIST number generation had an exceptionally good performance when the generated images are displayed. It is hard to tell if these are generated images or if they are part of the training set. The generated images sometimes have a bit more of blur to them, but certainly with more epochs and more training samples this could be minimized.

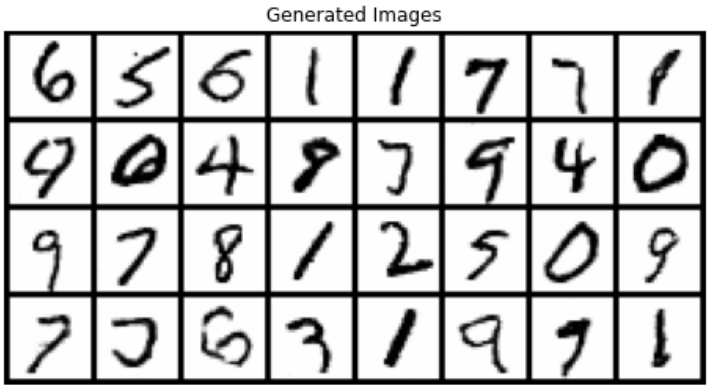

Figure 4.13. GAN generated MNIST digits

5 Conclusion

List of References

[1] Jure Zbontar, Florian Knoll, Anuroop Sriram, Tullie Murrell, Zhengnan Huang, Matthew J. Muckley, Aaron Defazio, Ruben Stern, Patricia Johnson, Mary Bruno, Marc Parente, Krzysztof J. Geras, Joe Katsnelson, Hersh Chandarana, Zizhao Zhang, Michal Drozdzal, Adriana Romero, Michael Rabbat, Pascal Vincent, Nafissa Yakubova, James Pinkerton, Duo Wang, Erich Owens, C. Lawrence Zitnick, Michael P. Recht, Daniel K. Sodickson, e Yvonne W. Lui. fastMRI: An Open Dataset and Benchmarks for Accelerated MRI. *arXiv:1811.08839 [physics, stat]*, December 2019. arXiv: 1811.08839.

[2] Jong Chul Ye. Compressed sensing MRI: a review from signal processing perspective. *BMC Biomedical Engineering*, 1(1):8, December 2019.

[3] Josh Patterson e Adam Gibson. *Deep Learning: A Practitioner's Approach*. O'Reilly, Beijing, 2017.

[4] Alec Radford, Luke Metz, e Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434 [cs]*, January 2016. arXiv: 1511.06434.

[5] Dominic Monn. Deep convolutional generative adversarial networks with tensorflow, 2017. Available at: https://www.oreilly.com/content/deep-convolutional-generative-adversarial-networks-with-tensorflow/. Last access on November 18th, 2020.

[6] R Nick Bryan. *Introduction to the Science of Medical Imaging*. Cambridge University Press, Cambridge, 2009.

[7] S. I. Kabanikhin. Definitions and examples of inverse and ill-posed problems. *Journal of Inverse and Ill-posed Problems*, 16(4), January 2008.

[8] C. J. Miosso, R. von Borries, e J. H. Pierluissi. Compressive sensing method for improved reconstruction of gradient-sparse magnetic resonance images. In *2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers*, pages 799–806, Pacific Grove, CA, USA, 2009. IEEE.

[9] Li Wan, Matthew Zeiler, Sixin Zhang, e Yann LeCun. Regularization of Neural Networks using DropConnect. page 12.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, e Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[11] Daeryong Kim e Bongwon Suh. Enhancing VAEs for collaborative filtering: flexible priors & gating mechanisms. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 403–407, Copenhagen Denmark, September 2019. ACM.

[12] P. C. Lauterbur. Image Formation by Induced Local Interactions: Examples Employing Nuclear Magnetic Resonance. *Nature*, 242(5394):190–191, March 1973.

[13] Michael Lustig, David Donoho, e John M. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, December 2007.

[14] D.L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, April 2006.

[15] Ian Goodfellow, Yoshua Bengio, e Aaron Courville. *Deep Learning*. MIT Press, 2016.

[16] L. A. Shepp e B. F. Logan. The Fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21(3):21–43, 1974.