



UNIVERSITAT OBERTA DE CATALUNYA (UOC)

MASTERS IN DATA SCIENCE

MASTER'S THESIS

AREA: MEDICINE AREA (TFM-MED)

Deep Convolutional Autoencoder for control brain MRI Development and Applications

Author: Adrián Arnaiz Rodríguez

Tutor: Baris Kanber

TFM Professor: Ferran Prados Carrasco

Barcelona, October 17, 2020

Créditos/Copyright



Este obra está bajo una [licencia de Creative Commons Reconocimiento, NoComercial, CompartirIgual 3.0 España](#).



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Spain License](#).

FICHA DEL TRABAJO FINAL

| | |
|-----------------------------------|--|
| Título del trabajo: | Deep Convolutional Autoencoder for control brain MRI: Development and Applications |
| Nombre del autor: | Adrián Arnaiz Rodríguez |
| Nombre del colaborador/a docente: | Baris Kanber |
| Nombre del PRA: | Ferrán Prados Carrasco |
| Fecha de entrega (mm/aaaa): | 01/2021 |
| Titulación o programa: | Máster en Ciencia de Datos |
| Área del Trabajo Final: | Area Medicina (TFM-Med) |
| Language: | English |
| Keywords | Deep Learning, Brain MRI, Autoencoder |

Dedication/Cite

To Be Done.

Acknowledgment

To Be Done. Si se considera oportuno, mencionar a las personas, empresas o instituciones que hayan contribuido en la realización de este proyecto.

Abstract

The analysis of brain MRI is critical for a proper diagnosis and treatment of neurological diseases. Improvements in this field lead to better health quality. Numerous branches can be still enhanced due to the nature of MRI recompilation: disease detection and segmentation, data augmentation, improvement in data collection, or image enhancement are some of them.

For several years, many approaches have been taken to address this. Machine Learning and Deep Learning emerge as very popular approaches to solve problems. Several kinds of data mining problems (supervised, unsupervised, dimension reduction, generative models, etc) and algorithms can be applied to the problem-solving of MRI. Besides, new emerging deep learning architectures for another kind of image task can be helpful. New types of convolution, autoencoders or generative adversarial networks are some of them.

Therefore, the purpose of this work is to apply one of these new techniques to T1 weighted brain MRI. We will develop a Deep Convolutional Autoencoder, which can be used to help some problems with neuroimaging. The input of the Autoencoder will be control T1WMRI and aims to return the same image, with the problematic that, inside its architecture, the image travels through a lower-dimensional space, so the reconstruction of the original image becomes more difficult. Thus, the Autoencoder represents a normative model.

This normative model will define a distribution (or normal range) for the neuroanatomical variability for the illness absence. Once trained with these control images, we will discuss the potential application of the autoencoder like noise reducer or disease detector.

Keywords: Deep Learning, Brain MRI, Deep Convolutional Autoencoder, Image denoising.

Resumen

El análisis de las resonancias magnéticas cerebrales es fundamental para un diagnóstico y tratamiento adecuados de las enfermedades neurológicas. Se pueden mejorar ámbitos del análisis debido a la naturaleza de la recopilación de resonancias: detección y segmentación de enfermedades, aumento de datos, mejora en la extracción o mejora de imágenes.

El aprendizaje automático y el aprendizaje profundo surgen como nuevas alternativas populares para resolver estos problemas. Se pueden aplicar varios enfoques de minería de datos y algoritmos para la resolución de problemas relacionados con la neuroimagen (supervisados, no supervisados, reducción de dimensionalidad, modelos generativos, etc.). Además, las nuevas arquitecturas emergentes de aprendizaje profundo, desarrollados para otro tipo de tareas de imagen, pueden ser útiles. Algunas de ellas son nuevos tipos de convolución, autoencoders o GAN.

Por lo tanto, el propósito de este trabajo es aplicar una de estas nuevas técnicas a resonancias cerebrales tipo T1. Desarrollaremos un Autoencoder convolucional profundo, que puede usarse para ayudar con algunos problemas de neuroimagen. La entrada del Autoencoder será el imágenes de control T1WMRI y tendrá como objetivo devolver la misma imagen, con la problemática de que, dentro de su arquitectura, la imagen viaja por un espacio de menor dimensión, por lo que la reconstrucción de la imagen original se vuelve más difícil. El autoencoder representa un modelo normativo.

Este modelo normativo definirá una distribución (o rango normal) para la variabilidad neuroanatómica para la ausencia de enfermedad. Una vez entrenado con imágenes de control, discutiremos la aplicación potencial del Autoencoder como reductor de ruido o detector de enfermedades.

Keywords: Aprendizaje profundo, Imágenes cerebrales de resonancias magnéticas, Autoencoder convolucional profundo, eliminación de ruido de imágenes.

Contents

| | |
|---|-------------|
| Abstract | ix |
| Resumen | xi |
| Content | xiii |
| List of Figures | xv |
| List of Tables | 1 |
| 1 Introduction | 3 |
| 1.1 Problem overview and relevance | 3 |
| 1.1.1 MRI general problems | 3 |
| 1.1.2 MRI Image enhancement | 5 |
| 1.1.3 Noise and artifact reduction with Deep Learning | 6 |
| 1.1.4 Our approach | 7 |
| 1.2 Personal motivation | 8 |
| 2 State of art: related works | 11 |
| 2.1 Overview | 11 |
| 2.2 Related works | 12 |
| 2.3 Volumes or slices? | 17 |
| 2.4 Network architectures for images | 19 |
| 2.4.1 Alexnet | 19 |
| 2.4.2 Residual networks | 19 |
| 2.4.3 Other skip-connection-based architectures | 22 |
| 2.5 Summary of related works | 24 |
| 3 Scope | 29 |
| 3.1 Hypothesis | 29 |

| | | |
|----------|---|-----------|
| 3.2 | Primary aims | 29 |
| 3.3 | Secondary aims | 30 |
| 4 | Planning and Methodology | 31 |
| 4.1 | Research plan | 31 |
| 4.2 | Methodology | 33 |
| 5 | Theoretical concepts | 37 |
| 5.1 | Data mining concepts | 37 |
| 5.1.1 | Deep Learning | 37 |
| 5.1.2 | Computer vision | 37 |
| 5.1.3 | Architectures | 37 |
| 5.1.4 | Autoencoder | 37 |
| 5.2 | Neuroimaging concepts | 37 |
| 5.2.1 | Magnetic Resonance | 37 |
| 5.2.2 | Magnetic Resonance Images Types | 37 |
| 6 | Techniques and tools | 39 |
| 6.1 | To Be Done | 39 |
| 6.2 | New paper-discovery frameworks | 39 |
| 7 | Project development | 43 |
| 7.1 | To Be Done | 43 |
| 7.2 | Data Life-cycle | 43 |
| 7.3 | Data Exploration | 44 |
| 7.4 | Data Preprocessing | 44 |
| 7.5 | Results | 45 |
| 8 | Conclusion and Outlook | 47 |
| 8.1 | To Be Done | 47 |
| | Bibliography | 47 |

List of Figures

| | | |
|-----|---|----|
| 1.1 | Brain MRI examples [1] | 4 |
| 1.2 | Noised and Denoised MRI [2] | 5 |
| 2.1 | Architecture of ResNet-VAE-based network of A. Myronenko. [3] | 14 |
| 2.2 | U-Net based architecture of network of J. V. Manjon et. al. [4] | 16 |
| 2.3 | Left: axial. Middle: sagittal. Right: coronal | 17 |
| 2.4 | C. Bermudez et. al. Denoising CAE+skip-connections architecture [5] | 18 |
| 2.5 | Residual Block Architecture [6] | 21 |
| 2.6 | Resnet architectures presented in [6] | 21 |
| 2.7 | U-Net architecture of original paper: O Ronneberger et. al. [7] | 23 |
| 2.8 | 2D slices from brain volume IXI ID 002. Different profiles can be seen. Source: myself | 25 |
| 2.9 | 2D slices from IXI ID 002. Slices from volume sides with no relevant information. Source: myself | 26 |
| 4.1 | CRISP-DM Cycle | 34 |
| 6.1 | Graph of connected papers for A. Myronenko 2018 [3] | 40 |
| 6.2 | A. Myronenko work [3] in Papers with code | 41 |

List of Tables

| | | |
|-----|--|--------------------|
| 2.1 | AlexNet Architecture [8] | 20 |
| 2.2 | Overview of studies for reconstruction based in Table 1 from D. Tamada [9] (In bold the autoencoder related architecture) | 27 |

Chapter 1

Introduction

Not Final Note: This color ('TBD' tag in latex) are going to be used in non-final versions of the memory to highlight the non-final sentences or things to be changed in the final version.

In this chapter we will introduce the main background, and aims of the project, basing it on its non-solved tasks and relevance.

1.1 Problem overview and relevance

1.1.1 MRI general problems

Neuroimaging in medicine allows studying the morphological features of the human brain. With the objective of improving the detection systems, diagnostic and treatment, correlations between the morphological features and the neurological disorders can be addressed in order to achieve that [10]. If we improve brain magnetic resonance image analysis 1.1, we will improve the detection systems and treatments for neurological diseases, so the social relevance of the field is very important.

The relevance of the project is also shown that there are a lot of branches in which neuroimaging analysis can improve. Machine learning, and more recently Deep Learning and Computer Vision, has irrputed in this field for helping in some tasks:

- **Disease detection: segmentation and classification.** There are still several problems

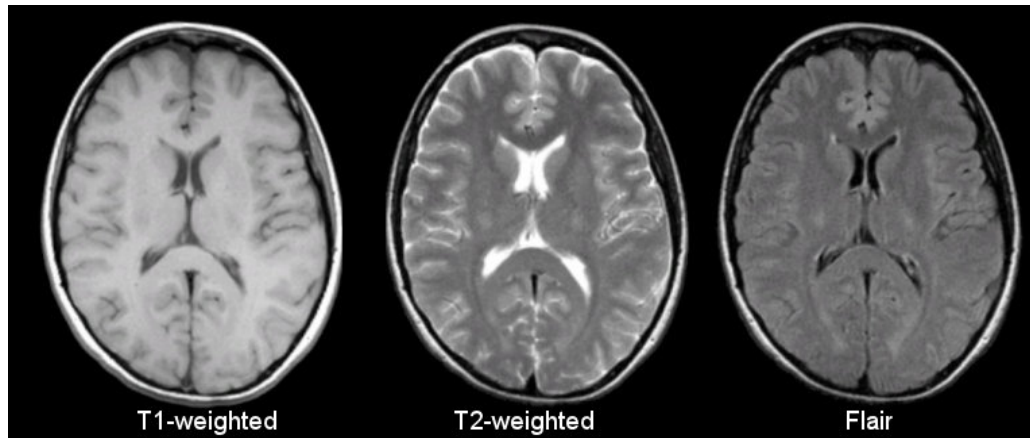


Figure 1.1: Brain MRI examples [1]

to solve in classification and segmentation problems. Brain MRIs are high dimensional, so we have to recruit big amount of images to properly develop a Machine Learning model that be able to achieve high accuracy. It is very difficult to recruit a large number of images, especially disease images. Even if it performs well, machine learning algorithms have been criticized due to the difficult of extract a clear knowledge of them (black-boxes). [3]. So a experimental approach is disease detection based on outliers from a normative model. Patients with pathologies will be outliers in the distribution build by the normative model (they will be out of normal range defined by the normative model) [11] [12]. It could be seen as an unsupervised anomaly detection technique, in which we don't need labeled data.

- **Data Augmentation.** Lack of data problem can be addressed by Data Augmentation techniques, which look for improve our Machine Learning models [13] and robustness of pipelines [4].
- **Improvement of data acquisition.** Recently high-impact **FastMRI**¹ release from Facebook for improving the speed in MRI scans [14].
- **Image enhancement.** Clinical evaluation is critical for good disease treatment. Experts and algorithms need good quality images to carry out their tasks. This is a problem we want to address, so we will explain it deeper in the document [9] [3].

However, some of these problems overlap. The advance in some of them leads to the advance in another. **Image reconstruction**, which is a mainly sub-problem of image enhancement, could help to achieve better results in data acquisition (i.e. reconstruct the image from less data

¹<https://fastmri.org/>

collected), unsupervised anomaly detection (i.e. reconstruction of the disease image differs more than the pathology-free one), data augmentation (i.e. reconstruction from pathology-free to abnormal and viceversa) and, obviously, image enhancement (i.e. reconstruction of cropped parts or reconstruction without noise and artifacts). **Thus, the main purpose for this project is to apply this reconstruction techniques for noise reduction (image enhancement) and data augmentation (lesion inpainting),** being another applications discussed for future work.

1.1.2 MRI Image enhancement

We are going to focus on the problem of image enhancement, specifically the **problem of image reconstruction**. With this reconstruction technique, the noise and artifacts will be reduced, see 1.2, and the image quality will be improved. If we achieved good performance in this task, we would research about how to apply this solution to unsupervised disease detection or data augmentation.

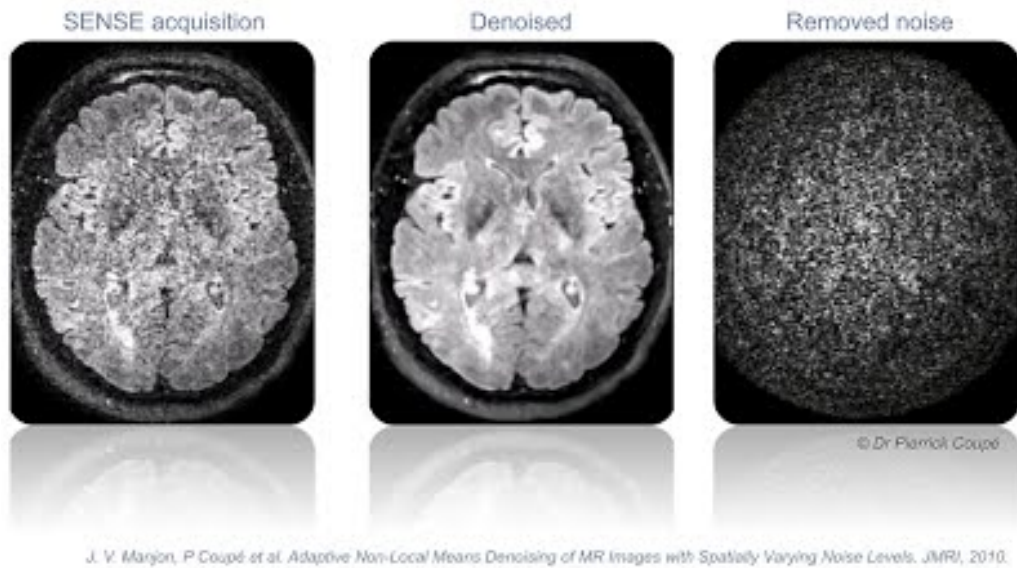


Figure 1.2: Noised and Denoised MRI [2]

Magnetic resonance images are collected with MR scans and the scan process, even though it is improved continually, adds some failures to the MR image. MR images have some random **noise** and **artifacts** due to this fact [15]. This noise and artifacts are present in the image due to different reasons: hardware-reasons (magnetic fields, etc.), body motion during scanning, thermal noise, weak signal intensity (which causes low signal-to-noise ratio), etc. The difference between noise and artifact is that noise can hide the characteristics of an image, whereas

artifacts appear to be characteristic but are not. If the 'problem' is structured, it is probably an artifact, while if it is random, it is probably noise.

1.1.3 Noise and artifact reduction with Deep Learning

Noise and artifact reduction is one of the main principal problems which are classically solved with reconstruction techniques. To address this problem many approaches have been done, all of them with some disadvantages. Advanced filtering methods [16] or retrospective correction approaches have been proposed, but, with the rise of **Deep Learning**, other methods have been proposed that take advantage of this approach. Deep Learning is very powerful in high-dimensional spaces and non-linear problems, so it can make a better job in feature extraction or information compression. Therefore, it will have good performance with images, where the **underlying structure of the images will be captured and foreign elements such as noise or artifacts can be eliminated through a noise-free reconstruction.**

There are some approaches to reduce the noise and artifacts in MRI images. An recent and outstanding review is made by D.Tamada [9]. In this paper D.Tamada summarize Deep Learning Architectures and applications to MRI. We notice the big relevance of denoiser MRI Deep Learning methods in this review. Although there are many methods, we will focus in brain MRI denoisers. A compilation of articles related with noise reduction is made in table 2.2.

There are some Deep Learning Architectures to address in the problem of denoise an brain MRI: Single-scale CNN, Denoising CNN, Autoencoders, and GAN-based architectures. **We will choose the Autoencoder Architecture for this project.**

Autoencoders [17] encode the input into a lower-dimensional space. It extracts important information from the higher-dimensional space, encodes it, and then decodes it to reconstruct the higher-dimensional spacer from the lower one. **This architecture will be deeply explained in 5 in further stages of the project.**

1.1.4 Our approach

In this project, we will design a **Deep Learning** autoencoder for reconstructing brain magnetic resonance images reducing noise/artifacts and inpainting lesions, and study its further implications and applications like data augmentation, disease detection or data acquisition improvement. In other words, we will train a autoencoder with disease-free neuroimaging data and, with this trained autoencoder, we could define a distribution (or normal range) for the neuroanatomical variability for the illness absence with the potential purpose of removing noise or look for diseases. Once trained, the autoencoder should be able to encode a input image and reconstruct it without the noise or lesson.

We want to highlight one main fact: the autoencoder do not remove noise or inpaint a lessons because it is trained like a denoiser or a image-filler. It do reconstruct MRI without noise or lessons because it is trained only with 'clean' and control MRI like target images, so it does not know how to reconstruct noise or lessons. Therefore, once the MRI is encoded in the latent space, the decoder will reconstruct a control clean MRI from this latent space.

If the main objective of the project is completed, we could research the application of this model in data augmentation and disease detection fields. In the case of **data augmentation**, we will then attempt to reconstruct magnetic resonance images from patients with brain pathologies, with a further view to using the autoencoder to generate 'pathology-free' versions of the said images. In the case of **disease detection**, we could take an approach like the one in [18] (creating a measure for the difference between input and output image and classify it as healthy or non-healthy based on this measure). Patients with pathologies will be outliers in the distribution build by the autoencoder (they will be out of normal range defined by the autoencoder) [11] [12]. This assumption of patients as outliers (based in [12]) is used in [18] for abnormal brain structure detection.

Of course, we will need **data**. For this project, we will use **T1-weighted MRI images of control subjects** (no disease). As the project is of fairly limited length, we won't need to detect disease as a principal objective, but only learn to reconstruct normal MRI images, so we won't need pathology images for this project. In addition, we will investigate whether our method of reconstruction can filter out noise and/or artefacts. So, in essence, we will have n 3D MRI volumes (n can be any number greater than 100) from healthy subjects, we will

preprocess it (data augmentation based on adding noise, remove parts...), and train our model to reconstruct the source MRI volumes. We will have access magnetic brain resonance images from control subjects for training the autoencoder. This data is arranged from different sources. We will use a set of T1 weighted control brain MR images for training the autoencoder. Then, we will consider to use another datasets in the experimentation stage:

- The [IXI dataset](#)² for training.
- Other data sources such as [Open Neuro](#)³ for experiment with autoencoder applications (reconstruction from disease image, etc).

1.2 Personal motivation

My personal motivation to carry out the project arises from several factors. My first steps in the world of Machine Learning were in the last year of my career at the University of Burgos. I was lucky enough to collaborate last year with the [ADMIRABLE](#)⁴ research department. The project that I did (in which we continue working) was on the [use of biomarkers extracted from the voice for the construction of classifiers that detect Parkinson's disease](#)⁵. The project include topics like **signal processing**, **supervised learning**, **unsupervised learning** and **transfer learning**. The project was very successful and we had a lot of impact at that time. We are currently in the process of meeting with the Burgos hospital to continue developing the model and the application (project and impact recompilation in [Github](#)⁶).

This project has fully opened me the doors the world of artificial intelligence and machine learning, which is a field of knowledge that I love. I have always liked math, problem-solving and since I started my career I love programming. Therefore, I find this field the ideal that aligns with my tastes and interests. As I said before, I have done lot of jobs with supervised learning or data analysis, but only with tabular data sets or text-datasets, so I wanted to break in the world of image processing and Deep Learning.

Then I worked half year in Ernst and Young, developing Machine Learning systems for RPA tasks (classify emails at Telefónica, Chatbot for Maxium or Fuzzy Name Matching for Xunta

²<https://brain-development.org/ixi-dataset/>

³<https://openneuro.org>

⁴<http://admirable-ubu.es/>

⁵<https://adrianarnaiz.github.io/TFG-Neurodegenerative-Disease-Detection/>

⁶<https://github.com/AdrianArnaiz/TFG-Neurodegenerative-Disease-Detection>

de Galicia).

In addition, I believe that the application of AI to medicine is one of the fields that may be of greater general interest to society. By advancing in the speed and quality of medical diagnoses and treatments, it will be possible to achieve health of higher quality, speed and accessibility for all. Also, computer vision and deep learning have helped to achieve big advances in this field nowadays.

Chapter 2

State of art: related works

2.1 Overview

The world relevance and impact of this problem is also shown in the related articles of this subject. The state of art of Deep Learning applied to brain MRI shows the relevance of this field. As we discussed in the introduction, in section 1, different deep learning techniques have been used to address the problems derived from brain MRI images: **classification healthy/disease, tumor segmentation, optimize data acquisition, data augmentation and image enhancement** are the principal ones.

Nevertheless, we must highlight that all of this problems have common points of works. One of them is the purpose of our project: **learn MRI representation for reconstruction**.

The advance in some of the questions leads to the advance in another. **Image reconstruction**, which is a mainly sub-problem of image enhancement, could help to achieve better results in:

- **Data acquisition**
 - Reconstruct the image from less data collected: faster scanning process [14].
- **Disease detection and segmentation**

- Unsupervised Anomaly Detection: detect diseases with non-labeled data: reconstruction of the disease image differs more than the pathology-free one [18].
- Tumor segmentation (widely known as BraTS [19]): encode for extract deep image features and decoder for reconstruction of dense segmentation mask [3].

- **Data Augmentation**

- Construction of pathology-free image from abnormal and viceversa [4] (i.e. lesion inpainting).
- Artificial MRI Generation [13] [20]¹.

- **Image Enhancement**

- Reconstruction of cropped parts.
- Reconstruction without noise and artifacts [21] [5], [22], [23].
- Definition enhancement: from low resolution to high resolution [24] [21].

We want to emphasize the actual relevance of this project. Solving MRI problems using Deep Learning isn't just about how to apply Deep Learning to another field. It is not just a Deep Learning experiment to demonstrate the power of this method. Solving problems with MRI diagnostics (classification, segmentation), MRI quality (MRI enhancement, data augmentation), or MRI acquisition are cutting edge issues in both the field of Computer Science and Healthcare (neuroimaging, neurological analysis, etc.).

2.2 Related works

We realize this in the overwhelming number of articles using different Deep Learning architectures for solving all kinds of problems with MRI. We will discuss papers addressing different problems but with one similarity: use of image reconstruction in some part of the process (preferably by using autoencoder-based solution). **However, the main purpose for this project is to apply this reconstruction techniques for noise reduction (image enhancement) and data augmentation (lesion inpainting).**

The main evidence of the big relevance and collaboration between Deep Learning and MR imaging is **FastMRI by Facebook AI**. In fact, lately, the focus is on **improving MRI**

¹<https://paperswithcode.com/paper/generation-of-3d-brain-mri-using-auto>

acquisition, with techniques based on collecting fewer data and using reconstruction techniques with Deep Learning with the aim of improving image quality and acquisition speed. The high-impact in the academic field of these kind of studies is based on Facebook AI works. Facebook AI is focused on **accelerating MR imaging** with AI, and it is his main goal in healthcare nowadays. They created **fastMRI**² [14], a set of models working with some benchmark datasets in order to accelerate the MR imaging acquisition. It is open source, and you can participate in the **challenge**³ with data from New York University. Recently, Facebook and **NeurIPS**⁴ announced that the best models and projects presented for this purpose, even from groups outside Facebook, will be invited to NeurIPS, one of the most important conferences on Neural Information Processing Systems.

To continue with the different studies using reconstruction methods for distinct purposes, we describe the use of reconstruction for helping **Tumor Segmentation**. This is another main problem in the state of the art. There is a global academic challenge using labeled brain tumor MRI for BRAin Tumor Segmentation called **BRATS** [19]. This competition is compound by a MRI dataset from T1, T1c, T2 and FLAIR MRI and the goal is make the segmentation of the distinct parts of the tumor. Using this data as a **benchmark**⁵, lots of different groups are making experiments each year to improve the results. One of these studies using reconstruction techniques is the current best outcome for BRATS 2018: **A. Myronenko** [3]⁶. Although their objective is the 3D segmentation of tumors, they use a curious architecture, shown in figure 2.1, that incorporates an encoder and two decoding branches: one for the creation of tumor segmentation masks and the other for the reconstruction of images. This branch of image reconstruction is only used during training as an additional guide to regularize the encoder part. The encoder is made by **ResNet** blocks (Group Norm+ReLU+Conv). The decoder is a **variational autoencoder** (VAE) made of the distribution layer and deconvolutional upsampling layers with Group Normalization and ReLU. 2 more parts are incorporated in their main loss function for tumor segmentation: Mean square error and Kullback–Leibler divergence of the reconstruction branch.

Another problem is to **classify whether an image belongs to a control or a patient**. A recent approach is based on the construction of normative models [11], and , therefore, the image reconstruction based in this normative model. **Pinaya et al.** [18] use this technique to identify abnormal patterns in neuropsychiatric disorders towards achieving **unsupervised**

²<https://fastmri.org/>

³https://fastmri.org/submission_guidelines/

⁴<https://sites.google.com/view/med-neurips-2020>

⁵<https://paperswithcode.com/task/brain-tumor-segmentation>

⁶<https://paperswithcode.com/paper/3d-mri-brain-tumor-segmentation-using>

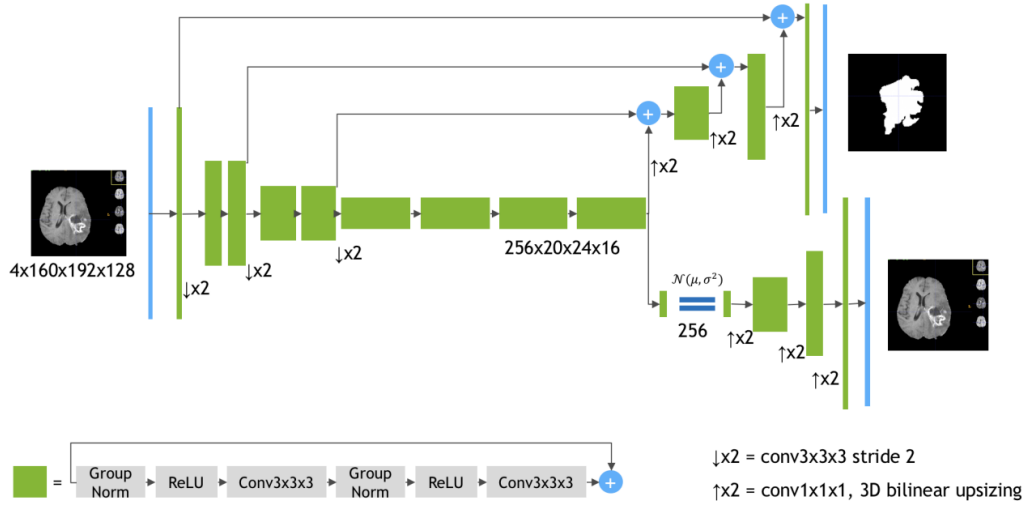


Figure 2.1: Architecture of ResNet-VAE-based network of A. Myronenko. [3]

anomaly detection, so we don't need labeled images from disease data. Classic methods and approaches based on **sMRI** (structural magnetic resonance imaging) can't get a good performance in abnormal brain structural detection because neuroanatomical alterations in neurological disorders can be subtle and spatially distributed. Another approach based on Machine Learning methods could improve performance because algorithms are sensitive to these subtle characteristics. The downside of this road is the need for a large amount of image data (control and disease) and that the models are black-boxes with no information on the critical characteristics used for the decision. They developed a **Deep semi-supervised Autoencoder**, which put unsupervised anomaly detection up for discussion. The goal of that study is to build an autoencoder which encode the structure of control brains. It means the autoencoder learn the normal distribution for healthy brains and the abnormal MR images would be outliers in that distribution. With this autoencoder defining a distribution for control patients, they define a **deviation metric** to measure the neuroanatomical deviation in patients. Patients with some disorder should be outliers in this distribution. The architecture and technique used in the experiment is the following:

- Architecture

- Semi-supervised autoencoder: reconstruction of the image and prediction of age and sex.
- 3 hidden layers with SELUs activation function.
- Output layer with Linear activation function.

- Loss function: MSE from reconstructed and original image + cross-entropy for age + cross-entropy for years + Unsupervised cross-covariance.
- 2000 epochs.
- ADAM optimizer (adaptive moment estimation) with adaptative learning rate.
- 64 samples mini-batches.
- Transformation of input data:
 - Add Gaussian noise to image (0, 0.1).
 - Feature scaling (normalization).
 - One-hot encoding for **sex** and **age** labels.

We continue with a special case: **lesion inpainting** [4]. It can be seen as a **data augmentation** task (reconstructing ‘pathology-free’ versions from patients with any brain disease). In the work of **José V. Majón et. al.** [4] the medial purpose is **the improvement of the behavior of current brain image analysis pipelines**. These pipelines are not robust to brain MR images with lesions. For example, a task such as brain part segmentation decreases its accuracy when dealing with lesions. They proposed a **3D UNet** like network to map the image with lesion to the inpainted image (target). The encoder is made by 3 blocks of a 3D Convolutional Layer with ReLU activation, Batch Normalization and max-pooling. For the decoder they used same architecture but upsampling instead of max-pooling and, in the last step, a tri-linear interpolation layer followed by a 3D convolution layer (with 8 filters) plus a ReLU and Batch normalization layers for upsampling the image. We can see the diagram of the architecture in figure 2.2. Everything was trained with MSE loss function. They use lesion masks to generate artificial training data. The use control cases masked out with lesion masks using the software lesionBrain [25].

Besides of all of these principal studies and objectives, there are so many more. **Théo Estienne et. al.**⁷ [26] in 2020 realize a project based in the study from A. Myronenko [3] which we explained before. They also research Deep Learning architectures for tumor segmentation (BraTS 2018) and use a autoencoder-based network with 2 decoder branches: one for tumor segmentation and another for reconstruction. This last branch is only used for encoder-regularization. They use a fully convolutional VNet architecture, with convolutional layers, ReLU activation function and an intra-block residual connection with the output of the first activated convolution of the corresponding block. They use also direct connections from encoder to decoder part.

⁷https://github.com/TheoEst/joint_registration_tumor_segmentation

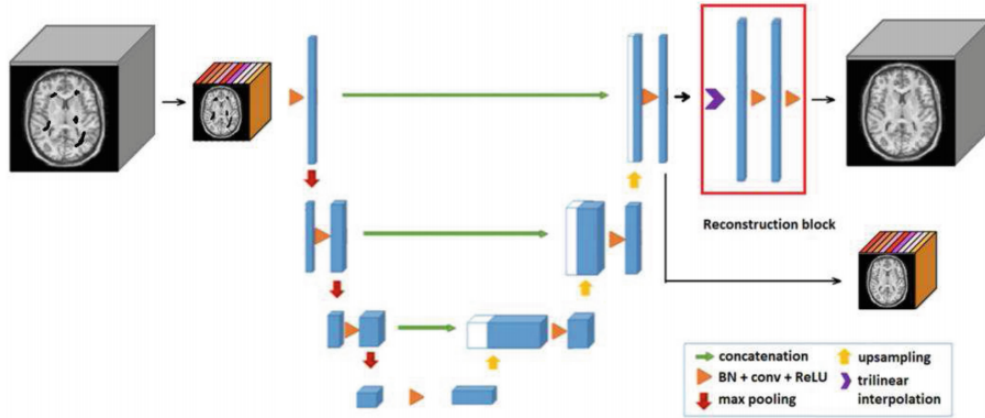


Figure 2.2: U-Net based architecture of network of J. V. Manjon et. al. [4]

We discover another study from Evan M. Yu et. al. [27]⁸ in which they try to learn volumetric representations from different parts of brain structure. They also use an autoencoder framework. Their architecture is composed by 2 components: a spatial transformer network (STN) and a convolutional autoencoder (CAE). The autoencoder is a kind of ResNet-based one, because it uses residual blocks with skip connections, instance normalization and Leaky ReLU activation function.

In order to finalize with the review of the studies of reconstruction applications in MRI, we want to explain one last paper. This work is not focused in MRI, but it achieves very good performance in many tasks like restoration, denoising, super-resolution or image inpainting. XJ. Mao et al [21]⁹ published in 2016 an study about CAE with symmetric skip-connections to address those objectives. They also use a residual based network which they call RED-Net. The main characteristics of this network are the skip connections (in which one layer from the encoder are added up to its symmetric layer in the decoder) and the lack of pooling layers. They don't use pooling layers due to pooling discards useful image details that are essential for these tasks. They use MSE loss function. Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) index are calculated for evaluation.

The studies and tasks just explained are the prominent and recent ones but there are other areas and analysis in which MRI restoration could help. Some of them are survival prediction [28], disease progression [29] or brain connectivity analysis [30].

⁸<https://paperswithcode.com/paper/a-convolutional-autoencoder-approach-to-learn>

⁹<https://paperswithcode.com/paper/image-restoration-using-convolutional-auto>

2.3 Volumes or slices?

Brain MR images are stored in volumes. It means it are stored as 3D volumes representing somebody's head. Some projects directly use 3D volumes to achieve the purpose (i.e. 3D tumor segmentation or 3D reconstruction). It is more complicated get good results in 3D than in 2D, because results are more relevant in the field.

When working in 2D we have to consider another decisions. The first one is **what profile of the volume should we use**. Brain MRI Volume has 3 different views: **axial** (from above the head), **sagittal** (from the side of the face, profile) and **coronal** (from behind the head). We can see the different views in figure 2.3

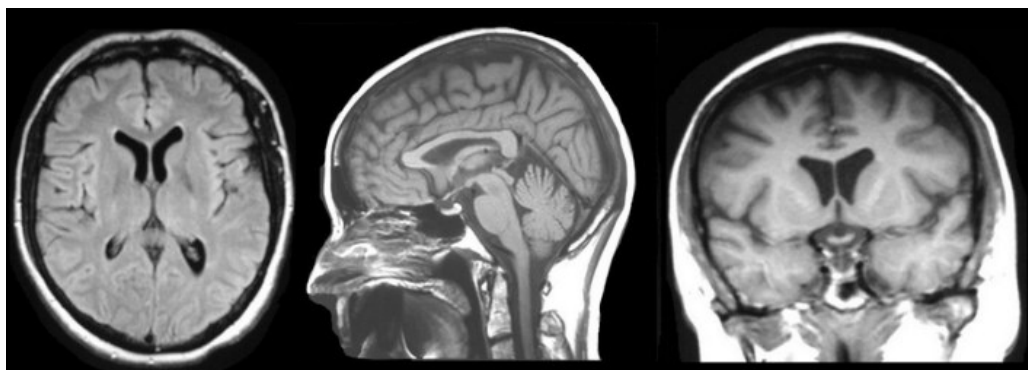


Figure 2.3: Left: axial. Middle: sagittal. Right: coronal

For non-isotropic acquisitions, we should ideally slice them so that the slices are high resolution. For example, if the **voxel** resolution is $1 \times 1 \times 5 \text{ mm}^3$, we should slice the volume so that the slices are $1 \times 1 \text{ mm}^2$ rather than $1 \times 5 \text{ mm}^2$ (or $5 \times 1 \text{ mm}^2$). The other issue to address is **what 2D slices from the volume has relevant information**. Some of the slices are slices of the extremes of the volume, and it didn't represent relevant information about the brain structure-

In this project, we are going to work with 2D slices due to time constraints. Therefore, in this section, we will discuss whether the projects mentioned above have used volumes or images, from which volume profile they have taken the images and how the ones containing important information have been chosen. **Our approach will be explained in SECTION 7.4.**

A. Myronenko [3] uses 3D volumes for the brain tumor segmentation task (BraTS 2018) with $1 \times 1 \times 1 \text{ mm}$ isotropic resolution and size $240 \times 240 \times 155$. Consequently, he does not have to get any profile or select slices. Pinaya et. al [18] use T1 weighted images, thus 2D slices. They don't say neither the profile used in the images, nor the method to select 2D slices with relevant information. To continue, Manjón et. al [4] propose "the first 3D blind inpainting method in

medical imaging” as they say. They use the same dataset that us (IXI), they work directly with 3D volumes, so neither profile nor slice election is done. They only preprocessed the volumes in order to normalize the voxels in to 1 mm³ voxel resolution. More recently, Théo Estienne et.al [26] also address the problem of 3D tumor segmentation of BraTS 2018. Evan M. Yu et. al. [27] uses 3D volumes of OASIS dataset to learn volumetric shape representations for brains structures.

As we can conclude, most projects address 3D volumes because his relevant implications, both in medical field and in deep learning field. However, we have found another articles in which they work with 2D slices. **C. Bermudez, et al** [5] uses 2D **axial** slices from BLSA dataset. All subjects were affine-registered to MNIs-space and intensity-normalized before 2D slices are selected. In addition, the only select a **single midline axial slice** from each volume. Finally they had 528 images with size 220 x 170 voxels. They use this 2D images to build 3 denoiser autoencoders with skip connections (one autoencoder for each level of noise added to train data). The architecture of this 2D denoiser brain mri autoencoder is shown in figure 2.4.

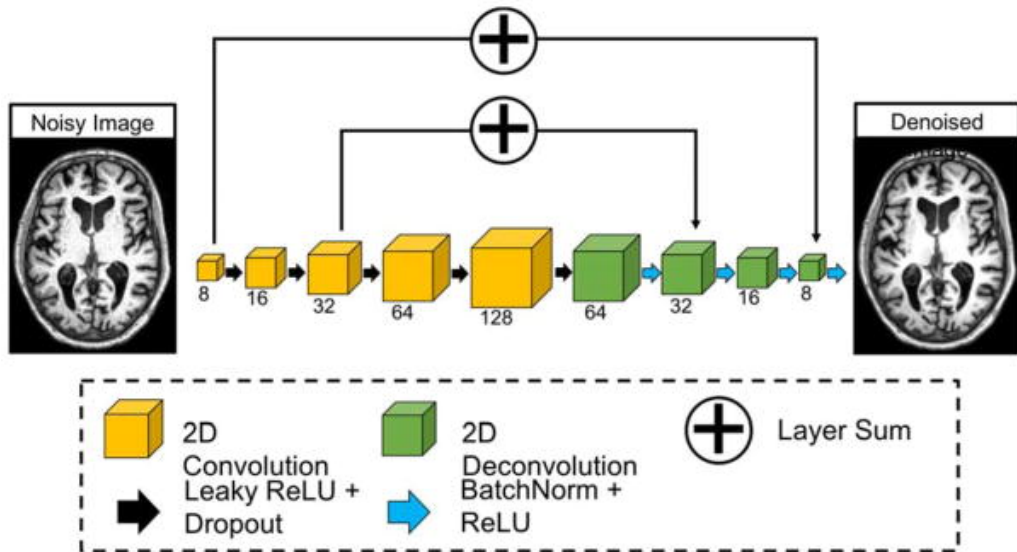


Figure 2.4: C. Bermudez et. al. Denoising CAE+skip-connections architecture [5]

2.4 Network architectures for images

The encoder part of the autoencoder should work as a feature extractor, so we can research the most known Deep Learning Architectures for images in order to use the same architecture or realize transfer learning (with frozen weights or not). We will start explaining the first approach to image processing with Deep Learning, then we will continue with residual network architectures (commonly used in related works) and we will conclude by explaining U-Net and V-Net, the other 2 commonly used networks for medical image segmentation.

2.4.1 Alexnet

First of all, we are going to introduce **Alexnet** [8], a deep convolutional neural network for image classification created by Alex Krizhevsky et. al in 2012 which, in that moment, was the best approach for ImageNet classification, improving by 10% of difference with the second best. It is very important because they apply Deep Learning and convolutional networks to the classification of images and they establish a standard from which new networks are created by adding improvements. The architecture, shown in table 2.1, is made by 5 convolutional layers and max pooling layer after convolutional 1, 2 and 5 layers, and 3 dense (fully connected) layers as classifier part. Rectified linear unit is used as a activation function (ReLU). Softmax function is used in the last layer in order to represent a probability distribution over the image classes. So AlexNet is characterized by **convolutional layers, max pooling layers, dense layers as classifier and ReLU activation function**.

Since the moment Alexnet appeared, the improvements made were almost everyone about go deeper in the layers and architecture. However, going deeper didn't solve another problems like vanishing gradient.

2.4.2 Residual networks

We realize that most of the works of the state of the art uses residual networks [3] [27] or networks with skip-connections [4] [21] [5], so we will introduce deeper than other architectures.

The emergence of this network rises from deep plain neural networks (as Alexnet) problems on training. Despite the general belief that the more layers these networks have the more it learns, an experiment made by He et. al. [6] shows that, in some cases, more layers are related

| Layer | Map | Tensor size | Kernel size | Stride | Activation | Params |
|---------------|-----|-------------|-------------|--------|------------|------------|
| Input Image | 1 | 227x227x3 | - | - | - | - |
| Convolution 1 | 96 | 55x55x96 | 11x11 | 4 | relu | 34,944 |
| Max pooling | 96 | 27x27x96 | 3x3 | 2 | relu | 0 |
| Convolution 2 | 256 | 27x27x256 | 5x5 | 1 | relu | 614,656 |
| Max pooling | 256 | 13x13x256 | 3x3 | 2 | relu | 0 |
| Convolution 3 | 384 | 13x13x384 | 3x3 | 1 | relu | 885,120 |
| Convolution 4 | 384 | 13x13x384 | 3x3 | 1 | relu | 1,327,488 |
| Convolution 5 | 256 | 13x13x256 | 3x3 | 1 | relu | 884,992 |
| Max pooling | 256 | 6x6x256 | 3x3 | 2 | relu | 0 |
| Dense 1 | - | 4096x1 | - | - | relu | 37,752,832 |
| Dense 2 | - | 4096x1 | - | - | relu | 16,781,312 |
| Dense 3 | - | 4096x1 | - | - | relu | 4,097,000 |
| Output Dense | - | 1000 | - | - | Softmax | - |

Table 2.1: AlexNet Architecture [8]

to less accuracy even using regularization techniques like L2 or Dropout. Deep plain networks suffer from performance degradation due to the loss of detail in deeper layers and vanishing gradient problem

This residual learning approach is introduced by Kaiming He et. al. in 2016 [6] with the goal of reducing the complexity of deep neural networks. The main innovation of residual networks is that the layers "learn residual functions with reference to the layer inputs, instead of learning unreferenced functions" [6] as they said. This network obtains better performance than before with less complexity in training.

In plain networks, the layers are built to approximate a mapping function from the image to a target: $H(x)$. This is equivalent to approximate the residual of this function: $F(x) = H(x) - x$ and then get the original function as $F(x) + x$. This kind of layer is called **Residual Block** in which we estimate the residuals and then we add the original input to get our original target function. We can see the architecture of this block in figure 2.5 , which is the picture of the original work. Thus, a **skip-connection** is include in the network. **This only consist in adding the input of a stack of layers to the output of this stack of layers.** We have to notice $F(x)$ and x could not have same dimension, so we have to multiply by a linear projection the original input x for matching dimensions.

This paradigm of residual learning gives rise to other ResNet-based architectures, which

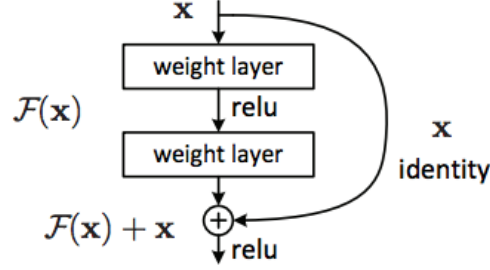


Figure 2. Residual learning: a building block.

Figure 2.5: Residual Block Architecture [6]

differs in number of layers and building block compositions. This kind of networks uses residual blocks (there are also several residual block architecture) with skip-connections embedded in some architecture. Two main examples are the ResNet34 and Resnet50 networks, which use different kinds of residual blocks. In addition, He et. al. presents some differentes ResNet architectures which we show in figure 2.6.

In addition, an Autoencoder made with residual blocks made by A. Myronenko [3] is shown in figure 2.1, in which green blocks represent residual blocks with group normalization. Residual blocks are also used in Yu et. al. [27] work.

| layer name | output size | 18-layer | 34-layer | 50-layer | 101-layer | 152-layer |
|------------|-------------|---|---|---|--|--|
| conv1 | 112×112 | 7×7, 64, stride 2 | | | | |
| | | 3×3 max pool, stride 2 | | | | |
| conv2_x | 56×56 | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$ |
| conv3_x | 28×28 | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$ | $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$ |
| conv4_x | 14×14 | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$ | $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$ |
| conv5_x | 7×7 | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$ | $\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ | $\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$ |
| | 1×1 | average pool, 1000-d fc, softmax | | | | |
| FLOPs | | 1.8×10^9 | 3.6×10^9 | 3.8×10^9 | 7.6×10^9 | 11.3×10^9 |

Figure 2.6: Resnet architectures presented in [6]

2.4.3 Other skip-connection-based architectures

As we explained before, deep plain networks suffer from performance degradation due to the loss of detail in deeper layers and vanishing gradient problem. Residual networks address this problems through residual blocks. But there is a inner idea in ResNets which other architectures also use: **skip-connections**. In ResNet skip-connections are added in order to estimate the residual function, but these kind of skip-connections can be added to networks with another purpose. Other networks like some Fully Convolutional Networks also use this skip-connections. Other networks like some Fully Convolutional Networks also use this skip-connections. For example, FCN-8 architecture has skip connections, in which some feature maps of the earlier layers are added to later layers.

Skip connections from one layer of the encoder to it symmetric layer of the autoencoder are added to the architecture with 2 purposes. First, to allow the signal to backpropagate straight to the lower layers and thus address the problem of gradient disappearance, facilitating deep network training and, thus, achieving improvements in restoration performance. Second, when we build a deeper network, low-detail of the image could be lost, making deconvolution difficult in recovering task. However, the skip connections pass through the feature maps which carry much image detail and helps deconvolution to recover the original image.

Some related works use this kind of architecture. C. Bermudez et. al [5] uses skip symmetric connections as can be shown in figure 2.4. This is a convolutional autoencoder with Leaky ReLU, in which the skip-connections add a layer in the encoder to their symmetric layer in the autoencoder (Like FCN but being symmetric).

However, some concrete architectures of CAE + skip-connections have been established due to their good results in some tasks. U-Net [7] is one of them, and a U-Net-Autoencoder-based architecture is used by Manjon et. al. [4] which we can see in figure 2.2.

2.4.3.1 U-Net and V-Net

U-Net [7] is one of the networks that has skip-connections between symmetric layers. U-Net is a Fully-Convolutional-based Network (FCN) that is mainly used for image segmentation, that's why Manjón [4] use a V-Net based architecture for inpainting. The principal 2 differences between a FCN and U-net are the **symmetry** of U-NET and the skip connections between the downsampling (encoder) path and the upsampling (decoder) path which use **concatenation**

operator instead of sum (sum operator is used in skip-connections in Fully Convolutional Networks).

Each of the 4 blocks used in the downsampling path is made up of 2 convolutional layers (3x3) with batch normalization and ReLU and another 2x2 MaxPooling layer. This block extract advanced features while reducing feature maps sizes. In the upsampling route, the 4 blocks used are made by a 2x2 upconvolutional layer and 2 other convolutional layers like those of the downsampling route to recover size of segmentation maps, the concatenation of the feature map of the symmetric layer of the encoder to give the location information from the encoding path to decoding path and a final 1x1 convolution. The original architecture of U-NET paper is illustrated in figure 2.7. As we can see, this original U-Net architecture is very similar to Manjon et. al. U-Net based autoencoder; see figure 2.2.

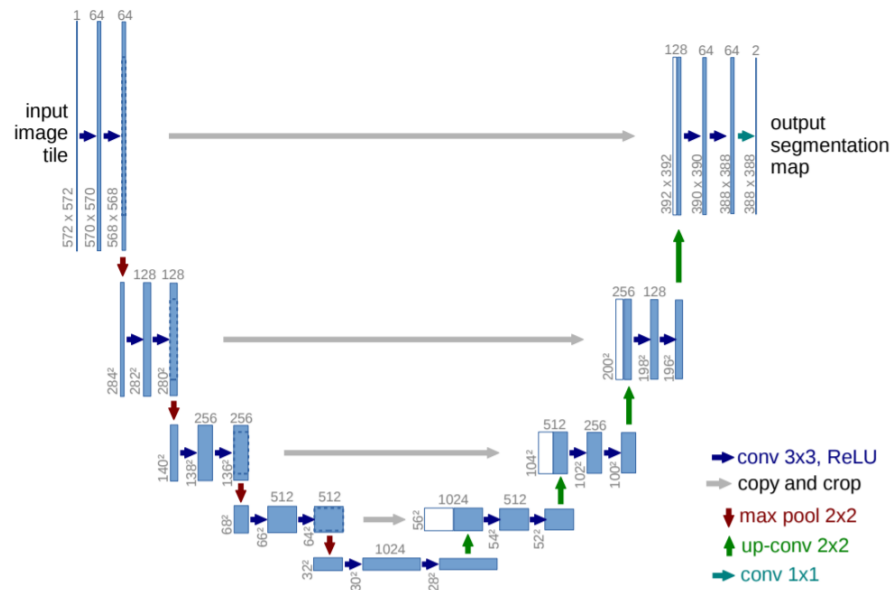


Figure 2.7: U-Net architecture of original paper: O Ronneberger et. al. [7]

U-Net has suffered some modifications, one of the most important is **V-Net**. This network was introduced by F. Milletari et. al [31]. V-Net is used for volumetric biomedical image segmentation and it gets a very good performance in this task. This network is used in the work of T. Estienne [26] to create a V-Net-based autoencoder to solve Brats 2018 challenge. V-Net combines the skip-symmetric-connections with residual blocks. It means that the encoder and decoder parts are built with residual blocks instead of convolutional blocks as U-Net. V-Net also changes the size of kernels and convolutions in respect to original U-Net, but the main change are the residual blocks. So, V-Net combines the 2 types of skip-connections we have spoken in this research of state of art: residual blocks and larger skip-connections between

symetrich layers.

2.5 Summary of related works

We have compiled some recent and prominent works in which they use reconstruction methods for different purposes. We have generally explained their architectures and approaches.

Although all the collected architectures are based on autoencoders, it has its differences in how the autoencoder is built. First of all, it differs in the main architecture of the blocks of the autoencoder. Different kind of networks like **ResNet**, **UNet**, **VNet**, **Simple CAE** or **AlexNet**. Also, in some of them use the Variational Autoencoder approach, even in [3] combine ResNet and VAE. Furthermore, there are additional architecture characteristic in which studies differ.

Regardless of the main architecture, one feature is shared by all related works (works that use Deep Learning for MRI reconstruction): **skip-connections**. In the works that use ResNet-based [3] [27] architectures, skip-connections are implicit in residual blocks. In this architecture the skip-connection is used for represents the residual estimation function. In U-Net [4] and CAE with skip-connections architectures [5] [21], the skip-connection is used with 2 purposes: allow the signal to backpropagate straight to the lower layers and pass the image details from the convolutional layers to the deconvolutional layers, fighting the low-detail loss [21]. In conclusion with the architecture research, **the related work use skip connections architectures: ResNet-base and CAE+Skip-Connection-based (U-Net-based and more)**.

The **loss function** used is also a critical issue. Most of the studies researched use pixel-wise *MSE*, which implicit improves the evaluation metrics *PSNR* and *SSIM*. In addition, other metrics are also used. *KLdivergence* is added to loss function when VAE is used or *cross – entropy* and *cross – covariance* are added when semi-supervised autoencoder is used [18]. However, we will discuss in this project the benefits of using *PSNR* or *SSIM* directly in loss function.

We are going to work with **2D slices of 3D brain MRI volumes**. It means that in our project we are going to reconstruct 2D brain MR images. In order to get 2D images from volumes, we have to get slices, as we can see in image 2.8. We can get many 2D images from one brain volume. So, in the preprocessing step, we firstly must choose **what brain MRI view we are going to work with**. For non-isotropic acquisitions, we should ideally slice them so

that the slices are high resolution: select slices where voxel dimension remains equal for the 2 slice dimensions (i.e. $1 \times 1 \times 5 \text{ mm}^3$ should transform into slices of $1 \times 1 \text{ mm}^2$ and not $1 \times 5 \text{ mm}^2$).

But main step in choosing 2D slices is not that, main step is choose slices which retrieves relevant information. A volume can be seen as a 3D head and some slices (i.e. from the sides) can not retrieve relevant information, because it will retrieve noise or bone parts, but it don't show information about brain structure. We can see this fact in the image 2.9, in which is shown the same volume of the image 2.8 but different slice. In order to get images with relevant information, we have recompiled some main methods in the state of art: **get fixed number of slices from all volumes, get the middle slice from the volume or develop ourselves a computer vision tool to evaluate the thickness (with opencv)** . Although the first approach seems the simplest, it is the most used for its good results.



Figure 2.8: 2D slices from brain volume IXI ID 002. Different profiles can be seen. Source: myself

Brain MRI preprocessing differs depending on the objective. First of all, some of the studies use the images of the dataset directly as the **target output** of the network. Other works **enhance the image quality and contrast** before sending it like target output. To continue, other preprocessing can be made when sending brain MRI to the input layers. **Downsampling** the input images could be also useful to reduce the number of parameters of the network. So the resolution of the input image should be balanced between usability (if it is very small is not useful at all) and trainability. We can **normalize the intensity** of pixels (values from 0 to 1 or mean 0 and standard deviation 1) is very important for a neural network. Normalization of the inputs helps the training of the network due to several reasons. One of the most important is that a target variable with a large spread of values may result in large

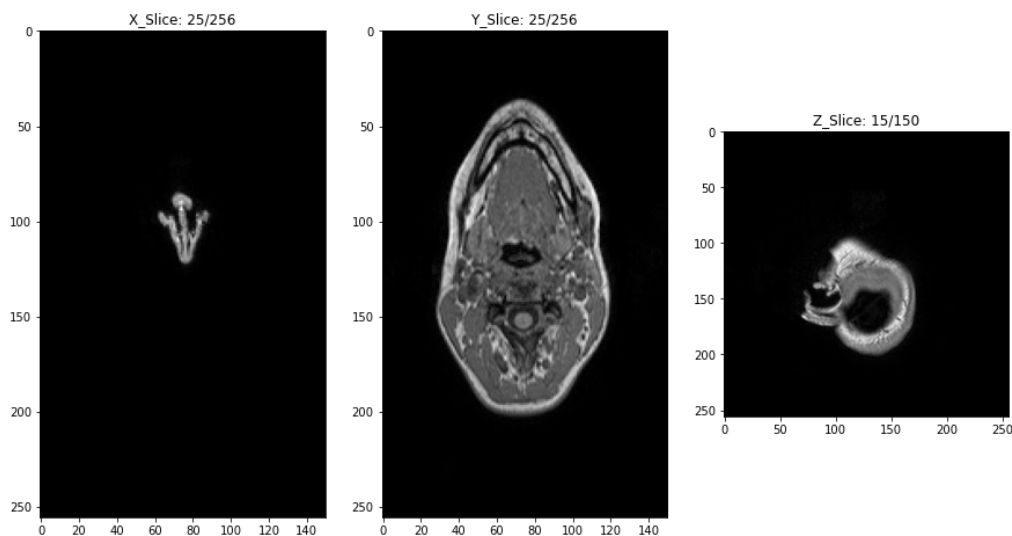


Figure 2.9: 2D slices from IXI ID 002. Slices from volume sides with no relevant information. Source: myself

error gradient values causing weight values to change dramatically, making the learning process unstable [32]. Besides, **affine transformations** like translation or rotation could be applied. In addition, we can realize **data augmentation** in real-time by adding Gaussian noise, other noise or cropping some parts (fixed rectangles or with lesion masks like lesionBrain [25]).

With this research of the art, we are ready to develop the next stages our approach to brain MRI reconstruction. We emphasize that the paper-discovery techniques used in this research are improved by new frameworks described in section 6.2.

Finally, we made a recompilation of papers reviewed in Table 1 of D. Tamada, 2020 [9] and by our own, with some removals and additions based in our goal of the project (see 2.2). We have just deeply explained some of them. From the table of D. Tamada we only obtain 1 autoencoder study [5], 3 sCNN and DnCNN approaches [33] [24] [34] and 1 GAN study. The other papers have been compiled by our own (Autoencoder based: [18] [3] [22] [21] [35] [27] [26], GAN-Autoencoder-based: [23]).

| Purpose | Year, Authors | Network |
|--|--|--|
| Autoencoders | | |
| Identify brain abnormal structural patterns | 2018, W. Pinaya, et al [18] | Semi-supervised autoencoder with SeLU and loss MSE+cross-variance |
| 3D Tumor segmentation | 2018, A. Myronenko [3] [Code available] | VAE for regularization to encoder (ResNet like) |
| Lesion inpainting | 2020, J. V. Manjón et al [4] | 3D UNet autoencoder with skip-connections and upsampling at end |
| General image denoising and super resolution | 2016, XJ. Mao et al [21] [Code available] | Residual CAE with symmetric skip connections |
| Learn Brain volumetric representation | 2018, Evan M. Yu et. al. [27] [Code available] | STN+Residual CAE with skip connections, IN and LReLU |
| 3D Tumor segmentation | 2020, T. Estienne [26] [Code available] | VNET Autoencoder for regularization to encoder. |
| Denoising for T1 weighted brain MRI | 2018, C. Bermudez, et al [5] | Autoencoder with skip connections |
| Medical image denoise | 2016, L. Gondara, et al [22] | Convolutional denoising autoencoder |
| Brain MRI denoise | 2019, N. Chauhan et al [35] | Convolutional denoising autoencoder with Fuzzy Logic filters |
| sCNN and DnCNN | | |
| Denoising for T1, T2 and FLAIR brain images | 2018, M. Kidoh, et al [33] | Single-scale CNN with DCT |
| Motion artifact reduction for brain MRI | 2018, P. Johnson, et al [36] | Single-scale CNN |
| Denoising for multishot DWI | 2020, M Kawamura et al [34] | DnCNN with Noise2Noise |
| GAN | | |
| Motion artifact reduction for brain MRI | 2018 BA. Duffy, et al [24] | GAN with HighRes3dnet as generator |
| Denoise 3D MRI | 2019, M. Ran et al [23] | Wasserstein GAN with Convolutional Autoencoder generator |

Table 2.2: Overview of studies for reconstruction based in Table 1 from D. Tamada [9] (In bold the autoencoder related architecture)

Chapter 3

Scope

In this chapter, we will establish the aims of the project. We have just spoken about the problem to be solved. So now we have to enumerate the concrete objectives of the project.

3.1 Hypothesis

We will build an autoencoder for reconstructing **T1-weighted brain MRI**. It will learn how to encode the underlying structure healthy brains in a lower-dimension space, and reconstruct the MRI from this space. Image quality will be improved reducing noise and artifacts and also could be used for lesion inpainting.

3.2 Primary aims

- To build a **autoencoder** that gets good results with control **T1-weighted brain MRI**: given a "clean" T1-WMRI, the autoencoder will return the same image as equal as we can to the original.
- Due to the nature of the autoencoder train, it will be able to remove noise or reconstruct hidden parts of the input MRI (making data-augmentation in real time).
- Research a good autoencoder architecture and parameters (loss function, batch-norm or not batch-norm, regularization, etc).

- Establish a good brain MRI pre-processing.

3.3 Secondary aims

- Develop the Deep Learning code using one of the most relevant framework, Python, and one of the best-known libraries: **Tensorflow, Keras, or Pytorch (To Be chosen in further stages)** .
- Use an agile methodology: SCRUM. This methodology should be used in the project. We will use the Zenhub tool of Github as a helper in the project management.
- Research the reconstruct applications for image enhancement, disease detection or data augmentation.

These next objectives will be addressed if the primary ones are reached. We could see these aims like a extra for the project. If we achieved good performance in this task, we would research about how to apply this solution to disease detection or data augmentation.

- Build a semi-supervised autoencoder.
- Build a tumor detection system (based on supervised learning or based in the output of the autoencoder [18]).
- Research GAN architectures for noise and artifact reduction.

Chapter 4

Planning and Methodology

In this chapter, we are going to discuss the scheduling for the project and the methodology used in this one.

4.1 Research plan

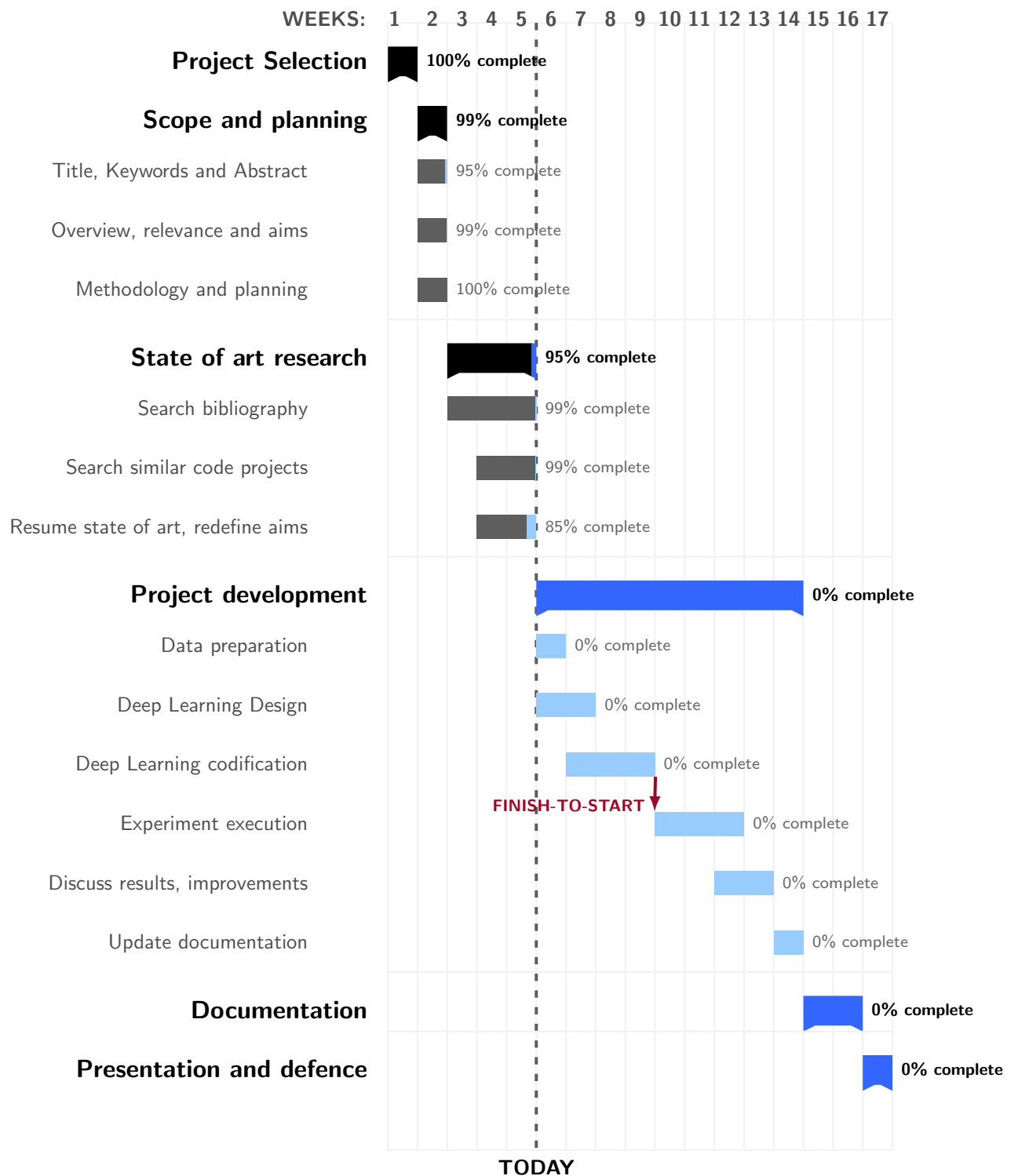
In this section, we are making a time planning for our project. Planning a project is a very important feature, because we can manage the time properly and we can keep a realistic task-calendar. For this purpose, we are going to elaborate a **Gantt Diagram**. This diagram is a very common resource used in project management [\[37\]](#).

Our diagram is a weekly Gannt Diagram. It has 17 weeks ([mm/dd/yyyy]):

- Week 1: from 09/14/2020 to 09/20/2020
- Week 17: from 01/01/2020 to 01/10/2021

It is built by all the main tasks that a master's degree final project must have and some personalized ones for this project. So we will have 6 big phases derived from project submits.

This diagram is up to the date of October 18, 2020. We show it in the next page.



4.2 Methodology

In this section we must choose a common academic Data Mining development methodology, in which there are described the phases, tasks and its relationships.

The description and nature of the project are very helpful at this point because the methodology used in the project will depend on the nature of it. The main characteristic of this project is its research-oriented purpose, so we can label the project as an **academic research project**. Nevertheless, the main objective of this research is to develop a software component (a Deep Convolutional Autoencoder). We can also describe the project as a **software project**. In addition, the project is located in the Machine Learning and Deep Learning areas. These areas are very related to Maths, Statistics, and Computer Science. In all of these fields, the aim is to analyze data in a quantitative way. We analyze how the variables are related, how the autoencoder performance with a concrete measure, how it trains getting concrete metrics (how it learns, time, overfitting...), etc. So our methodology should be **quantitative**. We will take a representative sample of brain MRI, we will train the autoencoder and inference the results to all the population. All this sample and inference techniques are addressed by the validation methods of Machine Learning (Train/test, Cross-validation to reduce bias, etc).

So, due to the nature of the project, we have to apply a methodology for an **quantitative academic research project for data mining software development**.

In a very summarized way, we will start researching the state of art, defining the problem, and proposing a model to solve the target problem. We will choose and prepare our data. Then we will develop the data mining software solution for this problem, evaluating each step. Finally, we will evaluate our model and get a conclusion for our hypothesis. Thus, **CRISP-DM methodology** embed all of these steps and it will be chosen as the project methodology.

The methodology that best suits our project is **CRISP-DM** [38]. The ***CR**oss-**I**ndustry **S**tandard **P**rocess for **D**ata **M**ining* is a framework used for creating and deploying machine learning solutions. Moreover, research and quantitative tasks can be embedded in the CRISP-DM phases (i.e. state of art research phase can fit into business understanding CRISP-DM phase and quantitative evaluation can fit into model evaluation).

As we know, agile methodologies are often used in software development. CRISP-DM is neither an agile methodology nor a waterfall one. This methodology has clear stages, but the

order of them is not strict and we could move forward and back whenever we need, in order to improve our data mining final model. In fact, this movement between phases is widely used. Also it has a iterative cycle, in which data, data preparation, modelling and evaluation are improved with the previous iteration feedback.

Figure 4.1 shows the phase dependencies and order. As we can see, the straight lines define the dependencies between phases as in a classical methodology. Nevertheless, We can see the circle and the two-arrowed straight lines that show the flexibility and the agile similarity of CRISP-DM.

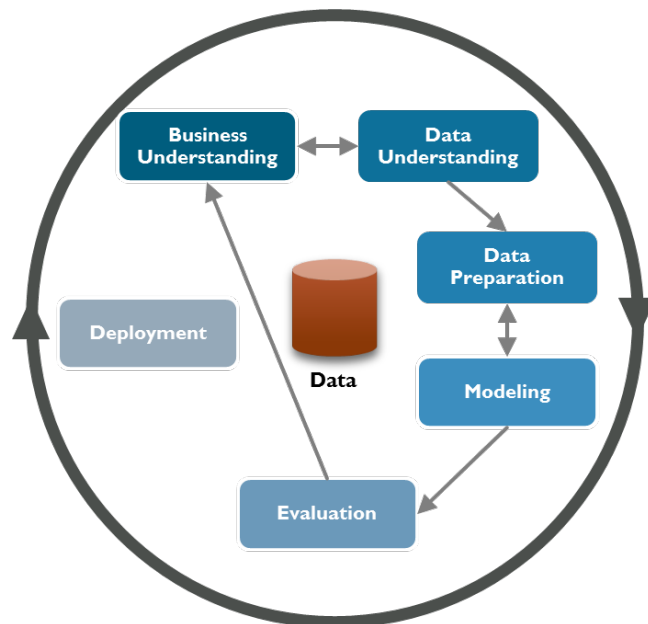


Figure 4.1: CRISP-DM Cycle

The phases of CRISP-DM [38] are the following (In later stages, references will be made to the specific sections of the document where the phases are carried out):

Business Understanding : deep analysis of the business needs. In this phase we can establish an objective. In our case, we can research the state of art for Deep Convolutional Autoencoder for brain MRI and propose a model based on this research.

Data Understanding : we should research the data sources as IXI¹, data quality and we should explore the data and its characteristics.

Data Preparation : Data should be cleaned, filtered, selected and integrated if necessary.

¹<https://brain-development.org/ixi-dataset/>

We could carry out tasks like preprocessing T1 weighted brain MRI or realize data augmentation. I will be explained in-depth in section [7.2](#).

Modeling : Specify the model to use and the architecture, parameters, etc. Maybe running several model architecture and hyper-parameter optimization to reach the most powerful model. So in can be an iterative process.

Evaluation : We must evaluate models properly to get meaningful conclusions. There are many techniques of model evaluation and it should be made carefully.

Deployment - Publication : As our main goal is academic research, this phase would be *Publication*. The tasks are: review the project and generate the final report.

Chapter 5

Theoretical concepts

To Be Done in other stages. I don't consider this state of the art research, but explanations about general concepts of the project

5.1 Data mining concepts

5.1.1 Deep Learning

5.1.2 Computer vision

5.1.3 Architectures

5.1.4 Autoencoder

5.2 Neuroimaging concepts

5.2.1 Magnetic Resonance

5.2.2 Magnetic Resonance Images Types

Chapter 6

Techniques and tools

6.1 To Be Done

6.2 New paper-discovery frameworks

In this state of art research, we have use new techniques and frameworks among the classical ones. [Connected Papers](https://www.connectedpapers.com/)¹ is a network science framework to improve the search of papers. There are also other platforms like [Papers With Code](https://paperswithcode.com/)² and [Distill](https://distill.pub/)³ that improve the experience of article discovering and article visualization-interaction.

I will explain some little examples of paper discovery using this tools. *Connected Papers* retrieve us a graph about the relationship of a given paper. A paper is related with another if one is cited by the other. In addition, the graph contains papers cited by the sucesors of the main one. So, the graph contains the most important prior works and Derivate works of our main paper, giving us a perfect tool for paper discovery. [One example](https://www.connectedpapers.com/main/37a18be8c599b781cc28b6a62d8f11e8a6a75169/3D-MRI-brain-tumor-segmentation-using-autoencoder-regularization/graph)⁴ made for A. Myronenko work [3] is shown in figure 6.1

Papers with code is a web page, in which are stored papers with it official code implementations. It also group works in different subjects of study, like Medical issues, image segmentation,

¹<https://www.connectedpapers.com/>

²<https://paperswithcode.com/>

³<https://distill.pub/>

⁴<https://www.connectedpapers.com/main/37a18be8c599b781cc28b6a62d8f11e8a6a75169/3D-MRI-brain-tumor-segmentation-using-autoencoder-regularization/graph>

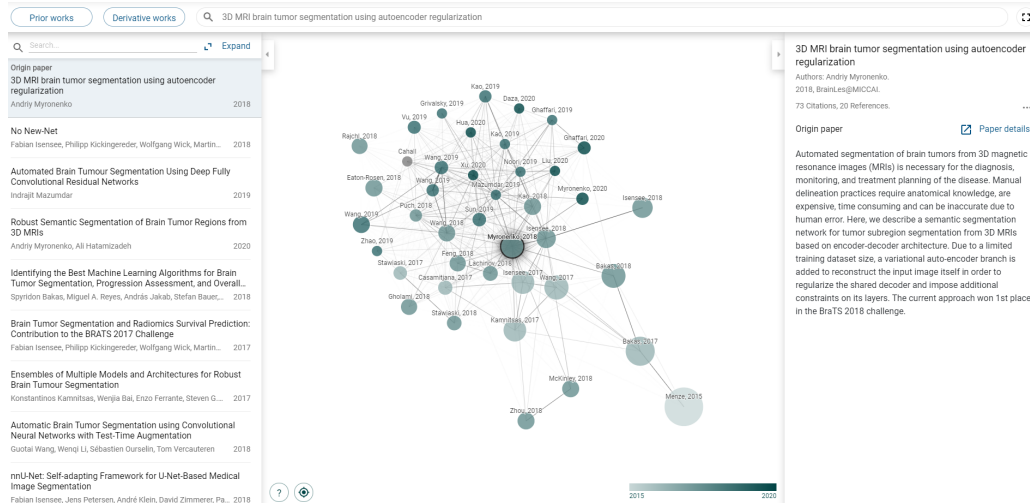


Figure 6.1: Graph of connected papers for A. Myronenko 2018 [3]

etc. In addition, it recompiles the benchmarks for different machine learning tasks, and it makes a ranking of the papers (with the code and pdf linked from the page). We show in figure 6.2 an example of the Myronenko work. We can see the abstract, the link of the paper and multiple links for implementations. Also, we can see the tasks and the benchmark results. There are more things that we can do and research in this framework.

3D MRI brain tumor segmentation using autoencoder regularization

27 Oct 2018 • Andriy Myronenko

Automated segmentation of brain tumors from 3D magnetic resonance images (MRIs) is necessary for the diagnosis, monitoring, and treatment planning of the disease. Manual delineation practices require anatomical knowledge, are expensive, time consuming and can be inaccurate due to human error... ([read more](#))



Code

[Edit](#)

| | | |
|---|-------|------------|
| IAmSuyogJadhav/3d-mri-brain-tumor-segmentation-using-autoencoder-regularization | ★ 178 | |
| black0017/MedicalZooPytorch | ★ 26 | PyTorch |
| sinclairjang/3D-MRI-brain-tumor-segmentation-using-autoencoder-regularization | ★ 5 | PyTorch |
| trungnhanuchiha/VAE_BRATS2018 | ★ 4 | |
| FBehrad/Brats2018 | ★ 0 | TensorFlow |

Tasks

[Edit](#)

- BRAIN TUMOR SEGMENTATION
- SEMANTIC SEGMENTATION
- TUMOR SEGMENTATION

Results from the Paper

[Edit](#)

| Get a GitHub badge | | | | | | |
|------------------------------------|------------|---------|-------------|--------------|-------------|-------------------------|
| TASK | DATASET | MODEL | METRIC NAME | METRIC VALUE | GLOBAL RANK | BENCHMARK |
| Brain Tumor Segmentation | BRATS 2018 | NVDLMED | Dice Score | 0.87049 | # 1 | Compare |

Figure 6.2: A. Myronenko work [3] in Papers with code

Chapter 7

Project development

TBD: The text wrote in this section is not official. This text are notes for further stages. At current stage, state of art, we only write the section of the related works

7.1 To Be Done

7.2 Data Life-cycle

- Capture: download from web.
- Storage: Locally.
- Preprocessing:
 - Cleaning and filtering: clean corrupted data, filter no-needed data.
 - MRI needed preprocessing (FreeSurfer in [18]).
 - Feature engineering: Normalize, downsampling.
 - Data augmentation: Noise addition, crop parts, ¿Rotations?
- Analysis: create autoencoder
- **Visualization: Optional - Make some visualizations from results, layers or behavior of the autoencoder. Ver las capas de los kernels como se activan con las entradas, ver resultados de capas intermedias, ver gráficas de callbacks.**

- Publication.

7.3 Data Exploration

First notebook: MRI data familiarization and nibabel familiarization. How to manage MRI, etc. Ids, spreadsheet of metadata, voxel sizes, shape sizes.

- $0.9375 \times 0.9375 \times 1.2 \text{ mm}^3 \rightarrow 576 \text{ volumes}$.
- $0.9766 \times 0.9766 \times 1.2 \text{ mm}^3 \rightarrow 5 \text{ volumes}$.

7.4 Data Preprocessing

Tell about voxel inspection to select profile

Tell about orientation checking (nifti header or subset checking)

Tell about how to get relevant slices

Tell about image preprocessing (translate, contrast, downsampling, normalize)

Tell about split train-test

Tell about data augmentation (Gaussian blurring, noise, crop, crop+inpaint)

- Select profile: voxel size
- intensity-normalized
- Select slices: relevant information
- Clean duplicates
- Improve target images: mean local, enhance contrast, etc
- Downsampling
- Pixel normalization: range [0-1]

- Data split: split train and test: stratify by sex, age and etc.
- Data augmentation in train
 - Add gaussian noise
 - Crop images

For non-isotropic acquisitions, we should ideally slice them so that the slices are high resolution. For example, if the voxel resolution is $1 \times 1 \times 5$ mm³, we should slice the volume so that the slices are 1×1 mm² rather than 1×5 mm² (or 5×1 mm²) 2. Overriding the above, we need to be consistent in which orientation we are slicing. In other words, if we are getting axial slices from one volume, we should make sure we get axial slices for all patients. Otherwise, the network will likely not train well.

7.5 Results

Make some filter exploration when noise, when cropped, when clear, etc. <http://deeplearning.net/tutorial/dA.html#running-the-code>¹

¹<http://deeplearning.net/tutorial/dA.html#running-the-code>

Chapter 8

Conclusion and Outlook

8.1 To Be Done

Bibliography

- [1] David C Preston. *Magnetic Resonance Imaging (MRI) of the Brain and Spine: Basics*. Case Western Reserve University, 2006.
- [2] José V Manjón, Pierrick Coupé, Luis Martí-Bonmatí, D Louis Collins, and Montserrat Robles. Adaptive non-local means denoising of mr images with spatially varying noise levels. *Journal of Magnetic Resonance Imaging*, 31(1):192–203, 2010.
- [3] Andriy Myronenko. 3d mri brain tumor segmentation using autoencoder regularization. In *International MICCAI Brainlesion Workshop*, pages 311–320. Springer LNCS, 2018.
- [4] José V. Manjón, José E. Romero, Roberto Vivo-Hernando, Gregorio Rubio, Fernando Aparici, Maria de la Iglesia-Vaya, Thomas Tourdias, and Pierrick Coupé. Blind mri brain lesion inpainting using deep learning. In Ninon Burgos, David Svoboda, Jelmer M. Wolterink, and Can Zhao, editors, *Simulation and Synthesis in Medical Imaging*, pages 41–49, Cham, 2020. Springer International Publishing.
- [5] Camilo Bermudez, Andrew J Plassard, Larry T Davis, Allen T Newton, Susan M Resnick, and Bennett A Landman. Learning implicit brain mri manifolds with deep learning. In *Medical Imaging 2018: Image Processing*, volume 10574, page 105741L. International Society for Optics and Photonics, 2018.
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.
- [8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

-
- [9] Daiki Tamada. Noise and artifact reduction for mri using deep learning. *arXiv*, abs/2002.12889, 2020.
- [10] Mohammed T Abou-Saleh. Neuroimaging in psychiatry: an update. *Journal of Psychosomatic Research*, 61(3):289–293, 2006.
- [11] Andre F Marquand, Ieab Rezek, Jan Buitelaar, and Christian F Beckmann. Understanding heterogeneity in clinical cohorts using normative models: beyond case-control studies. *Biological psychiatry*, 80(7):552–561, 2016.
- [12] Janaina Mourão-Miranda, David R Hardoon, Tim Hahn, Andre F Marquand, Steve CR Williams, John Shawe-Taylor, and Michael Brammer. Patient classification as an outlier detection problem: an application of the one-class support vector machine. *Neuroimage*, 58(3):793–804, 2011.
- [13] Changhee Han, Leonardo Rundo, Ryosuke Araki, Yujiro Furukawa, Giancarlo Mauri, Hideki Nakayama, and Hideaki Hayashi. Infinite brain tumor images: Can gan-based data augmentation improve tumor detection on mr images? In *Proc. Meeting on Image Recognition and Understanding (MIRU 2018), Sapporo, Japan*, 2018.
- [14] Florian Knoll, Jure Zbontar, Anuroop Sriram, Matthew J Muckley, Mary Bruno, Aaron Defazio, Marc Parente, Krzysztof J Geras, Joe Katsnelson, Hersh Chandarana, et al. fastmri: A publicly available raw k-space and dicom dataset of knee images for accelerated mr image reconstruction using machine learning. *Radiology: Artificial Intelligence*, 2(1):e190007, 2020.
- [15] Errol M Bellon, E Mark Haacke, Paul E Coleman, Damon C Sacco, David A Steiger, and Raymond E Gangarosa. Mr artifacts: a review. *American Journal of Roentgenology*, 147(6):1271–1281, 1986.
- [16] MA Balafar. Review of noise reducing algorithms for brain mri images. *methods*, 10:11, 2012.
- [17] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.
- [18] Walter HL Pinaya, Andrea Mechelli, and João R Sato. Using deep autoencoders to identify abnormal brain structural patterns in neuropsychiatric disorders: A large-scale multi-sample study. *Human brain mapping*, 40(3):944–954, 2019.

- [19] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2014.
- [20] Gihyun Kwon, Chihye Han, and Dae-shik Kim. Generation of 3d brain mri using auto-encoding generative adversarial networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 118–126. Springer, 2019.
- [21] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using convolutional auto-encoders with symmetric skip connections. *arXiv preprint arXiv:1606.08921*, 2016.
- [22] Lovedeep Gondara. Medical image denoising using convolutional denoising autoencoders. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 241–246. IEEE, 2016.
- [23] Maosong Ran, Jinrong Hu, Yang Chen, Hu Chen, Huaiqiang Sun, Jiliu Zhou, and Yi Zhang. Denoising of 3d magnetic resonance images using a residual encoder–decoder wasserstein generative adversarial network. *Medical image analysis*, 55:165–180, 2019.
- [24] Ben A Duffy, Wenlu Zhang, Haoteng Tang, Lu Zhao, Meng Law, Arthur W Toga, and Hosung Kim. Retrospective correction of motion artifact affected structural mri images using deep learning of simulated motion. *MIDL 2018 Conference*, 2018.
- [25] Pierrick Coupé, Thomas Tourdias, Pierre Linck, José E. Romero, and José V. Manjón. Lesionbrain: An online tool for white matter lesion segmentation. In Wenjia Bai, Gerard Sanroma, Guorong Wu, Brent C. Munsell, Yiqiang Zhan, and Pierrick Coupé, editors, *Patch-Based Techniques in Medical Imaging*, pages 95–103, Cham, 2018. Springer International Publishing.
- [26] Théo Estienne, M. Lerousseau, M. Vakalopoulou, Emilie Alvarez Andres, E. Battistella, Alexandre Carré, S. Chandra, S. Christodoulidis, M. Sahasrabudhe, Roger Sun, C. Robert, Hugues Talbot, N. Paragios, and E. Deutsch. Deep learning-based concurrent brain registration and tumor segmentation. *Frontiers in Computational Neuroscience*, 14, 2020.
- [27] E. M. Yu and M. R. Sabuncu. A convolutional autoencoder approach to learn volumetric shape representations for brain structures. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 1559–1562, 2019.
- [28] Spyridon Bakas, Mauricio Reyes, András Jakab, Stefan Bauer, Markus Rempfler, Alessandro Crimi, Russell Takeshi Shinohara, Christoph Berger, Sung Min Ha, Martin Rozycki,

- Marcel Prastawa, Esther Alberts, Jana Lipková, John B. Freymann, Justin S. Kirby, Michel Bilello, Hassan M. Fathallah-Shaykh, Roland Wiest, Jan Kirschke, Benedikt Wiestler, Rivka R. Colen, Aikaterini Kotrotsou, Pamela LaMontagne, Daniel S. Marcus, Mikhail Milchenko, Arash Nazeri, Marc-André Weber, Abhishek Mahajan, Ujjwal Baid, Dongjin Kwon, Manu Agarwal, Mahbubul Alam, Alberto Albiol, Antonio Albiol, Alex Varghese, Tran Anh Tuan, Tal Arbel, Aaron Avery, Pranjal B., Subhashis Banerjee, Thomas Batchelder, Kayhan N. Batmanghelich, Enzo Battistella, Martin Bendszus, Eze Benson, José Bernal, George Biros, Mariano Cabezas, Siddhartha Chandra, Yi-Ju Chang, and et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge. *CoRR*, abs/1811.02629, 2018.
- [29] Adrian Tousignant, Paul Lemaître, Doina Precup, Douglas L. Arnold, and Tal Arbel. Prediction of disease progression in multiple sclerosis patients using deep learning analysis of mri data. volume 102 of *Proceedings of Machine Learning Research*, pages 483–492, London, United Kingdom, 08–10 Jul 2019. PMLR.
- [30] Mohammadi Fatemeh. *Convolutional Autoencoder for Studying Dynamic Functional Brain Connectivity in Resting-State Functional MRI*. PhD thesis, Concordia University, April 2019.
- [31] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE, 2016.
- [32] Christopher M Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.
- [33] Masafumi Kidoh, Kensuke Shinoda, Mika Kitajima, Kenzo Isogawa, Masahito Nambu, Hiroyuki Uetani, Kosuke Morita, Takeshi Nakaura, Machiko Tateishi, Yuichi Yamashita, et al. Deep learning based noise reduction for brain mr imaging: tests on phantoms and healthy volunteers. *Magnetic Resonance in Medical Sciences*, pages mp–2019, 2019.
- [34] Motohide Kawamura, Daiki Tamada, Satoshi Funayama, Marie-Luise Kromrey, Shintaro Ichikawa, Hiroshi Onishi, and Utaroh Motosugi. Accelerated acquisition of high-resolution diffusion-weighted imaging of the brain with a multi-shot echo-planar sequence: Deep-learning-based denoising. *Magnetic Resonance in Medical Sciences*, pages tn–2019, 2020.

-
- [35] Nishant Chauhan and Byung-Jae Choi. Denoising approaches using fuzzy logic and convolutional autoencoders for human brain mri image. *International Journal of Fuzzy Logic and Intelligent Systems*, 19(3):135–139, 2019.
 - [36] PM Johnson and M Drangova. Motion correction in mri using deep learning. In *Proceedings of the ISMRM Scientific Meeting & Exhibition, Paris*, volume 4098, 2018.
 - [37] Irida da Cunha. *El trabajo de fin de grado y de máster: Redacción, defensa y publicación*. Editorial UOC, 2016.
 - [38] Rüdiger Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, pages 29–39. Springer-Verlag London, UK, 2000.