الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique

Master Thesis

Field: Electronics

Option: Electronics of Embedded Systems

Presented by:

Abdouhadi TCHAGBELE
&
Sid Ahmed HAMDAD

Using Deep Learning for MRI Stroke Lesion Segmentation

Supervised by:

Promoter: M. Yacine KABIR

Co-promoter: Mme Djamila NACEUR

Jury Members:

President: Prof. Abderrezak GUESSOUM

Examiner: Dr. Hamida BOUGHERIRA

Academic Year 2022/2023

Acknowledgments

First and foremost, we would like to thank the Almighty God for granting us the privilege and opportunity to study and pursue the path of science and knowledge.

The completion of this thesis was made possible thanks to the contribution of several individuals to whom we would like to express our deepest gratitude.

We would like to express our immense gratitude to Mr. Yacine KABIR and Mrs. Djamila NACEUR, our supervisors, who have accompanied us throughout this professional experience with great patience and pedagogy.

We also aspire to extend our thanks to the jury members president Pr Abderrezak GUESSOUM and examiner Dr. Hamida BOUGHERIRA who have agreed to review and evaluate our work.

We also see it fit to thank all the members of our respective families for their valuable assistance, support, and encouragement.

We are pleased to express our sincere thanks to all the teachers we have had during our training, particularly those in the Department of Electronics.

We desire to express our gratitude to the friends and colleagues who have provided us with their moral and intellectual support throughout our journey.

We truly appreciate the guidance and insights that we have received from all individuals we have encountered during our training. Their invaluable advice has not only helped us improve our academic and professional skills but also encouraged us to pursue our goals with greater confidence and determination. We are grateful for their support and encouragement, and we hope to maintain the connections we have established with them as we move forward in our careers.

Dedication

This work is dedicated to:

-To my dear parents, who have dedicated their lives to building mine, for their support, patience, tenderness, and affection, for everything they have done to help me reach this stage.

-To my brothers, for their constant support and encouragement throughout this thesis project. Their presence and support have been invaluable to me. Thank you from the bottom of my heart.
-To my dear friend and study partner Abdouhadi.
-To my dear friends who have supported and encouraged me.
-To all those who have helped me, directly or indirectly.

Sid Ahmed HAMDAD

I dedicate this modest work to:
- My dearest mother, the source of life, love, and affection,
- All my teachers and professors,
- My entire family, a source of hope and motivation,
- All my friends, including my dear friend and partner Sid Ahmed,
- Everyone who has helped me, whether directly or indirectly.

Abdouhadi TCHAGBELE

Abstract

The project aims to exploit Deep Learning techniques for the segmentation of MRI images to assist radiology specialists in the detection of strokes. The main aim is to improve the detection process using the advantages of artificial intelligence.

Several approaches were explored, including the creation of a customized CNN model, the use of learning transfer and the ensemble learning. An in-depth comparative study was carried out to evaluate the performance of the different models obtained, focusing in particular on Dice, IoU and Precision scores. Ultimately, after careful evaluation, two of our models, **Efficientnetb3-50** and **Model HT-FLAIR**, were selected and proved to be the best choice thanks to their better scores.

Keywords: Segmentation, Stroke ; MRI ; AI ; Deep Learning

Résumé

Le projet vise à exploiter les techniques de Deep Learning pour la segmentation des images IRM d'accidents vasculaires cérébraux (AVC) afin d'assister les spécialistes en radiologie dans la détection de ces AVC. L'objectif principal est d'améliorer le processus de détection en utilisant les avantages de l'intelligence artificielle. Plusieurs approches ont été explorées, notamment la création d'un modèle CNN personnalisé, l'utilisation du transfert d'apprentissage et l'apprentissage d'ensemble. Une étude comparative approfondie a été réalisée pour évaluer les performances des différents modèles obtenus, en se concentrant notamment sur les scores Dice, IoU et la Precision. Finalement deux de nos modèles à savoir « **Efficientnetb3-50** » et « **Model HT-FLAIR** » ont été retenus et se sont révélés être le meilleur choix de par leurs bons scores.

Mots clés : Segmentation, AVC ; IRM ; IA ; Apprentissage profond.

ملخص:

المشروع يهدف إلى استغلال تقنيات التعلم العميق (Deep Learning) لتقسيم صور الرنين المغناطيسي لمساعدة أطباء الأشعة في اكتشاف الجلطات. الهدف الرئيسي هو تحسين عملية الكشف باستخدام مزايا الذكاء الاصطناعي.

تم استكشاف عدة طرق، بما في ذلك إنشاء نموذج CNN مخصص، واستخدام نقل التعلم وتقنية التعلم المتعدد. تم إجراء دراسة مقارنة معمقة لتقييم أداء النماذج المختلفة المتحصل عليها، مع التركيز بشكل خاص على قياسات Dice و IoU وPrecision. في النهاية، تم اختيار نموذجينا Efficientnetb3-50 و Model HT-FLAIR ، بحيث اظهرا أنهما الأفضل بفضل قياساتيهما الجيدة.

الكلمات المفتاحية: تقسيم، جلطة دماغية، رنين مغناطيسي، ذكاء اصطناعي، تعلم عميق.

Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Networks
ATLAS	Anatomical Tracings of Lesions After Stroke
AVC	Accident Vasculaire Cérébral
BLSTM	Bidirectional Long Short-Term Memory
CAD	Computer-Aided Detection
CNN	Convolutional Neural Network
CONV	Convolution layer
CT	Computed Tomography
DSRPAI	Dartmouth Summer Research Project on Artificial Intelligence
DWI	Diffusion-Weighted Imaging
FC	Fully Connected layer
FRAIR	Fluid-attenuated Inversion Recovery
GUI	Graphical User Interface
HLSTM	Heterogeneous Long Short-Term Memory
IDE	Integrated Development Environment
IOU	Intersection Over Union
ISLES	Ischemic Stroke Lesion Segmentation
LSTM	Long Short-Term Memory
MDLSTM	Multi Dimensional Long Short-Term Memory
MLP	Multi-layer Perceptron

MRA	Magnetic Resonance Angiograph
MRI	Magnetic Resonance Imaging
NMR	Nuclear Magnetic Resonance
NIFTI	Neuroimaging Informatics Technology Initiative
PCA	Principal Component Analysis
PET	Positron Emission Tomography
PWI	Perfusion-Weighted Imaging
RMSprop	Root Mean Square Propagation
SPECT	Single Photon Emission Computed Tomography
SISS	Sub-Acute Stroke Lesion Segmentation
SVM	Support Vector Machine
TE	Echo Time
TR	Relaxation Time
WHO	World Health Organization

Contents

List of Figures

List of Tables

General Introduction

Stroke is one of the leading causes of death and disability in the world. According to the World Health Organization (WHO), approximately 15 million people have a stroke each year, 5 million die and 5 million are permanently disabled [12].

Stroke is a medical emergency that requires prompt and effective management to reduce the risk of death and disability. In order to diagnose this serious disease, devices such as Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) are used by radiologists.

MRI is an invaluable tool for stroke analysis, enabling visualization of brain lesions, monitoring disease progression and guiding treatment. MRI is a non-invasive technique that uses a magnetic field and radio waves to visualize the internal structures of the human body. However, the analysis of MRI images of stroke is a complex challenge due to the variability of lesions, the anatomical complexity of the brain and the need to precisely quantify lesions.

Fortunately, with advances in technology, and in order to be able to act quickly in the event of a stroke, research is being carried out into the application of Deep Learning in the field of medical imaging. Indeed, Deep Learning is a machine-learning technique that can detect specific anatomical structures in medical imaging data, such as MRI images of the brain. The use of Deep Learning for the segmentation of MRI stroke images enables the precise delineation of areas of infarction or cerebral lesions.

The deep artificial neural networks used in deep learning are trained on annotated MRI image data (masks), providing recognition models for specific patterns in stroke images. Using these patterns, segmentation algorithms can automatically identify areas of infarction or brain damage in MRI images, which can help medical professionals diagnose

and treat patients more quickly and accurately.

There are several types of CNN architecture for segmentation. In our work, we will use the U-Net architecture and the ATLAS v2 dataset as a first step, and the ISLES2015 dataset to tackle another approach.

The aim is to predict the presence or absence of a lesion and generate a corresponding mask

To structure our work, we have divided it as follows: Chapter 1 provides general information on stroke and MRI, Chapter 2 discusses artificial intelligence and its sub-branches for medical image segmentation, Chapter 3 is dedicated to previous studies conducted in this field, and finally Chapter 4 focuses on our own work.

Chapter 1

General information on stroke and MRI

1.1 Introduction

Stroke, is one of the most dangerous diseases in the world ranked in 2019 second with 11% of deaths recorded in the world after ischemic heart disease with 16% by the WHO [13]. In this chapter we will first discuss in a general way the MRI imaging that is used for medical imaging in case of a suspicion of a stroke in a patient. Then we will discuss stroke and its diagnosis with MRI. We will finish with the problems and research questions as well as the objectives and impact of our research.

1.2 MRI imaging

Magnetic resonance imaging (MRI) is an excellent medical imaging technique that has proven its effectiveness since its appearance. Indeed, it is a non-invasive technique based on the phenomenon of nuclear magnetic resonance physics (NMR) which consists in studying the changes in magnetization of the nuclei of a substance under the joint action of two magnetic fields, namely, a high static magnetic field and a rotating electromagnetic field of radio frequency for example [2]. It allows to visualize in 2D or 3D certain parts of the human body rich in water, among others the brain. It is widely used in the diagnosis and follow-up of stroke because of its high resolution and its ability to differentiate between normal and damaged brain tissue.

Figure 1.1: Image of an MRI machine [1].

1.2.1 How MRI works

The MRI technique is based on the use of a strong, uniform magnetic field to align the protons contained in the hydrogen atoms in the tissues of the human body. In the absence of an external magnetic field, when we take protons from a tissue sample, they are oriented in all directions.

During an examination, the patient is placed in the apparatus containing a strong and uniform magnetic field. The hydrogen protons are then influenced by the magnetic field and they all align themselves in the direction of the magnetic field (Figure 1.3). Subsequently, radio waves are sent through the body to disrupt the alignment of the hydrogen protons in the body tissue. This is the excitation. After a short time (in the order of a few milliseconds) the excitation is stopped, the hydrogen protons in the body tissues gradually return to their state of alignment with the magnetic field, thus emitting radio signals that are detected.

Through special antennas called receiver coils, which are placed around the body part to be imaged, the emitted radio signals are captured. They are then sent to a computer that processes them to produce detailed cross-sectional images of body tissue. These signals are measured by amplitude and frequency (frequency domain to image), and are converted into a gray-scale image.

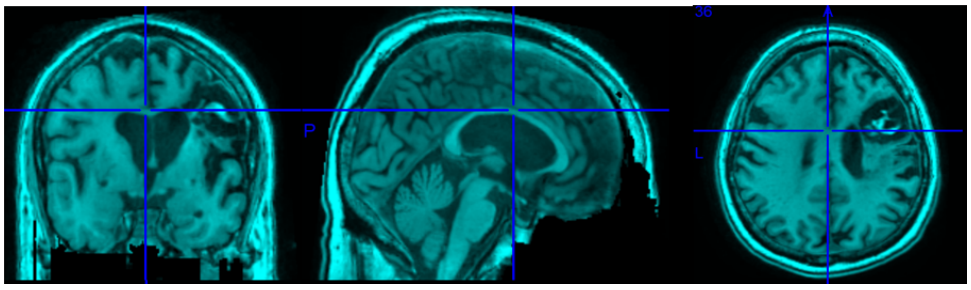


Figure 1.2: Example of brain images respectively from left to right, coronal, sagittal and axial section in T1 sequence displayed with MRIcon.

MRI is a technique capable of providing several types of images, such as structural, functional and diffusion images. Structural images show the anatomy of the brain or other parts of the body, while functional images can show brain activity or connectivity. Diffusion images show changes in the diffusion of water in body tissues, which can help locate brain damage caused by a stroke.

As a technique, MRI is non-invasive and does not produce ionizing radiation, which makes it safe for use in patients. However, MRI can be limited by cost, availability of equipment, the need to remain still during the examination (somewhat distressing for the patient) and the duration of an examination.

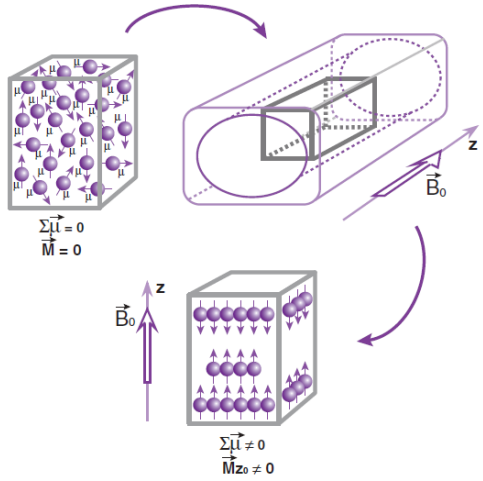


Figure 1.3: In the absence of an external magnetic field, the protons in a tissue sample are randomly oriented in all directions: the sum of the microscopic elemental magnetization vectors is zero and there is no macroscopic magnetization vector. Subjected to an external magnetic field (prevailing in the tunnel), the protons orient themselves along the direction of the latter (Oz) with the appearance of a macroscopic magnetization vector [2].

1.2.2 Use of MRI to diagnose stroke

MRI is a valuable tool for diagnosing stroke because it can provide detailed images of brain lesions, which appear as areas of abnormal signal in the brain. MRI can also be used to determine the cause of the stroke, such as blood clots or bleeding. During an examination, the radiologist can act on several parameters in order to obtain a type of image (weighted) with specific characteristics. By using a combination of these different weightings, doctors can obtain a complete view of the anatomical structures and potential pathologies in a patient's body.

a. T1 weighting

T1 weighting is a magnetic resonance imaging (MRI) sequence that uses a short relaxation time and a short echo time. This sequence is often used to visualize the soft tissues of the brain and is particularly useful for the analysis of the gray and white matter.

In the T1 sequence, tissues that contain a lot of hydrogen protons, such as gray matter, appear in white, while tissues that contain less hydrogen protons, such as white

matter, appear in dark gray. This weighting is therefore used to highlight the differences in proton density in soft tissue.

T1-weighted images are often used to identify brain lesions, such as tumors and infarcts. This sequence is also used to visualize blood vessels in the brain by injecting a contrast agent intravenously to produce contrast images.

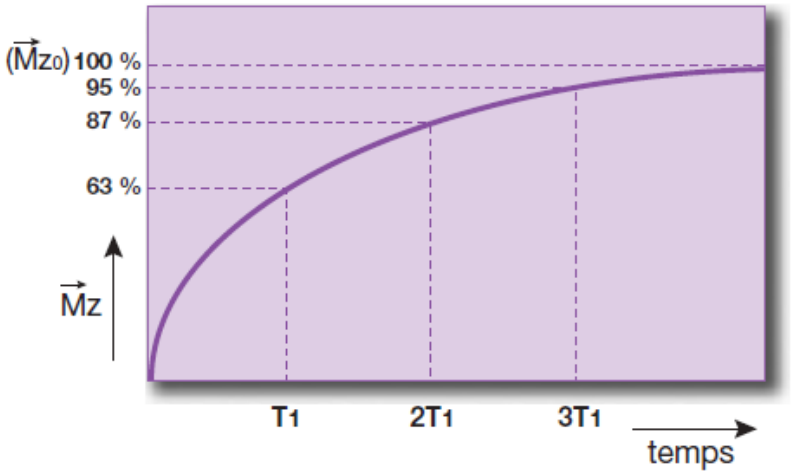


Figure 1.4: Exponential curve of longitudinal magnetization regrowth as a function of T1 [2].

b. T2 weighting

This weighting is often used to visualize soft tissue abnormalities such as brain lesions, edema, tumors, infections and inflammatory diseases. In the T2 sequence, tissues that contain a lot of water, such as white matter, appear as bright white, while tissues that contain less water, such as gray matter, appear as dark gray. Sequences with long TR and long TE provide more sensitive and specific T2 weighting.

The long TR relaxation time refers to the time between radio-frequency pulses, i.e. the time the MRI system takes to reset between each radio-frequency pulse. By increasing the repetition time, the longitudinal relaxation of hydrogen protons is complete, allowing for better tissue differentiation.

The long TE echo time is the time between the radio-frequency pulse and the detection of the signal. By prolonging the echo time, the transverse relaxation of hydrogen protons is complete, which allows a better distinction between tissues.

c. FLAIR weighting

FLAIR (fluid-attenuated inversion recovery) weighted images are a variant of T2

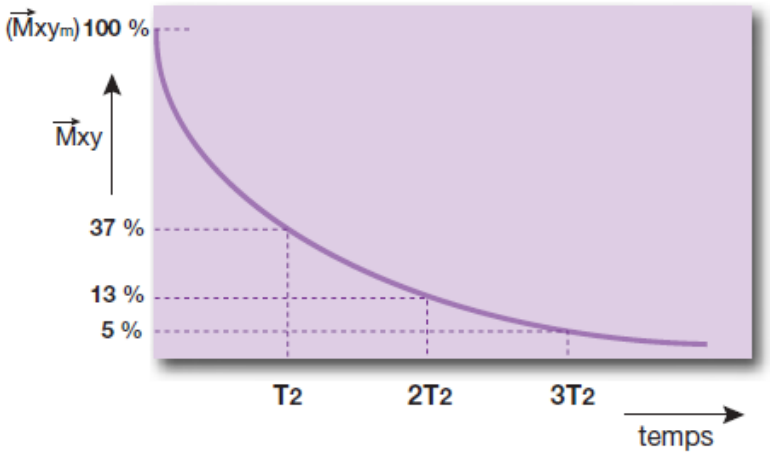


Figure 1.5: Exponential curve of transverse magnetization disappearance as a function of T2 [2].

weighting that removes the signal from the cerebrospinal fluid to highlight abnormalities in surrounding tissue. This weighting is often used to visualize brain lesions, tumors and inflammations.

d. T2* weighting

T2* weighted images are used to visualize tissues that contain iron particles, such as brain lesions caused by hemorrhages. This weighting uses a short TR and a short TE, and tissues that contain iron particles appear in black.

Both T2 relaxation time and T2* relaxation time are measures of hydrogen proton relaxation, but T2 is more sensitive to intrinsic tissue properties, while T2* is more sensitive to local magnetic fields and tissue heterogeneities.

e. DWI weighting

DWI (Diffusion-Weighted Imaging) weighted images are used to visualize the diffusion of water molecules in tissues. This weighting is often used to detect strokes and tumors.

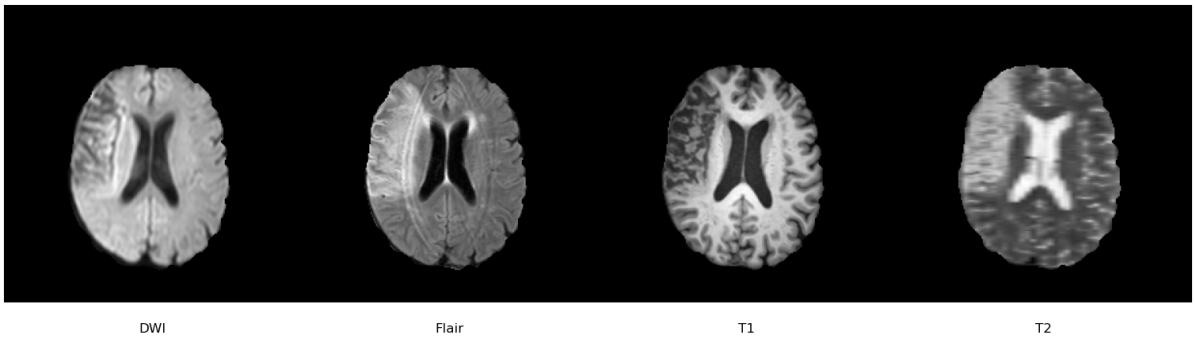


Figure 1.6: Different MRI modalities.

1.2.3 Different types of MRI used to diagnose stroke

There are several types of MRI used to diagnose stroke, each with its advantages and limitations.

a. Diffusion-weighted MRI (DWI)

This technique allows visualization of brain damage in the early hours of stroke by detecting changes in water diffusion in brain tissue. Areas of ischemic brain tissue appear as a hyper-intense signal due to reduced water diffusion. This technique is particularly useful for detecting acute ischemic stroke [14].

b. Perfusion MRI (PWI)

This technique visualizes areas of the brain that are perfused in an altered manner due to stroke. It uses dynamic MRI sequences to measure how much blood reaches the brain tissue and how fast it flows. This technique is useful for assessing the extent of altered brain perfusion in ischemic stroke. Like DWI, it is among the most widely used techniques when analyzing the pathophysiology of brain parenchyma during acute stroke [15].

c. Magnetic resonance angiography (MRA)

This technique uses MRI sequences to visualize blood vessels in the brain, either directly by adjusting the input parameters or by injecting a contrast agent such as gadolinium [16]. It is useful for detecting blood clots in brain vessels that can cause ischemic stroke.

d. T2-weighted MRI

This technique allows visualization of areas of brain tissue that have suffered loss of substance due to stroke. It is useful for detecting brain lesions that are no longer acute but are already healing.

1.2.4 Limitations of MRI for Stroke Diagnosis

Although MRI is a powerful imaging technique for diagnosing stroke, it also has some limitations such as

- Motion artifacts: MRI images can be affected by patient motion during the exam,

which can lead to artifacts that can make image interpretation difficult.

- MRI contraindications: Some people cannot have an MRI because of contraindications, such as the presence of pacemakers or certain types of metal prostheses.

- Spatial resolution limitations: Although MRI offers high spatial resolution, it may not be sufficient to visualize small brain lesions or to distinguish between different types of brain tissue.

- Time required for the examination: MRI generally requires more time than other imaging techniques to perform the examination, which may limit its availability for patients requiring emergency intervention.

- Cost and availability: MRI is an expensive imaging technique and is not always available in all health care centers.

1.2.5 Use of advanced image processing techniques to improve stroke diagnosis with MRI

The use of advanced image processing techniques can improve the accuracy and reliability of MRI stroke diagnosis. These techniques can extract more precise and detailed features from MRI images, which can help physicians diagnose stroke more quickly and accurately.

One of the most promising image processing techniques for improving stroke diagnosis is Deep Learning. Deep Learning is a form of machine learning that uses artificial neural networks to analyze MRI images and extract features relevant to stroke diagnosis. These features can be used to develop algorithms for automatic segmentation of MRI images, which can help detect and characterize brain lesions associated with stroke.

In addition, image processing techniques can also be used to improve the quality of MRI images by correcting motion artifacts and improving the spatial resolution of images.

The use of these techniques can help physicians diagnose strokes more quickly and accurately, which can improve clinical outcomes for stroke patients. However, these techniques require special expertise and resources, which may limit their use in some health care centers.

1.3 Stroke lesions

A stroke, or cerebral vascular accident, can cause significant damage to the brain. Stroke lesions refer to areas of brain tissue that have been damaged due to an interruption of blood flow to the brain. Stroke lesions can be caused by blood clots (ischemic stroke) or bleeding (hemorrhagic stroke) [3].

The location of brain lesions can affect the symptoms and consequences of stroke. Sophisticated imaging techniques such as MRI can be used to visualize stroke lesions and track their progression over time. Stroke lesions can have significant functional consequences, including speech, motor and cognitive impairments [3], which may require specialized rehabilitation and management.

1.3.1 Types of stroke lesions

There are two main types of stroke lesions: ischemic and hemorrhagic.

Ischemic stroke is the most common type of stroke and is caused by an interruption of blood flow to the brain due to a blood clot or narrowing of the cerebral arteries. This leads to a shortage of oxygen and nutrients, which can damage brain cells and result in cell death. Ischemic lesions are responsible for approximately 80% of stroke cases [3].

Hemorrhagic injuries, on the other hand, are caused by a rupture of a blood vessel in the brain, which can lead to a buildup of blood in the brain and compression of surrounding brain tissue. Hemorrhagic lesions account for approximately 20% of all strokes. Despite their rarity, 40% are fatal within the first month, 50% of which occur within the first 48 hours [3].

It is important to note that ischemic and hemorrhagic lesions may co-exist in some complex strokes, and may be associated with different clinical presentations and functional sequels. Stroke lesions may also vary in size, shape and location, which may have implications for clinical presentation and functional outcome.

1.3.2 Location of stroke lesions

The location of stroke lesions can have a significant impact on the patient's symptoms and function, as each area of the brain is responsible for a specific function [3]. Ischaemic

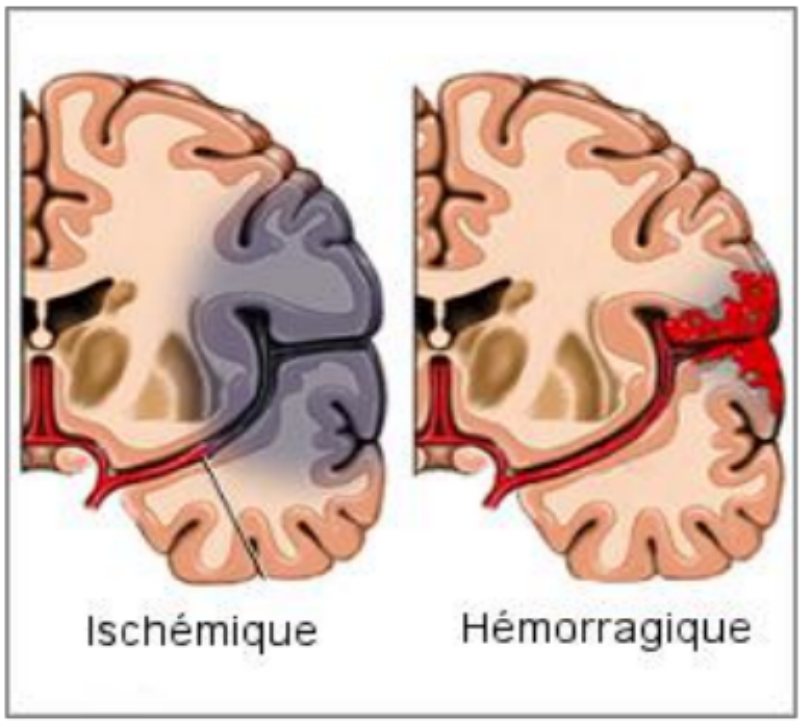


Figure 1.7: The two types of stroke [3].

and hemorrhagic lesions can occur in different areas of the brain, and their location can be identified using imaging techniques such as MRI.

Stroke lesions can be classified according to their location, for example as cortical lesions (affecting the outer layer of the brain) or sub-cortical lesions (affecting the sub-cortical regions of the brain). Lesions can also be classified according to their anatomical location, for example into lesions of the brain-stem, cerebellum, frontal, parietal, occipital or temporal lobes.

The location of stroke lesions can have important implications for symptoms and functional outcomes. For example, damage to the frontal cortex may affect motor skills and movement planning, while damage to the parietal cortex may affect sensation and spatial perception. Lesions of the brain-stem can be particularly severe and can affect vital functions such as breathing and blood circulation.

By understanding the location of stroke lesions, health professionals can better predict symptoms and functional consequences, and plan appropriate management to help patients recover their function.

1.3.3 Imaging of stroke lesions

Imaging plays a crucial role in the detection and characterisation of stroke lesions. The most common imaging techniques used to visualise stroke lesions are magnetic resonance imaging (MRI) and computed tomography (CT).

MRI images can provide detailed information about the structure of the brain, including ischaemic, hemorrhagic, or mixed lesions. MRI is also able to detect lesions very early, before they are detectable by other imaging techniques. Different types of MRI can be used to visualise different aspects of stroke lesions, such as brain perfusion or water diffusion in brain tissue [3].

Computed tomography (CT) is a less sensitive imaging technique than MRI for the detection of stroke lesions, but it can be faster and more readily available in emergency situations [3]. CT uses X-rays to produce cross-sectional images of the brain, which can help identify hemorrhagic lesions and areas of brain oedema.

Other imaging techniques, such as positron emission tomography (PET) and single photon emission computed tomography (SPECT), can also be used to visualise stroke lesions and assess brain function.

1.3.4 Evolution of stroke lesions

The course of stroke lesions depends on a number of factors, such as the size, location and type of the lesion, and the speed with which the patient is managed. The course of ischaemic lesions differs from that of hemorrhagic lesions.

Ischaemic lesions usually develop slowly, over a period of hours or days. The size and severity of the lesion may increase during this period, which may lead to worsening symptoms and complications. In some cases, the lesion may partially or completely resolve if blood flow is restored quickly, for example by thrombolysis or thrombectomy [3]. However, if the lesion is large or if treatment is delayed, permanent sequels may occur.

Hemorrhagic lesions progress more rapidly than ischaemic lesions, as they result in direct compression of brain tissue and increased intracranial pressure. The hemorrhage can also cause inflammation and an immune response, which can further damage brain

tissue. Prompt management is essential to minimise damage and prevent life-threatening complications.

1.3.5 Causes of stroke lesions

Stroke lesions are caused by an interruption of blood supply to the brain. This interruption can be due to two main types of stroke: Ischemic and hemorrhagic stroke [3] . Just as other factors can increase the risk of developing a stroke, including:

a. Age

Age is an important risk factor for stroke. The risk of stroke increases with age due to the deterioration of blood vessels over time. Arteries become stiffer and less elastic, which can lead to plaque buildup and narrowing of the blood vessels. In addition, the arteries can also become more fragile and more likely to rupture, leading to a brain hemorrhage.

The risk of stroke doubles for every decade of life after age 55. Strokes are therefore more common in older people. However, strokes can occur at any age, even in young people.

b. Gender

Although men younger than 75 years of age have a slightly higher risk of stroke than women, male sex is not considered a major risk factor. In contrast, women have a higher risk of subarachnoid hemorrhage. Women have a longer life expectancy than men, which increases their risk of stroke. In addition, the decline in estrogen at menopause is associated with elevated cholesterol and triglycerides, which are risk factors for stroke.

c. Family history

People with a family history of stroke have a higher risk of developing stroke.

d. Cardiovascular disease

Heart disease such as high blood pressure, cardiac arrhythmia, heart failure, and heart valve disease can increase the risk of stroke.

e. Diabetes

People with diabetes have a higher risk of stroke because of complications associated with the disease. One of the major complications of high blood sugar in diabetes is

protein glycation and macroangiopathy. This weakens the artery walls and promotes atherosclerotic plaque formation.

f. Lifestyle

Lifestyle risk factors such as smoking, poor diet, physical inactivity, and alcohol abuse can also increase the risk of stroke.

1.3.6 Functional consequences of stroke lesions

The functional consequences of stroke lesions can be highly variable depending on the location and extent of the lesion. Symptoms may be immediate or may appear gradually over time.

Stroke lesions can result in loss of motor, sensory and cognitive functions. Motor functions may be impaired by partial or total paralysis of one or more limbs, decreased muscle strength, loss of coordination or balance problems. Sensory functions, such as vision, hearing or touch, may also be affected. Cognitive functions, such as memory, attention, concentration and decision making, may also be impaired.

The functional consequences of stroke injuries can have a significant impact on the quality of life of the patient and family. They can result in loss of independence, decreased ability to work, to perform activities of daily living, and to participate in social activities.

Rehabilitation is essential to help patients recover their function and quality of life after a stroke. Rehabilitation techniques may include physical therapy, occupational therapy, cognitive rehabilitation and speech therapy. Health care professionals often work as a team to develop a personalized rehabilitation plan for each patient based on their individual needs.

1.4 Research problem and questions

Segmentation of MRI images of stroke is a crucial area of research in medical neuroimaging. Segmentation involves identifying and delineating anatomical structures of interest in the image. It is used to identify lesions in the brain.

For segmentation of stroke lesions, manual segmentation by an expert is considered the gold standard method. However, this method is time consuming and dependent on

the level of expertise of the operator. Therefore, with the evolution of technologies, it is necessary to develop automatic and accurate segmentation methods for MRI stroke images. Classical segmentation techniques such as thresholding segmentation, region segmentation, active contour segmentation, region growing segmentation have shown their limits in terms of accuracy in this domain. So, lately the research is turned to the use of Deep Learning.

Deep Learning is a machine learning technique that has revolutionized medical image segmentation. Deep neural networks have the ability to learn representative features from images by using deep network architectures to identify and extract features from the image. This technique has led to more accurate results in the segmentation of medical images, including MRI images of stroke.

However, there are still challenges and research questions in segmenting stroke MRI images with Deep Learning. For example, the identification of different lesion areas is complex due to inter- and intra-patient variability, as well as the presence of other brain pathologies that may disrupt the segmentation. Thus, research questions arise on how to improve the accuracy and robustness of segmentation algorithms for stroke.

Another important research question is the generalizability of these algorithms to other types of stroke and other brain pathologies. Indeed, most current segmentation algorithms have been developed for a specific type of stroke, which limits their use in other types of lesions. There is therefore a need to develop segmentation algorithms that can be used for different types of stroke and other brain pathologies.

Finally, the performance of Deep Learning segmentation algorithms is very sensitive to the training data. The quality and quantity of training data can have a significant impact on segmentation performance. Thus, the research question arises on how to obtain high quality training data in sufficient quantity to develop accurate and robust segmentation algorithms for stroke.

Segmentation of MRI images of stroke with Deep Learning is an evolving research question. The major challenge is to succeed in developing accurate and robust segmentation algorithms for stroke that can be generalized to different types of lesions and other brain pathologies.

1.5 Objectives and Impact of the research

The ultimate goal of research on MRI stroke image segmentation with Deep Learning is to improve the accuracy and speed of brain lesion segmentation, which is essential for early and accurate stroke diagnosis.

By using Deep Learning for MRI image segmentation, the researchers also aim to solve segmentation problems that persist in conventional image processing techniques. Conventional approaches are often prone to errors due to the complexity of MRI images, inter- and intra-observer variability, and the need for significant processing time. Deep Learning can overcome these problems through deep learning from large annotated MRI image datasets. The impact of this research is significant because it can contribute to better management of stroke patients, enabling faster and more accurate medical intervention. This can lead to a reduction in stroke-related mortality and disability, as well as improved quality of life for patients.

In addition, using Deep Learning to segment MRI images can help reduce the workload of medical professionals, who may be able to diagnose strokes faster and more efficiently. This can also reduce the costs associated with healthcare by reducing the length of hospitalizations and treatments.

Finally, research on MRI stroke image segmentation with Deep Learning can have a broader impact on the fields of medicine and medical imaging, providing more efficient approaches to image segmentation and clinical decision making based on image analysis.

1.6 Conclusion

This chapter gave a global view on stroke and MRI by starting with an introduction on magnetic resonance imaging, its functioning, its use for stroke diagnosis and its limitations. Then we discussed the lesions of stroke, the different types of stroke, their locations and their causes. Finally, we presented our problematic, some questions about the research in the field and the impact that our research will have. In the next chapter we will discuss artificial intelligence, especially its application in the medical field.

Chapter 2

Artificial Intelligence for Medical Image Analysis

2.1 Introduction

Medical imaging represents an imperative tool for the diagnosis and treatment of diverse pathologies, However the accurate interpretation of medical images necessitates a considerable degree of proficiency and experience, with such abilities being available in a select few people having progressed specialization within the field, The advent of artificial intelligence (AI) has resulted in a significant paradigm shift in the field of medical imaging by enabling automated and accurate analysis of medical images.

In recent times, artificial intelligence (AI) methods, specifically deep learning, have demonstrated promising outcomes in the field of medical image analysis. These techniques have facilitated a significant improvement in detecting and segmenting various abnormalities, including tumors, lesions, and other anomalies.

This chapter gives insights into the role of AI and deep learning in medical image analysis, particularly stroke lesion analysis from brain MRI images, we aim to provide a comprehensive understanding of the applications and limits of deep learning in this area.

2.2 Overview of artificial intelligence and machine learning

2.2.1 Definition of Artificial Intelligence and Machine Learning

Artificial Intelligence (AI) refers to the ability of machines to perform tasks that generally necessitate human intelligence, including visual perception, speech recognition, decision-making, and natural language processing. AI includes a broad range of techniques and approaches, including machine learning and deep learning.

Machine learning (ML) is a subset of AI that involves the development of algorithms that can learn from data and improve their performance over time. In ML, a computer system is trained on a large dataset of examples and then uses that training to make predictions or decisions on new data. ML techniques include supervised learning, unsupervised learning, and reinforcement learning.

Deep Learning (DL) is a branch of Machine Learning (ML) that employs artificial

neural networks (ANN) to model and address complex problems. Deep Learning algorithms are based on the way human brain function, they learn to recognize patterns in data through the processing of vast amounts of information across interconnected layers. Deep learning has demonstrated an outstanding level of success across a diverse range of applications, including image and speech recognition, natural language processing, and recommendation systems.

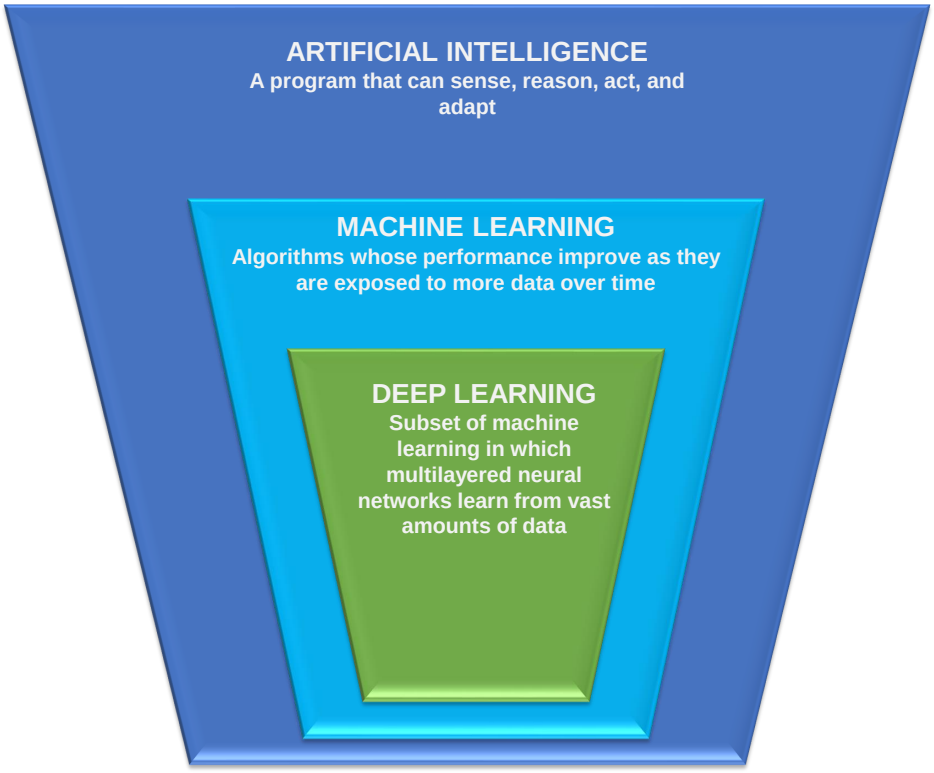


Figure 2.1: Artificial Intelligence-Machine learning-Deep Learning.

2.2.2 Brief history of AI and Machine Learning

The roots of AI can be traced back to the 1940s, particularly 1942, when the American Science Fiction author Isaac Asimov published his brief story *Runaround*. The plot of *Runaround*—a story about a robot created by the engineers Gregory Powell and Mike Donovan—evolves around the *Three Laws of Robotics:* (1) a robot may not harm a human being or, through inaction, permit a human being to come to harm; (2) a robot must comply to the orders given to it by humans except where such orders would conflict with the First Law; and (3) a robot must protect its own existence as long as such protection does not strife with the First or Second Laws. Asimov's work inspired generations of

scientists in the field of robotics, AI, and computer science—among others the American cognitive scientist Marvin Minsky (who afterward co-founded the MIT AI laboratory).

Alan Turing, an English mathematician developed a code breaking machine called *The Bombe* during World War II to decipher the Enigma code used by the German army. *The Bombe*, which was about 7 by 6 by 2 feet large and had a weight of about a ton, is generally Considered as the first working electro-mechanical computer, *The Bombe* broke the code in a way that was previously impossible for human mathematicians. This breakthrough made Turing wonder about the intelligence of such machines. leading him to publish his seminal article in 1950, titled *"Computing Machinery and Intelligence"*. In this article, he described how to create intelligent machines and in particular how to test their intelligence. This Turing Test is a benchmark to identify the intelligence of an artificial system, if a human is interacting with another human and a machine and unable to distinguish the machine from the human, then the machine is said to be intelligent.

Six years later, in 1956, the term Artificial Intelligence was officially coined by Marvin Minsky and John McCarthy (a computer scientist at Stanford), who organized the *Dartmouth Summer Research Project on Artificial Intelligence* (DSRPAI) at Dartmouth College in New Hampshire. This event, funded by the Rockefeller Foundation, brought together the founding fathers of AI, including the computer scientist Nathaniel Rochester, who later designed the IBM 701, the first commercial scientific computer, and mathematician Claude Shannon, who founded information theory. The purpose of DSRPAI was to establish a new field of research aimed at creating machines that could replicate human intelligence by bringing together researchers from different disciplines. [17]

2.3 Role of AI in medical image analysis

During the 1980s AI was applied to diagnostic imaging. Users first define explicit parameters and features of the imaging based on expert knowledge. For instance, the shapes, areas, histogram of image pixels of the regions-of-interest (i.e., Tumor regions) which can be extracted. A subset of the available data entries is allocated for training purposes, while the remainder is reserved for testing in order to effectively train the machine learning algorithm, the implementation of a specific algorithm is selected, such

as principal component analysis (PCA), support vector machines (SVM), convolutional neural networks (CNN), Upon completion of training, the algorithm becomes capable recognizing and classifying a given test image.

The primary focus of research in diagnostic imaging has been on detection, with computer-aided location (CAD) frameworks being created as early as the 1980s. Traditional machine learning algorithms were applied on image modalities like CT, MRI, and mammography. Despite considerable research efforts, the real clinical applications of CAD systems were not fruitful. Several large trials showed that CAD provided no noteworthy advantage and, in some cases, even reduced radiology accuracy, driving to higher rates recall and biopsy.

the new era of deep learning in AI has shown significant improvements in the research area. for example, a deep learning algorithm developed by Ardila et al used a patient's prior and current CT volumes to predict the risk of lung cancer. The model achieved a state-of-the-art performance 94.4% area under the curve (AUC) on 6716 national lung cancer screening trial cases and performed similarly on an independent clinical validation set of 1139 cases. In contrast, low-dose CT conventional screening involves several potential adverse outcomes according to cancer.org: false-positive exams, overdiagnosis, complications of diagnostic evaluation, increase in lung cancer mortality, and radiation exposure. One false-positive exam example provided on the web site was 60%. Overdiagnosis was estimated at 67%. There is also radiation induced risk to develop lung cancer or other types of cancer later in life. AI-based diagnosis methods demonstrate the potential to mitigate these risks.

Deep learning algorithms have become increasingly important in the field of radiology imaging analysis. These algorithms have demonstrated remarkable advancements in various image modalities such as CT, MRI, PET, and ultrasonography etc. and different tasks like tumor detection, segmentation, disease prediction etc. In comparison to conventional machine learning algorithms, deep learning has shown substantial performance improvements. deep learning learns from large sets of image examples and metadata. However, it might take much less time, as it does not depend on domain expertise, which usually takes years to develop. As the traditional AI requires predefined features and have shown plateauing performance over recent years. With the current success of AI and deep learning in image research, it is anticipated

that AI will continue to dominate the field of image research in radiology.[18]

2.4 Types of deep learning algorithms used in medical image analysis

Sequential modeling is a technique in machine learning and artificial intelligence that analyzes and understands data sequences. It captures temporal dependencies and patterns to make predictions or generate new sequences. Recurrent Neural Networks (RNNs) are commonly used for sequential modeling, as they retain information from previous steps and use it to influence the current step. RNNs pass hidden states or memory cells from one step to the next, allowing them to capture long-term dependencies. This enables the model to learn from the entire sequence and perform tasks like language modeling, speech recognition, and time series analysis. By employing sequential modeling, we can uncover insights, leverage patterns, and apply it to applications such as text generation, sentiment analysis, and speech synthesis. In addition to exploring the main types of deep learning algorithms used in medical image analysis, we will also cover in this chapter some of the most commonly employed deep learning algorithms.

2.4.1 Convolutional Neural Networks (CNNs)

Over the past few years, deep learning has gained widespread recognition and application in the field of medical image processing. Among the various deep learning methods used in this domain, Convolutional Neural Network (CNN) emerges as a prominent and widespread method. The CNN methodology has not only exceeded past methodologies but has also attained incomparable accuracy in tasks involving the classification and segmentation of images. Additionally, it plays a crucial role in transfer learning methodologies for image processing. it can be said that without the development and expansion of CNN, there will be no great prospect of transfer learning in the field of medical image processing. In fact, a significant amount of transfer learning approaches are based on CNN. Within the architecture of CNNs[19], several essential components can be identified:

a. Convolutional Layers

In Convolutional Neural Networks (CNNs), the convolution layer follows the input layer and plays a crucial role in extracting features from the input. The convolutional kernel, similar to a filter in signal processing, scans the image and detects specific features. Different convolutional layers extract various features from the input image. CNNs typically consist of multiple convolutional layers, with earlier layers extracting basic features and later layers extracting more advanced features. For example, when identifying a cat, the first layer may detect edges and lines, while the second layer may identify specific parts like the cat's eyes, nose, and ears. This layer-by-layer approach allows the CNN to learn and recognize the features associated with a cat. The standard structure of convolutional layers involves key parameters such as kernel size, input image size, zero-padding, stride, and output feature map size. The kernel, represented by a matrix, serves as the feature extractor. Padding is used to add extra pixels of zero value around the input image, enabling deeper convolutional neural networks. The convolution operation applies the kernel to the input image, typically with a stride of one. However, different strides can be used for different performance requirements[19].

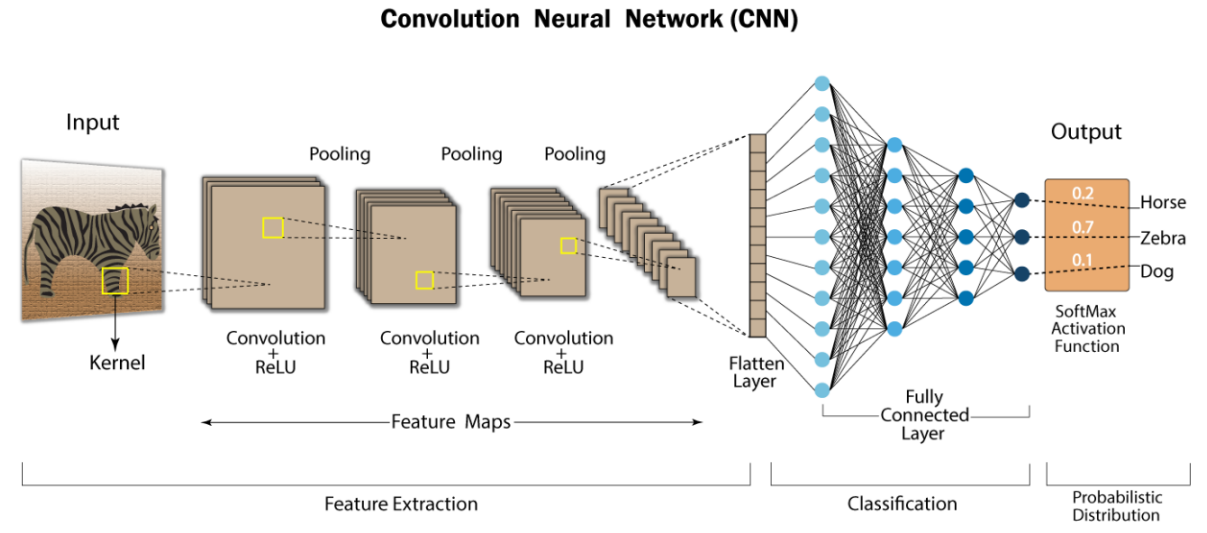


Figure 2.2: Basic structure of typical convolutional neural networks [4].

b. Pooling Layers

Pooling layers play a role in downsampling the feature maps generated by convolution operations. They take the larger-sized feature maps and shrink them to lower-sized feature maps. While shrinking the feature maps, they always preserve the most dominant features (or information) at each pooling step. The pooling operation is performed by

specifying the pooled region size and the stride of the operation, similar to the convolution operation. Different types of pooling techniques are used in various pooling layers, such as max pooling, min pooling, average pooling, gated pooling, tree pooling, etc. Among these, Max Pooling is the most commonly used technique[20].

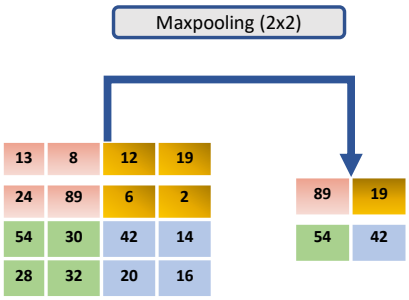


Figure 2.3: Max pooling illustration.

c. Activation Functions

The primary role of an activation function in any neural network-based model is to transform the input to the corresponding output. The input value is determined by computing the weighted sum of the neuron's inputs and adding a bias if present. Essentially, the activation function determines whether a neuron will activate or remain inactive based on the given input, producing the appropriate output. In CNN architecture, non-linear activation layers are employed after each learnable layer (such as convolutional and fully connected layers). These non-linear activation layers introduce non-linearity to the CNN model, allowing it to learn and map inputs to outputs in a non-linear fashion. An important characteristic of an activation function is its differentiability, which facilitates error backpropagation for model training. The following are the commonly used activation functions in deep neural networks, including CNN.

In neural networks,there are several commonly used activation functions. The sigmoid activation function maps the input to a range between 0 and 1, making it suitable for binary classification tasks. The hyperbolic tangent (tanh) function provides outputs in the range of -1 to 1 and is commonly used in classification problems and hidden layers. The rectified linear unit (ReLU) function, which sets negative values to zero, is widely used in deep learning due to its effectiveness in handling the vanishing

gradient problem. To address the drawbacks of ReLU, the leaky ReLU function introduces a small slope for negative values. Lastly, the softmax function is employed in multi-class classification to generate class probabilities that sum up to 1. These activation functions play a crucial role in introducing non-linearity and shaping the output of neural networks.[20]

d. Fully Connected Layers

Typically, in CNN architectures used for classification, the final part (or layers) primarily consists of fully-connected layers. These layers establish connections between each neuron in a layer with every neuron in the previous layer. The last layer of fully-connected layers functions as the output layer, serving as the classifier for the CNN architecture. The fully-connected layers belong to the category of feed-forward artificial neural networks (ANNs) and adhere to the principles of a traditional multi-layer perceptron neural network (MLP). These FC layers receive input from the last convolutional or pooling layer, which takes the form of a collection of metrics known as feature maps. To facilitate processing, these metrics are flattened, creating a vector that is subsequently passed into the FC layer. This process concludes in generating the final output of the CNN.[20]

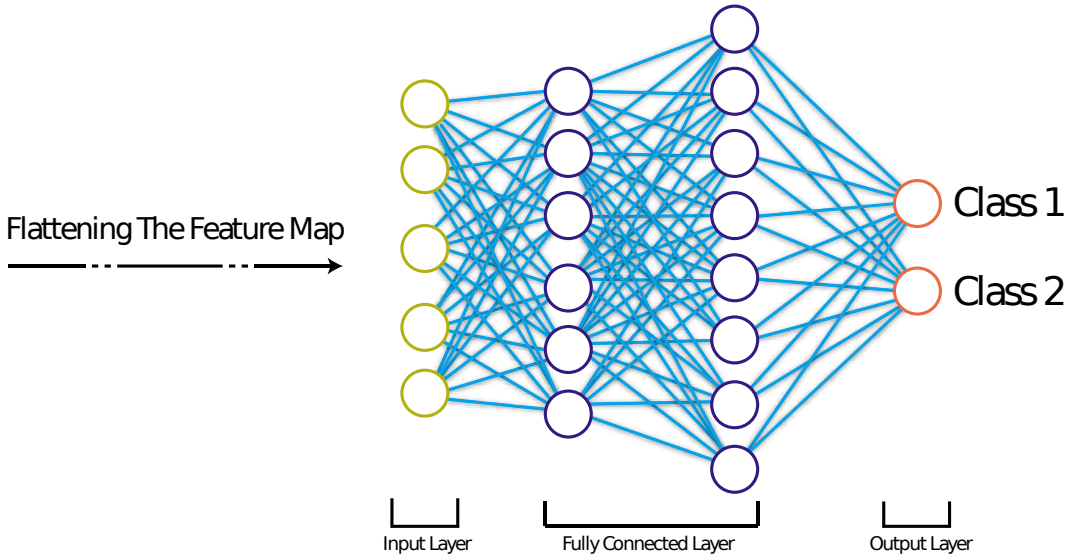


Figure 2.4: The architecture of Fully Connected Layers.

e. Optimizers

Optimizers are essential in Convolutional Neural Networks (CNNs) as they improve the model's performance by adjusting weights during training. They minimize the loss function by iteratively updating the weights based on gradients. Popular optimizers include Stochastic Gradient Descent (SGD), Adam, and RMSprop. These algorithms ensure the CNN learns relevant features and generalizes well to new data.[21]

2.4.2 Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM)

RNNs possess the capability to recognize sequences, with the weights of the neurons distributed across all dimensions. Variants like LSTM, BLSTM, MDLSTM, and HLSTM have demonstrated state-of-the-art accuracies in tasks such as character recognition, speech recognition, and various natural language processing problems. RNNs excel at capturing temporal dependencies, but they face challenges such as gradient vanishing and a reliance on large datasets [22].

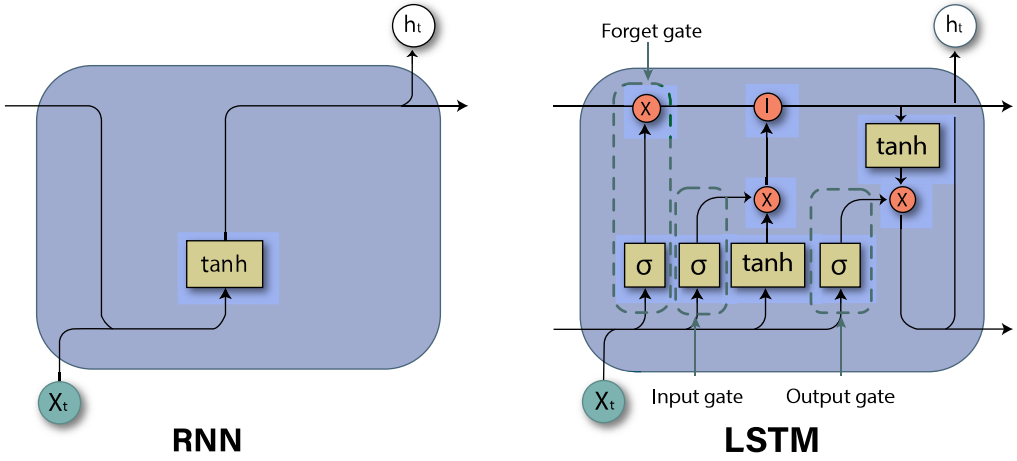


Figure 2.5: RNN and LSTM cells.

Key aspects of RNNs and LSTMs include:

a. Sequential Modeling

Sequential modeling is a technique in machine learning and artificial intelligence that analyzes and understands data sequences. It captures temporal dependencies and patterns to make predictions or generate new sequences. Recurrent Neural Networks (RNNs) are commonly used for sequential modeling, as they retain information from

previous steps and use it to influence the current step. RNNs pass hidden states or memory cells from one step to the next, allowing them to capture long-term dependencies. This enables the model to learn from the entire sequence and perform tasks like language modeling, speech recognition, and time series analysis. By employing sequential modeling, we can uncover insights, leverage patterns, and apply it to applications such as text generation, sentiment analysis, and speech synthesis.

b. Recurrent Layers

Recurrent layers in RNNs contain recurrent connections that enable information flow through time. These layers maintain a hidden state that carries information across sequential steps. LSTMs are a type of recurrent layer that use memory cells and gates to selectively remember or forget information over long sequences.

c. Training and Backpropagation Through Time

RNNs are trained using backpropagation through time, which is an extension of the standard backpropagation algorithm for feedforward neural networks. It involves unfolding the recurrent structure over time and applying the chain rule to calculate gradients and update weights.

d. Applications in Medical Image Analysis

RNNs and LSTMs have shown promising results in various medical image analysis tasks. They can be used for tasks such as time series classification, disease progression prediction, medical signal analysis, and sequential medical image analysis, where the temporal aspect of the data is crucial.

2.4.3 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) are a type of deep learning algorithm that consists of two neural networks: a generator and a discriminator. GANs are primarily used for generating new data samples that resemble the training data. Key aspects of GANs include:

a. Generator Network

The generator network in a GAN takes random noise as input and generates synthetic data samples. It learns to map the noise to the target data distribution by generating samples that are increasingly difficult for the discriminator to distinguish from real data.

b. Discriminator Network

The discriminator network in a GAN acts as a binary classifier, distinguishing between real and generated data samples. It learns to differentiate between real and fake samples by optimizing a binary classification objective. The discriminator's role is to provide feedback to the generator, encouraging it to generate more realistic samples.

c. Adversarial Training

GANs are trained in an adversarial manner, with the generator and discriminator networks competing against each other. The generator aims to generate realistic samples to fool the discriminator, while the discriminator aims to correctly classify real and fake samples. This adversarial training process encourages both networks to improve and reach a state where the generated samples are indistinguishable.

2.5 Applications of deep learning in stroke lesion analysis

2.5.1 Automated Lesion Segmentation

Automated lesion segmentation plays a crucial role in medical imaging analysis, primarily in identifying and delineating stroke lesions in MRI or CT scans. Deep learning algorithms, specifically convolutional neural networks (CNNs), have demonstrated significant advancements in this domain, offering accurate and efficient segmentation outcomes. Convolutional neural networks (CNNs) have been successfully applied to segment brain stroke lesions from MRI images. These deep learning models leverage their ability to learn complex spatial patterns and features within the images, enabling accurate identification and delineation of stroke lesions. By training CNNs on large datasets of labeled MRI images, the models learn to recognize and differentiate between healthy brain tissue and stroke-affected regions. This allows for precise localization and segmentation of stroke lesions, providing valuable information for diagnosis, treatment planning, and monitoring of stroke patients. The use of CNNs in brain stroke lesion segmentation has shown promising results in terms of accuracy and efficiency, facilitating improved clinical decision-making and patient care in the field of neurology.

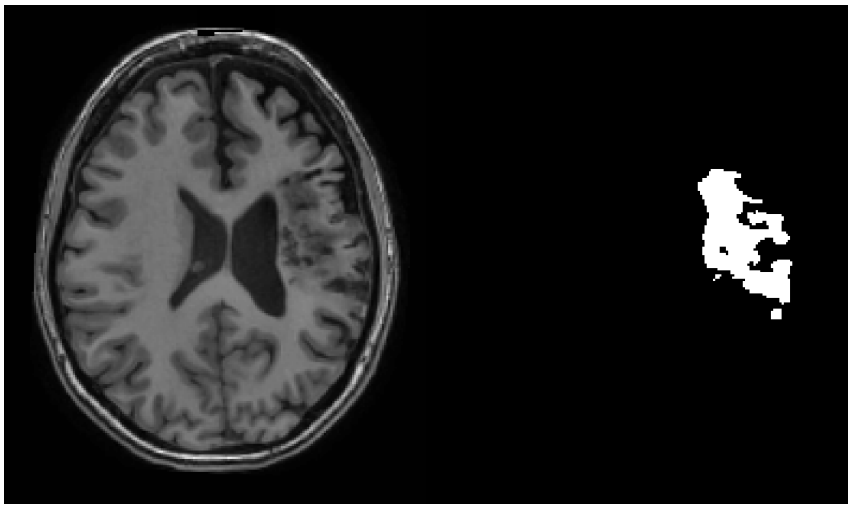


Figure 2.6: MRI Image and Mask.

2.5.2 Predicting Clinical Outcomes

The accurate prediction of patient outcomes is of utmost importance in healthcare, particularly in the context of stroke management. Deep learning models have shown great promise in utilizing stroke lesion characteristics to forecast patient prognosis, recovery, and treatment response. By leveraging these models, healthcare professionals can make informed decisions regarding treatment and devise optimal rehabilitation strategies for stroke patients.

Predicting clinical outcomes using deep learning extends beyond stroke management. It has applications in various medical domains, such as:

a. Cancer Prognosis

Cancer Prognosis: Deep learning models can analyze tumor characteristics and patient data to predict the likelihood of disease progression, recurrence, and survival rates.

b. Heart Disease Risk Assessment

By analyzing patient medical records, lifestyle factors, and genetic information, deep learning models can help assess the risk of cardiovascular events, such as heart attacks and strokes.

c. Mental Health Diagnosis

Deep learning techniques can aid in predicting mental health conditions, such as depression and schizophrenia, based on patient behavior patterns, speech analysis, and

neuroimaging data.

d. Diabetes Management

Deep learning algorithms can analyze patient data, including blood glucose levels, dietary patterns, and physical activity, to predict glycemic levels and aid in personalized diabetes management.

These examples highlight the vast scope of deep learning's potential applications in forecasting clinical outcomes throughout various medical disciplines. By harnessing the power of these models, healthcare professionals can enhance decision-making, elevate the quality of patient care, and optimize treatment strategies.

2.5.3 Quantitative Lesion Analysis

Quantitative lesion analysis using deep learning has revolutionized stroke assessment and research. deep learning models effectively extract significant features from stroke lesion images, thus enabling precise quantification of size, location, volume, and shape attributes. These measurements aid in clinical assessment and facilitate research studies by providing objective and standardized data. Deep learning-based quantitative lesion analysis enhances stroke severity assessment, treatment response evaluation, and prognosis. It also contributes to understanding stroke mechanisms, assessing therapeutic interventions, and developing predictive models. Overall, deep learning empowers quantitative lesion analysis, improving stroke care and advancing stroke research.

2.6 Advantages and limitations of deep learning for stroke lesion analysis

The utilization of deep learning techniques has emerged as a highly promising approach for the purpose of analyzing stroke lesions, presenting a host of advantages in comparison to conventional methods. Recent advancements in computational models have employed neural networks to extract significant features from imaging data, facilitating precise and effective analysis of stroke lesions. Nonetheless, similar to all methodologies, the deep learning approach has inherent restrictions. In the present

discourse, we shall examine the advantages and limitations pertaining to deep learning techniques in the context of stroke lesion analysis. By comprehending these factors, we can gain a more profound understanding of the promising advantages and hurdles that are related to the implementation of deep learning algorithms for the analysis of stroke lesions and the significant consequences it has on both clinical practice and research endeavors.

2.6.1 Advantages of Deep Learning

Deep learning offers several advantages for stroke lesion analysis. These include:

a. Handling complex and high-dimensional data

Deep learning models excel at analyzing intricate patterns and relationships within stroke lesion data, enabling more accurate analysis.

b. Learning hierarchical representations

Deep learning models automatically learn hierarchical features, capturing both low-level and high-level characteristics of stroke lesions for more comprehensive analysis.

c. Automation and efficiency

Once trained, deep learning models can automate lesion analysis tasks, saving time and reducing the risk of human error. They can efficiently analyze large datasets.

d. Generalizability

Deep learning models can generalize knowledge learned from one dataset to new and unseen data, making them adaptable to diverse stroke lesion datasets.

These advantages make deep learning a powerful tool for enhancing stroke lesion analysis, improving stroke management, and advancing research in this field.

2.6.2 Data Requirements and Limitations

Deep learning for stroke lesion analysis faces challenges in data requirements, including the need for large labeled datasets and potential biases in training data. Interpretability of deep learning models can be limited, making it difficult to explain their decisions. Strategies like data augmentation, transfer learning, and model interpretability techniques help mitigate these limitations.

2.6.3 Computational Requirements

Deep learning models for stroke lesion analysis require substantial computational resources and infrastructure. The computational complexity and memory requirements of deep learning algorithms can be demanding, necessitating powerful hardware and sufficient storage capacity. Potential limitations in terms of available computational resources and hardware capabilities should be considered when training and deploying deep learning models for stroke lesion analysis.

2.7 Conclusion

In this chapter, we discussed the role of AI and ML in medical image analysis, focusing on Deep Learning algorithms. Deep Learning has revolutionized medical image analysis by enabling automated and accurate interpretation of complex medical images. It offers advantages such as improved diagnosis and patient care.

However, Deep Learning has limitations, including the need for large datasets, potential biases, interpretability challenges, and computational requirements. Mitigating these limitations requires techniques like data augmentation, transfer learning, and interpretability methods.

In summary, AI and Deep Learning have transformed medical image analysis, but further research is needed to overcome existing challenges and fully utilize their potential in improving healthcare outcomes.

Chapter 3

State of the Art Methods

3.1 Introduction

The use of Deep Learning, a machine learning technique based on artificial neural networks inspired by the human brain, offers promising prospects for the analysis of MRI images of stroke. This technique allows to assist a radiologist in his task or to obtain the first diagnoses in case of absence of an experienced radiologist because strokes are cases that require a fast management.

In this state of the art, we explore the methods that are used for medical image segmentation. We first focus on traditional methods, then we give an overview on current state-of-the-art methods followed by related work. We conclude with our proposed method for stroke image segmentation.

3.2 Traditional Methods for medical images Segmentation

Prior to the real emergence and gain in popularity of Deep Learning around the 2010s, traditional methods were used by researchers for segmentation of medical images.

3.2.1 Thresholding segmentation

The thresholding segmentation method consists in applying a threshold to the image to separate the pixels of the area of interest from the rest of the image. Different thresholding techniques can be used, such as thresholding based on pixel intensity or thresholding based on textural features.

A number of thresholding algorithms have been proposed using global and local techniques. Global methods apply a threshold to the entire image, while local thresholding methods apply different threshold values to different regions of the image. The choice of threshold value is based on the neighborhood of the pixel to which the thresholding is applied [23]. Thresholds used in these algorithms can be selected automatically or manually. Manual selection of a suitable threshold value requires knowledge and sometimes experimental trials, while automatic selection combines image information to obtain the adaptive threshold value.

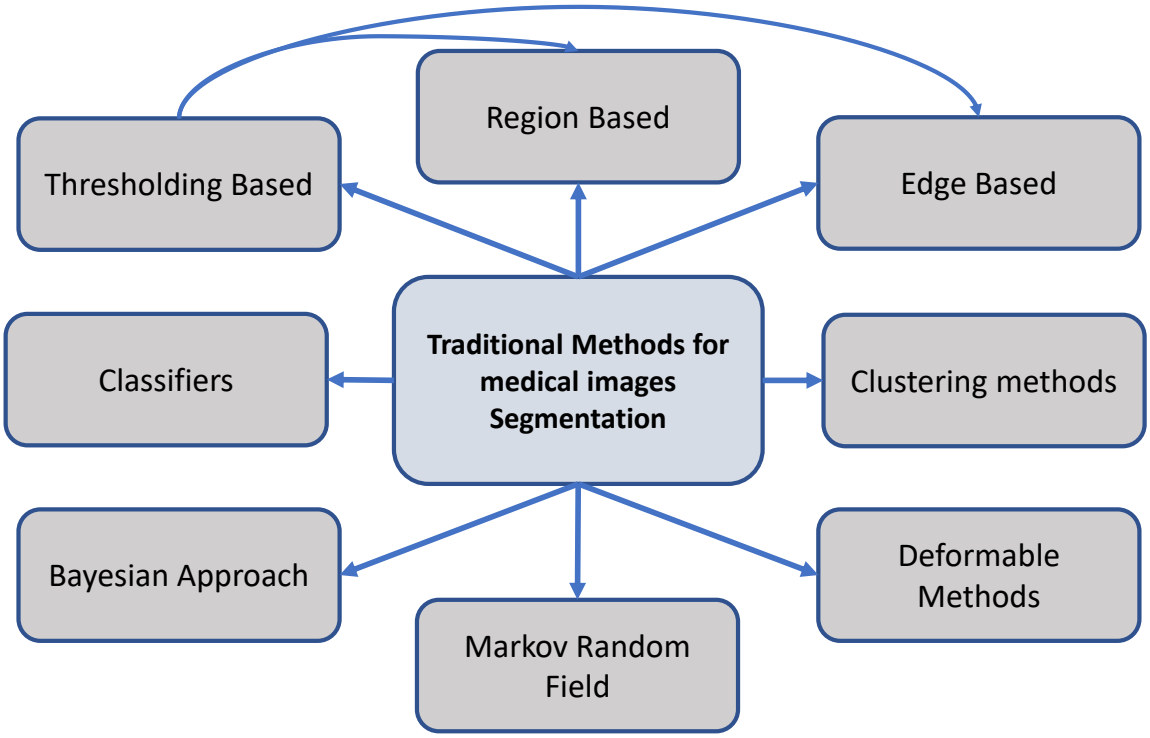


Figure 3.1: Traditional Methods for Stroke Lesion Segmentation.

The use of global thresholding is only effective if the image to be segmented contains a constant background and a single object. When dealing with images with a diversity of backgrounds and objects, local thresholding is a good choice [24], although it is time consuming.

The Otsu algorithm is a very common global thresholding technique and he obtains the threshold values using an image histogram. But there are also other popular thresholding techniques such as Kapur, Tsai, Kittler and Illingworth, Huang, Yen and others [23].

Depending on the information used to define the threshold values, thresholding-based algorithms can be classified into three categories: edge-based algorithms, region-based algorithms and hybrid algorithms.

3.2.2 Clustering/Unsupervised Methods

Clustering is a technique for grouping objects, either physical or abstract, to form classes. The objects in a class must be similar and dissimilar to another class to ensure efficient clustering. Indeed, the ultimate goal is to minimize inter-class similarity and

maximize inta-class similarity [25]. This technique is a type of unsupervised learning, as it is used to cluster data without the use of previously defined labels or target variables.

Algorithms and techniques have been proposed by researchers. K-means, Mean Shift, fuzzy-c means, Gaussian Mixture Models (GMM), Spectral Clustering, Affinity Propagation are some of the most used algorithms.

3.2.3 Deformable Methods

Compared to the other types we have seen before, these are more flexible and can be used for complex segmentations. The algorithms treat the structure boundary as the final state of the initial contours. The approach that these algorithms follow can be thought of as modeling the evolution of the curves. Depending on how they are used to track the moving contour, deformable models can be classified into two categories: parametric models and geometric models [24].

3.2.4 Bayesian approach

The Bayesian approach to medical image segmentation is a probabilistic approach that combines a priori information and image observations to estimate the posterior distribution of segmentation variables. This provides robust segmentation results and allows for the uncertainty associated with the segmentation task. However, it can be computationally intensive due to the need to compute conditional probability distributions and perform probabilistic inferences. Numerical optimization and sampling techniques, such as Monte Carlo methods, can be used to solve these problems.

The Bayesian approach uses directed graphs, which are data structures used to represent dependency relationships between variables, such as pixels or regions of interest.

In the Bayesian category of image segmentation, we can cite three main approaches. These approaches are illustrated in figure (3.2) below.

3.2.5 Markov Random Field

This method essentially uses an undirected graph that determines the Markov values of some arbitrary variables contained in a graphical model. MAP is quite similar to the

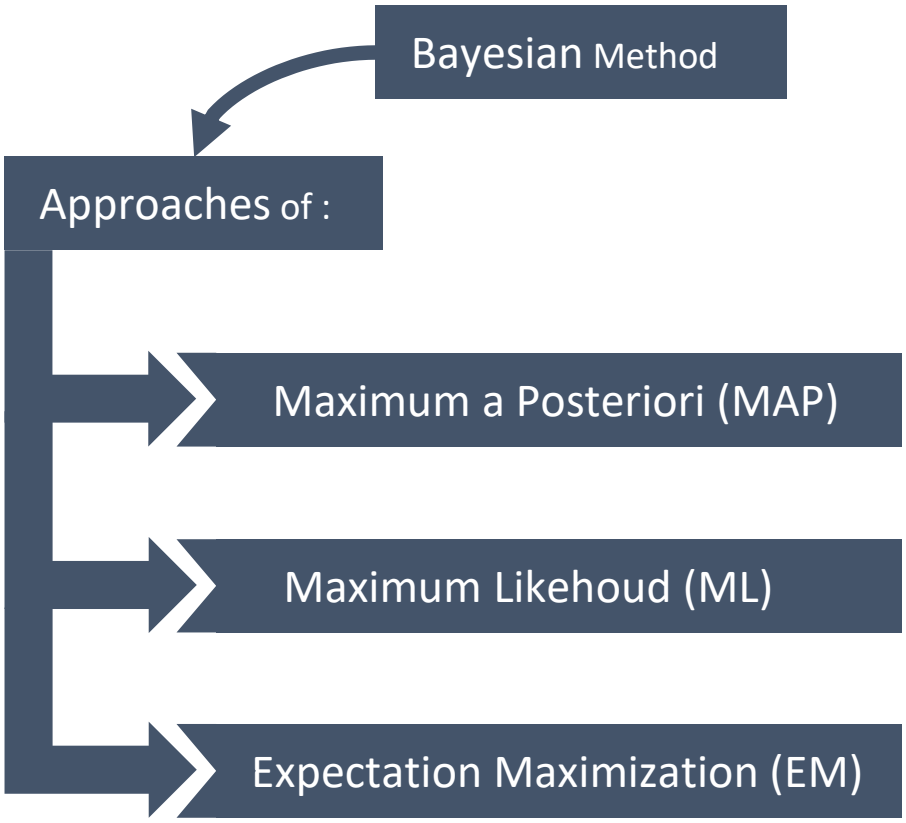


Figure 3.2: Bayesian Approaches.

Bayesian approach in terms of representation. The non-orientation of its graph is the only difference compared to the Bayesian method which is composed of directed graphs [26].

3.2.6 Classifiers

It is an approach to classify images by creating a feature space derived from the image. This space is then partitioned into different regions according to a function defined in this space. The feature space can be thought of as encompassing the entire range of a specific function used for classification purposes, for example, image intensity. Classification methods are based on pattern recognition. This approach is also referred to as a supervised method because it uses previously annotated and manually segmented data for the automatic segmentation process [26]. Figure (3.3) below shows the different methods that use this approach.

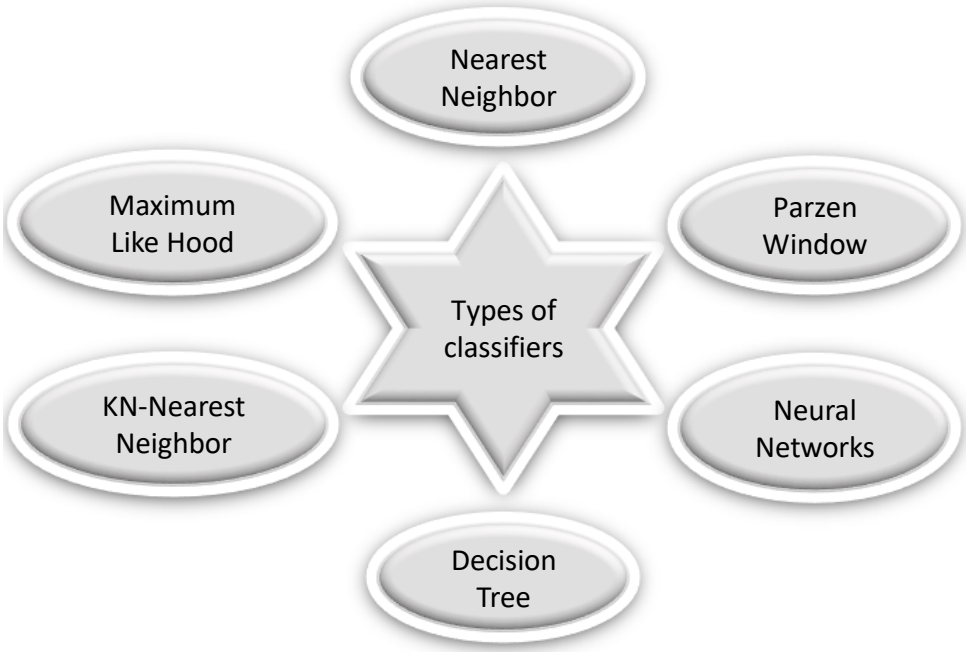


Figure 3.3: Types of classifiers.

3.3 Modern methods for medical image segmentation

Modern methods for medical image segmentation use advanced approaches and innovative techniques to extract and identify regions of interest in the images we wish to segment. Among these methods, we find the use of convolutional neural networks (CNNs), which are able to learn relevant features from image data. CNNs are often combined with more complex architectures such as residual neural networks (ResNet) or encoder-decoders such as U-net [27] which is widely used in the medical field to improve the accuracy of segmentation. Nevertheless, there are other solutions such as FCN (Fully Convolutional Network) [28], PSPNet or DeepLabV3+ which offer good results.

Another modern approach is the use of transfer learning based methods, where a pre-trained model on a large dataset is tailored to the specific segmentation tasks of medical images. This allows the knowledge and features learned from large general image datasets to be used to improve medical image segmentation with smaller, specialized datasets. This is a time-saving approach to training.

3.4 Related works

3.4.1 Related works based traditional methods

In 2016, X. Zhang et al. developed a multi-scale 3D Otsu segmentation algorithm based on dimension decomposition [29]. This algorithm uses the spatial information of the image and works by successive iterations. Each iteration uses the output of the previous iteration as input. The process starts by applying a Laplacian filter to obtain multiple scales of the image, then 3D Otsu thresholding is applied to generate segmentation maps. Finally, these maps are combined to obtain a final segmented image. The main advantages of this algorithm lie in its improvement of image segmentation, its reduction of noise when thresholding at two or more levels, and its reduction of the time complexity usually encountered in other thresholding algorithms.

Jiangdian Song et al in 2016 [30], proposed an approach for accurate segmentation of lung lesions from CT scans for lung cancer research and clinical care. The study presents a novel automatic segmentation approach called TBGA, which is based on a three-step framework. This method is notable for its ability to detect lesions with high sensitivity (96.35%) and to achieve segmentation accuracy comparable to manual segmentation. Evaluations conducted on the LIDC-IDRI and in-house clinical datasets confirm these promising results. Compared to existing methods, TBGA demonstrates a significant improvement in segmentation accuracy. In addition, this method allows for rapid lesion segmentation, with an average time of less than 8 seconds.

In their study, Kesavamurthy and SubhaRani [31] explored various image analysis technologies for processing MRI images of brain tissue. They developed a semi-automated method that accurately depicted damaged regions of brain tissue, thereby helping to improve clinical diagnosis and treatment. To do this, they used a Canny-based edge detection algorithm to extract damaged areas of brain tissue. This approach allowed them to accurately differentiate the boundaries of adjacent normal brain tissue as well as the skull.

In 1999 N.A. Mohamed et al [32] performed a study on the application of fuzzy set theory to medical imaging, more precisely to brain image segmentation. Their study proposes a fully automatic method based on a modified Fuzzy c-mean (FCM)

classification algorithm to obtain image clusters. They also incorporated Markov Random Field (MRF)-inspired techniques to filter the images and improve the filter parameters at each iteration of the clustering process. The authors applied their method to noisy CT images and single-channel MRIs. They recommended the use of a textured MRI-based over-segmentation methodology and a user-guided interface to obtain the final clusters. One of the applications of their method concerns the prediction of head injury recovery by taking into account the partial volume. They observed that the system stabilizes after a number of iterations, with the contours of the region reflecting the value of the partial volume.

Y. Kabir et al [33] (2007), conducted a study on the automatic segmentation of stroke lesions on multiple MR sequences. They exploited the fact that lesions manifest differently depending on the MR modality used. To utilize this feature of lesions on different MR modalities, the authors proposed a multimodal Markov random field model that simultaneously integrates all MR modalities. The results obtained by their multimodal method are compared with those of a one-dimensional segmentation applied to each MRI sequence individually. This technique showed promising results. Finally, the authors developed an atlas of blood supply territories to facilitate the identification of stroke subtypes and associated functional deficits.

3.4.2 Related works based modern methods

Joshi, S. et al [34] (2018), proposed an approach based on dilated convolutions to do the segmentation. They used only the DPWI MRI image as it generates the most data and is the most useful for assessing the lesion. Their CNN model is inspired by encoder-decoder architecture. Diluted convolutions are used, as they found that dilation reduces the parameters while increasing the field of view of the convolution kernels. The dice score and jacard score obtained are 0.85 and 0.78 respectively.

Liu, L. [35], proposed in 2019 in their study, a new deep convolutional neural network (Res-CNN) to automatically segment acute ischemic stroke lesions from multi-modality MRIs. their network is based on the U-shaped structure and uses a residual unit to mitigate the degradation problem. They use multimodality to exploit complementary MRI information. Seven neural networks were thoroughly evaluated on two acute ischemic stroke datasets. Res-CNN outperformed the other six networks,

both in single and multimodality. The mean Dice coefficient and Hausdorff distance of their method are 74.20% and 2.33 mm respectively.

Islam, M. [36], have developed a deep learning approach for the automated segmentation of intracerebral hemorrhage (ICH) from 3D CT scans. Using the ICHNet model, the authors developed an innovative approach by combining a dilated convolution neural network (CNN) with hypercolumn features. This model makes it possible to sample a limited number of pixels and concatenate the corresponding features from different layers. By focusing on the brain region and ignoring the background during training, the model improves temporal convergence and accuracy by focusing only on healthy and damaged brain tissue. To solve the problem of class imbalance, equal pixel sampling is performed for each class. In addition, the use of a 3D conditional random field (3D CRF) smoothes the predicted segmentation as a post-processing step. ICHNet demonstrated 87.6% accuracy in hemorrhage segmentation, comparable to that of radiologists.

3.5 Our proposal

Our final goal is to propose a model capable of segmenting MRI images of stroke lesions. We therefore propose several approaches for comparative study after results have been obtained.

3.5.1 Proposed CNN model

We propose a model called **Model HT**, which is based on the U-net architecture (refer to Figure 3.9). It consists of **80** layers and has a total of **31,058,113** parameters, out of which **13,184** parameters are not involved in the training process.

For this model, we changed input shape to 128x128x1 for datasets on ATLAS V2.0 and 256x256x1 for ISLES 2015. We used Upsampling2D and then ConvTranspose2D to see the influence of both on the model. The encoder (see 3.6) is activated by Relu. We used a binary output because this is a binary segmention, hence the use of Sigmoid as the output activation function.

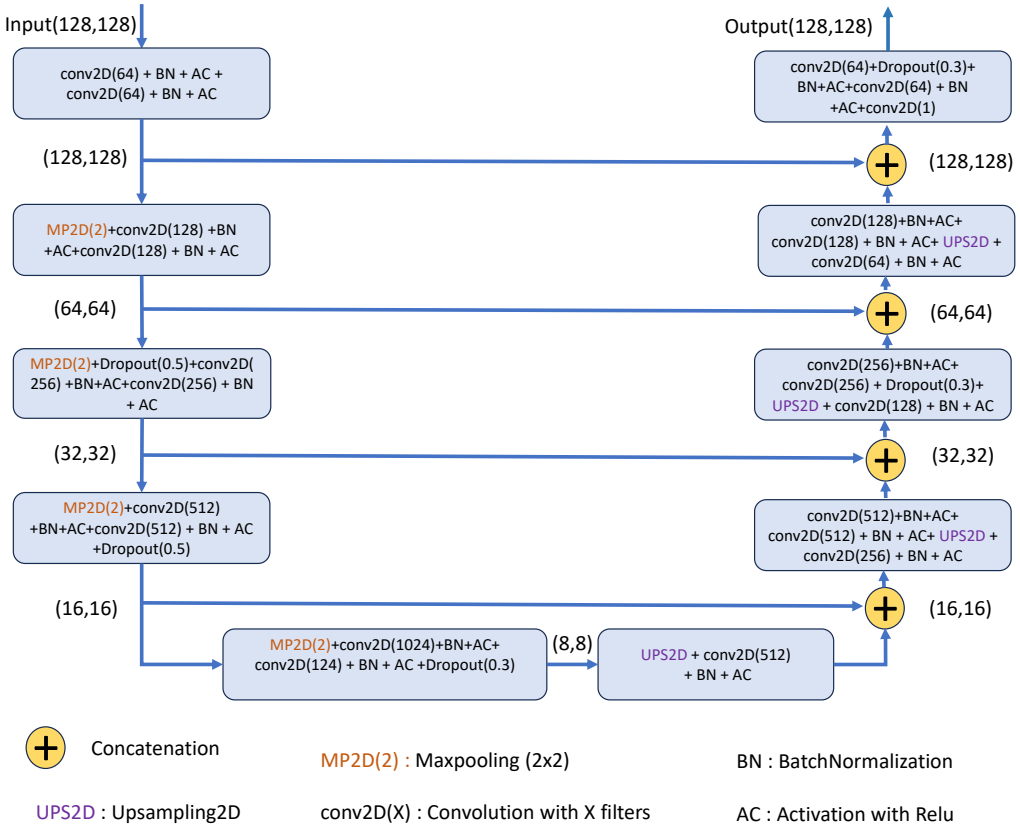


Figure 3.4: Proposed U-Net architecture

The difference between our architecture and the reference architecture is that we've modified the input size, added Batchnormalization and dropouts.

An overview of the program :

```
def unet_2D_Stoke(pretrained_weights = None, input_size = (128, 128, 1)):
    inputs = Input(input_size)

    #Contraction path
    conv1 = Conv2D(64, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(inputs)
    conv1 = BatchNormalization()(conv1)
    conv1 = Activation('relu')(conv1)
    conv1 = Conv2D(64, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv1)
    conv1 = BatchNormalization()(conv1)
    conv1 = Activation('relu')(conv1)
    pool1 = MaxPooling2D(pool_size=(2, 2))(conv1)

    conv2 = Conv2D(128, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(pool1)
    conv2 = BatchNormalization()(conv2)
    conv2 = Activation('relu')(conv2)
    conv2 = Conv2D(128, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv2)
    conv2 = BatchNormalization()(conv2)
    conv2 = Activation('relu')(conv2)
    pool2 = MaxPooling2D(pool_size=(2, 2))(conv2)
    drop2 = Dropout(0.3)(pool2)

    conv3 = Conv2D(256, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(drop2)
    conv3 = BatchNormalization()(conv3)
    conv3 = Activation('relu')(conv3)
    conv3 = Conv2D(256, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv3)
    conv3 = BatchNormalization()(conv3)
    conv3 = Activation('relu')(conv3)
    pool3 = MaxPooling2D(pool_size=(2, 2))(conv3)

    conv4 = Conv2D(512, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(pool3)
    conv4 = BatchNormalization()(conv4)
    conv4 = Activation('relu')(conv4)
    conv4 = Conv2D(512, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv4)
    conv4 = BatchNormalization()(conv4)
    conv4 = Activation('relu')(conv4)
    drop4 = Dropout(0.5)(conv4)
    pool4 = MaxPooling2D(pool_size=(2, 2))(drop4)
```

Figure 3.5: Encoder part

```
conv5 = Conv2D(1024, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(pool4)
conv5 = BatchNormalization()(conv5)
conv5 = Activation('relu')(conv5)
conv5 = Conv2D(1024, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv5)
conv5 = BatchNormalization()(conv5)
conv5 = Activation('relu')(conv5)
drop5 = Dropout(0.5)(conv5)
```

Figure 3.6: Bottleneck part

```
up6 = Conv2D(512, 2, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(UpSampling2D(size = (2,2))(drop5))
up6 = BatchNormalization()(up6)
up6 = Activation('relu')(up6)
merge6 = concatenate([drop4,up6], axis = 3)
conv6 = Conv2D(512, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(merge6)
conv6 = BatchNormalization()(conv6)
conv6 = Activation('relu')(conv6)
conv6 = Conv2D(512, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv6)
conv6 = BatchNormalization()(conv6)
conv6 = Activation('relu')(conv6)

up7 = Conv2D(256, 2, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(UpSampling2D(size = (2,2))(conv6))
up7 = BatchNormalization()(up7)
up7 = Activation('relu')(up7)
merge7 = concatenate([conv3,up7], axis = 3)
conv7 = Conv2D(256, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(merge7)
conv7 = BatchNormalization()(conv7)
conv7 = Activation('relu')(conv7)
conv7 = Conv2D(256, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv7)
drop7 = Dropout(0.3)(conv7)

up8 = Conv2D(128, 2, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(UpSampling2D(size = (2,2))(drop7))
up8 = BatchNormalization()(up8)
up8 = Activation('relu')(up8)
merge8 = concatenate([conv2,up8], axis = 3)
conv8 = Conv2D(128, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(merge8)
conv8 = BatchNormalization()(conv8)
conv8 = Activation('relu')(conv8)
conv8 = Conv2D(128, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv8)
conv8 = BatchNormalization()(conv8)
conv8 = Activation('relu')(conv8)

up9 = Conv2D(64, 2, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(UpSampling2D(size = (2,2))(conv8))
up9 = BatchNormalization()(up9)
up9 = Activation('relu')(up9)
merge9 = concatenate([conv1,up9], axis = 3)
conv9 = Conv2D(64, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(merge9)
drop9 = Dropout(0.3)(conv9)
conv9 = BatchNormalization()(drop9)
conv9 = Activation('relu')(conv9)
conv9 = Conv2D(64, 3, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(conv9)
conv9 = BatchNormalization()(conv9)
conv9 = Activation('relu')(conv9)

conv10 = Conv2D(1, (1, 1), activation='sigmoid')(conv9)
model = Model(inputs = [inputs], outputs = [conv10])
```

Figure 3.7: Decoder part

3.5.2 Transfer learning

This is a widespread approach that works well in many areas. So we propose a transfer based on : Resnet34, VGG19 and Efficientnetb3. To achieve this, we use the **Segmentation Models** library [5].

The "Segmentation Models" library is a popular library for deep learning applied to image segmentation. It provides ready-to-use implementations of various advanced segmentation models, such as U-Net, PSPNet, FPN, LinkNet and others. The library is based on TensorFlow and Keras, making it easy to use and integrate into deep learning projects. Of course, we'll be using U-Net (see Figure 3.7). The library uses existing architectures (backbones) such as Resnet34, inceptionv3 (25 backbones) available for each architecture (U-Net, PSPNet, FPN, LinkNet) for the encoder part (Downsample) and use the characteristics of the target architecture to develop the decoder part (Upsample).

Some of the main features of the Segmentation Models library :

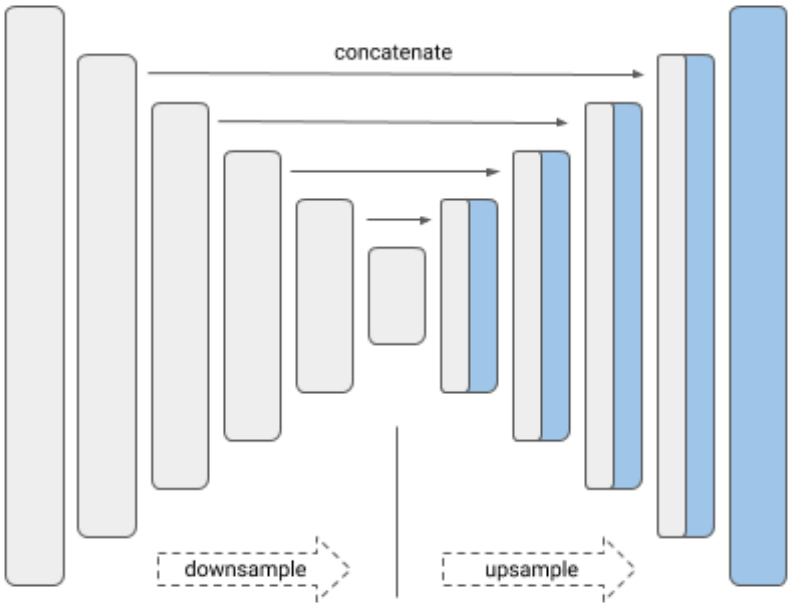


Figure 3.8: U-Net architecture proposed for segmentation models library [5].

a. Segmentation model implementations

The library provides complete implementations of several popular segmentation models, making them easy to use without having to code the model from scratch.

b. Pre-training on large datasets

Library models are often pre-trained on large datasets such as ImageNet, enabling good performance even with a small training dataset.

c. Various model architectures

The library offers a variety of segmentation model architectures, each with its own advantages and suitable for different types of segmentation problem.

d. Performance evaluation functions

The library also provides functions for evaluating the performance of segmentation models, such as the Jaccard index measure (IoU), the area under the ROC curve (AUC-ROC).

e. Ease of use

The Segmentation Models library is designed to be easy to use and to integrate into existing deep learning projects. It follows a clear code structure and provides detailed examples and tutorials to facilitate learning.

3.5.3 Ensemble Learning

Ensemble learning is an approach that aims to combine the predictions of several models to achieve better overall performance than a single model. The fundamental idea behind ensemble learning is that multiple models, if diverse and competent, can deliver more accurate and robust predictions.

3.6 U-Net Architecture

U-Net is an architecture dedicated to computer vision tasks, and more specifically to the problem of semantic segmentation. Semantic segmentation consists in assigning to each pixel of an image a label of a corresponding class. As each pixel must be predicted, this is called dense prediction. U-net is one of the most widely used neural network architectures for medical image segmentation. It is a fully convolutional neural network model that was developed by Olaf Ronneberger, Phillip Fissher and Thomas Brox in 2015.

The U-Net architecture is composed of two parts forming a U, hence its name. The first is the contraction part, commonly known as the encoder. It is used to extract image features. It is characterized by a succession of convolution layers followed by subsampling layers (max pooling) to progressively reduce the spatial resolution of the image while increasing the number of feature channels. This reduces the number of network parameters.

The second is the symmetrical expansion part, known as the decoder. It enables precise localization thanks to upsampling.

This symmetrical expansion can be achieved with :

Upsampling2D :

Upsampling2D is a simple upsampling operation that increases the spatial resolution

of features by repeating values in spatial dimensions. For example, if a feature has a size of 2x2, Upsampling2D will resize it to a feature of size 4x4 by repeating the values in each dimension. This method is fast and requires no additional parameter learning. Nevertheless, it can lead to a loss of fine detail, as it does not take into account the spatial relationships between features.

Conv2DTranspose:

Conv2DTranspose is an upsampling operation that uses learned filters to increase the spatial resolution of features. Unlike Upsampling2D, Conv2DTranspose learns the filters during model training. This operation performs a reverse convolution, where the filter weights are adjusted to reverse the convolution process. Conv2DTranspose captures spatial relationships between features and retains fine detail. However, it can be more computationally expensive and requires the learning of many parameters.

An important feature of U-Net is the use of the ReLU (Rectified Linear Unit) activation function between layers, which introduces non-linearity into the model. This enables it to learn more complex representations and better model spatial relationships.

Finally, the U-Net output layer generally uses a softmax activation function, which provides probabilities for each class of interest in the segmentation problem.

The advantages of U-NET

U-Net is an architecture that solves the problem of large data demand for classical networks, as it has proven effective with small datasets.

It preserves the initial output dimensions.

As deconvolution is performed on the decoder side, it avoids the bottleneck problem encountered with an auto-encoder architecture and thus avoids loss of features.

Disadvantages of U-NET

U-Net tends to be specialized and may have difficulty generalizing to data other than that on which it was trained.

The model can be sensitive to class imbalances and present difficulties in representing small regions of interest.

The input size of images is generally fixed, which can lead to loss of information or

degradation of image quality during resizing.

These drawbacks can be mitigated, but must be taken into account when using U-Net or any other model.

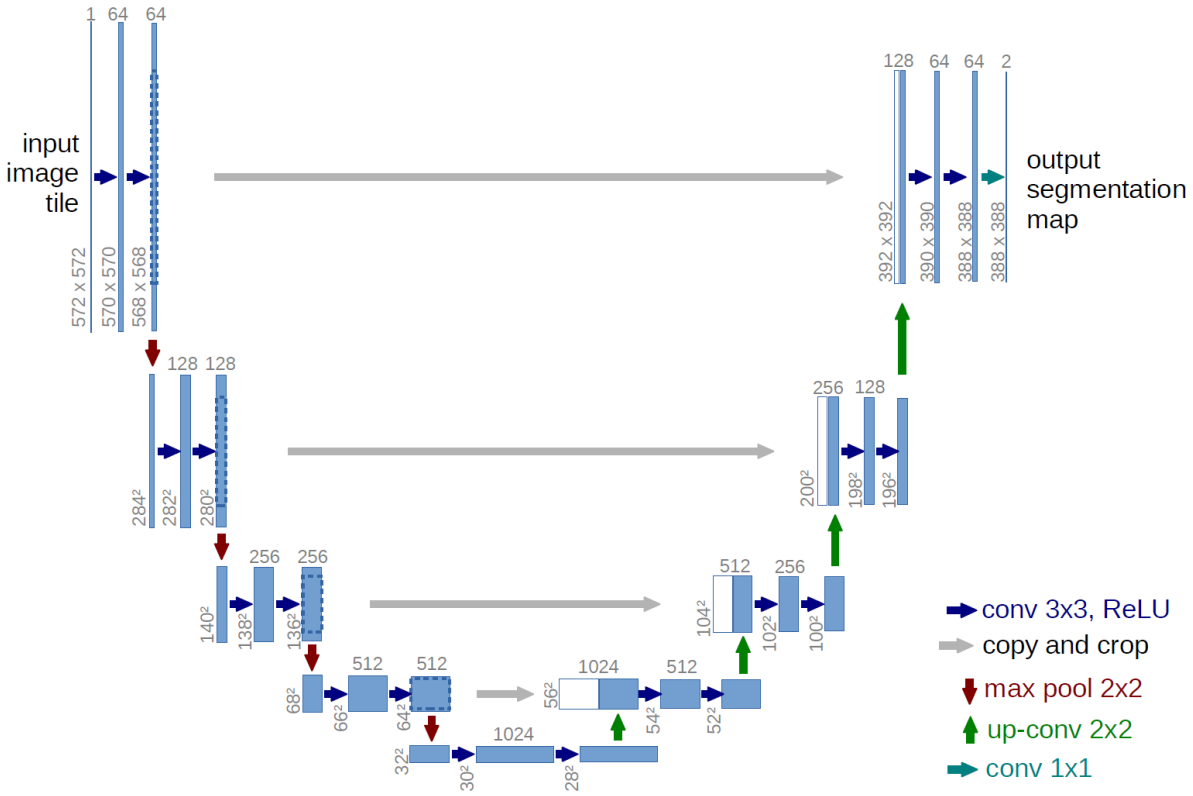


Figure 3.9: U-Net architecture [6].

3.7 Conclusion

In this chapter, we reviewed the traditional methods used for medical image segmentation, as well as related work in the field. We found that traditional approaches were often based on image processing algorithms and required significant manual intervention. Thanks to recent advances in machine learning, new approaches based on deep neural networks have emerged. In this context, we propose a number of approaches for more robust and accurate segmentation. Our methods are based on the use of advanced neural network architectures and ensemble learning techniques.

Chapter 4

Methodology: Implementation and Results

4.1 Introduction

In this chapter, we will describe in detail the implementation of our proposed methods and present the results obtained. We'll start by explaining our methodological approaches, detailing the algorithms, models and techniques we used. Next, we'll discuss data collection and preparation, explaining how we obtained the data and how we prepared it for training and evaluating our models. We will then present the experimental setup, detailing the specific parameters used for our models, as well as any other important considerations, such as the use of hardware gas pedals. We will explain how we trained our models and evaluated their performance using appropriate metrics. Finally, we will analyze the results obtained, identifying trends, patterns and important findings. We will discuss the strengths and weaknesses of our approaches, possible limitations and avenues for future improvement. We will also compare our results with other similar work in the literature to highlight the specific contributions of our approaches.

4.2 Tools used in our work

To realize our project we had to use open source tools. Here is a non-exhaustive list of the tools we used.

4.2.1 Kaggle platform

Kaggle is a popular online platform for data scientists, researchers and machine learning competition enthusiasts. It offers a dynamic community where users can collaborate, share knowledge and participate in data science competitions.

Kaggle has a vast collection of datasets from various fields, such as medical imaging, financial data, social networks and more. Users can explore these datasets, download them and use them for their own projects.

Kaggle offers an interactive notebook environment based on Jupyter, enabling users to write and run Python code (and other languages) directly on the platform. Notebooks are a convenient way of sharing analyses, visualizations and models with the community.

With Kaggle, we have access to a GPU that allows us to accelerate our calculation times. We have 30 hours a week. We've done all our work on this platform.

4.2.2 ITK-SNAP

ITK-SNAP is a free and open-source software used in medical imaging for segmentation, visualization, and analysis. It offers a user-friendly interface for researchers and clinicians working with MRI, CT, and PET scans. Its key features include image segmentation with manual and semi-automatic tools, 3D visualization with advanced viewing options, image registration for aligning datasets, quantitative analysis of segmented regions, and extensibility through an API for custom modules and scripts. ITK-SNAP is widely used in medical research and clinical settings for tasks like brain tumor segmentation and treatment planning. Its intuitive interface and powerful features make it a valuable tool in the field of medical imaging.

4.2.3 Programming language: Python

Python is a programming language widely used in the field of artificial intelligence (AI), thanks to its simplicity, flexibility and rich library of tools and frameworks.

4.2.4 Python libraries

1. **Numpy**

Numpy is the essential Python library for numerical computation and data analysis. It offers multidimensional array data structures, efficient mathematical functions, tools for array manipulation and integration with other scientific computing libraries. With Numpy, we can perform complex mathematical operations on arrays, such as vector and matrix operations, advanced indexing, data selection and filtering. Numpy is also renowned for its speed of execution, making it an essential tool for performance-intensive calculations.

2. OpenCV

OpenCV (Open Source Computer Vision Library) is an open source software library dedicated to computer vision and image processing. It offers advanced features such as object detection, motion tracking, facial recognition, image segmentation and much more. OpenCV is widely used in fields such as robotics, augmented reality, surveillance and industrial automation.

3. Matplotlib

Matplotlib is a widely used data visualization library in Python. It offers a simple interface for creating a wide variety of graphs, from line graphs to bar charts and box plots. It lets you customize every aspect of the graphs, including axes, legends, colors and styles. The library is highly flexible and can be used for simple data visualization as well as more advanced tasks, such as 3D graphics and animation.

4. Keras

Keras is a popular open-source library for deep learning that makes it easy to create, train and deploy machine learning models. With a simple, intuitive interface, Keras enables developers to quickly build neural networks and experiment with different architectures. It is based on TensorFlow, giving it excellent performance and flexibility. Thanks to its popularity, Keras has an active community that shares resources, tutorials and pre-trained models, facilitating the development and learning process for users.

5. Tensorflow

TensorFlow is an open-source machine learning library developed by Google in 2011. It provides tools and features for creating and training machine learning models, in particular deep neural networks. TensorFlow offers great flexibility and a wide range of

features for manipulating and transforming data, and for deploying models on different platforms. It is widely used in the field of artificial intelligence and is one of the most popular frameworks for machine learning.

5. Sklearn

Scikit-learn is a Python library widely used in the field of machine learning. It offers a wide range of machine learning algorithms and features for data preprocessing, cross-validation, evaluation metrics, clustering and dimensionality reduction. We have used its metrics to evaluate the performance of our models.

6. Anaconda

Anaconda is a popular distribution for Python programming and data science. It includes a package manager, a virtual environment and an integrated development interface (IDE) called Anaconda Navigator. Anaconda Navigator is not a full-fledged IDE, but rather a user-friendly graphical interface for managing packages, environments and projects in Anaconda. So we used the Spyder IDE to develop our interfaces in Python with Anaconda.

6. Spyder

Spyder is an IDE specially designed for data scientists. It offers a complete development interface with a code editor, variable explorer, debugger, IPython integration and data science-specific features.

6.PyQt5

PyQt5 is a powerful Python library for creating graphical user interfaces (GUI) for desktop applications. It is based on the Qt library, a popular framework for cross-platform user interface development.

4.3 Organization charts

An MRI image of the human brain can be obtained in several modalities (T1, T2, FLAIR, DWI, ...), each modality presenting different characteristics but the location of the lesion remaining the same. We will therefore apply monomodal and multimodal segmentation to our proposed approaches.

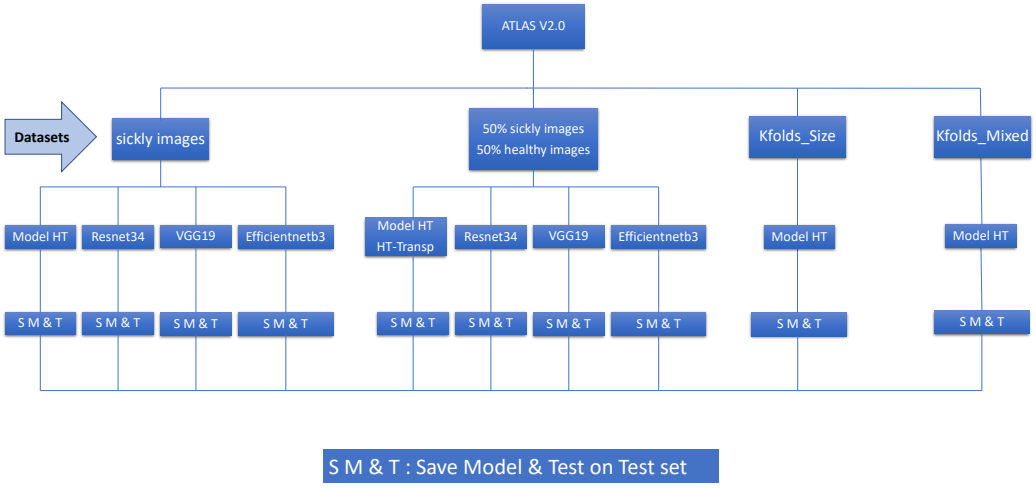


Figure 4.1: Flowchart when using the ATLAS V2.0 dataset.

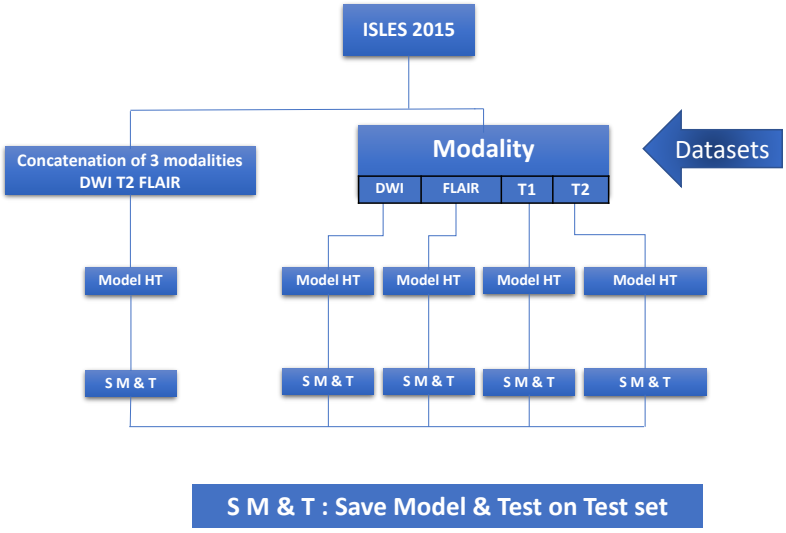


Figure 4.2: Flowchart when using the ISLES 2015 dataset.

4.4 Description of the datasets used for the study

In our work we used two large datasets, ATLAS V2.0 and ISLES 2015.

4.4.1 NIfTI format

The NIfTI (Neuroimaging Informatics Technology Initiative) format is a file format used to store and share medical imaging data, in particular brain imaging data. It is a standard format widely used in the field of neuroimaging.

The NIfTI format is based on the Analyze file format, but has been enhanced to be more flexible and efficient. It supports 3D volumetric data and can also store 4D temporal data.

The structure of the NIfTI format consists of two files: a header file (extension .nii or .hdr) and a data file (extension .img or .nii).

The header file contains important information about the imaging data, such as spatial dimensions, voxel spacing, orientation, data type, units of measurement, etc. It also defines the order of the dimensions, which can be changed by the user. It also defines the order of dimensions in the data (e.g. X, Y, Z for 3D volumetric data) and provides additional metadata.

The data file contains the intensity values of the voxels themselves, which represent the imaging information. Values can be stored in different formats, such as signed integer, unsigned integer, float, etc., depending on the data type specified in the header.

We'll be using this format in the course of our work, as our datasets are in this format.

4.4.2 ATLAS V2.0 dataset

ATLAS v2.0 is an MRI (Magnetic Resonance Imaging) data set used in stroke rehabilitation research. Accurate segmentation of brain lesions is of crucial importance in quantifying lesion burden and facilitating accurate image processing. ATLAS v2.0 is designed to overcome the limitations of previous methods. This dataset includes a larger number of 3D MRI scans of T1w strokes (1271 in total), as well as manually segmented lesion masks. It is divided into three distinct sets: the training set (655 samples), the test set (300 samples with hidden masks) and the generalization set (316 fully hidden samples). The aim of creating ATLAS v2.0 is to enable the development of more robust algorithms using this larger sample set. The hidden test and generalization datasets facilitate unbiased evaluation of algorithm performance by organizing segmentation challenges.

As shown in figure 4.1, we designed two new datasets. To contain the material, we worked in 2D. As the dataset is in 3D, we extracted the z-axis slices (axial slices), which are rich in lesion information, to obtain our 2D images. The shape of the 3D images is 197x233x189, so we have 190 slices for each image.

1. Construction of a dataset containing only lesioned images (sickly images)

A model's effectiveness in learning and predicting depends on the quality, quantity and diversity of the data on which it has been trained. In order to keep only those slices that are useful for our future model, we extracted from each 3D image of the **655**, the slices containing at least one lesion pixel. This yielded **32958** images and their corresponding mask.

We split our initial dataset of **32958** images into train, validation and test sets. We used a split of **80**%, **15**% and **5**% respectively. The training set was used to fit the model. The validation set, was used to evaluate model performance during training. The test set, was used to evaluate the final performance of the model in an unbiased way.

2. Construction of a dataset containing 50% sickly images 50% healthy images

In order to allow our models to generalize, we have included non-injured images in

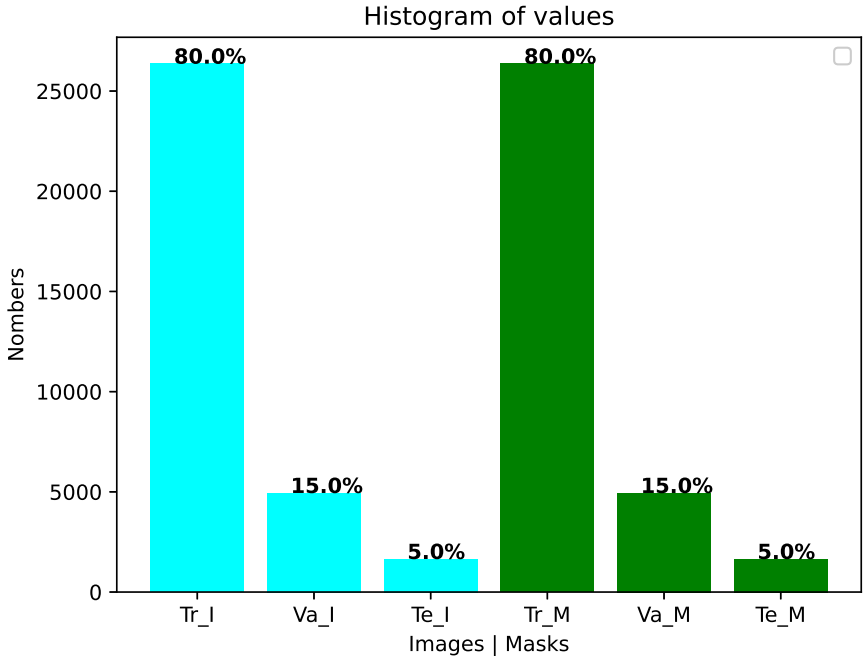


Figure 4.3: Histogram of dataset split into Train, Validation and Test set.

our dataset. To avoid overloading the dataset with uninformative images, we relied on the probabilistic lesion localization provided by the researchers who designed the ATLAS V2.0 dataset (see figure 4.4). We took the uninjured images between slice **35** and **140** (see figure 4.5). This enabled us to eliminate images from low-probability areas as well as completely black images. We obtained a dataset containing **65978** images, including the **32958** injured images, i.e. **33020** uninjured images.

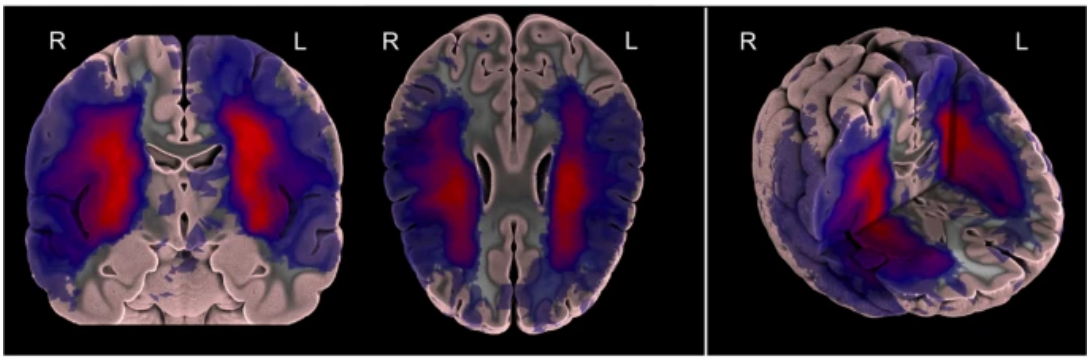


Figure 4.4: Probabilistic distribution of lesions in the ATLAS R1 dataset [7].

We split our initial dataset of **65978** images into train, validation and test sets. We used a distribution of **90**%, **5**% and **5**% respectively.

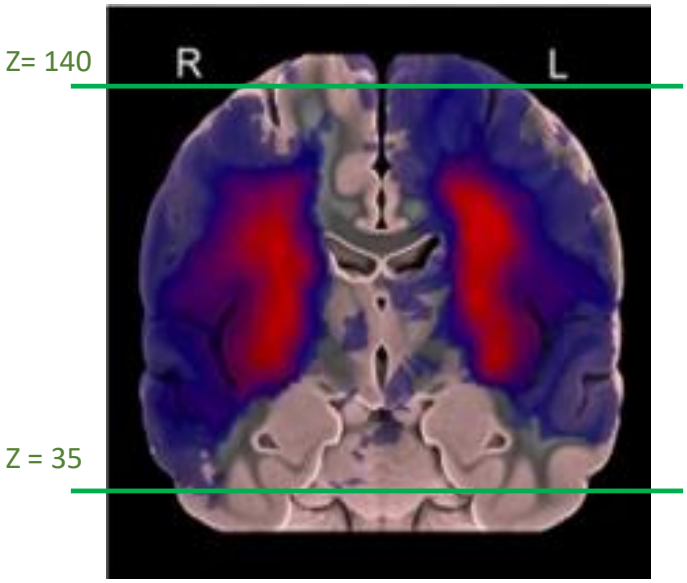


Figure 4.5: Region of interest for dataset.

4.4.3 ISLES 2015 dataset

The ISLES 2015 dataset was part of a medical image segmentation challenge focused on comparing methods for segmenting ischemic stroke lesions from multi-spectral MRI images. This challenge took place at the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI) in 2015, from October 5th to 9th.

For our analysis, we utilized the SISS dataset provided in the ISLES challenge, which consisted of 64 MRI images. Among these images, 28 were allocated for training, while the remaining 36 were designated for testing. Each case within the dataset included T1-weighted (T1), T2-weighted (T2), Diffusion-Weighted Imaging (DWI) with b = 1000, and FLAIR MRI sequences.

The segmentation process was conducted manually by an experienced medical doctor. Lesions were classified as sub-acute infarcts when there was a simultaneous presence of pathologic signals in both FLAIR and DWI images. This presence indicated the existence of vasogenic and cytotoxic edema, accompanied by visible swelling caused by increased water content. Infarct lesions with signal changes due to hemorrhagic transformation

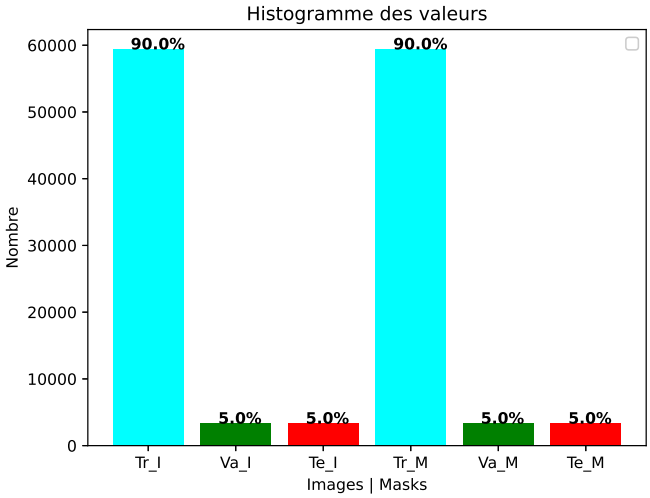


Figure 4.6: Histogram of dataset split into Train, Validation and Test set.

were also taken into consideration. However, acute infarct lesions (with only DWI signal for cytotoxic edema and no FLAIR signal for vasogenic edema) and residual infarct lesions showing gliosis and scarring after the infarction (without DWI signal for cytotoxic edema and no evidence of swelling) were excluded from the analysis. [37].

To accommodate the hardware limitations we had to deal with, we made the decision to primarily work with 2D images. Consequently, we transformed the multi-spectral MRI images, originally sized at 230x230x154, into 2D images. This conversion resulted in two new datasets. The first dataset consists of 2D images with a resolution of 230x230x1, while the second dataset contains 2D images with a resolution of 230x230x3.

Firstly, we sliced the images based on the presence of non-black pixels in the T1-weighted image along the z-axis. From there, we extracted the corresponding slices from the T2, DWI, FLAIR modalities, and their respective masks. This process resulted in **3946** images for each of the four modalities and the associated masks,Subsequently, we partitioned our dataset into training, validation, and testing sets, with proportions of **80**%, **15**%, and **5**% respectively.

Secondly, we constructed a second dataset by concatenating the T2, DWI, and FLAIR modalities together, this process resulted in **3946** images with three color channels similar to rgb but with first channel T2, DWI as second channel and Flair in third channel.

4.5 Preprocessing of the MRI images

Image pre-processing for training stroke lesion segmentation models is a crucial step in guaranteeing accurate and reliable results. Our datasets have already undergone a number of pre-processing steps, including registration, intensity normalization, spatial reorientation and resampling.

These pre-processing steps can vary according to the specific characteristics of the dataset and the requirements of the segmentation model used. Their main aim is to improve image quality, reduce artifacts and optimize the results of stroke lesion segmentation. We made a few additional steps:

- Image resizing to 128x128xC instead of 197x233x1. C is chosen according to the input characteristics of the model to be trained.

- 90° counter-clockwise rotation to respect the desired orientation.

- Normalization to have pixels between 0 and 1, in float32 format.

4.5.1 Data Augmentation

Data augmentation is a technique commonly utilized in machine learning and computer vision tasks to increase the size and diversity of a training dataset. It involves applying a variety of transformations and modifications to the existing data, creating new images that are slightly different from the original ones. The purpose of data augmentation is to enhance the model's ability to generalize and improve its performance when presented with unseen or slightly altered data during the testing or deployment phase.

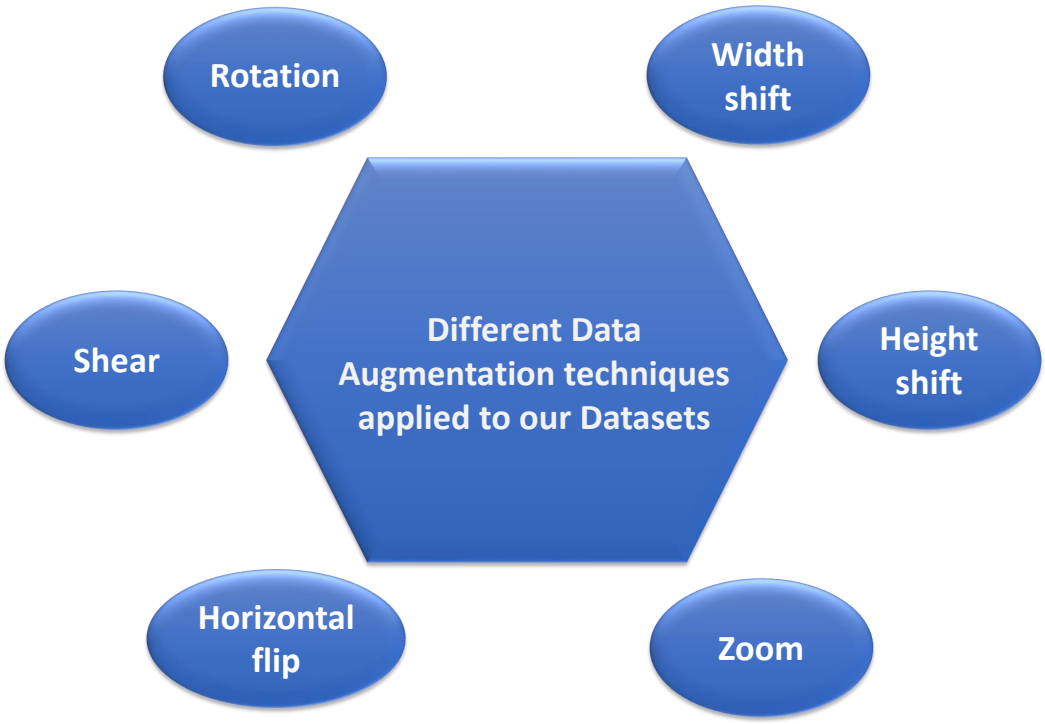


Figure 4.7: Diagram showing different data augmentation techniques used.

We have used several parameters that define the types and degrees of augmentation applied to our datasets. We will explore each of these parameters to gain a deeper understanding of the augmentation techniques employed:

a. Rotation

This parameter specifies the maximum angle in degrees for random rotations applied to the images. It allows the model to generalize better by recognizing objects from different angles.

b. Width and Height

These parameters control the range, as a fraction of the total width or height, within which random horizontal or vertical shifts can be applied to the images. Shifting the images helps the model learn invariance to translation and improves its ability to detect objects at different positions.

c. Shear

Shear transformations introduce geometric deformations by slanting the images along the horizontal or vertical axis.

d. Zoom

This parameter determines the range for random zooming, allowing the model to learn to recognize objects at different scales.

e. Horizontal flip

this parameter enables random horizontal flipping of the images. This augmentation simulates the presence of objects in mirrored orientations, enhancing the model's ability to handle horizontal reflections.

By artificially expanding the dataset and introducing variations, data augmentation helps prevent overfitting and enables the model to learn more robust and representative features.

4.6 Implementation details of the deep learning model used for stroke lesion segmentation

We're going to explain the details of our implementations in relation to the datasets used. First, ATLAS and then ISLES2015.

4.6.1 Models trained on ATLAS V2.0 dataset

1. Models trained on dataset containing only sickly images

a. Our Model CNN / Model HT

This model is based on the U-Net architecture and has a total of 80 layers. It has 31,054,145 parameters, with 11,776 untrainable parameters due to the use of BatchNormalization. We used Adam as an optimizer with a learning rate lr = 0.001.

We combined Binary Crossentropy Loss and Jaccard Loss (Loss = (0.4 * Bce) + (0.6 * Jaccard loss)) for the loss function. This is a customized loss function. We've named this model "**Model HT-OSI**" for future reference.

BatchNormalization is a technique for normalizing activations in neural networks, normalizing mini-training batches to accelerate convergence and improve learning stability.

b. Transfer Learning using Resnet34, VGG19 and Efficientnetb3 as backbones

With the Segmention-models library, we have implemented 3 transfer models still based on the U-Net architecture and using other architectures (Resnet34, VGG19 and Efficientnetb3) for the encoding and decoding part of the network. These models have the following characteristics:

Name	Numbers of parameters	Numbers of non-trainable parameters	BatchNorm-alization	Optimizer	Loss fonction
Resnet34-OSI	24,456,154	17,350	Yes	Adam	dice_loss + (1 * binary_loss)
VGG19-OSI	29,061,969	4,032	Yes	Adam	dice_loss + (1 * binary_loss)
Efficientnetb3-OSI	17,867,113	89,280	Yes	Adam	dice_loss + (1 * binary_loss)

OSI : Only Sickly Images dataset

Table 4.1: Characteristics of the three models.

2. Models trained on dataset containing 50% sickly images and 50% healthy images

Here we've used models with the same architectures and Characteristics, except that we've used a new dataset. So in the end we got 4 new models named: **Model HT-50, Resnet34-50, VGG19-50 and Efficientnetb3-50**. 50 to refer to the dataset used.

3. Models trained on Kfold dataset

The K-Fold Cross-Validation technique is a commonly used method for evaluating the performance of segmentation models. This technique divides the dataset into K subsets (or "folds") of similar size. To create these K subsets, we first used a distribution based on lesion size and then a mixed distribution. In our case K = 5 and once we've obtained the 5 folds we use the current fold for validation and the other 4 for training. At the end,

we calculate the average of our metrics on our test dataset, which remains unchanged.

We have obtained two models that we have named : **Model HT-Kfolds-Size** and **Model HT-Kfolds-Mixed**

4.6.2 Models trained on ISLES 2015 dataset

The ISLES 2015 dataset is a multi-modality dataset, which is an asset for experimenting with different approaches. On this dataset we used only our HT Model to train it on several datasets in order to obtain a better model.

1. Model trained on Concatenation of 3 modalities DWI T2 FLAIR dataset

Based on the differences between the modalities, we extracted the DWI, Flair and T2 modalities and concatenated them to obtain input images for the three-channel model (128x128x3). After training, we obtain a model that we call "**Model HT-DFT2**" for future use.

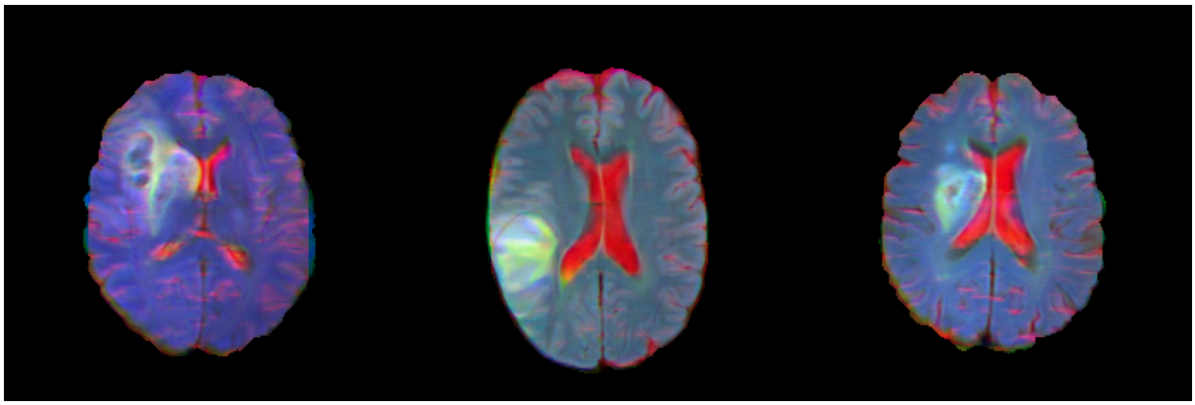


Figure 4.8: Concatenation of DWI T2 FLAIR image modalities.

2. Models trained on modalities DWI FLAIR T1 T2 dataset

As each modality has different characteristics, we trained our HT model on the four modalities separately. This gave us four output models. We named them: **Model HT-DWI, Model HT-FLAIR, Model HT-T1, Model HT-T2** to identify them.

4.6.3 Using of ensemble Learning

To improve performance, we thought about using the ensemble learning . To achieve this, we opted for majority voting. We chose four models (Model HT-DWI, Model HT-FLAIR, Model HT-T1, Model HT-T2) based on the modalities (DWI, Flair, T1, T2) having the same output mask during training, we made a prediction of the image on the 4 models followed by an aggregation on the mask produced (not in terms of probability).

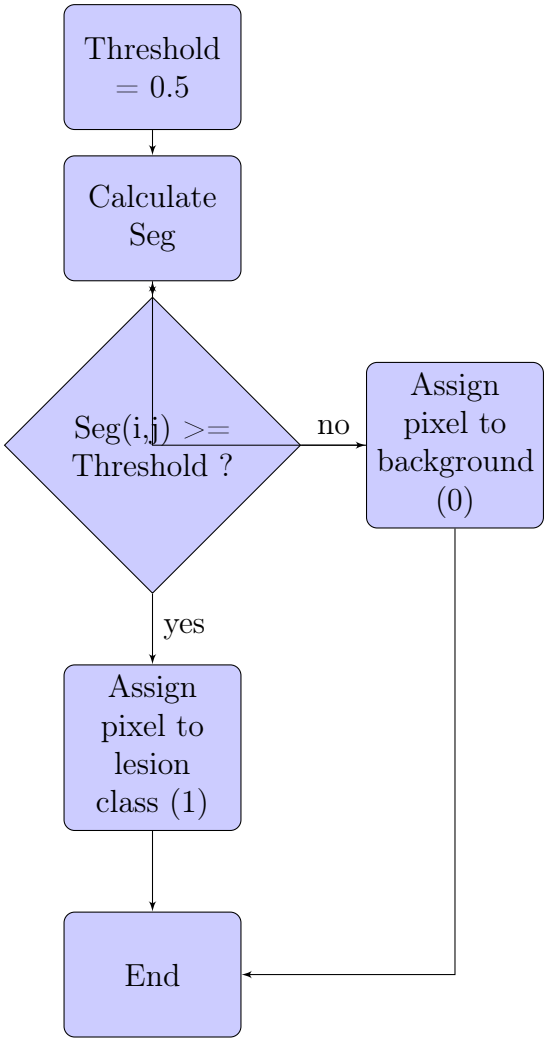


Figure 4.9: Organigramme algorithmique

For the threshold of 0.5, Seg (i,j) >= Threshold means that at least two models predicted that the pixel is part of brain lesion.

4.7 Performance evaluation metrics used for the models

There are several metrics used to evaluate models in different domains and tasks. In our work, we used the following performance evaluation measures:

4.7.1 Intersection over Union

Intersection over Union (IoU), also known as Jaccard Index, is a measure used to assess the quality of a segmentation or mask by comparing the similarity between prediction and ground truth. The IoU formula is given by :

$$IoU = \frac{\text{Zone of intersection}}{\text{Zone of union}}$$

where the zone of intersection represents the common area between the prediction and the ground truth, and the zone of union represents the total area covered by the prediction and the ground truth.

The Intersection over Union (IoU) formula can also be expressed in terms of true positives (TP), false positives (FP) and false negatives (FN):

$$IoU = \frac{TP}{TP + FP + FN}$$

Figure 4.10: Visual illustration of TP, FP and TN [8].

4.7.2 The Dice coefficient

The Dice coefficient, also known as the Sørensen-Dice similarity coefficient, is a measure commonly used to evaluate the similarity between two sets (masks) during segmentation. The formula for the Dice coefficient is as follows:

$$Dice = \frac{2 \times \text{Intersection}}{\text{Total pixels in first set} + \text{Total pixels in second set}}$$

where intersection represents the number of pixels in both the first set and the second set.

The Dice coefficient ranges from 0 to 1, where a value of 1 indicates a perfect match between sets, and a value of 0 indicates no match.

The formula for the Dice coefficient as a function of true positives (TP), false positives (FP) and false negatives (FN) is as follows:

$$Dice = \frac{2 \times TP}{2 \times TP + FP + FN}$$

4.7.3 Recall

Recall, also known as true positive rate, is a measure used to assess a model's ability to correctly detect true positive examples. Recall measures the proportion of true positive examples that have been correctly identified by the model. It is particularly useful when detecting positives is more critical than minimizing false positives.

It is calculated using true positives (TP) and false negatives (FN) as follows:

$$Recall = \frac{TP}{TP + FN}$$

4.7.4 Precision

Precision, also known as positive predictive value, is a measure used to assess the accuracy of positive predictions made by a model. Precision measures the proportion of positive examples predicted by the model that are actually positive. It is particularly

useful when minimizing false positives is more critical than detecting all true positives.

It is calculated using true positives (TP) and false positives (FP) as follows:

$$Precision = \frac{TP}{TP + FP}$$

4.8 Results of the experiments and analysis of the findings

4.8.1 Model results trained on ATALS V2.0 dataset

Dataset	Name	Mean IoU %	Mean Dice %	Mean Precision %	Mean Recall %
Dataset OSI	Model HT-OSI	90.69	66.29	85.25	67.24
	Resnet34-OSI	89.19	62.62	85.55	62.23
	VGG19-OSI	90.67	67.08	84.57	67.81
	Efficientnetb3-OSI	91.53	71.70	84.61	73.07
	Ensemble_OSI	91.30	66.50	90.91	64.29
Dataset 50_50	Model HT-50	90.34	80.80	92.08	82.36
	Model HT-50-Transp	90.63	81.64	93.35	81.95
	Resnet34-50	90.32	81.04	93.71	81.27
	VGG19-50	90.41	80.83	92.42	82.35
	Efficientnetb3-50	91.25	83.42	92.23	84.54
	Ensemble_50	91.12	81.91	95.73	81.02
Dataset Kfolds	HT-Kfolds-Size	91.71	69.69	86.54	70.08
	HT-Kfolds-Mixed	91.86	69.70	87.56	69.61

The OSI and Kfolds datasets contain 1648 test images and the 50_50 dataset contains 3300 test images.

Table 4.2: ATLAS Models.

4.8.2 Discussion of models trained on the ATALS V2.0 dataset

In this study, we evaluated and compared the performance of several segmentation models we propose for the analysis of MRI stroke images. Key performance metrics, such as Dice, IOU, precision and recall, were used to assess the prediction quality of each model. The main objective of this analysis is to determine the best segmentation model for MRI stroke images based on these performance metrics.

Analyzing the results of the different models evaluated, we find that « Efficientnetb3-50 » shows the highest performance in terms of Dice, IOU and precision. This model achieved an average Dice of 83.42, an average IOU of 91.25 and an average precision of 92.23. These results indicate a high ability of the model to accurately detect and segment stroke lesions in MRI images.

The HT-50 model also performed well, with an average Dice of 80.80, an average IOU of 90.34 and an average precision of 92.08. Although its performance is slightly inferior to that of the « Efficientnetb3-50 » model, it remains a solid choice for the segmentation of MRI images of stroke.

Dice, precision and recall have enabled us to observe that our models trained on the dataset containing only injured images have difficulty generalizing, since the test dataset contains all images (injured and uninjured). Although they all perform well in training, we discard them because generalization is important.

As for the models trained with the Kfolds technique, although the similarity (mean IoU) is acceptable, recall shows us that these models produce a high number of false negatives (FN). That's why we can't choose them.

In the end, the dataset containing 50% sickly images and 50% healthy images enabled us to achieve good scores and validate the effectiveness of our three proposed approaches (our CNN model, Transfer learning and Ensemble learning). Although the four models obtained on this dataset all performed well, we chose **Efficientnetb3-50** as the first, followed by **Ensemble_50** model. Our CNN **HT-50** model perfomed well but it ranks last among models trained on the 50_50 dataset. VGG19-50 and Resnet34-50 can also be used, as the performance gap is not huge. However Using conv2DTranspose instead of Upsampling2D has enabled us to improve our architecture and obtain Model HT-50-Transp, which ranks third after Efficientnetb3-50 and Ensemble_50.

Ideally, we'd like to use all four together to make an overall prediction because the precision of this technique is the highest.

4.8.3 Model results trained on ISLES 2015 dataset

Dataset	Name	Mean IoU %	Mean Dice %	Mean Precision %	Mean Recall %
Dataset DFT2	Model HT-DFT2	88.16	87.01	96.60	87.21
Datasets DWI | FLAIR | T1 | T2	Model HT-DWI	89.23	86.56	94.44	87.09
	Model HT-FLAIR	91.93	86.42	95.28	86.95
	Model HT-T1	88.93	80.41	96.54	80.44
	Model HT-T2	86.90	79.73	94.92	79.80
Datasets DWI | FLAIR | T1 | T2	Ensemble DFT1T2	90.99	87.91	92.82	89.36

The test dataset contains 199 images that the models didn't see during training or validation.

Table 4.3: ISLES Models.

4.8.4 Discussion of models trained on the ISLES 2015 dataset

Stroke MRI image analysis is crucial for the detection and accurate segmentation of brain lesions. In this study, we evaluated and compared the performance of five different segmentation models: Model "HT-DFT2", "HT-DWI", "HT-FLAIR", "HT-T1", "HT-T2" and a model based on the set of models called "Ensemble DFT1T2". The aim was to determine the best-performing model in terms of Dice, IOU, precision and recall.

The results obtained reveal that the "Model HT-FLAIR" achieved the best overall performance. It had an average IOU of 91.93 and an average Dice of 86.42. These values indicate that the model is capable of accurately capturing the contours of brain lesions. In addition, the "Model HT-FLAIR" has a high mean precision of 95.28, indicating that the majority of the model's positive predictions are correct. Average recall is also high, suggesting that the model is able to identify most of the true positives among all the real true positives.

Comparatively, the other models also performed well, but with some variation. For example, the "Model HT-DFT2" achieved a slightly higher average Dice of 87.01, but its average IOU was slightly lower at 88.16. The "HT-DWI", "HT-T1" and "HT-T2" models also performed respectably, but with slightly lower scores compared with "Model HT-FLAIR". The "Ensemble DFT1T2" set of models showed an improvement over the individual models, with a mean IOU of 90.99 and a mean Dice of 87.91.

The results of this study highlight the importance of choosing the right type of MRI images for segmentation of cerebral stroke lesions. In our case, FLAIR sequence images

gave the best performance in terms of Dice, IOU and Precision. This can be attributed to the high-contrast properties of FLAIR images, which enable better distinction between normal brain tissue and lesions.

Nevertheless, it should be noted that each model has its own strengths and weaknesses depending on the characteristics of the stroke images. Consequently, the choice of segmentation model should be based on the specific needs of each study and the nature of the lesions sought.

In conclusion, the **"Model HT-FLAIR"** stands out as the best choice for segmentation of MRI stroke images, with superior performance in terms of Dice, IOU and Precision. However, other models such as "Ensemble DFT1T2" may also be considered, depending on the specific needs and characteristics of the MRI stroke images used in each study.

4.8.5 Some performance graphs for model training

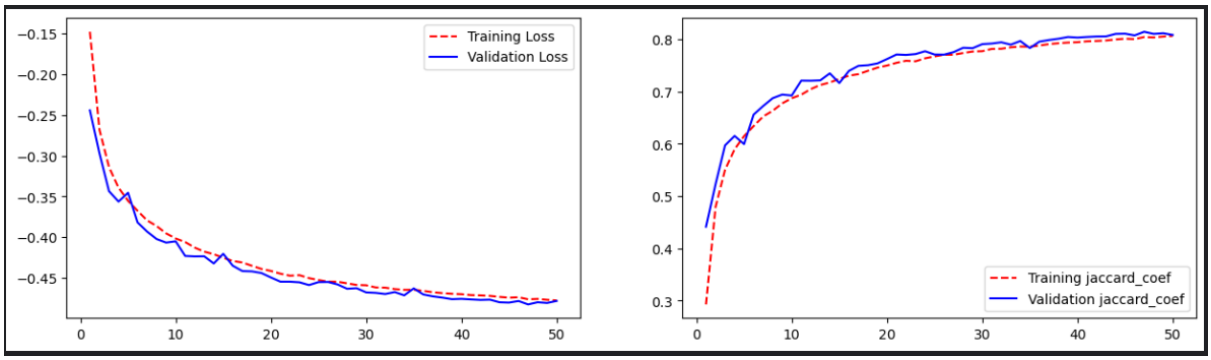


Figure 4.11: Example of ATLAS V2.0 Model performance.

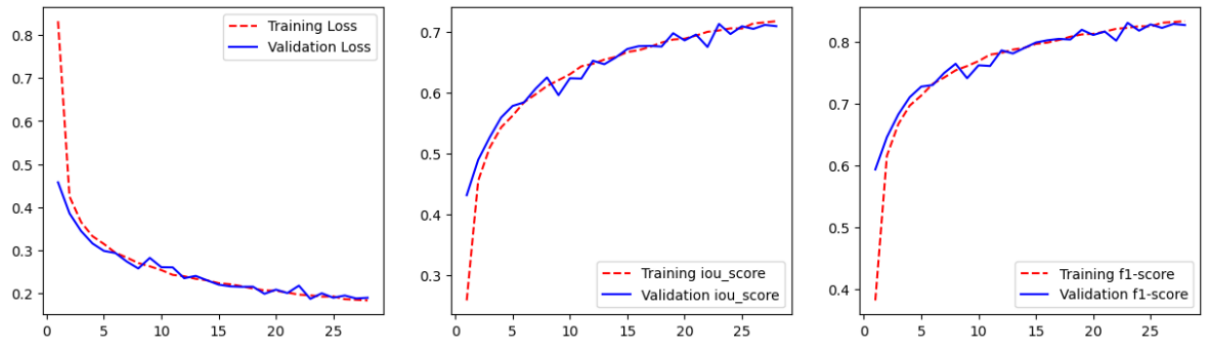


Figure 4.12: Example of metric evolution during training.

4.8.6 Some test segmentation results

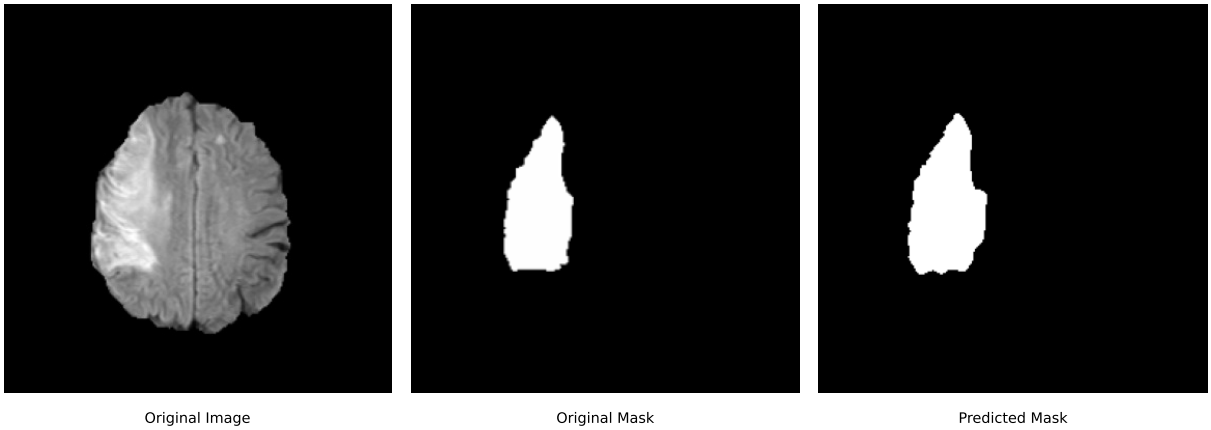


Figure 4.13: Example of Flair modality segmentation (ISLES 2015).

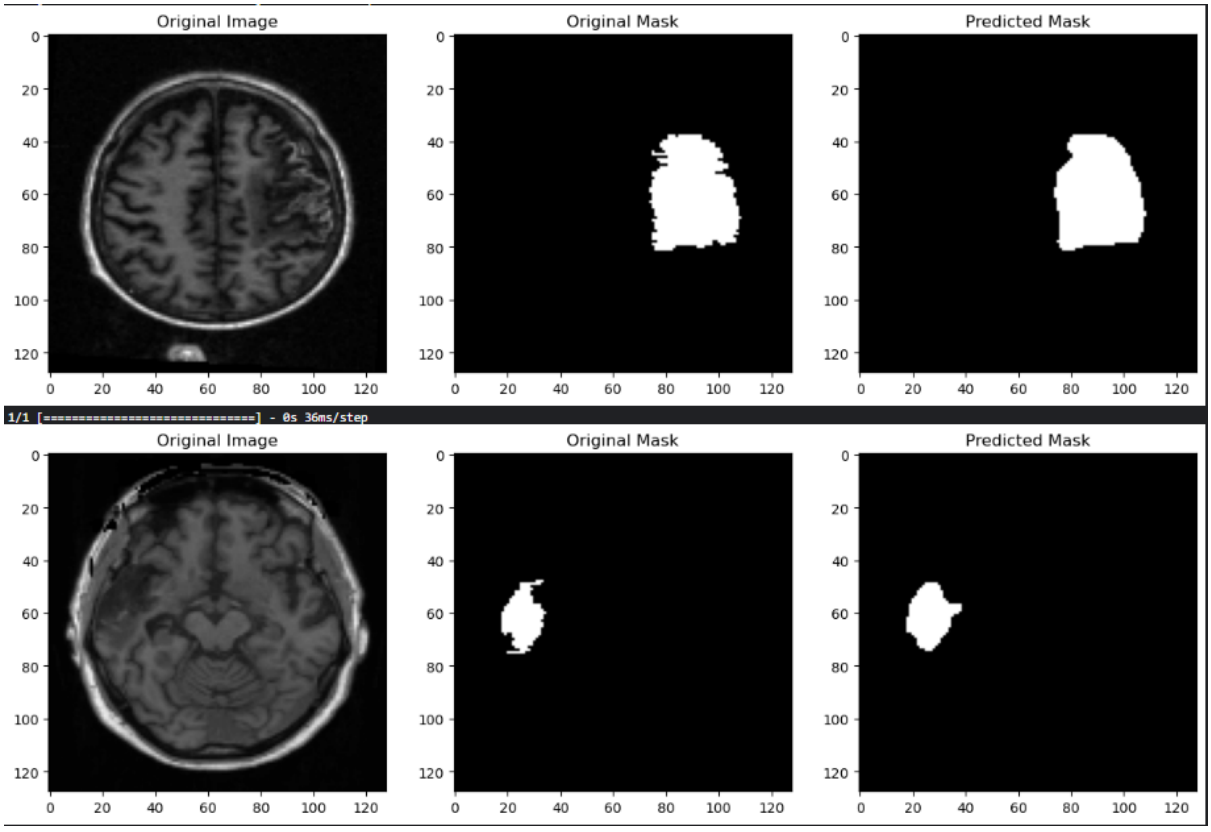


Figure 4.14: Example of ATLAS V2.0 image segmentation.

4.9 Graphical user interface

4.9.1 2D image segmentation interface

To facilitate man-machine communication, we offer a graphical interface. This interface has three display sections : the image to be segmented, the predicted mask and the mask overlay on the image. It also features buttons :

- Resolution selection drop down list to select between 128x128 and 256x256 depending on selected model input.

- Select Image to select an image from a directory.

- Select backbone drop down list for selecting image preprocessing according to the selected model.

- Select Model to select the desired model.

- Predict to start prediction.

- Reset to reset the interface to its initial state.

- Exit to quit the interface.

- The two buttons are used to go to the next or previous image.

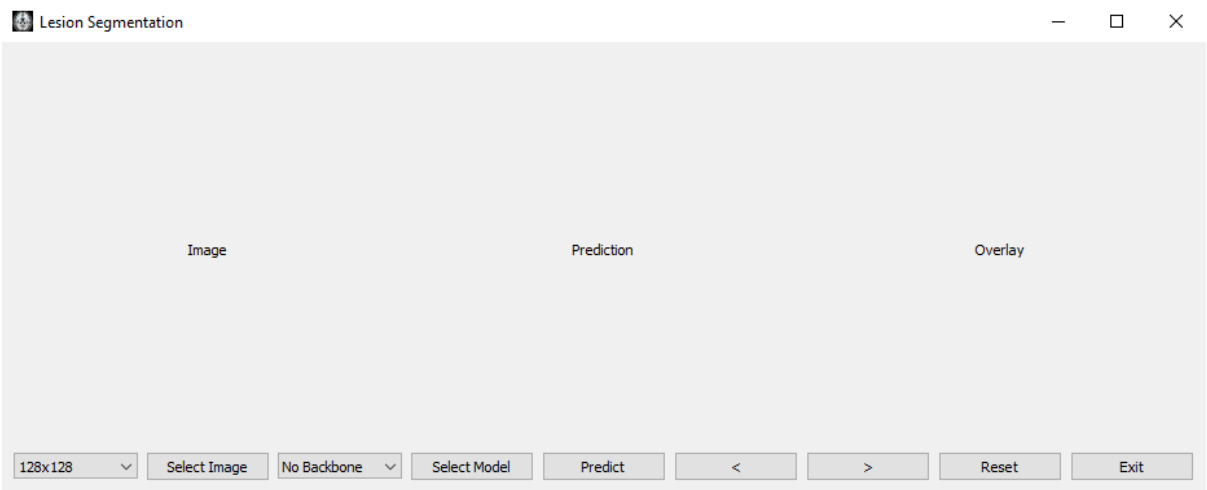


Figure 4.15: Our GUI

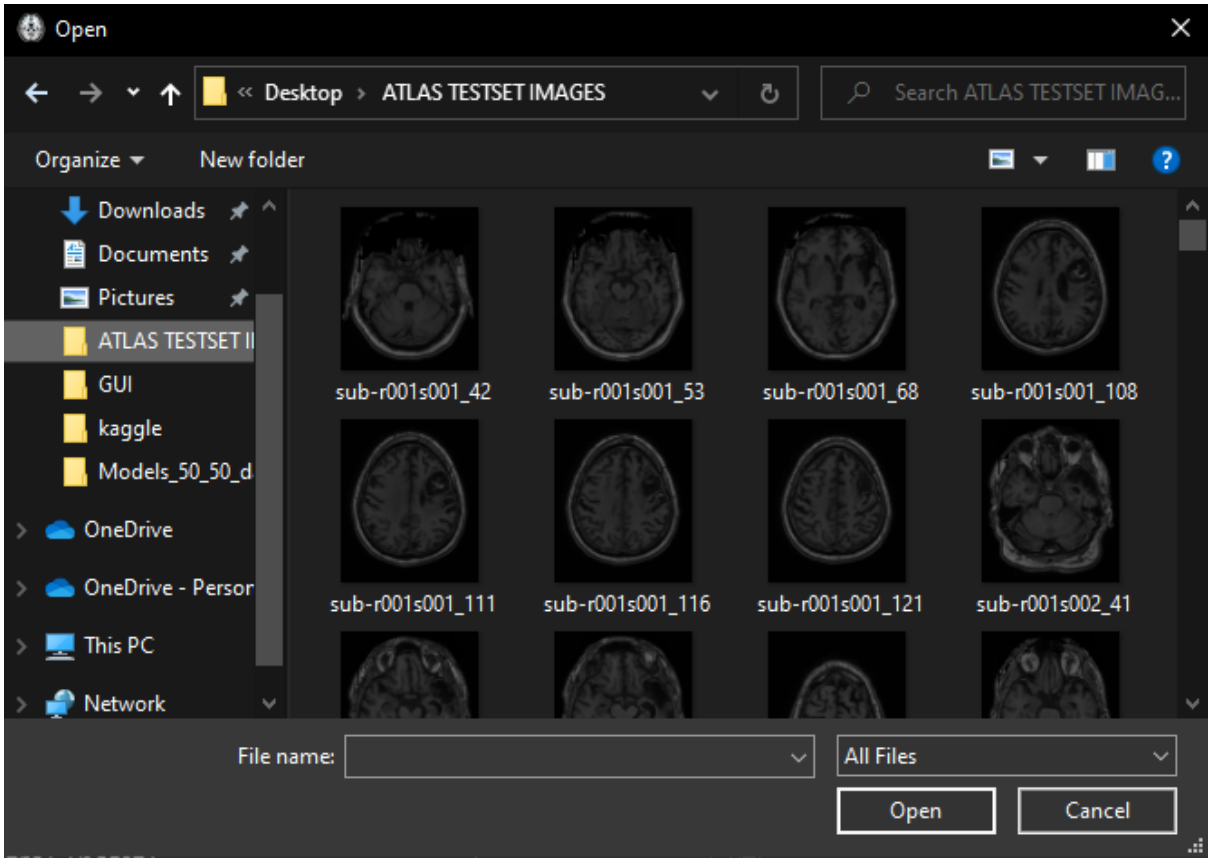


Figure 4.16: Select Image

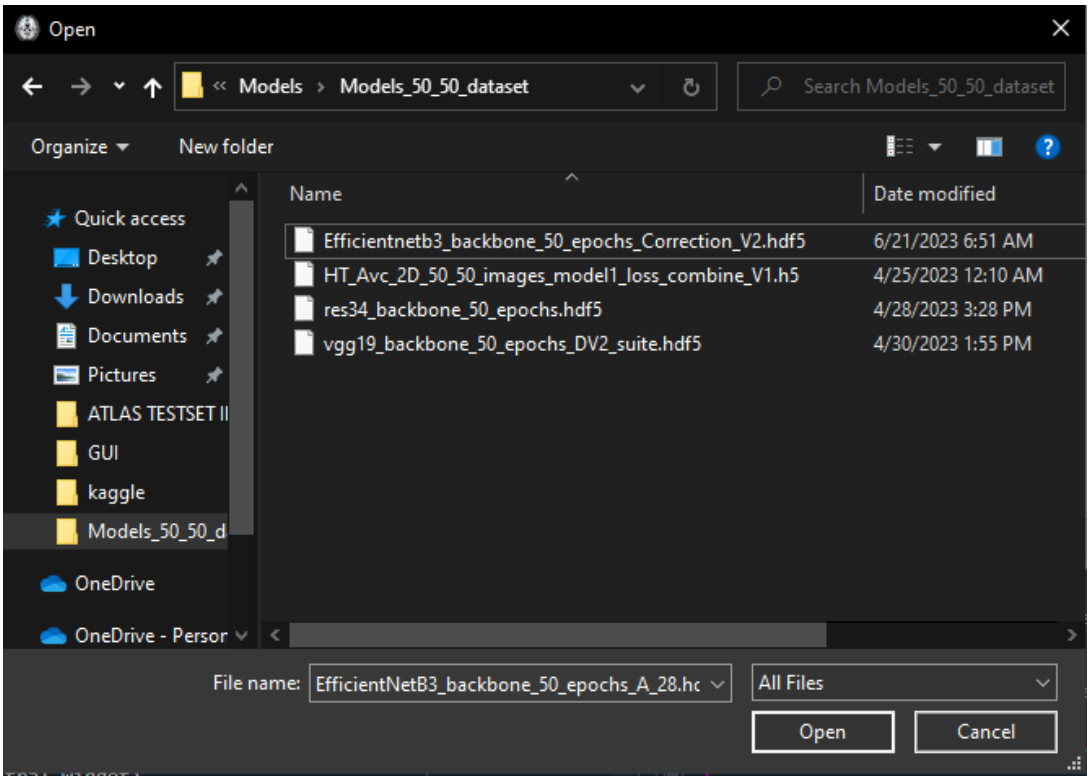


Figure 4.17: Select Model

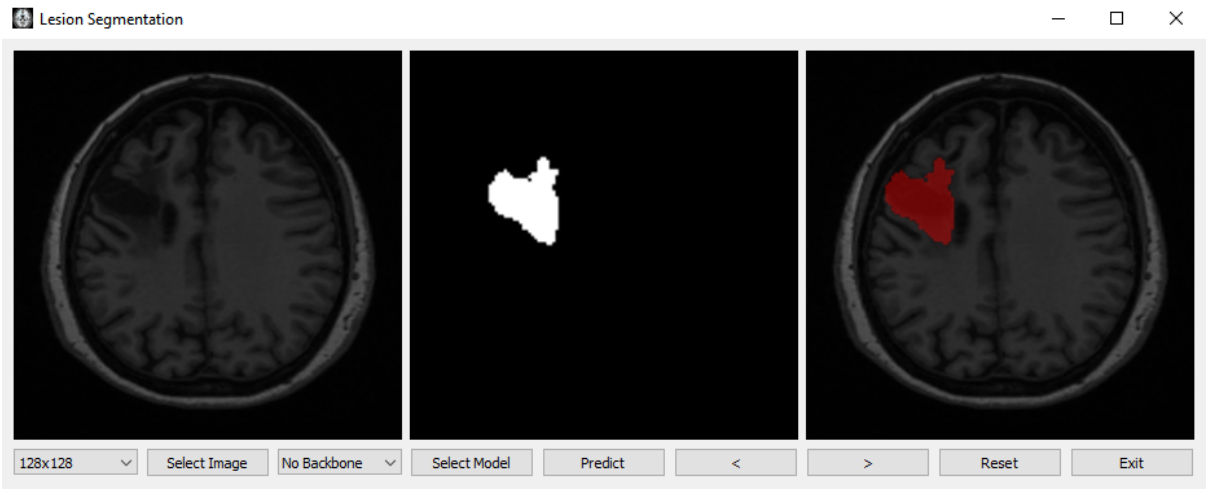


Figure 4.18: Prediction with Model HT-50

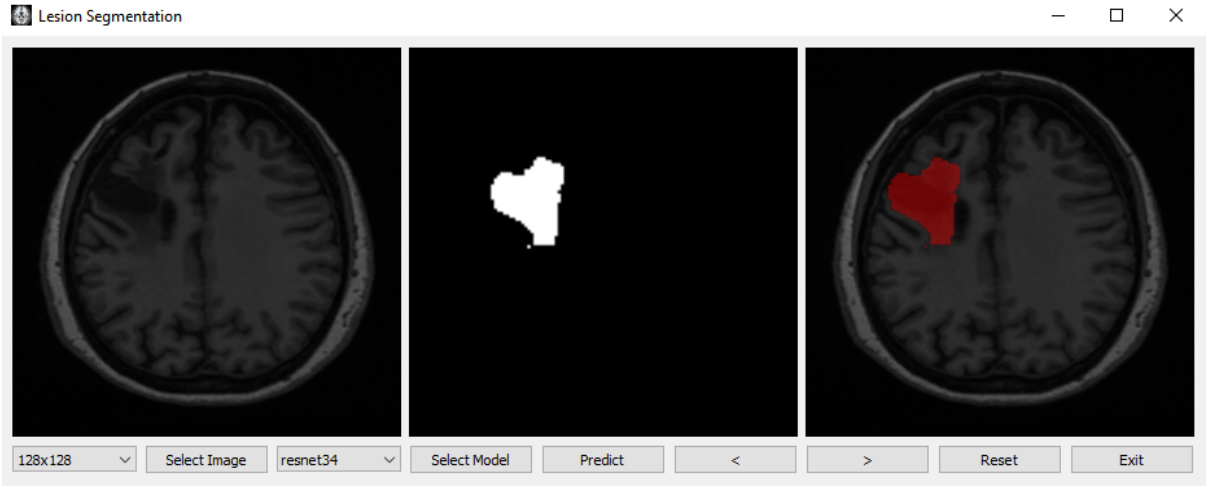


Figure 4.19: Prediction with Model Resnet34-50

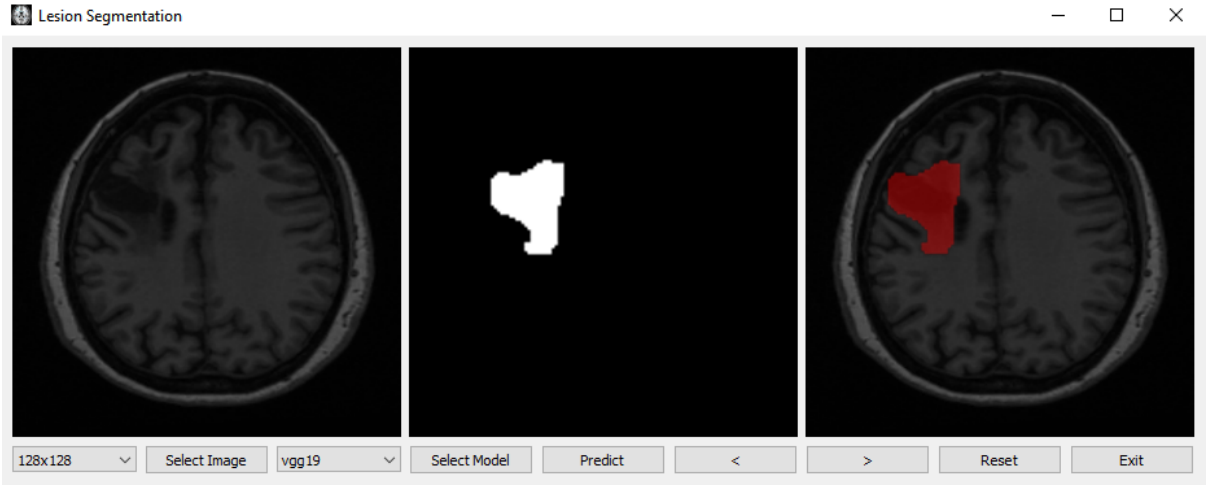


Figure 4.20: Prediction with Model Vgg19-50

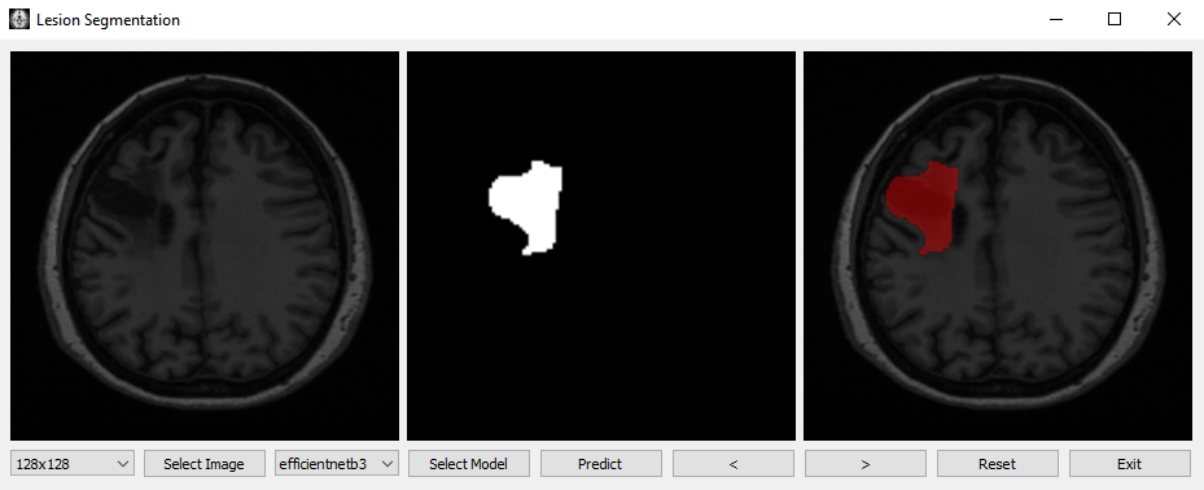


Figure 4.21: Prediction with Model Efficientnetb3-50

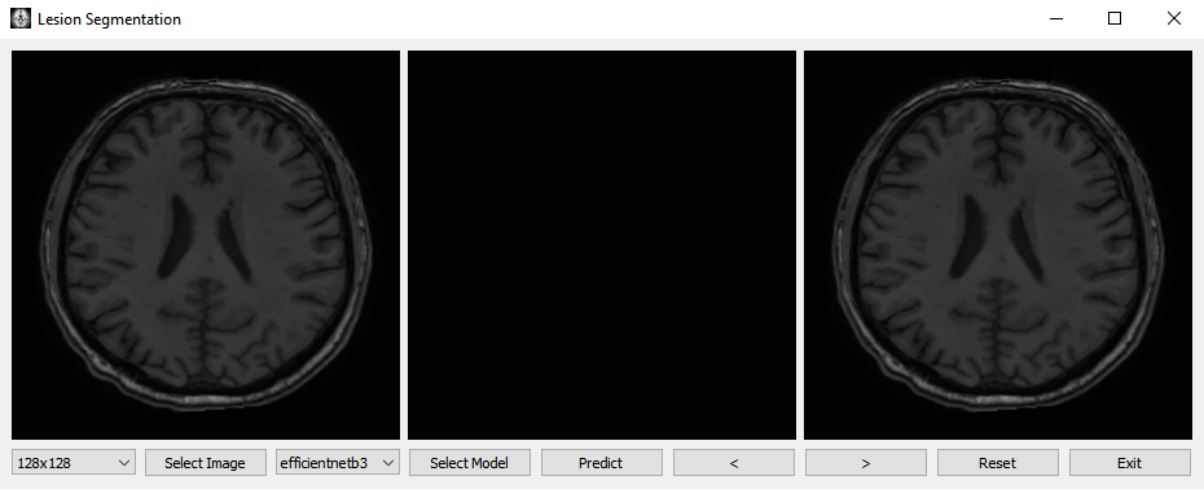


Figure 4.22: No lesion in this image

4.9.2 2D to 3D images converter

We have developed two separate software applications to facilitate the generation of 3D mask images from predicted 2D images and from 2D images back to 3D. Due to package incompatibility between nibabel, which is used for converting 3D Nifti images to 2D slices, and TensorFlow, which is utilized to load our trained model for generating predictions, we had to utilize two distinct environments.

Here's how the workflow operates:

a. In the first application, called NII file slicer, we open an MRI image in Nifti format and extract 2D slices from the z-axis. These slices are then saved in a designated folder.

b. In the second application, named Lesion Segmentation on multiple PNG images

at once, we open the folder containing the 2D slices obtained in the previous step. This application is responsible for making predictions on these slices, and the resulting predictions are saved in another folder.

c. Finally, we revisit the first application, NII file slicer. In this step, we select the folder that contains the prediction images generated in the second application. Using these prediction images, we generate a Nifti file. This file can be saved and opened using any software capable of reading the Nifti format.

By following this process, we enable the seamless transformation from 3D Nifti images to 2D slices, performing predictions on those slices, and ultimately reconstructing a 3D mask image in the Nifti format for further analysis and utilization in compatible software applications.

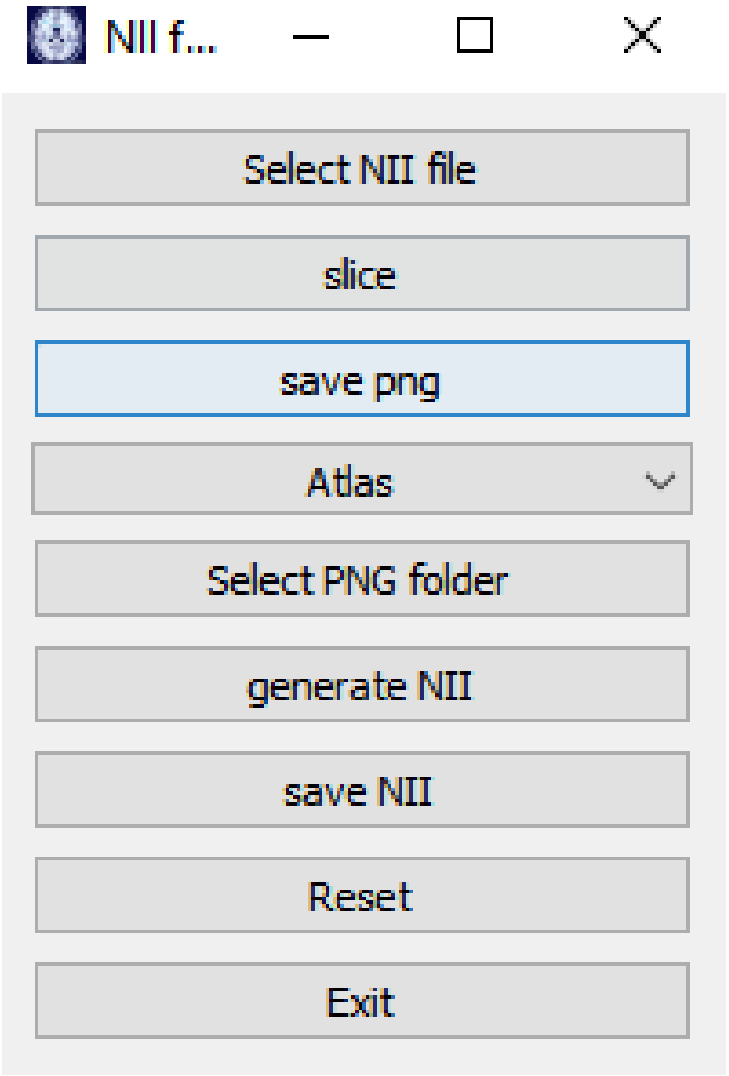


Figure 4.23: NII file slicer

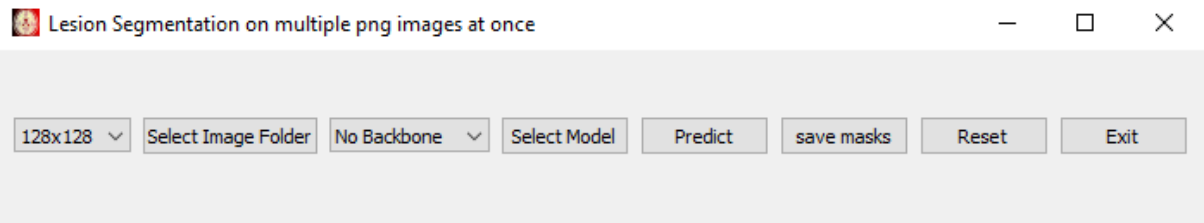


Figure 4.24: Lesion segmentation on multiple png images at once

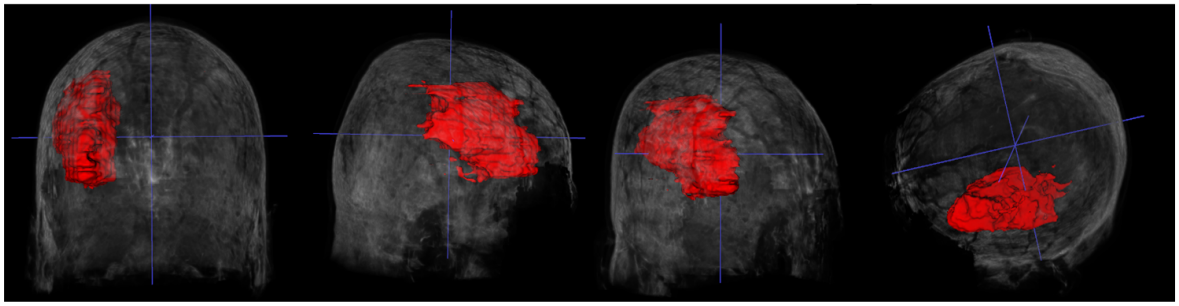


Figure 4.25: 3D View of segmented lesion

4.10 Conclusion

In this chapter, we discussed in detail the implementation of our segmentation study for MRI stroke images. We began by discussing the datasets used, which comprise previously annotated MRI stroke images. These datasets are essential for training and evaluating our segmentation models. After training and testing the models, we carried out an in-depth analysis of their performance using metrics such as Dice, IOU, precision and recall obtained on the test set. These metrics enable us to assess the quality of the segmentations produced by the models and their ability to accurately detect brain lesions in stroke MRI images.

Performance analysis revealed that the **Efficientnetb3-50** model (dataset ATLAS V2.0) and the **Model HT-FLAIR** (dataset ISLES 2015) achieved the best results in terms of Dice, IOU, Recall and Precision.

The "Ensemble DFT1T2" set showed an improvement over the individual models, highlighting the importance of combining multiple sources of information for more accurate segmentation.

The results obtained provide valuable information for the selection of the best segmentation model in the context of stroke imaging, highlighting the importance of

using FLAIR images and the Learning ensemble of models for accurate segmentation.

The results provide a solid foundation for further research in the field of stroke MRI image segmentation, paving the way for improved stroke patient care.

General Conclusion

Stroke remains one of the most dangerous diseases, and finding an efficient and autonomous way to detect it early without a great deal of human intervention is a priority, as it's a very tedious task for a specialist. Our aim was therefore to use the power of artificial intelligence through Deep Learning to propose a segmentation model for MRI stroke images. In order to achieve our objective, we had to propose several models based on several approaches, such as the proposal of our CNN model based on U-Net, Transfer Learning and Ensemble Learning.

We had to use two main datasets (ATLAS V2.0 and ISLES 2015), which enabled us to build other datasets. In the end, we obtained 18 models. Many of the models performed well, but especially as we're in the healthcare field, a proposed solution requires good accuracy.

We've found that transfer learning with Effitientnetb3 as backbone is effective, but the problem is that it takes a lot of time to train. We therefore used the Dice coefficient, IOU, Precision and Recall metrics to select our final models, which were « **Efficientnetb3-50** » and « **Model HT-FLAIR** » according the datasets used.

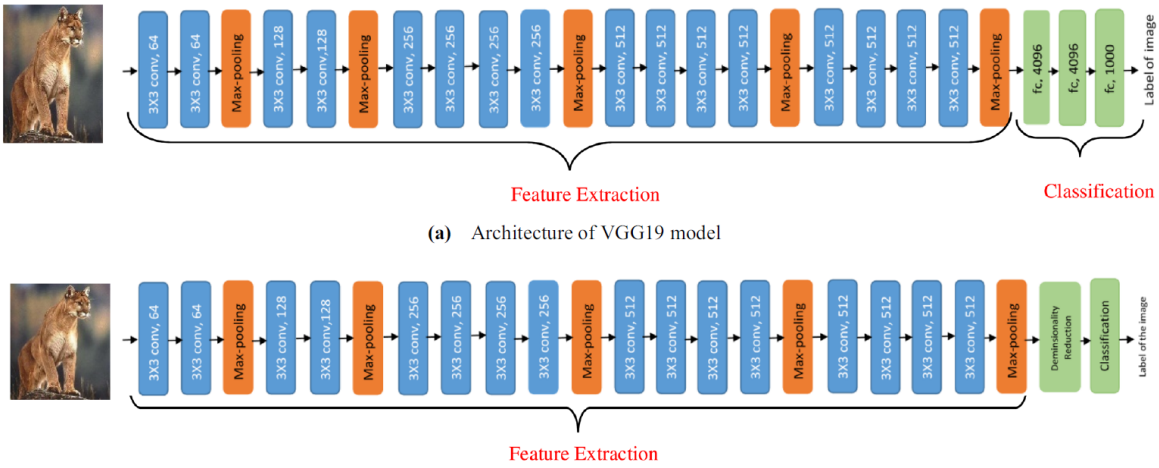
As we mentioned earlier, the medical field demands a very high level of efficiency from the solutions we propose. This is why we plan to continue our work by proposing new approaches to improve performance:

- Working in 3D if we can get the necessary equipment,

- Combining traditional approaches and Deep Learning,

- After segmentation we locate the region of the brain that was damaged using brain atlas map.

Appendix



(a) Architecture of VGG19 model

(b) Ensemble of deep feature extraction using VGG19 model and machine learning classification

Figure 4.26: Architecture of VGG19 [9].

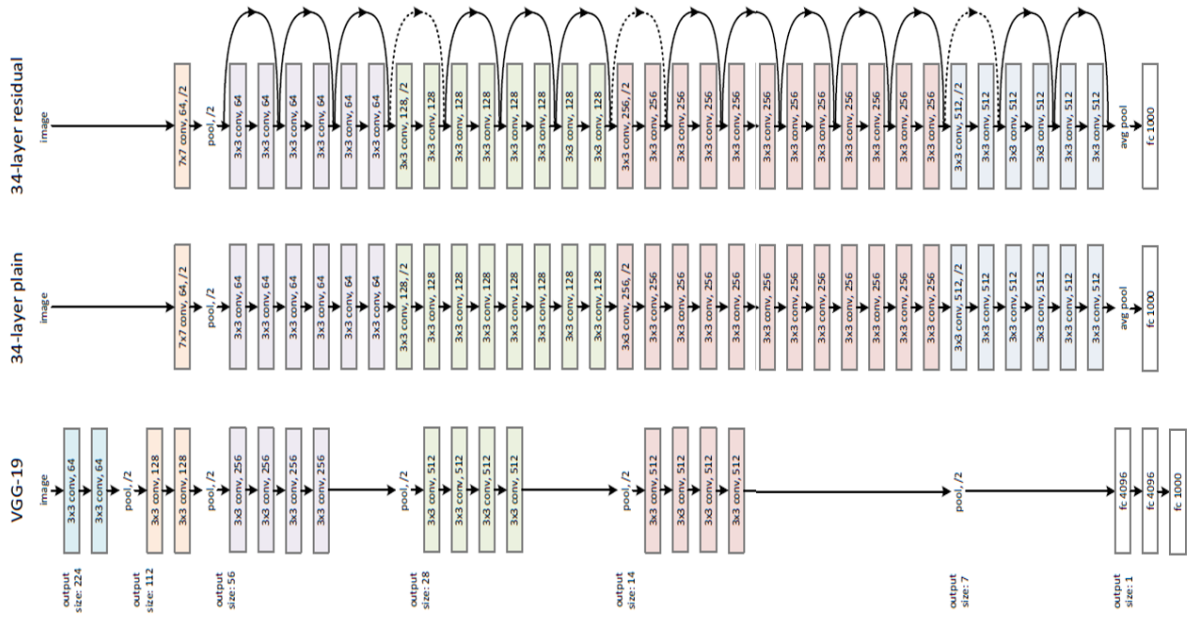


Figure 4.27: Architecture of Resnet34 [10].

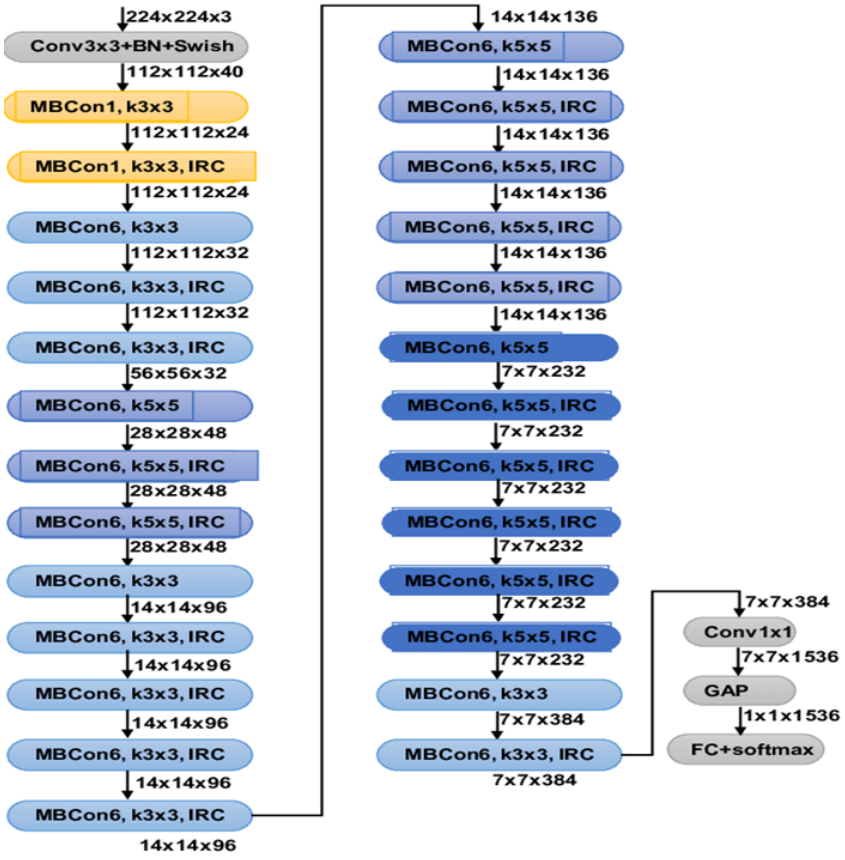


Figure 4.28: Architecture of Efficientnetb3 [11].

Bibliography

[1] centre hospitalier des quatre villes. scanner et irm. https://www.ch4v.fr/activites-medico-techniques/scanner-et-irm/, 2019. Accessed on 01 May 2023.

[2] Kastler B, Vetter D, Patay Z, and Germain P. Comprendre l'irm manuel d'auto-apprentissage. *Elsevier Masson SAS*, pages 6–27, 2011.

[3] Amélie Dauba. *Les conséquences d'un accident vasculaire cérébral et le rôle du pharmacien d'officine dans leur prise en charge*. PhD thesis, Université de Bordeaux, 2018.

[4] Nafiz Shahriar. What is convolutional neural network (cnn)? https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5, 2023. Accessed on 10 june 2023.

[5] Pavel Iakubovskii. Segmentation models. https://github.com/qubvel/segmentation_models, 2019.

[6] University of Freiburg LMB. Convolutional networks for biomedical image segmentation. https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/, 2015. Accessed on 24 March 2023.

[7] S.-L. Liew, J. M. Anglin, N. W. Banks, M. Sondag, K. L. Ito, and A Kim, H.and ... Stroud. A large, open source dataset of stroke anatomical brain images and manual lesion segmentations. *Scientific Data*, 5, 2018.

[8] Kukil. Intersection over union (iou) in object detection & segmentation. https://learnopencv.com/intersection-over-union-iou-in-object-detection-and-segmentation/, 2022. Accessed on 07 June 2023.

[9] Monika Bansal, Munish Kumar, Monika Sachdeva, and Ajay Mittal. Transfer learning for image classification using vgg19: Caltech-101 image data set. *Journal of ambient intelligence and humanized computing*, pages 1–12, 2021.

[10] 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.

[11] Haikel Alhichri, Asma S Alswayed, Yakoub Bazi, Nassim Ammour, and Naif A Alajlan. Classification of remote sensing images using efficientnet-b3 cnn model with attention. *IEEE access*, 9:14078–14094, 2021.

[12] Accident vasculaire cérébral. https://www.emro.who.int/fr/health-topics/stroke-cerebrovascular-accident/index.html, 2023. Accessed on 23 March 2023.

[13] Les 10 principales causes de mortalité. https://www.who.int/fr/news-room/fact-sheets/detail/the-top-10-causes-of-death/, 2020. Accessed on 30 April 2023.

[14] BAUMANN P.S, HAGMANN P, and MARQUET P. De l'eau et des neurones... irm de diffusion, cartographie cérébrale et brèves perspectives psychiatriques. *L'information psychiatrique*, 86(2010/6):497, 2012.

[15] Hamon M, Marié RM, Clochon P, Coskun O, Constans JM, Viader F, Courthéoux P, and Baron JC. Relation quantitative des modifications de la diffusion et de la perfusion au sein du parenchyme cérébral au cours de l'accident ischémique aigu. *Journal of neuroradiology*, 32(2):118, 2005.

[16] Vermandeland Maximilien, Dewalle AS, Palos G, Christian Vasseur, and Jean Rousseau. Recalage et mise en correspondance d'images tomographiques et

de projection. cas de l'angiographie par résonance magnétique (arm) et de l'angiographie par rayons x (arx). *Traitement du Signal et de Image*, 20(2), 2003.

[17] Kaplan Haenlein. A brief history of artificial intelligence: On the past, present, and future of artificial intelligence. *California Management Review*, pages 2–3, 2019.

[18] Tang X. The role of artificial intelligence in medical imaging research. *BJR Open*, page 2, 2020.

[19] Jian Wang, Hengde Zhu, Shui-Hua Wang, and Yu-Dong Zhang. A review of deep learning on medical image analysis. *Springer Science*, pages 350–354, 2020.

[20] Anirudha Ghosh, Sufian A, Farhana Sultana, Amlan Chakrabarti, and Debashis De. Fundamental concepts of convolutional neural network, 01 2020.

[21] Humera Shaziya. A study of the optimization algorithms in deep learning, 03 2020.

[22] Suganyadevi S, Seethalakshmi V, and Balasamy K. A review on deep learning in medical image analysis. *International Journal of Multimedia Information Retrieval*, 11:19–38, March 2022.

[23] Senthilkumaran N and Vaithegi S. Image segmentation by using thresholding techniques for medical imaging. *Computer Science I& Engineering: An International Journal*, 6(1):page 3, 2016.

[24] Ma Z, Tavares JMRS, Jorge RMN, and Mascaranhas T. A review of algorithms for medical image segmentation and their applications to the female pelvic cavity. *Comput Method Biomech Biomed Eng*, 13(2):233–246, 2010.

[25] Abdel-Maksoud E, Elmogy M, and Al-Awadi R. Brain tumor segmentation based on a hybrid clustering technique. *Egyptian Informatics Journal*, 16(1):71–81, 2015.

[26] Saleha Masood, Muhammad Sharif, Afifa Masood, Mussarat Yasmin, and Mudassar Raza. A survey on medical image segmentation. *Current Medical Imaging*, 11(1):3–14, 2015.

[27] Ronneberger O., Fischer P., and Brox T. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, page 234–241, 2015.

[28] Long J., Shelhamer E., and Darrell T. Fully convolutional networks for semantic segmentation. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[29] Zhang X., Zhao H., Li X., Feng Y., and Li H. A multi-scale 3d otsu thresholding algorithm for medical image segmentation. *Digit. Signal Process.*, page 186–199, 2016.

[30] Jiangdian Song, Caiyun Yang, Li Fan, Kun Wang, Feng Yang, Shiyuan Liu, and Jie Tian. Lung lesion extraction using a toboggan based growing automatic segmentation approach. *IEEE Transactions on Medical Imaging*, 35(1):337–353, 2016.

[31] T Kesavamurthy and S SubhaRani. Pattern classification using imaging techniques for infarct and hemorrhage identification in the human brain. *Calicut Medical Journal*, 4(3):1–5, 2006.

[32] N.A. Mohamed, M.N. Ahmed, and A. Farag. Modified fuzzy c-mean in medical image segmentation. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, volume 6, pages 3429–3432, 1999.

[33] Yacine Kabir, Michel Dojat, Benoît Scherrer, Florence Forbes, and Catherine Garbay. Multimodal mri segmentation of ischemic stroke lesions. In *2007 29th annual international conference of the IEEE engineering in medicine and biology society*, pages 1595–1598. IEEE, 2007.

[34] Shubham Joshi and Sonal Gore. Ishemic stroke lesion segmentation by analyzing mri images using dilated and transposed convolutions in convolutional neural networks. In *2018 fourth international conference on computing communication control and automation (ICCUBEA)*, pages 1–5. IEEE, 2018.

[35] Liangliang Liu, Shaowu Chen, Fuhao Zhang, Fang-Xiang Wu, Yi Pan, and Jianxin Wang. Deep convolutional neural network for automatically segmenting acute ischemic stroke lesion in multi-modality mri. *Neural Computing and Applications*, 32:6545–6558, 2020.

[36] Mobarakol Islam, Parita Sanghani, Angela An Qi See, Michael Lucas James, Nicolas Kon Kam King, and Hongliang Ren. Ichnet: intracerebral hemorrhage (ich) segmentation using deep learning. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I 4*, pages 456–463. Springer, 2019.

[37] ISLES 2015 Challenge. http://www.isles-challenge.org/ISLES2015/, 2015. Accessed on 04 April 2023.