Universitetet
i Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study programme/specialisation: Computer Science	Spring semester, 2019 Open/~~Confidential~~
Author: Luca Tomasetti	.... (signature of author)

Programme coordinator: Professor Kjersti Engang

Supervisor(s): Professor Kjersti Engang & Professor Kathinka Dæhli Kurz

Title of master's thesis:

Segmentation of infarcted regions in Perfusion CT images by 3D deep learning

Credits: 30

Keywords:
 	
   Perfusion CT, Ischemic stroke, Image segmentation, Machine Learning, Deep Neural Network, Convolutional Neural Network, U-Net.
	Number of pages: 82
	
	   + supplemental material/other: 77 pages of Appendix A-C in addition to *ThesisCode.7z*
	
	                Stavanger, 13/06/2019

University of Stavanger

Faculty of Science and Technology
Department of Electrical Engineering and Computer Science

Segmentation of infarcted regions in Perfusion CT images by 3D deep learning

Master's Thesis in Computer Science
by
Luca Tomasetti

Internal Supervisor
Kjerst Engan

External Supervisor
Kathinka Dæhli Kurz

June 15, 2019

"All those moments will be lost in time, like tears in rain."

Roy Batty, Blade Runner.

Abstract

This thesis explores different Convolutional Neural Network (CNN) approaches to classify and segment infarcted regions from images taken through a Computed Tomography Perfusion (CTP) from patients of the Stavanger's hospital (SUS) affected by an ischemic stroke. Also, it evaluates the accuracy and the loss functions of the images analyzed through CNN. Furthermore, a segmentation approach, based on a U-Net model, is tested to create, from scratch, a unique image containing a summary of the section of the brain investigated with the different infarcted regions prediction. The purpose of this thesis work is to find a fast and effective method to help doctors in their decisions during these delicate and problematic situations.

Acknowledgements

I would like to genuinely thank both my supervisors, Kjersti Engan and Kathinka Dæhli Kurz, for helping me in the past six months to realize this work, and for expressing their passion and their dedication for this thesis.

I would also like to express my sincere gratitude to all my friends, constantly present at all times during the realization of this thesis; I'm really thankful of all the moments spent together, thank you.

Finally, a special mention goes to all members of my family, who were always there to support me in all decisions, good and bad, made during my entire life. I know that it can be very hard to understand some of my decisions, but you were, and will always be there to help me make the right ones. This thesis was realized just for you, to show all my love and my appreciation that I couldn't demonstrate in the last two years, far away from home.

Grazie da profondo del cuore.

Contents

Abbreviations

AIF   **A**rterial **I**nput **F**unction

ANN   **A**rtificial **N**eural **N**etwork

AP   **A**verage **P**recision

CBF   **C**erebral **B**lood **F**low

CBV   **C**erebral **B**lood Volume

CT   **C**omputed **T**omography

CTP   **C**omputed **T**omography **P**erfusion

CNN   **C**onvolutional **N**eural **N**etwork

CVST   **C**erebral **V**enous **S**inus **T**hrombosis

DICOM **D**igital **I**maging and **CO**mmunications in Medicine

DNN   **D**eep **N**eural **N**etwork

FDR   **F**alse **D**iscovery **R**ate

FFNN   **F**eed **F**orward **N**eural **N**etwork

FNR   **F**alse **N**egative **R**ate

FOR   **F**alse **O**mission **R**ate

FPR   **F**alse **P**ositive **R**ate

IRF   **I**mpulse **R**esidue **F**unction

mAP   **m**ean **A**verage **P**recision

MLP   Multi-**L**ayer **P**erceptron

MRI   **M**agnetic **R**esonance **I**maging

MSE   **M**ean **S**quared **E**rror

MTT   **M**ean **T**ransit **T**ime

NPV   **N**egative **P**redicted **V**alue

SGD   **S**tochastic **G**radient **D**escent

SVD   **S**ingle **V**alue **D**ecomposition

SUS   **S**tavanger **U**niversitets**S**jukehus

TDC   **T**ime **D**ensity **C**urve

TIA   **T**ransient **I**schemic **A**ttack

TTP   **T**ime-**T**o-**P**eak

WHO   **W**orld **H**ealth **O**rganisation

1

Introduction
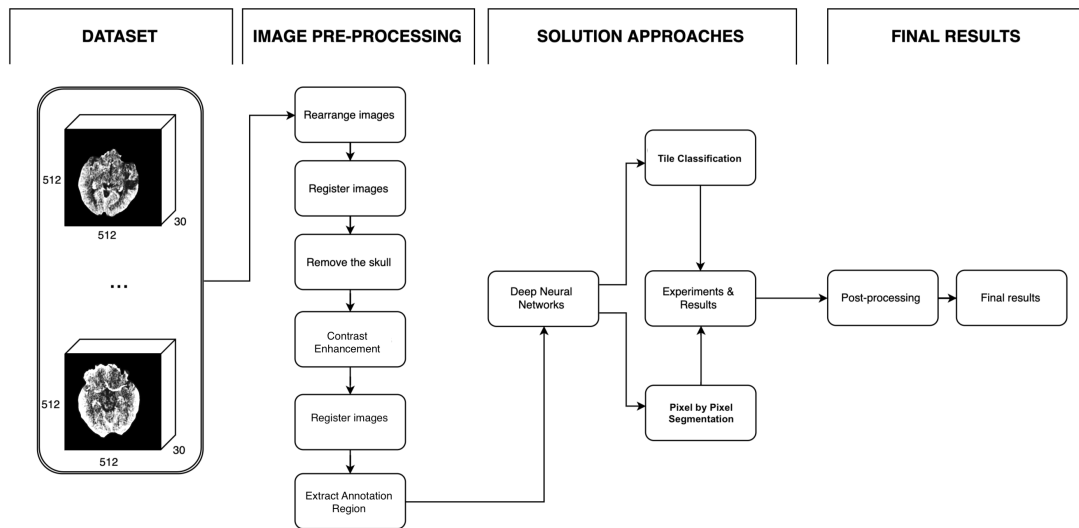
Figure 1.1: Overview of the structure approach.

1.1   Motivation

A cerebral stroke can occur if the flow of oxygen-rich blood to a portion of the brain is blocked [1]. Cerebral strokes are a severe neurological condition that can cause lasting brain damage, long-term disability, and death. It can lead to a drastic change of life for the affected patient and expensive healthcare treatments for society.

Cerebral stroke is the third cause of death among adults in Norway [2] and the second most common cause worldwide [3]. Despite the incidences that have been significantly reduced over the past 20 years in the entire world, in Norway, over 13000 patients were admitted to a hospital with a stroke in 2016 [4]. Worldwide, the number of registered deaths due to stroke was almost 6 million over the 56.9 million deaths just in 2016 [5]: a stroke causes over 10% of the annual deaths worldwide. One in six people will be affected by a stroke during their life [6]; approximately 66% of these will have a functional disability as a result of a stroke while mortality is 17% within the first three months after the stroke. Every year in Norway, between 350 and 500 people die of stroke [7].

For medical doctors, dealing with a cerebral stroke is very complicated. Time is the fundamental factor during the early stage of the treatment, as stated by the World Health Organization (WHO), which says that a stroke can be described as a *"neurological deficit of cerebrovascular cause that persists beyond 24 hours or is interrupted by death within 24 hours"* [8]. Hence, after a patient is admitted at SUS, he/she is immediately taken for a CT Perfusion scan (CTP) to find out the best treatment for him/her. CTP images, with the support of parametric maps, explained in detail in Sec. 2.1.4, are helpful to visualize the damaged area of the brain and to assess if the damage is reversible or irreversible.

1.2   Problem Definition

The aforementioned fact gives a small but general overview of the reasons why a stroke can lead to severe disease for human beings. Over the last decades, new ways are explored that can discover quickly and automatically the location of the stroke's area and segment the different affected regions of the brain.

The thesis presents a fast mechanism to easily detect the infarcted areas of the stroke based on a series of CTP images, illustrated in Sec. 2.1.3. Plus, the thesis proposes two methods to classify or to segment the various regions, (penumbra and core) essential to help doctors in their decisions of how to treat a patient, detected inside the stroke's area. The first approach proposed to accomplish this purpose, is based on the reconstruction of brain sections with different predicted labels using the Tile Classification solution,

as explained in detail in Chap. 4, while the other approach relies on a Pixel by Pixel Segmentation to create the various regions of the brain, as defined in Chap. 5. All implemented methods are based on a Deep Neural Network (DNN) approach, described in detail in Sec. 2.2.

1.3 Outline

Fig. 1.1 presents an overview of the various steps involved during the realization of this thesis work. Besides this chapter, which gives a brief introduction about the problem and motivation, the thesis is structured in 6 different chapters:

- The second chapter presents a technical and medical background for a better understanding of the thesis's topics and problems.

- The third chapter describes the analyzed dataset in detail, presents the pre-processed steps involved and discuss the other contributions to prepare the input for the architectures.

- Chapter four explains the first proposed method in detail, based on Computational Neural Network (CNN), specifying the reasons why this approach was chosen; furthermore, it describes the experiments evaluated.

- Chapter five presents the second proposed method, which was implemented using an U-Net structure, plus various experiments and relative evaluations.

- The sixth chapter presents a discussion on the results achieved during the experiments of the various implemented methods and a comparison between them. It also discusses the possible future directions of this thesis

- The last chapter presents the conclusion of the work.

"Deep Learning is a superpower. With it, you can make a computer see,
synthesize novel art, translate languages, render a medical diagnosis,
or build pieces of a car that can drive itself. If that isn't a superpower,
I don't know what is."

Andrew Ng

2

Background

The topics presented in this chapter give a brief explanation of medical and technical background; they are useful to understand the reasons why the methods proposed are chosen and how they might be necessary for future clinical decisions. More detailed information of the arguments covered in this chapter can be found here [1, 9–14].

2.1 Medical Background

As explained in the introduction, strokes can occur if the flow of oxygen-rich blood to a portion of the brain and medical doctors must, therefore, act very quickly if a patient is affected by a stroke. The treatment window for thrombolysis is 4,5 hours from symptom onset [15]. The patient's condition could only get worst if the blood flow is not restored as soon as possible. A crucial reason for doctors to act rapidly is because a patient can lose up to 1.9 million neurons, 14 billion synapses, and 12 km nerve fibers every minute from the time of the stroke happens [16].

Two main categories of strokes are a hemorrhagic and ischemic stroke. Both of them lead to brain malfunction in the affected area. However, an ischemic stroke is 10-times more frequent than a hemorrhagic stroke in a Western Country; the hemorrhagic stroke is considered to have higher mortality rate compared to ischemic stroke [17]. Another

possible category of stroke is a transient ischemic attack (TIA), also called mini-stroke. TIA is a short episode of neurological dysfunction generated by a lack of blood flow in the brain without tissue death. In TIA, symptoms usually resolve within 1 hour. The phenomenon of a TIA is a risk factor for eventually causing a stroke [18].

In the later chapters there will be a focus only on ischemic stroke, since analyzed patients in the thesis were affected by that; the other category, hemorrhagic stroke, is just briefly mentioned in the next section.

2.1.1   Hemorrhagic Stroke

A bleeding artery inside the brain causes a hemorrhagic stroke. The pressure generated from the leaked blood damages brain cells. The stroke can happen in two different forms: if the bleeding is located in a vessel inside the brain itself, it is called intracerebral. Otherwise, if the bleeding occurs outside the brain tissue but still inside the skull, it is called subarachnoid hemorrhage. Fig. 2.1 shows an example of a hemorrhagic stroke: the aneurysm in a cerebral artery breaks open, which causes bleeding around the brain, leading to the death of brain tissues caused by the mass-effect of the blood clot in a constricted system like the skull, assuming surgical decompression or interventional treatment of the affected artery doesn't occur immediately.
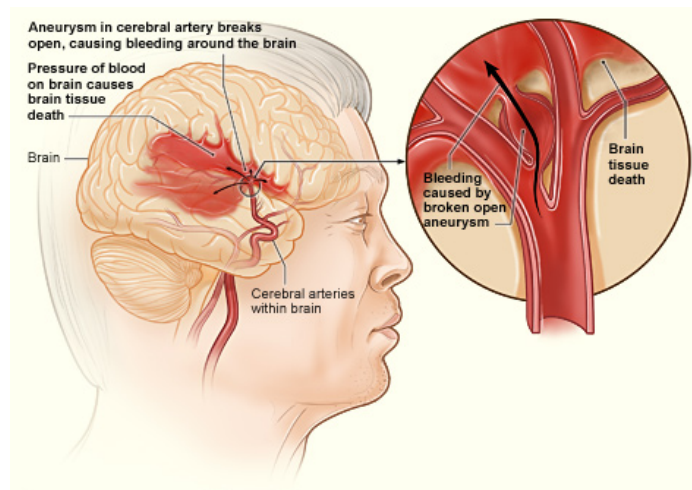
Figure 2.1: The illustration shows how a hemorrhagic stroke can occur in the brain. An aneurysm in a cerebral artery breaks open, which causes bleeding in the brain. The pressure of the blood causes brain tissue death. *The figure is reprinted in unaltered form from Wikimedia Commons, File: Stroke_ischemic.jpg, licensed under CC-PD-Mark.*

2.1.2 Ischemic Stroke

An ischemic stroke might happen because of the loss of blood supply to a part of the brain. There can be four different reasons why this happen:

- Thrombosis: a blood clot inside a blood vessel that obstructs the normal flow of it through the circulatory system [19].

- Embolism: an obstruction due to an embolus, a blockage-causing piece of material inside a blood vessel, coming from another part of the body [20].

- Systemic hypoperfusion: is a state where not enough blood flow goes to the tissues of the body as a result of problems with the circulatory system [21].

- Cerebral venous sinus thrombosis (CVST): it indicates the presence of a blood clot in the dural venous sinuses, which diminish blood in the brain [22].

A visual representation of an ischemic stroke is given in Fig. 2.2; if a blood clot breaks off from plaque buildup in a carotid artery, it can travel into an artery in the brain. The clot can block blood flow for some parts of the brain, causing brain tissue death. Patients affected by an ischemic stroke immediately suffer from various symptoms. According to the site of obstruction, the symptoms can be paresis in the arm, legs or face, difficulties in speaking and understanding the speech, a possible visual loss and unconsciousness [15, 23].
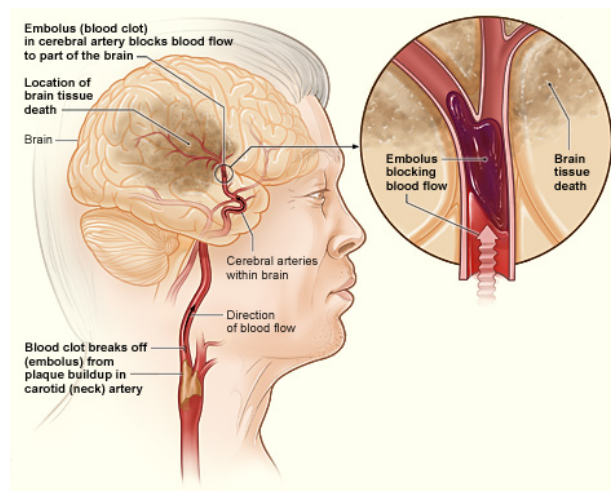
Figure 2.2: The illustration shows how an ischemic stroke can occur in the brain. If a blood clot breaks away from plaque buildup in a carotid (neck) artery, it can travel to and lodge in an artery in the brain. The clot can block blood flow to part of the brain, causing brain tissue death. *The figure is reprinted in unaltered form from Wikimedia Commons, File: Stroke_ischemic.jpg, licensed under CC-PD-Mark.*

The brain region affected by an ischemic stroke can be characterized in two different areas based on the severity of ischemia: penumbra and core. The core denotes the area which is already infarcted or is irrevocably destined to infarct regardless of reperfusion. The penumbra is described by the area of an "*ischemic tissue potentially destined for infarction, but it is not irreversibly injured and the target of any acute therapies*" [24]. Hence, penumbras are areas of the brain with too little blood supply to revive the neuronal function of the tissue, but enough blood supply to prevent the cells from being damaged; they can be saved with the appropriate treatments. Nonetheless, penumbra areas can degenerate and become core areas after a certain amount of time, if blood flow is not restored as soon as possible; that is the main reason why time is critical when doctors are dealing with patients affected by strokes.

2.1.3 Computed Tomography Perfusion

When a person arrives at the hospital for treatment, doctors need to understand the medical situation in the shortest time possible. The European Stroke Organization guidelines recommend: "*brain imaging with Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) in all suspected stroke or transient ischemic attack (TIA) patients*" [25]. The thesis has a focus only on CT images; thus, there will be no explanations for MRI.

Images of cross-sections of the human body are produced from data obtained by measuring the attenuation of x-rays along with a large number of lines through the cross-section [9]. The first commercial CT scanner was available in 1971 [26]. These days, CT scans have a fundamental role in investigating and diagnosing strokes. CT is a rapid and cheap method, plus it has only few contraindications. Acute infarcts are not always detectable, and the body is exposed to a percentage of radiation [27].

The set analyzed during the thesis is formed by images obtained with a CT Perfusion technique (CTP). The dataset from this examination contains more relevant information compared with the information detectable by the eyes alone. This method uses an iodinated contrast agent injected in cubital veins to enhance contrast in the tissue. The same sections of the brain are repeatedly scanned during the passage of a contrast medium from the arteries through the capillaries to the veins and then into the venous sinuses [14]. There is not a fixed number of images per patient; it depends on many factors, such as the age of the patient, the volume of the brain, and the radiation dose. For the Siemens machines used at SUS, it is possible to set a limit for the CT Dose Index Volume (CTDIvol), which represents the dose for a specific scan protocol considering the radiation dose. According to Food and Drugs Administration (FDA) recommendations,

the sensible dose warning for CTDIvol is 1000 mGy [28]; however, the typical dose for CTP brain scan is 500 mGy [29].
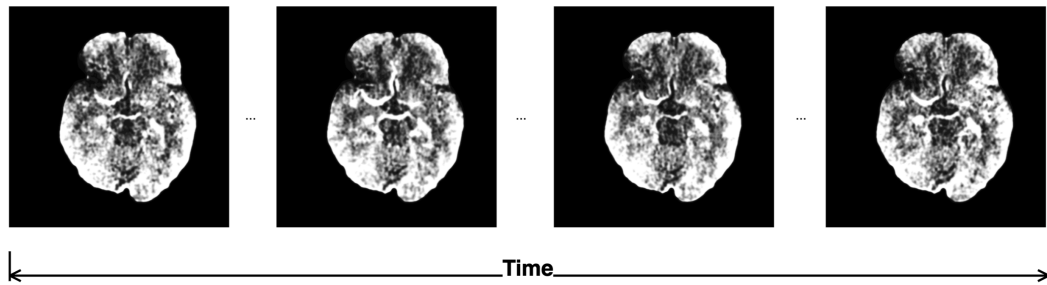
Figure 2.3: Overview of a brain section over time during the injection of the contrast agent. Useful to create the TDC.

Fig. 2.3 displays an overview of a brain section after the pre-processing process, explained in detail in Chap. 3. Each section has a fixed number of different images showing how the injected contrast agent is moving across the brain over a period of time, highlighting the contrast for each pixel. Typically, the period of propagation and detection of the contrast agent is approximately between 30 and 40 seconds. However, scan images of the same area are collected in a 50 seconds window of time to allow a better understanding of the difference between the injection and the start time point of the scanning, since it is an essential feature for the final result [14]. CTP images are taken with a high frequency of time (1 second) during the first part of the scanning, roughly for the first 20 seconds, because of the importance of the first part of the injection, while during the second part of the scan the images are taken with a 3 seconds frequency.

2.1.4 Parametric Maps derived from CTP

Parametric maps derived from CTP series allow doctors to discover rapidly if a patient has an ischemic problem. The parametric maps are formed by evaluating different sections of the brain during the injection of the contrast agent to highlight if there are penumbra areas and core infarction areas in the analyzed brain. The passage of the contrast agent is recorded over time, through time versus contrast concentration curve [30]; this curve is referred to as a time density curve (TDC). The curve represents the intensity of each pixel of the image during the passage of the contrast agent over a period of time [14]. Fig. 2.3 displays an example of a set of images, after performing pre-processing steps, of the same brain section during the injection of the contrast agent, over time. Several TDC equal to the number of pixels in the first image are created.

An example of TDC is presented in Fig. 2.4; the x-axis displays the time elapsed after the start of the scanning (in seconds), the y-axis denotes the relative enhancement level
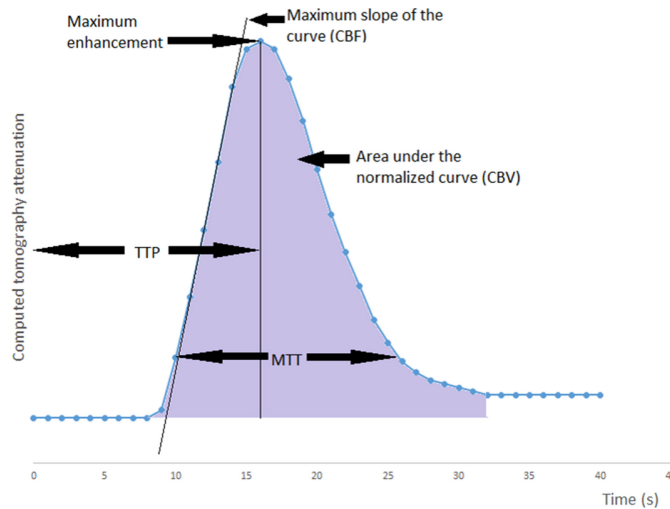
Figure 2.4: The CTP Time Density Curve; it shows the different measurements for the creation of the parametric maps. *The figure is reprinted in unaltered form from: "Radiological imaging in acute ischaemic stroke" [14] under the consensus of the author.*

measured in Hounsfield Units (HU) [31]. Generally, TDC is generated from images acquired in quick succession. Analyzing the TDC in different ways, various measures of perfusion can be calculated for each image pixel. The ratios derived include a variety of color-coded parametric maps; these maps are meant to help visualize an acute stroke [32]; the comparison of these maps helps understand the area affected by a stroke if present.

Time-To-Peak
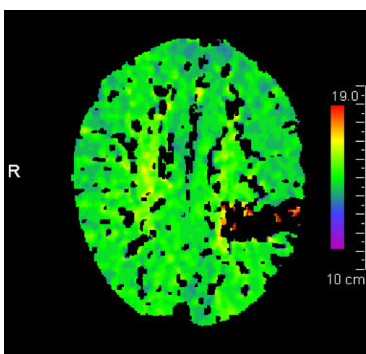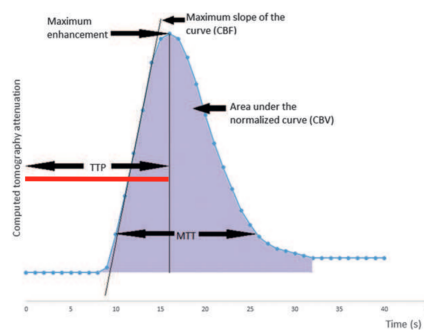
(a) TTP   (b) TTP on the TDC

Figure 2.5: Different visualizations of TTP. *The figures are reprinted from: "Radiological imaging in acute ischaemic stroke" [14] under the consensus of the author.*

Time-To-Peak (TTP) represents the time from the start of the contrast injection to the peak of enhancement in the tissue. It shows immediately if the patient has an ischemic problem. An example of this representation is given in Fig. 2.5. Fig. 2.5(a) shows the TTP of all pixels from a section of the brain during the injection of a contrast agent, while Fig. 2.5(b) is the representation of one pixel over the TDC. The black

area inside the brain (Fig. 2.5(a)) symbolizes an ischemic area, that may contain both penumbra and infarct core: the contrast agent reaches the affected area later compared to a non-damaged part of the brain; hence, it is not adequately shown in the image.

Cerebral Blood Volume

(a) CBV (b) CBV on the TDC
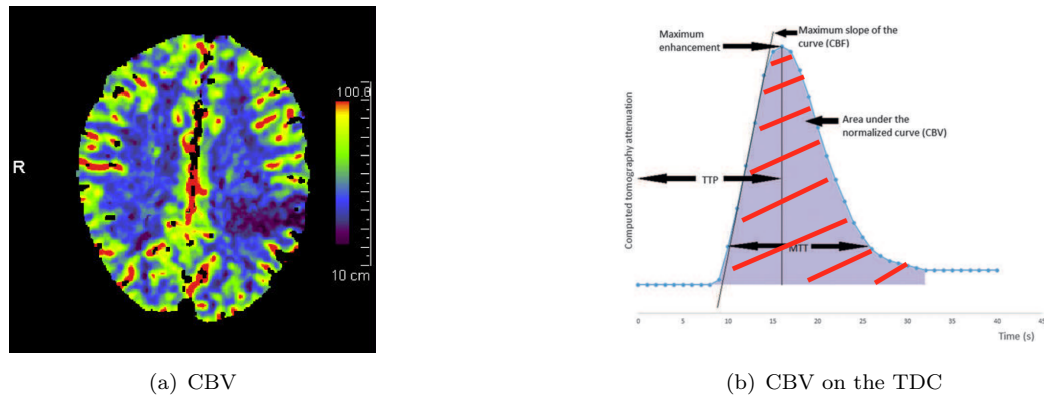
Figure 2.6: Different visualizations of CBV. *The figures are reprinted from: "Radiological imaging in acute ischaemic stroke" [14] under the consensus of the author.*

Cerebral Blood Volume (CBV) is defined as the volume of the blood per unit of brain tissue. It is measured as milliliters of blood per 100g of brain tissue (ml/100g). In non-damaged brain tissue, the CBV should be approximately around 4-5ml/100g; however, if the blood volume results below 2.5 ml/100g, it indicates infarcted tissue [33]. While it is shallow in the core area, there is a compensatory increase in the penumbra zone. The CBV is represented by the integral of the TDC, which produces the area below the TDC. Fig. 2.6 shows both its representation: all its pixels of the brain section (2.6(a)) and its visualization over the TDC (2.6(b)). The violet area inside the brain in Fig. 2.6(a) defines a possible section for an infarcted area.

Cerebral Blood Flow

Cerebral Blood Flow (CBF) represents the volume of blood flow per unit of brain tissue per minute. It is commonly measured in milliliters of blood per minute per 100g of brain tissue (ml/100g/min). Normally, in a patient without any ischemic problem, the value of CBF is around 50-60ml/100g/min [34]. However, if the brain has an ischemic problem, the flow is reduced in both penumbra and core areas, as it is possible to observe from Fig. 2.7. If CBF is reduced and CBV is normal or slightly reduced, the tissue ischemia is likely to be reversible; if CBF and CBV are markedly reduced or if TTP is not measurable, the tissue may be infarcted [35].

(a) CBF

(b) CBF on the TDC
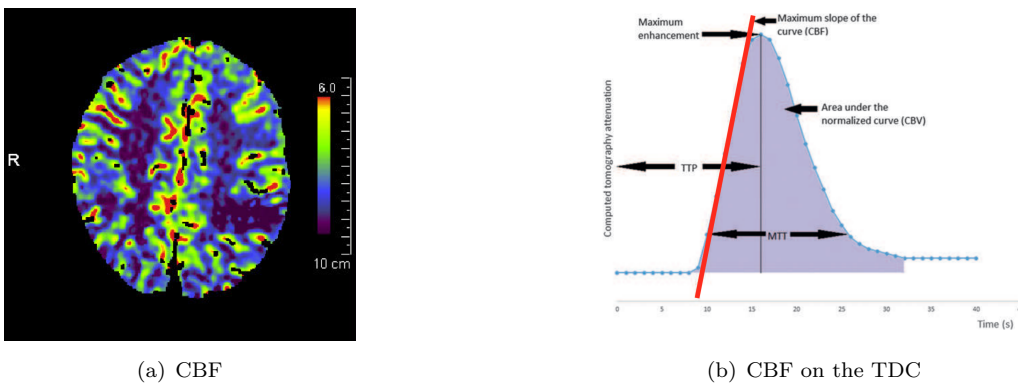
Figure 2.7: Different visualizations of CBF. *The figures are reprinted from: "Radiological imaging in acute ischaemic stroke" [14] under the consensous of the author.*

Mean Transit Time

Mean Transit Time (MTT) is the time taken by the contrast agent to pass through the brain tissue. It is described as the average transit time of blood through a brain region, estimated in seconds. Fig 2.8(a) shows the representation of MTT over the TDC. If the MTT is raised as compared to the healthy side, ischemia/infarction is present [35].

(a) MTT on the TDC

(b) Parametric map of TMax.
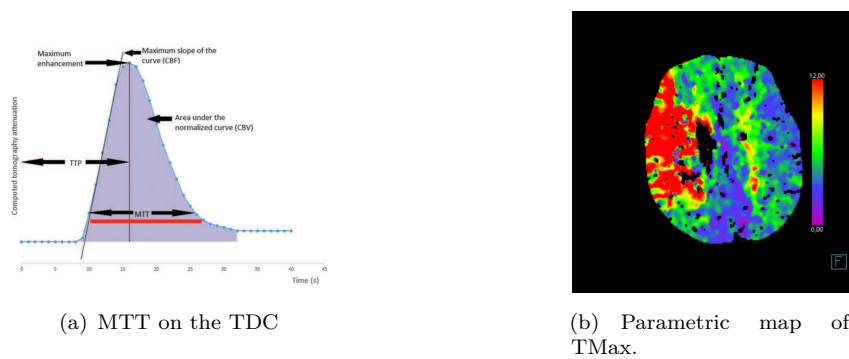
Figure 2.8: MTT on the TDC and a parametric map of TMax. *The figures are reprinted from: "Radiological imaging in acute ischaemic stroke" [14] under the consensous of the author.*

TMax

Time-to-maximum (TMax) displays the time taken by a contrast agent to reach and traverse areas of the brain. Together with the TTP, it is a good measure of contrast arrival time to the tissue. "*The tissue time-enhancement curves are deconvolved with the arterial input function (AIF) by using the single value decomposition (SVD) method to produce an impulse residue function (IRF)*" as stated by [36]. TMax is calculated from

the TTP of the IRF curve, where TMax $= 0$ reflects normal blood supply in normal tissue without delay. Fig 2.8(b) shows an example of a TMax map.

2.2   Technical Background

The section gives a general overview of Deep Neural Networks (DNN) and the theory behind it; moreover, a brief explanation of various methods used for this thesis, based on Convolutional Neural Network (CNN), is presented. However, first, the section tries to answer a fundamental question related to the thesis itself:

- Why are Machine Learning and Deep Learning so popular?
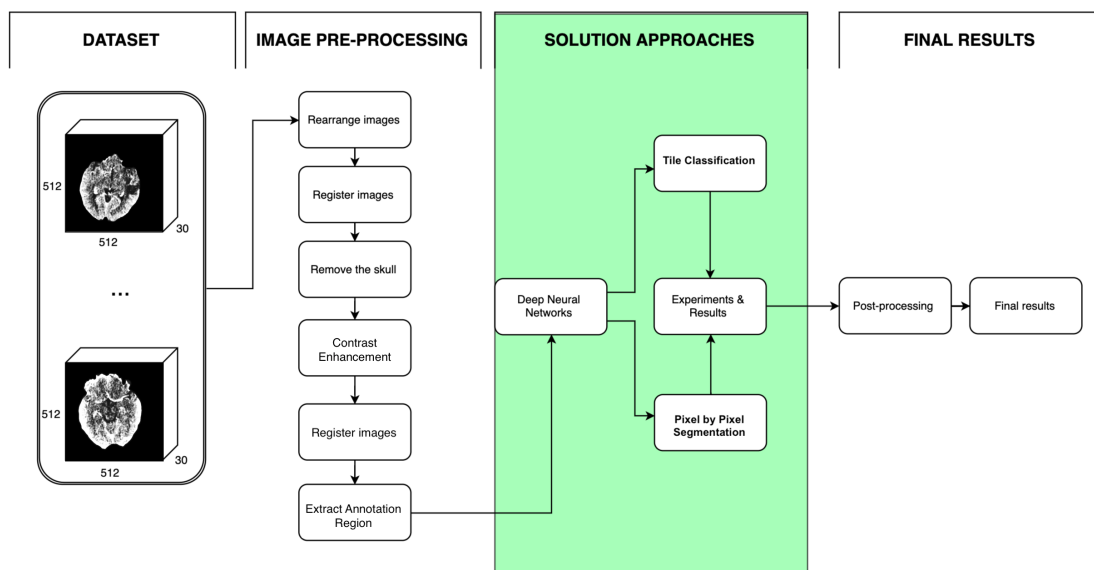
Figure 2.9: Focus of the technical background section.

In modern days, machine learning (ML) and deep learning (DL) technologies are becoming an essential branch in Computer Science. From web search algorithms implemented by Google [37], passing through DL applications for helping doctors during medical diagnosis to finally arrive at AlphaGo, the first ML software that was capable of defeating Lee Sedol, the world champion of Go, a popular board game [38]. ML and DL software are rapidly influencing and changing the day-to-day life in a way that was not even imaginable 30 years ago. However, conventional ML techniques were limited and dependent on the choice of features to extract from data and to use these features in a ML framework. Lately, a new field in the ML community was introduced: representation learning (RL), "*a learning representations of the data that make it easier to extract useful information when building classifiers or other predictors*" [39]. RL allows a system to automatically discover the representations needed for feature detection or classification from raw data.

2.2.1 Linear Regression

To understand Neural Networks (NN), the first step is to introduce the problem of regression. Linear regression is a linear combination of input components, as defined in equation 2.1, given a data set $\mathbf{x} = [x_1, \ldots, x_n]$, the task is to predict the real valued target $\mathbf{y}$, where $\mathbf{w} = [w_1, \ldots, w_n]$ is the weight vector and $\mathbf{b}$ is the bias:

$$\mathbf{y} = \mathbf{b} + x_1 w_1 + \cdots + x_n w_n = \mathbf{b} + \sum_{i=1}^{n} x_i w_i \tag{2.1}$$

2.2.2 Neural Network

The biological operations of specialized cells, the neurons, inspired the term Neural Network (NN). A neuron is an electrically excitable cell that has a large number of inputs received from other neurons via dedicated connections called synapses. Furthermore, some connections may be "strengthened" or weighted in a different way than other connections. A neuron can produce different outputs based on its activity.

The equivalent of a neuron in a NN is called "artificial neuron". It receives a set of weighted inputs, processes their sum with a specific activation function θ, and forward that result to the next artificial neuron in the network. Equation 2.1 is essential for a NN because it gives a general overview of the addressed problem. An artificial neuron receives a specific input x_i from the previous neuron in the network; x_i is weighted by $\mathbf{w_i}$ which determines how to response to the data plus a biased value $\mathbf{b_i}$. Equation 2.2 defines the output of an artificial neuron, where θ is the activation function that receives in input the sum of the weighted inputs.

$$\hat{\mathbf{y}} = \theta \left(\sum_i \mathbf{w_i} x_i + \mathbf{b_i} \right) \tag{2.2}$$

Fig. 2.10(a) shows a visual representation of a simple NN. Three different input x_i are feeding the artificial neuron output $\mathbf{y}$ with three different weight values w_i. The artificial neuron is located in one layer. Fig. 2.10(b) reproduces a NN with three layers connected with each other. This representation is called the Feed Forward Neural Network (FFNN). The FFNN contains one hidden layer and an output layer. The input layer consists of six artificial neurons, and the output layer has four artificial neurons. No calculation is required during the load of the input layer; thus building the FFNN would consist of implementing two computational layers. The inputs in the hidden layer are fully connected to the artificial neurons in the input layer. Moreover, a full connection is

presented between the artificial neurons of the hidden layer and the artificial neurons in the output layer.
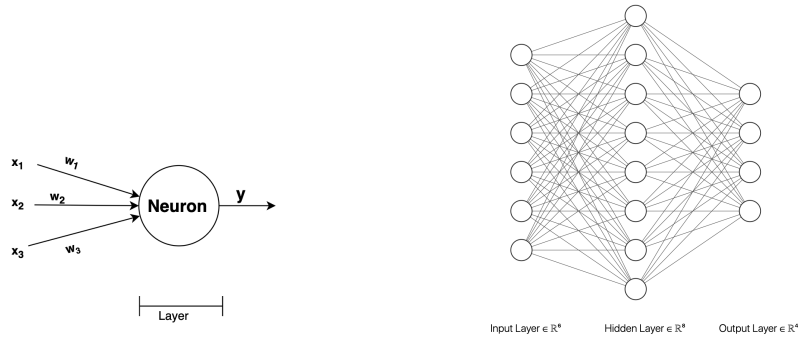
(a) Example of NN with a single layer.

(b) Example of NN with multiple layers (FFNN).

Figure 2.10: Overview of NNs with different layers.

Activation Functions

The choice of an activation function in NN has a significant effect on the training dynamics and task performance [40]. During the evaluation of the developed approaches, two different activation functions were used:

- Rectified Linear Unit (ReLU) [13].

- Sigmoid function [10].

ReLU is one of the most popular choices for an activation function because it is straightforward to implement, and it shows good results during the training [40]. ReLU is very important due to its speed during training. Also, ReLU provides a straightforward nonlinear transformation. The function is defined as the maximum between 0 and a given element x.

$$\text{ReLU}(x) = \max(0, x)$$
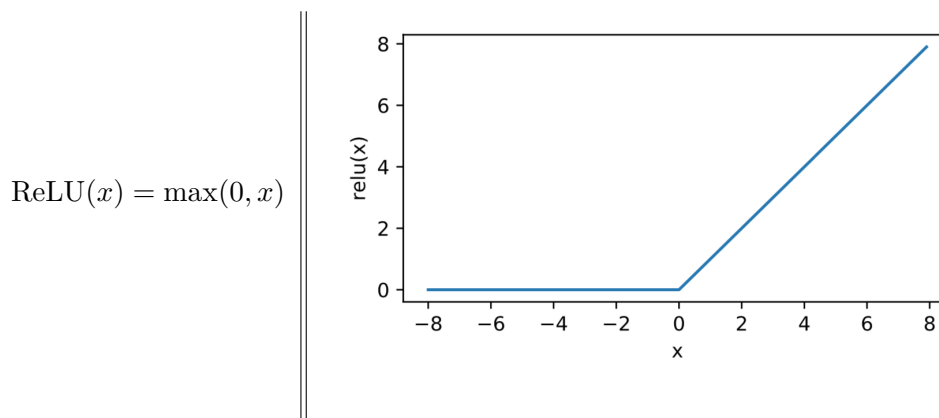
Differently, the sigmoid function transforms its inputs, values in $\mathbb{R}$, into the interval $[0, 1]$, which is fundamental for some particular models:

$$\text{sigmoid}(x) = \frac{1}{1+\exp(-x)}$$
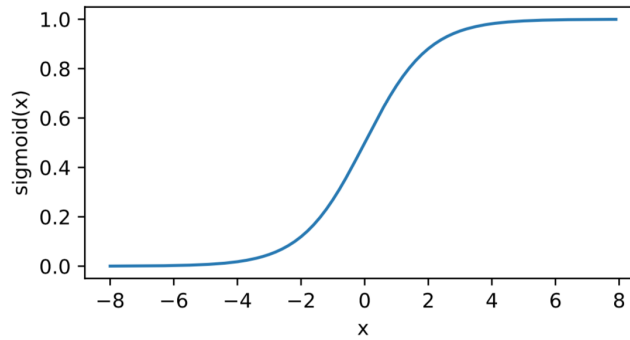
Loss Functions

To predict its output, NN trains its data through Equation 2.2. The terminology "training data" is used to determine the parameters of a model that minimize the error between the predicted output and the real output. The usual way to measure the error between the two outputs is to use a loss function. The architectures implemented during the thesis work (Chap. 4 and Chap. 5), are using two different loss functions based on the output that is generating:

- Categorical crossentropy [41];

- Dice loss [42].

Categorical crossentropy The categorical crossentropy loss function is used for the approaches described in Chap. 4 based on Convolutional Neural Network (CNN). This function is mostly used to train a CNN to output a probability a specific number of classes greater than two (C). It measures the probability error for classification tasks where the classes are mutually exclusive. It is also called Softmax Loss because it can be described as a Softmax activation plus a Cross-Entropy loss.

$$\text{cat\_CE} = -\sum_{i}^{C} g_i \, \text{softmax}(\log(s_i)) \tag{2.3}$$

where g_i is the ground truth while s_i is the CNN score for each class i in C

The gradient of the cat_CE is:

$$\nabla \text{cat\_CE} = \frac{\partial}{\partial s_i}(\text{cat\_CE}) = \left(\frac{e^{s_i}}{\sum_{j}^{C} e^{s_j}} - 1 \right) \tag{2.4}$$

Dice loss it is a function used to measure the similarity of two samples; it is based on the dice coefficient. This loss function was implemented for the approach described in Chap. 5, which is based on a U-Net method because it is commonly used in image segmentation, to compare predicted output against masks in medical applications [43]. The output of this method is an image; thus, the function compares a different portion of the volume of every image. The formula for the dice coefficient D between two binary volumes can be written as:

$$D = \frac{2 \sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2} \tag{2.5}$$

where the sums run over the N voxels, of the predicted segmentation volume p_i and the ground truth volume g_i [42]. The dice loss is defined as:

$$dice\_loss = 1 - D \tag{2.6}$$

The gradient of the dice coefficient is:

$$\nabla D = \frac{\partial D}{\partial p_j} = \left[\frac{g_j \left(\sum_i^N p_i^2 + \sum_i^N g_i^2 \right) - 2 p_j \left(\sum_i^N p_i g_i \right)}{\left(\sum_i^N p_i^2 + \sum_i^N g_i^2 \right)^2} \right] \tag{2.7}$$

Optimization algorithms

Optimization algorithms help the model to minimize the loss function in a neural network. The goal of any optimization function is to find the weight vector $\mathbf{w}$ and the bias term $\mathbf{b}$, given a collection of data $\mathbf{x}$ and a vector containing the corresponding target values $\mathbf{y}$, that associate each data x_i with an approximation $\hat{y}_i$ of its corresponding label y_i with a minimum error in the approximation. A famous example of an optimization algorithm is backpropagation. During the implementation of different architectures, it was used the stochastic gradient descent (SGD) function and the adaptive moment estimation (Adam) to optimize the models.

Backpropagation It is a mechanism to calculate the gradient of the loss function involved; it is essential in the calculation of the weights involved in the network [44]. It is used to adjust the weights during the training of the model in order to minimize the error of the output.

The importance of backpropagation was discovered after the release of an article in 1986 [45]. The paper describes different neural networks with a backpropagation implementation that shows a faster result compared to the other approaches; more details of this algorithm on [44].

Stochastic Gradient Descent   The stochastic gradient descent is an iterative method for optimizing a differentiable objective function, a stochastic approximation of gradient descent optimization. SGD is famous for large scale optimization but has slow convergence asymptotically due to the inherent variance [46]. The equation of SGD is used to minimize an objective function is given in the form of a sum:

$$Q(w) = \frac{1}{n} \sum_{i=1}^{n} Q_i(w) \tag{2.8}$$

where the parameter w that minimizes $Q(w)$ is to be estimated; each function Q_i is associated with the ith observation in the data set (used for training).

Adaptive Moment Estimation   Adaptive Moment Estimation (Adam) is an algorithm for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments [47]. Empirical results demonstrate that Adam works well in practice and compares favorably to other stochastic optimization methods, thus it is used in one of the models to check its efficiency.

2.2.3   Deep Learning

The term *deep learning* (DL) refers to a class of ML algorithms that uses representation learning methods with multiple levels of representation. These levels are obtained by composing nonlinear but straightforward modules that transform the description at one level into an image at a higher, slightly more abstract level [44]. All layers manipulate the output from the previous layer, and they use it as input; plus, the learning method is divided into two main approaches: supervised learning, like classification and regression, and unsupervised learning, like pattern analysis and clustering.

Supervised Learning

The supervised learning approach is used to predict a target given some input data [10]. hence, the goal is to create a model f_γ that maps an input $\mathbf{x}$ to a prediction $f_\gamma(\mathbf{x})$. The target, also called label, is generally denoted with $\mathbf{y}$. The approach implements a model

$f_\gamma(\mathbf{x})$ with a supervision set of targeted input $(\mathbf{x_i}, \mathbf{y_i})$, where each input $\mathbf{x_i}$ is matched up against its correct label. The supervision learning is used to select the best parameter γ. This approach is used and described in details in Sec. 4.3 and Sec. 5.4.

Unsupervised Learning

The unsupervised learning refers to a learning method without a teacher or truth labels, also known as self-organization and a way of modeling the probability density of inputs [48]. The method is not used in this work, thus it will be explained further. For a more detailed explanation, it is possible to consult [48].

2.2.4   Deep Neural Networks

While a standard NN consists of many simple connected artificial neurons, each producing a sequence of real-valued activations, *"a Deep Neural Network (DNN) is an artificial neural network (ANN), which has multiple, often many, layers between the input and output layers"*, as stated in a fundamental article about DNN [49]. Various implementations of DNN have been creating promising results in numerous problems, like image recognition [50, 51], speech recognition [52] or even predicting the effects of mutations in non-coding DNA on gene expression and disease [53, 54].

The final goal of a DNN is to find the weight vector $\mathbf{W}$ and bias term $\mathbf{b}$, given a collection of input $\mathbf{x}$ and a vector containing the corresponding target values $\mathbf{y}$, that associate each input $\mathbf{x_i}$ with a prediction $f_\gamma(\mathbf{x_i})$ of its corresponding target $\mathbf{y_i}$. Every layer of a DNN, except the input layer, is produced by the output of the previous layer. In a linear algebra annotation, each layer can be defined as:

$$o = \mathbf{W}\mathbf{x} + \mathbf{b} \tag{2.9}$$

where $\mathbf{W}$ is the weight vector, $\mathbf{x}$ is the input of the layer, as defined in Sec. 2.2.3, and $\mathbf{b}$ represents the bias vector.

Fig 2.11 displays a possible visual aspect of the structure of a DNN with four hidden layers and one output layer. This representation is called Multi-Layer Perceptron (MLP). The input layer contains six artificial neurons, and the output layer has two artificial neurons. As for a standard NN, no calculation is needed for the input layer; only five computational layers are necessary to build this network. Artificial neurons in the input layer are fully connected to the inputs in the hidden layer. In the same way, the artificial neurons in all the hidden layer are fully connected to the artificial neurons of

Input Layer ∈ ℝ⁶   Hidden Layer ∈ ℝ⁸   Hidden Layer ∈ ℝ⁸   Hidden Layer ∈ ℝ⁸   Hidden Layer ∈ ℝ⁸   Output Layer ∈ ℝ⁴
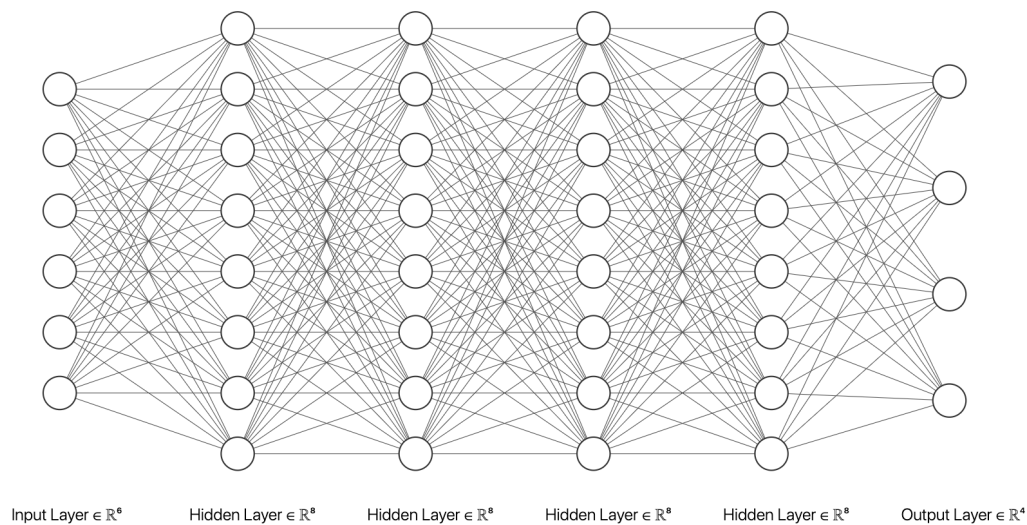
Figure 2.11: Example of Deep Neural Network (MLP).

the successive hidden layer. Additionally, the artificial neurons in the output layer are fully connected to the artificial neurons of the last hidden layer.

2.2.5 Convolutional Neuron Network

For image processing and computer vision problems, the training of MLP can be hard because of the dense connections between artificial neurons did not allow them to scale efficiently. A Convolutional Neural Network (CNN) solves this complication because it convolves each input with a detector (kernel), and thus it is sensitive to the same feature everywhere.

Biological processes inspired CNNs because the connectivity pattern between neurons resembles the organization of the animal visual cortex [11]. The visual cortex contains a vast number of cells responsible for identifying light in overlapping sub-regions of the visual field, the receptive fields. These cells behave as filters over the input; the more complex cells have larger receptive fields [12].

In the last few years, a large number of researches in computer vision and pattern recognition have highlighted the capabilities of CNN, achieving state-of-the-art performances on challenging tasks such as classification, segmentation and object detection. This success has been attributed to its ability to learn a hierarchical representation of raw input data, without relying on handcrafted features. Additionally, several papers have shown that it can also deliver outstanding performance on challenging visual classification assignments [55]. Furthermore, according to [56], *"the ability of multilayer backpropagation networks*
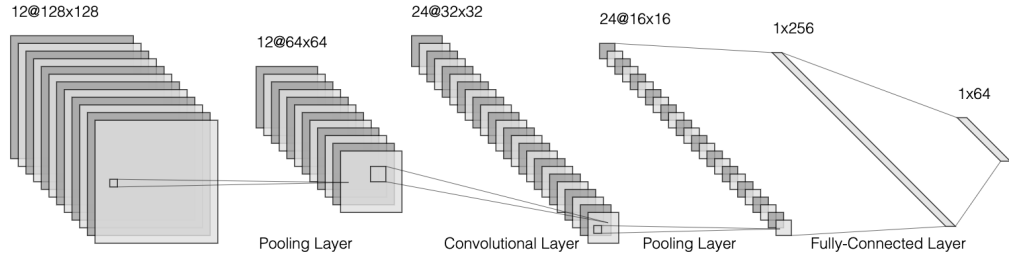
Figure 2.12: Typical block diagram of CNN.

to learn complex, high-dimensional, nonlinear mappings from extensive collections of examples makes them obvious candidates for image recognition or speech recognition tasks".

As a standard NN, CNN consists of an input layer plus an output layer linked by a non-fixed number of hidden layers (Fig. 2.12). Usually, hidden layers are represented by convolutional layers, activation functions, pooling layers, fully-connected layers and normalization layers [57].

2.2.6 Layers of CNN

This section explores different typologies of layers typically used during the creation of a CNN architecture; all these layers were implemented in the distinct architectures described in Chap. 4 and in Chap. 5.

Convolutional Layers

CNNs base their foundation on convolutional layers. This layer consists of a set of small learnable filters. The operation of this layer is expressed by convolving each filter over the entire input and computing a dot products between the input at any position and the entries of the filter.

$$g(x,y) = \omega * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} \omega(s,t) f(x-s, y-t) \tag{2.10}$$

Equation 2.10 defines a convolution operation where $g(x, y)$ is the filtered image, $f(x, y)$ represents the original image and the filter kernel is ω . Each element of the filter kernel is examined between $-a \leq s \leq a$ and $-b \leq t \leq b$.

Transposed Convolutional Layers

A transposed convolutional layer, also called deconvolutional layer, behaves in the opposite way of a standard convolutional layer. If an input X_i is parsed through a convolutional layer generates an output Y_i, if the resulting output Y_i is given to a transposed convolutional layer, which has the same structure and parameters of the convolutional layer used to create Y_i, it generates as a final result the initial input X_i [58]. This type of layer is fundamental to build the architecture described in Chap. 5 because it is necessary to use a transformation or an operation that goes in the opposite direction of a regular convolution.

Pooling Layers

Like the convolutional layer, a pooling layer computes the output for each element in a fixed-shape window of input data. The pooling layer decreases the resolution of the window to prevent misleading noise and distortion pixels. Max pooling and average pooling are the two existing methods.
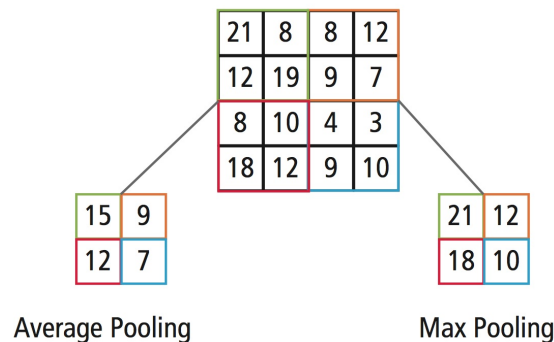
Figure 2.13: Pictorial representation of max pooling and average pooling. *The figure is reprinted in unaltered form from: "Using convolutional neural networks for image recognition" [12].*

Fig. 2.13 shows an example of these two techniques: the input matrix has a size 4x4. According to [12]: *"for 2x2 subsampling, the 4x4 image is split into four non-overlapping matrices of size 2x2. In the case of max pooling, the maximum value of the four values in the 2x2 matrix is the output. Otherwise, in the case of average pooling, the average of the four values is the output".*

Fully-Connected Layers

Usually, fully-connected layers are used as the last layer in a CNN, after a certain amount of convolutional and pooling layers. The layer connects all its artificial neurons to every artificial neuron in the other layer. The result is a flat matrix useful to classify images.

Normalization Layers

A normalization layer uses the distribution of the summed input to a neuron over a mini-batch of training cases to calculate a mean and variance which are then adopted to normalize the summed data to that neuron on each training case, which can significantly reduce the training time in feed-forward neural networks [59]. Also, the normalization layer produces, for each layer, a system to learn by itself a little bit more independently of other layers.

Dropout Layers

A dropout layer is used to reduce overfitting in NN by preventing complex co-adaptations on training data. According to the Oxford dictionary, the overfitting is *"the production of an analysis that corresponds too closely or exactly to a particular set of data, and may, therefore, fail to fit additional data or predict future observations reliably"*.

2.2.7 U-Net

U-Net is a CNN architecture for fast and precise segmentation of images. It was first described and implemented by a group of researchers from the University of Freiburg (Germany) in 2015 [60]. As stated in their main article: *"it relies on the heavy use of data augmentation to work with the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization"* [60]. A visual example of the possible architecture of the network is shown in Fig. 2.14.

This innovative structure gives the possibility to give in input an image or a series of images and to receive in output a full label image. This architecture achieved very interesting and promising results on different biomedical segmentation applications. The pixel by pixel segmentation approach (Chap. 5) uses this structure to receive in output a brain section image containing the different infarcted region inside the brain.
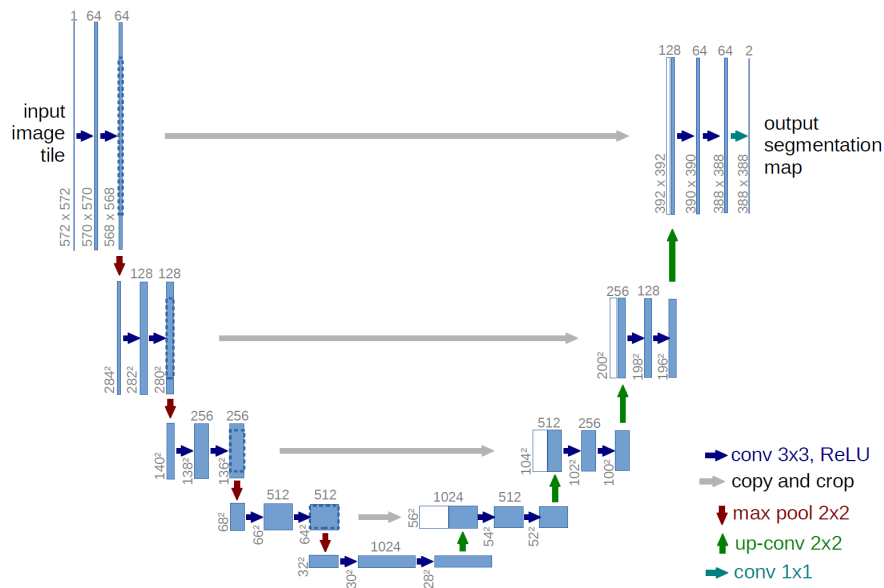
Figure 2.14: U-Net architecture. *The figure is reprinted in unaltered form from [60].*

U-Net Layers

The left part of the U-Net architecture is quite similar to any standard CNN's structure. The important modification happens on the right side, which is called "the expansive path". As stated in the original article:

> "*every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution ('up-convolution') that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer, a 1x1 convolution is used to map each 64-component feature vector to the desired number of classes. In total the network has 23 convolutional layers*" [60].

2.2.8 Statistical Information

During training and testing evaluations of the various models proposed, different statistical information were calculated and analyzed based on the resulting confusion matrix of each predicted brain section. Table 2.1 shows a representation of a 2x2 confusion matrix for one class. A confusion matrix is represented as a table with two columns and two rows that presents the number of false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). It helps to visualize and to calculate the performance

of any supervised learning algorithm highlighting how good a model predicts different outputs.

Predicted Class	Actual Class	
	True Positive (TP)	False Positive (FP)
	False Negative (FN)	True Negative (TN)

Table 2.1: Representation of 2x2 confusion matrix

The evaluation of each method is based on various statistical information; each equation lies in the values of the corresponding confusion matrix:

- **Accuracy**: a description of systematic errors, a measure of statistical bias; it is calculated as

$$\text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2.11}$$

- **Recall**: it measures the proportion of actual positives that are correctly identified, also called True Positive Rate (TPR). The equation to calculate it is:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2.12}$$

- **Precision**: it is the fraction of relevant instances among all instances. It is also called Positive Precision Value (PPV) and it's defined as:

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2.13}$$

- **Specificity**: also called True Negative Rate (TNR); it measures the ratio of actual negatives that are correctly identified. It is calculated as follow:

$$\text{TPR} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{2.14}$$

- **Negative Predictive Value** (NPV): it is defined as:

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}} \tag{2.15}$$

The values of the NPV oscillate between 0 (worst possible value) and 1 (best possible option).

- **False Negative Rate** (FNR): it's the complementary of the recall measurement; it is calculated as:

$$\text{FNR} = \frac{\text{FN}}{\text{FN} + \text{TP}} \tag{2.16}$$

- **False Positive Rate** (FPR): it measures the ratio between the true positive values and the the number of real negative cases in the data with the following equation:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{2.17}$$

- **False Discovery Rate** (FDR): it is the complementary of the precision. The equation to calculate it is:

$$\text{FDR} = 1 - \text{PPV} = \frac{\text{FP}}{\text{FP} + \text{TP}} \tag{2.18}$$

- **False Omission Rate** (FOR): it is the complementary of the negative prediction value; the formula to measure it is:

$$\text{FOR} = -\text{NPV} = \frac{\text{FN}}{\text{FN} + \text{TN}} \tag{2.19}$$

- **F1 score**: the formula takes in consideration both the precision and the recall of the test to compute the final score; it is a measure of a test's accuracy. It is calculated as:

$$\text{F1 score} = 2\frac{\text{PPV} * \text{TPR}}{\text{PPV} + \text{TPR}} \tag{2.20}$$

2.2.9   Statistical Metrics

Moreover, it was also calculated the Jaccard index, the mean average precision (mAP), and the mean squared error (MSE) for the images created. These values are generated to produce a significant overview of the outcomes and to have a better judgment of the various information.

The Jaccard index is used to gauge the diversity and similarity of two sample sets A, B. The equation to calculate it is:

$$\text{Jaccard}(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2.21}$$

The mAP is another statistical metric that returns a percentage of how accurate is the similarity between two sets. It calculated the mean of the average precision (AP) of each output image; the AP equation is defined using the precision and recall values illustrated before:

$$\text{AP} = \sum_n (\text{TPR}_n - \text{TPR}_{n-1})\text{PPV}_n \tag{2.22}$$

where TPR_n and PPV_n are the recall and precision at the nth threshold [61]. Additionally, the mAP is given as the following equation:

$$\text{mAP} = \frac{\sum_k \text{AP}_k}{K} \tag{2.23}$$

where K is the number of brain section per patient.

Finally, the MSE is a statistical measurement to calculate the average of the squares of the errors; it measures the quality of the produced outputs based on the variance and bias compared with the real result. It is calculated as follow:

$$\text{MSE} = \frac{\sum_{i=1}^{n}(x_i - \widehat{x}_i)^2}{n} \tag{2.24}$$

where n is the number of prediction, x_i and $\widehat{x}_i$ are respectively the real and the produced output.

2.2.10   K-Fold Cross-Validation

The last statistical analysis implemented during the training and testing of the various architectures is called K-Fold Cross Validation [62]. This technique is used to generalize how accurately a predictive model will perform in practice. The best way to properly understand the general idea behind K-Fold Cross-Validation, is to described it using as an example the dataset of 11 patients used during the thesis. For K times, where K is equal to the number of patients, a random patient is chosen without any duplication during the selection of a patient. Subsequently, the dataset is first split into two different subsets (folds): the input subset, which contains images of all the patients except for images of the selected patient, and the testing subset, which contains the remaining images, the ones related with the selected patient. Afterward, the input subset is randomly split again: the resulting subsets are the training subset and the validation subset, which consists of a tenth of the input subset randomly selected for the validation step during the training of the model. The validation set is equivalent to a 10% of the training dataset. It is used to find the "optimal" number of hidden units or determine a stopping point for the backpropagation algorithm; it is fundamental during training epochs of a model.

3

Dataset & Image pre-processing
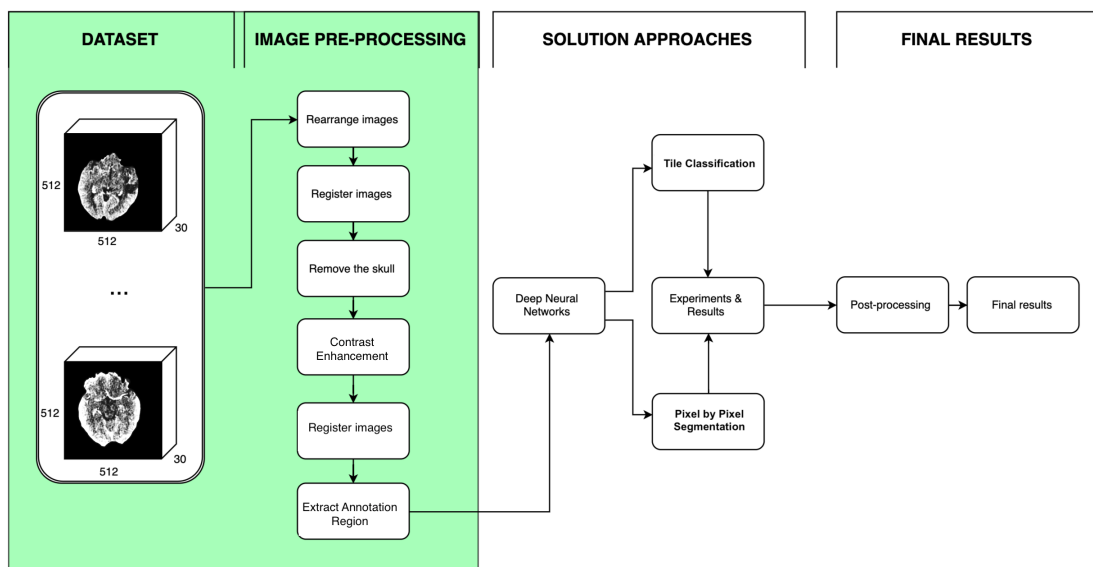
Figure 3.1: Focus of chapter three.

3.1 Dataset
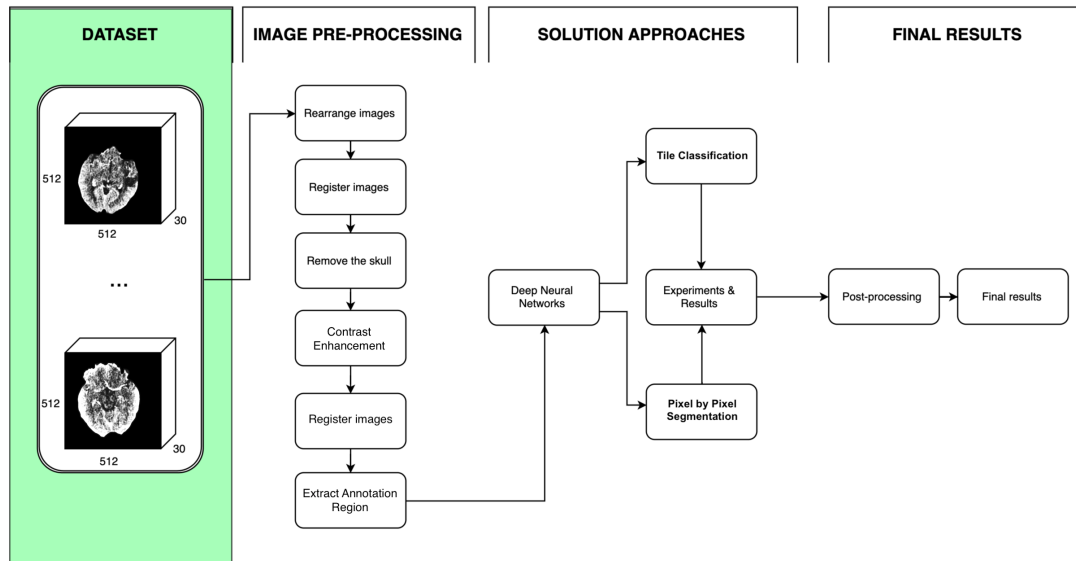
Figure 3.2: Focus of the Dataset section.

The dataset, used during the thesis work, consists of 4800 CTP images and 160 parametric maps of 11 anonymous patients obtained at SUS between 2014 and 2015. Each parametric map contains information from 30 different images over time. All the images of each patient generate a 4D (3D + time) examination of a CT Perfusion. For each patient, there are CTP images as well as parametric color-coded maps that describe the blood perfusion in the brain. A summary of the information of patients is given in Table 3.1. All patients have suffered significant strokes; therefore they are excellent candidates to analyze during the thesis because these type of strokes have different categories of ischemia within the ischemic area, which it is essential for the understanding of the disease.

ID	Age	Sex	Maps	Hemisphere affected	Comments
1	64	Male	19	Right	
2	56	Female	13	Left	Old infarct in the right side.
3	67	Female	13	Right	Old infarct in the right side.
4	69	Male	13	Right	Old infarct in the right side.
5	65	Male	13	Right	
6	77	Female	13	Left	Bolus not optimal.
7	87	Female	13	Left	
8	70	Male	13	Right	
9	63	Female	22	Left	
10	67	Female	14	Left	
11	83	Male	14	Right	

Table 3.1: Information of the 11 patients analyzed.

All images for the patients in this thesis were created and saved during injection of 40 ml iodine-containing contrast agent (Omnipaque 350 mg/ml) and 40 ml isotonic saline in a cubital vein with a flow rate of 6 ml/s; the scan delay was four seconds. [63]

3.1.1 DICOM Standard

Images extrapolated from a CTP scan are saved as DICOM files, the international standard to transmit, store, retrieve, print, process, and display medical imaging information [64]. The structure of a DICOM data object contains several attributes: name, ID, time of the acquisition, and also one unique attribute containing the image pixel data. The DICOM Standard has a disadvantage related to data entry: "*A significant disadvantage of the DICOM Standard is the possibility of entering probably too many optional fields. This disadvantage is mostly showing in the inconsistency of filling all the areas with the data. Some image objects are often incomplete because some variables are left blank, and some are filled with incorrect data*" [65]. For the sake of the privacy, personal information of each patient was made anonymous.

A lot of free libraries and software are available to display a DICOM data object; during this project, the MATLAB library was used to access data from the DICOM files because the pre-processing phase is based on the work of two former students of the University of Stavanger [63, 66]. Additionally, the extrapolation of the manual annotation regions was performed on different brain sections to have a "gold standard" or ground truth for the output using various Python libraries.

3.1.2 General Overview of the Dataset

The 2D images of the image-sets are 512x512 pixels, corresponding to a resolution of 120 pixels/cm; pixels have a bit depth of 12 bits per pixel. Each image's location has a 5mm distance from each other. The CTP examinations that are the foundation of this thesis, consist of 390, 420 or 660 images, resulting in respectively 30 time-series of 13, 14 or 22 parametric maps.

Fig. 3.3 shows a general representation of the dataset of DICOM images of one patient. The time-series is represented in the vertical axis while the volume-series is indicated in the horizontal axis. Each 3D volume on the horizontal axis highlights a section of the head, displaying both the skull and the brain. Steps of pre-processing are performed to remove the skull, extrapolate the brain from these CTP images, and rearrange them. Each time-series of images is realized from the same section and position of the brain over a period of time; this information is saved in one of the DICOM standard fields.
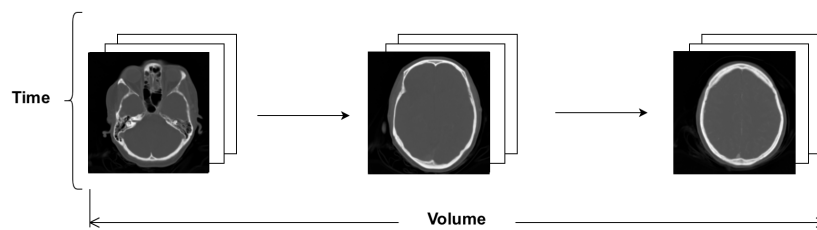
Figure 3.3: Example of images of one patient.

3.1.3 Annotated Regions

In addition to the parametric maps, a set of manually annotated images was given to represent the ground truth or final results for the CTP images. Each manually annotated image is associated with a specific brain section. The annotations are corresponding to the various affected regions of a brain's section due to an ischemic stroke, such as penumbra and core. The supervised doctor of the thesis elaborates the annotation after a study of the different parametric maps of the CTP scans for each patient.

Each annotated image contains zero, one or multiple green perimeters: an external border coincides with a possible penumbra area whereas a perimeter inside another one represents a potential core area of an ischemic stroke. Fig. 3.4 shows an example of one of the manually annotated images correlates with a parametric map; the image contains a vast penumbra region in the right hemisphere and a small core area, approximately in the center of the brain, inside the penumbra region.
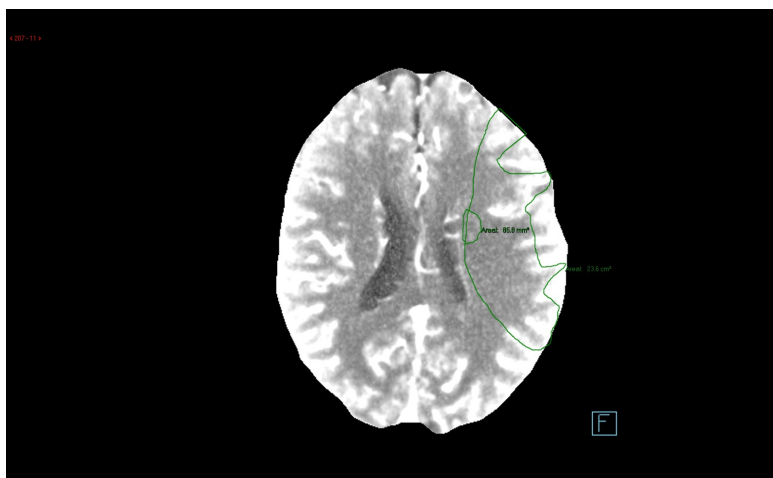
Figure 3.4: Example of manually annotated brain section.

3.2 Image pre-processing
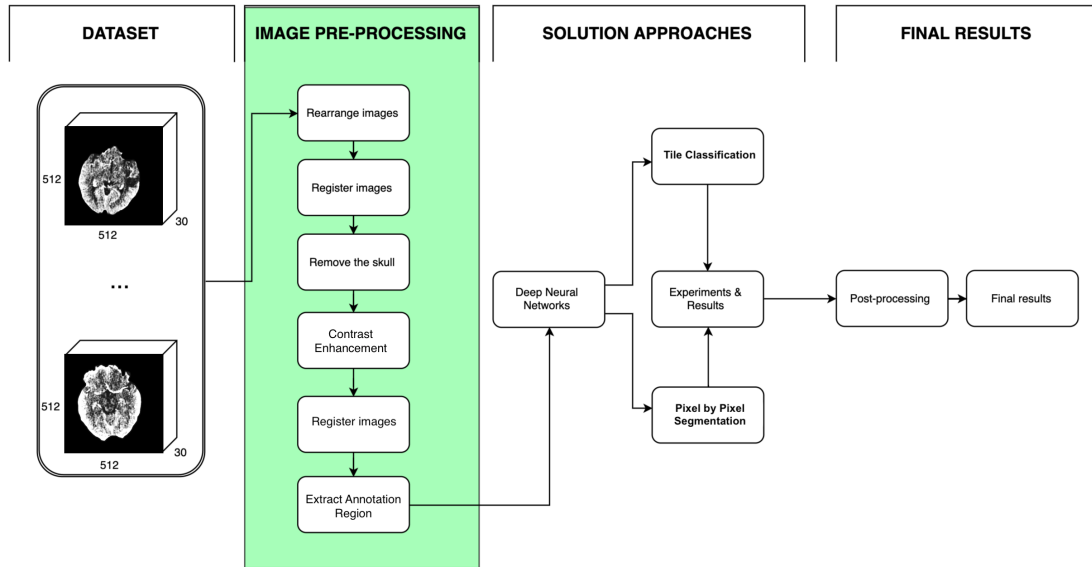
Figure 3.5: Focus of the image pre-processing section.

This section presents and discusses all the steps involved during the pre-processing of the images. Most of the methods used during the pre-processing were based on the work of two former students of the University of Stavanger (UiS) [63, 66], except for the contrast enhancement and the extraction of the annotated regions, which were developed primarily for this thesis. The steps involved in the pre-processing are:

1. Rearrange images.

2. Image registration.

3. Skull removal.

4. Contrast enhancement.

5. Second image registration.

6. Extract annotated regions.

3.2.1 Rearrange Images

The first step of the pre-processing is to rearrange images to sort them temporally. For each image, acquisition time and slice location contained in the DICOM header are read. This step is vital for the project because it splits the dataset of images in different groups based on the slice location and acquisition time.

3.2.2   Register Images

The image registration step is used to coordinate many images where they may have been acquired at a different time or viewing points [67]. Images have been registered because there is always a chance that, during the acquisition of CT perfusion images, a patient may have moved to create images not aligned with each other. CTP images are registered using a similarity transformation to align all images in every slice spatially. The first temporal image in each slice is set as the fixed image, and the 29 remaining images are registered based on the first temporal image. The method involved is iterative, and it uses an optimizer to get the best similarity among images; the registration is intensity-based and, also, it uses phase correlation to estimate an initial geometric transformation. For more details, check [63, 66].

3.2.3   Skull Removal

The skull is useful during the registration process. However, the skull area has a very high intensity in the images, and for many tasks, it can be an advantage to remove it before further processing. This step is important as it may affect the extracted features; hence, the removal may lead to better and more distinct features [68]. The skull removal technique is based on an algorithm suggested by Hovland [63], a former student at the University of Stavanger. The algorithm is based on four main passages that are applied to each image. A visualization of the algorithm is given in Fig. 3.6.
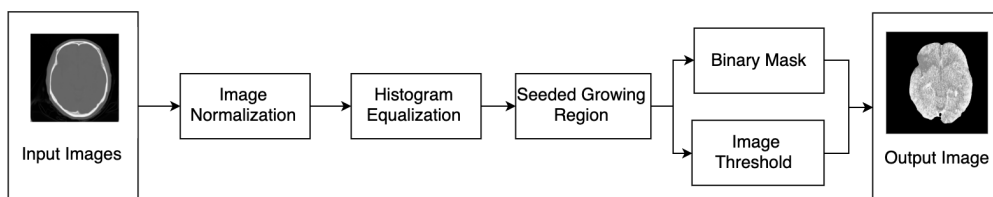
Figure 3.6: Skull removal algorithm.

A summary of the steps is displayed below, for a more detailed explanation about the algorithm, check [63].

1. **Image normalization**: the technique is useful to expand the grayscale for images to share a similar range. It is implemented using the following equation:

$$I_N = (I - \text{Min}) \frac{\text{newMax} - \text{newMin}}{\text{Max} - \text{Min}} + \text{newMin} \qquad (3.1)$$

where I expresses an image with n-dimensional grayscale levels with intensity values in range $[\text{Min}, \text{Max}]$. I_N represent the normalized image, with intensity value in range $[\text{newMin}, \text{newMax}]$.

2. **Histogram equalization**: after the normalization, images are subjected to a histogram equalization, which increases the global contrast leading to a better distribution of the intensities in the histogram. This step is useful since the images have a uniform foreground.

3. **Seeded growing region**: this step is fundamental to segment each image. The initial position of the region is the seed point $[250, 250]$ chosen as the pixels with the maximum intensity level newMax; the seeded growing region expands by examining neighboring pixels of the initial seed point.

4. **Binary mask**: this is the final construction from the resulting seeded growing region; the mask represents the borders of the skull, thus to finally strip the skull, apply the mask over the image. If the seeded growing region doesn't return a satisfactory binary mask for one or more images, a different method is used to generate proper binary masks: the image is compared with a specific threshold in order to create a binary image from the operation, then small objects are removed from the binary image, and the holes are filled.

3.3 Contributions to the pre-processing steps

The section presents a discussion and an overview of the contributions made for this thesis regarding the development of two pre-processing steps to prepare the proper dataset for the models and various techniques used to overcome a limitation in the dataset.

3.3.1 Contrast Enhancement

This step enhances the contrast and the differences within the intensity regions inside the brain to emphasize how much each pixel is different from the others over time. Fig. 3.7 presents an example of a brain section (a) after skull removal, (b) displays the same brain section after contrast enhancement. Three steps are involved in the improvement of the contrast of the images:

- First, filter the image with a 2D Gaussian smoothing kernel with a standard deviation of 0.5 to reduce the gaussian random noise inside an image; this is an essential step before a contrast stretching.
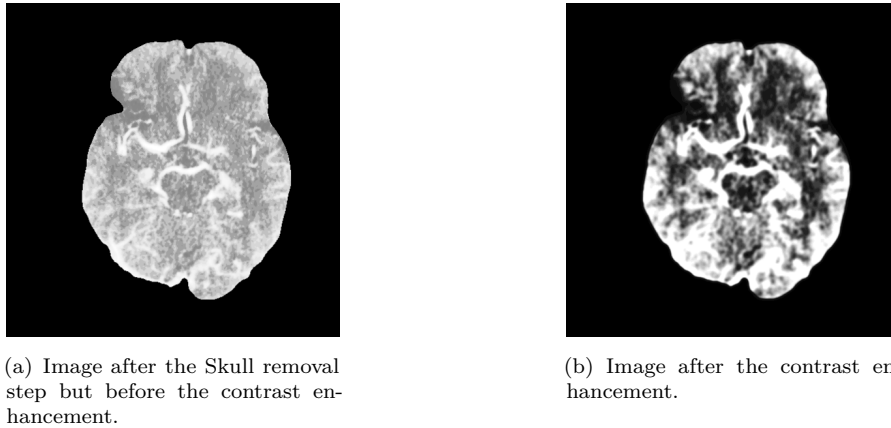
(a) Image after the Skull removal step but before the contrast enhancement.

(b) Image after the contrast enhancement.

Figure 3.7: Difference between the same brain section before and after the contrast enhancement.

- The second step performs a histogram equalization on the resulting image from the previous step;

- Finally, an image normalization technique is applied, based on the Myklebust's algorithm [66].

3.3.2 Extract Annotated Regions

The last step of the pre-processing part involves the manual annotations extraction of the penumbra and core regions. First, each manually annotated image (Fig. 3.4), is crop and resize to 512x512; then it is set as the fixed image and registered with the other 30 time-series images, already registered in the first step.

After the registration, the annotated area in the image is extracted to create a new image that contains the different regions of the brain. All these regions represent the various classes to classify with the CNN approaches. Fig. 3.8 displays a manually annotated image after the extraction of the different classes:
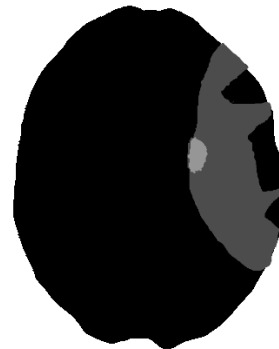
Figure 3.8: Brain section after the extraction of the different regions.

- The white label corresponds to the background class;

- the black label shows the area of the brain;

- the dark grey displays the penumbra region;

- the light grey portion gives an overview of the core area.

Finally, after all these steps, the dataset is ready to be trained using one of the models developed for this work and explained in the next chapters.

3.3.3 Limitation of the dataset

Unfortunately the dataset is composed only of image-sets from 11 patients; however, only 10 patients were adopted during the training of the distinct models: patient with ID 1 was not included due to the peculiar structure of its rearranged CTP images which have a different number of groups compared to its manually annotated images. This problem leads to confusion during the association of the different rearranged groups with the manually annotated images; hence, a decision was made to exclude the patient 1 from the dataset.

The sample size is not adequate for any DNN because a model needs to be trained with a large and suitable number of elements. For this reason, an augmented dataset is also generated from the current one. A series of vectors create the dataset used during the training of the various methods. A vector is composed of 30 small portions of 16x16 pixels, called tile, plus an output. The 30 tiles refer to the first image in the time-series, plus the same part of the other 29 images in the time-series. The vectors dataset is created using a sliding window technique, Fig. 3.9 represents a visual explanation of the sliding window technique method to overcome the dataset limitation. Starting from position $[0, 0]$ in the image until position $[512, 512]$ with a stepping size of 4 pixels, with a sliding window technique, a red square, which represents the 16x16 tile, is extrapolated from each image plus the image containing the annotation region. The final result of this step is a training vector that contains $(16x16 \times N + 1)$, where N is the number of brain sections per patient, the input, while the $(+1)$ indicates the 16x16 tile extrapolated from the image with the annotated regions, which corresponds to the output.

Each vector created is labeled with an output, which is different based on the approach used. The outcome of the Tile Classification approach, described in Chap. 4, is an integer that corresponds to the region class with the most considerable number of pixels inside the tile; while the output of the pixel-by-pixel segmentation, defined in Chap. 5, is the complete vector that represents the extrapolated tile. The output integer for the Tile Classification approach is generated by summing up together the number of pixel-based on their different intensity: the pixel value with the highest number of elements is chosen as the labeled class. Classes have a selection priority, thus if two or more classes have the
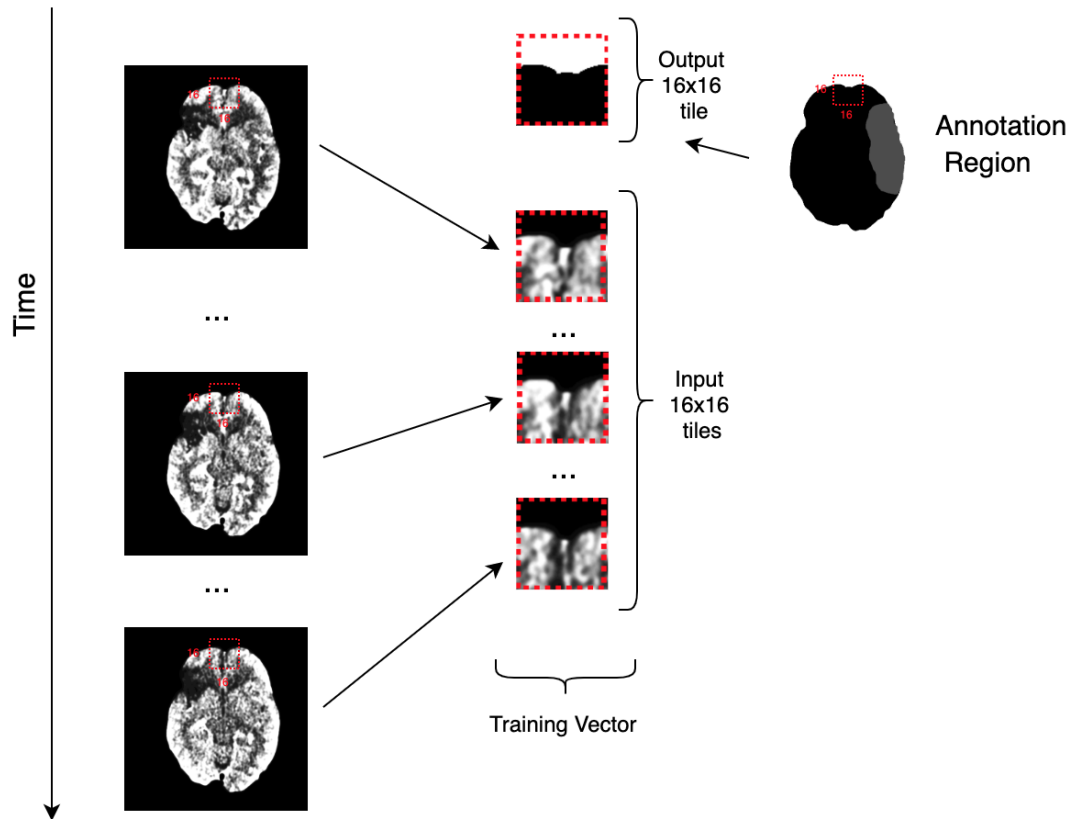
Figure 3.9: Example of the sliding window technique.

same number of pixels inside a tile, the choice is made based on the priority: first *core*, then *penumbra*, *brain* and finally *background*. The decision of this order is necessary to maintain simplicity among the process and to increase the number of core and penumbra classes. An improvement for future work could be to find another method to decide the specific class of a tile: a possible way could be to add new classes for those tiles that have a similar number of pixels for two or more classes.

At the end of the process, the number of vectors extrapolated using this technique is more suitable to train a model based on a DNN. The total number of vectors is 803453; the number of vectors labeled with a background class is 70500, while the ones with a brain class are 596039. Vectors labeled with the penumbra class are 127692; vectors with a core label are 9222. The extraction of background tiles is limited because overweighting the dataset with too many images of the same class; thus, the number of background vectors was fixed to 500 per each brain section in the volume.

Data Augmentation

Considering that the number of core tiles is inadequate in the dataset (1.1%) compared to the other classes extrapolated, there is a chance that the trained models will not be able to detect that particular class. Besides the standard dataset, it was also created another dataset with the same method as the other one with the addition of some data augmentation techniques. For each image labeled with the core class, five different methods were applied to augment the sample size:

- Rotate the time-series images of 90, 180 and 270 degrees counterclockwise.

- The image tiles are flipped upside down.

- Flipped from left to right.

The augmented dataset is formed of 848849 number of vectors in total. 70500 are labeled with a background class, 596039 represents a brain class, 127692 belong to the penumbra class, while the remaining 55338 are vectors labeled with the core class, leading to 6.5% of the vectors are marked with the core class.

To not create confusion for the next chapters, the augmented dataset will be called "Dataset 2", while the dataset without any data augmentation technique will be called "Dataset 1".

4

Tile Classification Approach
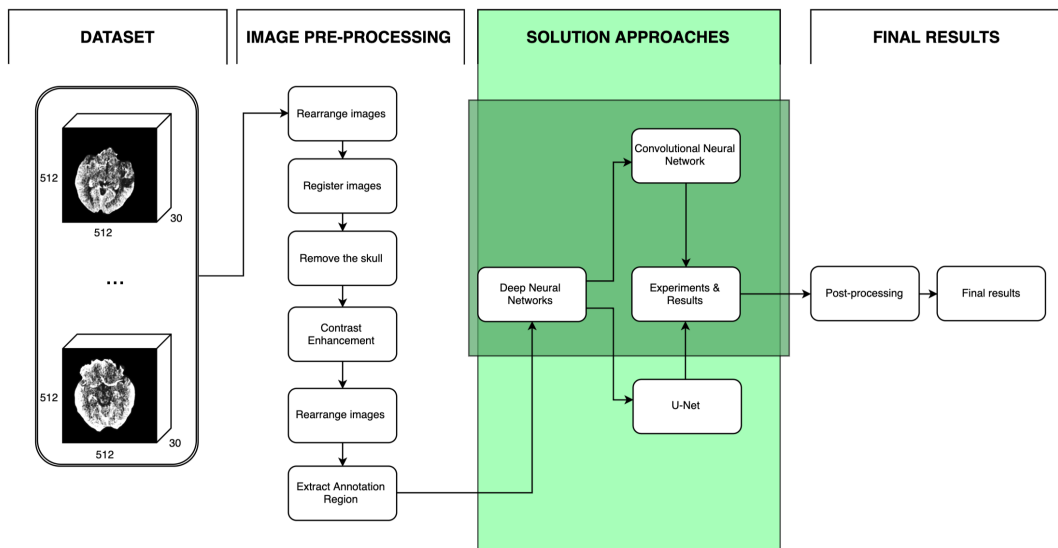
Figure 4.1: Focus of chapter four.

4.1 Introduction

The chapter explores three different architectures to test which one yields the best result during the classification. The structure of these methods is similar among each other except for smaller changes, explained in detail in corresponding sections. Fig. 4.2 displays an overview of the involved steps after the creation of the input until the generation of the predicted output, passing through the selection of the architecture and the post-processing steps. The three Tile Classification architectures are described in detail, highlighting the differences and the different results achieved, based on statistical information and confusion matrices. Additionally, visual examples are presented to show the predicted brain section of the patient with ID 2. All the other predicted images for all the patients, with their corresponding confusion matrices and statistical information, are presented in Appendix A.
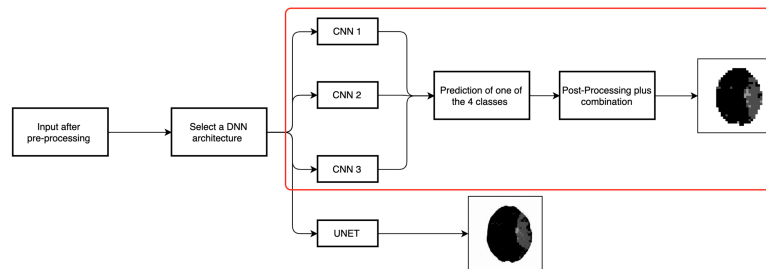
Figure 4.2: Overview of the input and output section for the CNN architectures.

The Tile Classification approach is based on CNN architecture. The general idea is to use the dataset of 4D vector images, described in detail in Sec. 3.3.3, as the input; then train one of the proposed models with a predefined number of epochs equal to 50 and, after that, receive as output a value that represents one of the four classes that characterize an image. Each output is transformed into a 16x16 image and merged with the other productions to create the final image. All predicted images in this chapter display pixelation due to the merging implementation of the various tiles.

4.1.1 Existing Approaches/Baselines

A large number of studies were made to predict and detect regions inside a brain affected by different diagnoses. A lot of recent researches are related to a brain tumor, lesion segmentation, and the prediction of Alzheimer's disease [69–73]. Architectures describe in the next sections are following a similar approach to the researches mentioned above but with two important differences: the dataset and the application of the model. All previous studies based their dataset on MRI images, and these researches aimed to

discover Alzheimer's diseases or brain tumors. The proposed architecture is using CTP images over a period of time, as explained in detail in Chap. 2 and Chap. 3; to the authors' knowledge, there is no literature using NN on CTP images for stroke detection or segmentation of stroke areas.

4.2 Post-processing

This section investigates the post-processing technique used to generate the final images from the predicted tiles. The post-processing steps are an essential part of the entire process. They allow combining the different tiles predicted during the test of the distinct models to have a better overview of their overall accuracy. The three architectures presented for the Tile Classification approach are adopting the same post-processing technique: the output of a model is a list containing four numbers representing the four probabilities of the classes that form every ground truth image; the highest value in the list corresponds to the class predicted by a model. This integer is expanded into a 16x16 matrix containing the same value, transformed into an image and then combined it with all the other matrices of the same brain section; an example of the involved steps is given in Fig. 4.3. This process is performed for every tile generated during the testing of each model.
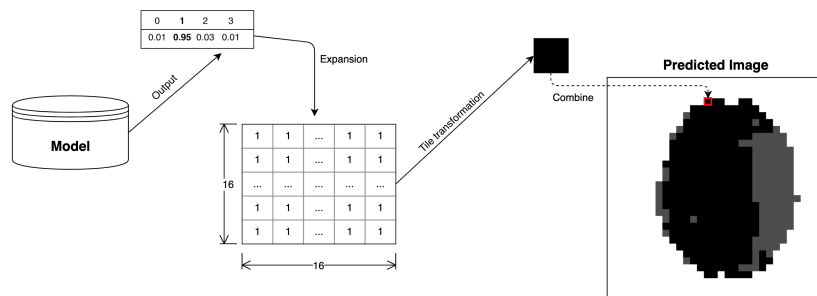
Figure 4.3: Example of post processing steps for the Tile Classification approach.

Each class is mapped to a specific output integer and a particular value in the RGB color code to simplify the process of transforming a matrix into a tile image:

- The background class corresponds to the integer 0, and it is mapped with white color: (255,255,255) RGB decimal code.

- The brain class is related to the integer 1, and it is mapped with black color: (0,0,0) RGB decimal code.

- The penumbra class coincides with the integer 2; it is mapped with a dark gray color: (76,76,76) RGB decimal code.

- The last class, core, is mapped with the integer 3 and it corresponds to light gray color: (150,150,150) RGB decimal code.

4.3 Proposed Architectures

The section explores the *K*-Fold Cross-validation for patient 2, the predicted confusion matrices, and all statistical results for the model analyzed with the dataset without any modification ("Dataset 1") and the augmented dataset ("Dataset 2"). The results of all the other patients are presented in Appendix A. The following list delineates all various classes renamed and adopted in the next architectures for a better understanding:

- *I*: Background class;

- *II*: Brain class;

- *III*: Penumbra class;

- *IV*: Core class.

All results are based on the training of 50 epochs (steps); at the end of every epoch, a validation step is performed to evaluate the loss of the model. The loss function implemented for these architectures was the categorical cross-entropy, while the optimizer function used was the stochastic gradient descent (SGD), and the adaptive moment estimation (Adam).

4.3.1 Architecture 1
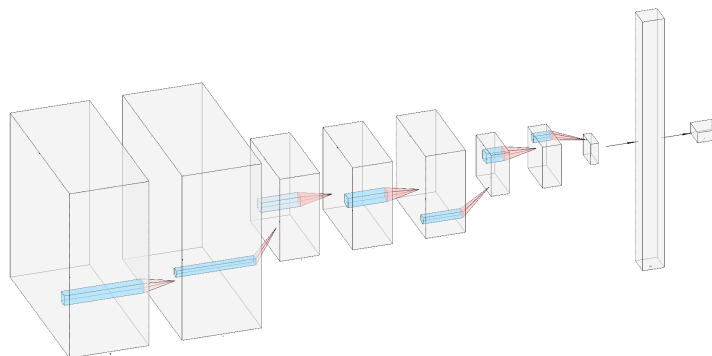
Figure 4.4: General structure for the Tile Classification architecture 1.

Fig. 4.4 displays a general overview of the first architecture of the Tile Classification approach. The network contains nine hidden layers plus the output layer. A summary of the architecture, plus the parameters involved, are presented in Table 4.1. The first layer, a convolutional layer, takes in input a volume of 30 images of dimension 16x16 pixels; the input is convolved with a ReLU activation function, a kernel filter of size (3,3,3) and batch normalization. Subsequently, the second layer represents a convolutional layer with a ReLU activation function plus a batch normalization operation. The third layer represents a max-pooling operation of size (2,2,2) over the time-series images plus a dropout operation. Layers four and five contain convolutional layers similar to the first two layers but with an input of size 8x8x15. The next layer executes a max-pooling operation plus a dropout operation to reduce the overfitting probability. Layer seven is a convolutional layer with a 4x4x7 input with the same components of the other convolutional layers. Layer eight contains the last max-pooling layer before the fully-connected layers. Fully-connected layer compresses the input from three dimensions to a one-dimensional vector of length 100. The output layer contains four artificial neurons which yield the probability of the four classes of the brain for the time-series input. The number of total parameters is 203032; trainable parameters are 202680 while non-trainable are 352.

Architecture 1		
#	**Layer**	Param.
1	conv3D	448
	batch_norm	64
2	conv3D	13856
	batch_norm	128
3	max_pool3D	0
	dropout	0
4	conv3D	27680
	batch_norm	128
5	conv3D	27680
	batch_norm	128
6	max_pool3D	0
	dropout	0
7	conv3D	55360
	batch_norm	256
8	max_pool3D	0
	dropout	0
9	flatten	0
	dense	76900
	dropout	0
10	dense	404

Table 4.1: Layers summary of architecture 1.

4.3.2 Architecture 2
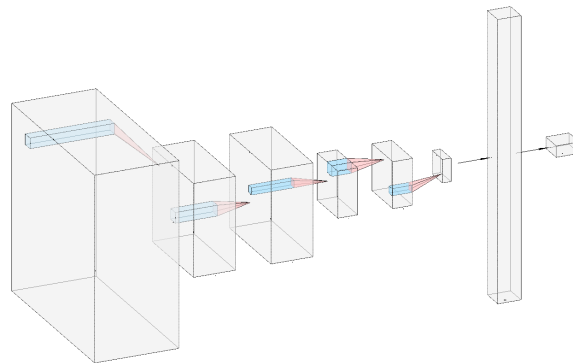
Figure 4.5: General structure for the Tile Classification architecture 2.

The second architecture has a more compressed structure compared to the first one (Sec. 4.3.1) because it has only seven hidden layers plus the final output layer, as shown in Fig. 4.5. The main difference with the first architecture, besides the number of layers, lays on the kernel size used during convolutional layers: instead of using the default kernel size of (3,3,3), it was implemented a kernel size of (3,3,N) where N is equivalent to the number of image in depth of the volume. Hence, the kernel matrix checks and evaluates the entire time-series volume of the brain section simultaneously.

A summary of layers for this architecture is displayed in Table 4.2. A convolutional operation forms the first layer with a kernel size of (3,3,30), a ReLU activation function and a batch normalization at the end. Layer two uses a max-pooling operation with a window size of (2,2,2) on the input plus a dropout function. The next four layers are a reiteration of the first two layers with different input sizes. The last hidden layer flats the output in a one-dimensional vector of length 100, making use of the dropout function to prevent overfitting and pass the output to the final layer,

Architecture 2		
#	**Layer**	Param.
1	conv3D	4336
	batch_norm	64
2	max_pool3D	0
	dropout	0
3	conv3D	138272
	batch_norm	128
4	max_pool3D	0
	dropout	0
5	conv3D	553024
	batch_norm	256
6	max_pool3D	0
	dropout	0
7	flatten	0
	dense	76900
	dropout	0
8	dense	404

Table 4.2: Layers summary of architecture 2.

which yields the four probabilities for the respective classes. The number of total parameters is 773384; trainable parameters are 773160 while non-trainable are 224. The number of parameters involved in the architecture is almost four times bigger than the number of parameters of the first architecture, which means that the computational time of each epoch is relatively slower for the analyzed methods.

4.3.3 Architecture 3

Figure 4.6: General structure for the Tile Classification architecture 3.

The last architecture constructed presents a structure similar to the second one (Sec. 4.3.2) but with two fundamental differences in it: the window size of the first max-pooling layer is (2,2,*N*) and the number of layers. Fig. 4.6 exhibits an overview of the architecture, which contains eight hidden layers plus the output. Layer one is a convolutional layer with a kernel size of (3,3,*N*), a ReLU activation function and a batch normalization operation. The second layer has an essential role in the architecture because it is the max-pooling layer with a window size of (2,2,*N*), where *N* is equal to the number of time series in the input volume. After this operation, the depth of the time series volume input is shrunk to a vector of dimension one. Layers three, four, and six execute convolutional operations with a ReLU function, a kernel size of (3,3,1) and a final batch normalization at the end of each layer. The fifth layer performs a max-pooling operation with a window size of (2,2,1) in the same way as layer seven. The last two layers implement a flatten and a dense procedure to create an output of 4 values, to show different probabilities for each class. The number of total parameters is 63800; trainable parameters are 63312 while non-trainable are 488. This architecture presented contains the smallest amount of parameters among the architectures; hence, each epoch takes approximately 400 seconds to execute.

Architecture 3		
#	**Layer**	Param.
1	conv3D	4336
	batch_norm	64
2	max_pool3D	0
	dropout	0
3	conv3D	4640
	batch_norm	128
4	conv3D	9248
	batch_norm	128
5	max_pool3D	0
	dropout	0
6	conv3D	18496
	batch_norm	256
7	max_pool3D	0
	dropout	0
8	flatten	0
	dense	25700
	batch_norm	400
	dropout	0
9	dense	404

Table 4.3: Layers summary of architecture 3.

4.4 Experimental Setup and Dataset

Even after dividing the dataset in smaller tiles and using data augmentation techniques, described in Sec. 3.3.3, the sample size is still limited. For these reasons, before the training and testing of all proposed architectures, the dataset is split using a *K*-fold Cross-validation, as illustrated in Sec. 2.2.10. Additionally, four experiments with distinct optimizer functions and datasets were implemented to test all different architectures, to compare them, and to demonstrate which one presents the best performances.

In the first experiment, the various architectures were tested with "Dataset 1" (no data augmentation) and the SGD optimizer function; on experiment 2, the "Dataset 2" (augmented dataset) was used for the training of the distinct models; however, during the tests, the "Dataset 1" was used to predict the brain sections in order to prevent

false results. Experiment 3 and 4 were executed with the Adam optimizer function, respectively with "Dataset 1" for both training and testing and "Dataset 2" for training and "Dataset 1" for testing.

4.5   Analysis of results for a single test patient

In this section, a visualization comparison of predicted images with the distinct experiments is shown for each architecture to empathize over the similarities and differences of the various experiment results; plus, the section presents an analysis of confusion matrices and the numerous statistical information for two distinct datasets, and two different optimizer functions, evaluated through the various architectures. Results shown in the next pages are indicative only for the patient with ID 2 due to the large size of all the predicted brain sections. The results of all the other patients are presented in detail in Appendix A.1 for the first architecture, Appendix A.2 for architecture two and Appendix A.3 for the last architecture.

4.5.1   Partial results for patient 2 with Architecture 1

Visualization Comparison

This section presents a visual comparison of one brain section for the patient with ID 2, given in Table 4.4, to understand how the presented architecture performs with the different experiments. The choice of this particular brain section is due to the presence of the totality of the classes involved. The first column in the table represents a brain section's ground truth using an image after extracting the manual annotations, as defined in Sec. 3.3.2. The second column shows a predicted image with "Dataset 1" and the SGD optimizer function (experiment 1); the third column exhibits a predicted image with the same dataset as the second column, but the model is trained with an Adam optimizer function (experiment 3). The last two columns display the predicted section with the augmented dataset for both the optimizer function SGD (experiment 2) and Adam (experiment 4). All other brain sections of the patient are presented in Table A.1 in the Appendix A together with all other patients. Comparing all Tile Classification results, the experiment that produces better accuracy in correlation with the ground truth image is the one with the augmented dataset and the Adam optimizer (experiment 4), even if specific images contain tile classified with the core or penumbra class.

As it can be observed in Table A.1, Tile Classification results for the first two slices, both for "Dataset 1" and "Dataset 2", are very different from the ground truth images;

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam

Table 4.4: Example of brain section comparison for patient 2 with different techniques of the first architecture.

this is due to imperfections in the skull removal in the pre-processing part. The skull removal algorithm is not perfect; nevertheless, it performs well and shows good results for most of the sections; the only unsuitable outcomes are some of the first sections where the eyes make the skull less obvious to detect, among a few of the patients. Despite this inconvenience, the shape of various classes is almost maintained for both training methods. Data augmentation sections manifest a larger number of tiles classified with the core class compared with sections generated with the "Dataset 1". This event is related to the fact that a model has an extended number of samples that corresponds to a core class during its training: the augmented dataset can lead to a higher grade of confusion in the prediction of core areas compared to the other dataset. This phenomenon is confirmed also in all images in Table 4.4. The architecture proposed with the different datasets and optimizer is not perfect, as it is possible to evince from the samples in the previous table. A significant number of false-positive penumbra and core tiles are spotted in random places in the brain region, which can lead to possible wrong or not accurate decisions for medical doctors.

Analysis with different Datasets

Table 4.5 presents a comparison of the various statistical information for different classes in distinct brain sections and their average for both standard and augmented datasets and the two optimizer functions. Achieved results are quite similar between the two datasets even if it is possible to notice some variances on average results in penumbra and core classes, respectively. Accuracy and precision exhibit promising outcomes, while the F1 score metric shows impressive achievements for the first two classes but average results for penumbra and especially for core classes. Table 4.6 shows the overall confusion matrix for all sections of the brain analyzed for patient with ID 2 with the normal dataset ("Dataset 1"), while Table 4.8 displays the final confusion matrix for the same patient with the augmented dataset ("Dataset 2"); both tables present values for the SGD optimizer function. Similarly, Tables 4.7 and 4.9 exhibit confusion matrices for two different datasets with the Adam optimizer function. The predicted core class (*IV*)

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.958	0.968	0.045	0.018	0.035	0.084	0.965	0.955	0.982	0.916
		II	0.939	0.893	0.116	0.098	0.038	0.046	0.962	0.884	0.902	0.954
		III	0.971	0.757	0.057	0.367	0.028	0.003	0.972	0.943	0.633	0.997
		IV	1.0	0.5	0.5	0.5	0.0	0.0	1.0	0.5	0.5	1.0
		Average	**0.967**	**0.780**	**0.179**	**0.246**	**0.025**	**0.033**	**0.975**	**0.821**	**0.754**	**0.967**
	Adam	I	0.964	0.973	0.034	0.021	0.04	0.066	0.96	0.966	0.979	0.934
		II	0.94	0.891	0.137	0.08	0.03	0.053	0.97	0.863	0.92	0.947
		III	0.969	0.75	0.045	0.383	0.03	0.002	0.97	0.955	0.617	0.998
		IV	1.0	0.333	0.5	0.75	0.0	0.0	1.0	0.5	0.25	1.0
		Average	**0.968**	**0.737**	**0.179**	**0.283**	**0.025**	**0.03**	**0.975**	**0.821**	**0.692**	**0.970**
Data Augment.	SGD	I	0.966	0.974	0.029	0.022	0.044	0.056	0.956	0.971	0.978	0.944
		II	0.951	0.914	0.095	0.077	0.03	0.038	0.97	0.905	0.923	0.962
		III	0.976	0.775	0.146	0.291	0.018	0.008	0.982	0.854	0.709	0.992
		IV	0.999	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
		Average	**0.973**	**0.666**	**0.318**	**0.348**	**0.023**	**0.026**	**0.977**	**0.682**	**0.652**	**0.974**
	Adam	I	0.962	0.972	0.037	0.02	0.039	0.071	0.961	0.963	0.98	0.929
		II	0.945	0.904	0.096	0.095	0.038	0.039	0.962	0.904	0.905	0.961
		III	0.978	0.798	0.111	0.277	0.018	0.006	0.982	0.889	0.723	0.994
		IV	0.999	0.333	0.0	0.8	0.001	0.0	0.999	1.0	0.2	1.0
		Average	**0.971**	**0.752**	**0.061**	**0.298**	**0.024**	**0.029**	**0.976**	**0.939**	**0.702**	**0.971**

Table 4.5: Comparison of the statistical information for prediction on Patient 2 based on two different datasets.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8455	154	2	0
	II	330	3367	35	0
	III	67	287	612	1
	IV	0	1	0	1

Table 4.6: Confusion Matrix for Patient 2 with the normal dataset and SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8553	177	3	0
	II	259	3283	25	0
	III	44	341	622	1
	IV	0	2	1	1

Table 4.7: Confusion Matrix for Patient 2 with the normal dataset and Adam optimizer.

in Table 4.8, which displays the results for experiment 2, contains a relevant number of elements than the corresponding class in Table 4.6. The confusion matrix in Table 4.6 was computed using the first experiment setup; while the experiment 2 relays on the "Dataset 2", which consists of a more significant number of sample for the core class compared with "Dataset 1". The confusion matrix created from experiment 3 (Table 4.7) shows a better result than the others, which corresponds well with what was indicated by the visualization example in Table 4.4.

Fig. 4.7 shows four plots representing different accuracy results achieved with particular methods described in previous sections. Plots are displaying how the training and validation accuracy is evolving during the 50 epochs as it is possible to evince from Fig.

		Actual Class			
		I	II	III	IV
Pred.	I	8597	190	5	0
Class	II	198	3447	88	0
	III	56	169	554	2
	IV	1	3	2	0

Table 4.8: Confusion Matrix for Patient 2 with the augmented dataset and SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred.	I	8525	169	4	0
Class	II	296	3440	67	0
	III	31	191	579	0
	IV	0	7	1	2

Table 4.9: Confusion Matrix for Patient 2 with the augmented dataset and Adam optimizer.

4.7(c) and Fig. 4.7(d), during the training with the Adam function, the accuracy (red lines) stabilizes faster compared to the other two implementation. The different learning curves in the plots are expected since Adam optimizer achieves similar results as the SGD function but in a smaller amount of computational time [47]. As expected, due to the small number of data in the validation set, the validation curves are not smooth.
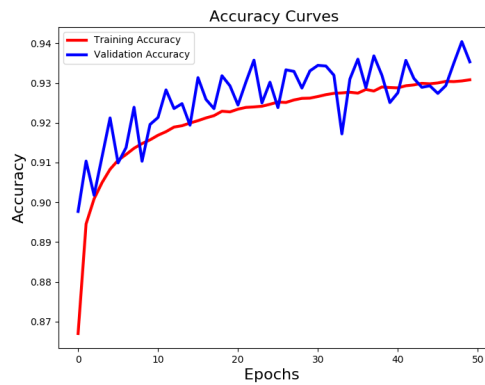
(a) Accuracy plot for normal dataset and SGD.
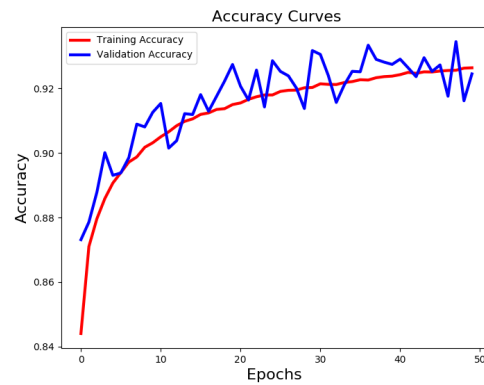
(b) Accuracy plot for augmented dataset and SGD.
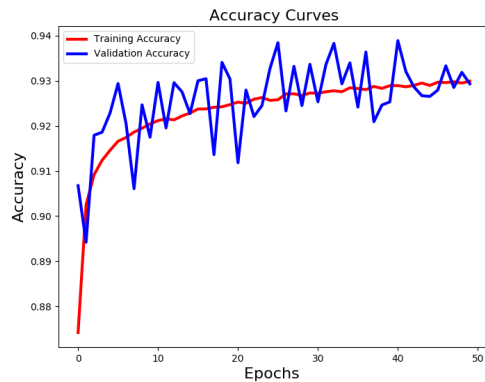
(c) Accuracy plot for normal dataset and Adam.
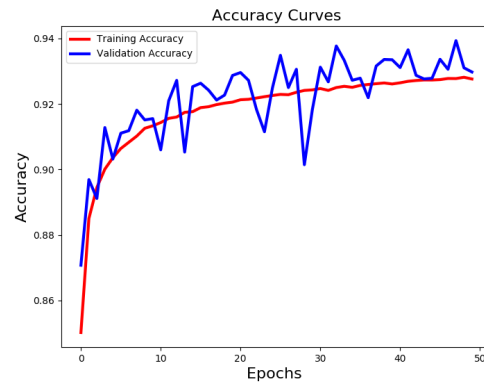
(d) Accuracy plot for augmented dataset and Adam.

Figure 4.7: Different accuracy plots for Patient 2 for the architecture one.

4.5.2   Partial results for patient 2 with Architecture 2

Visualization Comparison

Table 4.10 displays the same brain section of the various predicted images for the second architecture. The most similar predicted image with the ground truth brain section is the one realized with the augmented dataset and the SGD optimizer function (experiment 2). Furthermore, images created with the augmented dataset are displaying promising outputs since the shapes of the different classes look very similar to the ground truth's classes despite the last predicted image contains a small number of tiles inside the brain classified as a background (white tiles).

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam

Table 4.10: Example of brain section comparison for patient 2 with different techniques of the second architecture.

Images displayed in Sec A.2 present some peculiarities: in some portions of the border between the background and the brain, the outcome tile is labeled with the penumbra class creating a bizarre effect in the final image. This fact is notable in the majority of the predicted images, with both datasets and the two optimizer functions. A possible reason for this aftereffect is the limited amount of samples for the border regions between different classes, which can confuse the train of the various models.

Analysis with different Datasets

Table 4.21 displays the different statistical information for the second architecture. The overall results are very close to the first architecture with small variations in some metrics. F1 score is almost 10% below to the first architecture; the precision with the only exception on the result with an Adam optimizer and an augmented dataset (experiment 4), is around 15% lower than the first architecture. With an augmented dataset, the SGD function shows a similar result in the recall because of the better score within the core class, while the other methods perform poor results compared to other architectures.

Confusion matrices extrapolated from this architecture have similar structures as the other confusion matrices for architecture one. They all show different interpretations

		Actual Class			
		I	II	III	IV
Pred. Class	I	8496	199	9	0
	II	274	3278	60	0
	III	86	324	578	2
	IV	0	4	2	0

Table 4.11: Confusion Matrix for Patient 2 with the normal dataset and SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8481	169	1	0
	II	288	3383	41	0
	III	83	257	607	2
	IV	0	0	0	0

Table 4.12: Confusion Matrix for Patient 2 with the normal dataset and Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.957	0.968	0.041	0.024	0.047	0.078	0.953	0.959	0.976	0.922
		II	0.935	0.884	0.139	0.092	0.035	0.054	0.965	0.861	0.908	0.946
		III	0.964	0.705	0.109	0.416	0.033	0.006	0.967	0.891	0.584	0.994
		IV	0.999	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
	Average		**0.964**	**0.639**	**0.322**	**0.383**	**0.029**	**0.035**	**0.971**	**0.678**	**0.617**	**0.966**
	Adam	I	0.959	0.969	0.042	0.02	0.038	0.08	0.962	0.958	0.98	0.92
		II	0.943	0.9	0.112	0.089	0.035	0.044	0.965	0.888	0.911	0.956
		III	0.971	0.76	0.065	0.36	0.027	0.003	0.973	0.935	0.64	0.997
		IV	1.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
	Average		**0.968**	**0.657**	**0.305**	**0.348**	**0.028**	**0.032**	**0.975**	**0.695**	**0.633**	**0.968**
Data Augment.	SGD	I	0.954	0.965	0.048	0.021	0.04	0.091	0.96	0.952	0.979	0.909
		II	0.938	0.892	0.11	0.106	0.042	0.044	0.958	0.89	0.894	0.956
		III	0.968	0.724	0.138	0.375	0.027	0.007	0.973	0.862	0.625	0.993
		IV	0.999	0.2	0.0	0.889	0.001	0.0	0.999	1.0	0.111	1.0
	Average		**0.965**	**0.695**	**0.074**	**0.348**	**0.028**	**0.036**	**0.973**	**0.926**	**0.652**	**0.965**
	Adam	I	0.961	0.971	0.022	0.036	0.073	0.045	0.927	0.978	0.964	0.955
		II	0.945	0.9	0.145	0.05	0.018	0.056	0.982	0.855	0.95	0.944
		III	0.972	0.75	0.126	0.344	0.023	0.007	0.977	0.874	0.656	0.993
		IV	0.997	0.1	0.0	0.947	0.003	0.0	0.997	1.0	0.053	1.0
	Average		**0.969**	**0.68**	**0.073**	**0.344**	**0.029**	**0.027**	**0.971**	**0.927**	**0.656**	**0.973**

Table 4.13: Comparison of the statistical information for prediction on Patient 2 based on two different datasets for the second architecture.

among methods: the confusion matrix in Table 4.12 does not have any tile for the core class, but it is the one who predicts the larger number of tiles for class three (penumbra). Table 4.15 shows a few errors after the training; however, it contains the biggest number of errors for the predicted core class. Table 4.11 and Table 4.14 display an average distribution of corrected predictions and errors among all classes.

Lastly, Fig. 4.8 presents four learning curves for the methods proposed, after 50 epochs. The curves show similar shapes as the four leaning curves related to the first architecture results in Fig. 4.7. Each epoch has an average computational time of 600 seconds for experiment 1 and experiment 2; 670 seconds on average for the other two experiments, for a total of almost 9 hours of training. All plots show similar curves and results during

		Actual Class			
		I	*II*	*III*	*IV*
	I	8423	170	10	0
Pred.	*II*	329	3390	74	0
Class	*III*	96	241	561	0
	IV	4	6	6	2

Table 4.14: Confusion Matrix for Patient 2 with the augmented dataset and SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8659	302	22	0
Pred.	*II*	121	3255	51	0
Class	*III*	71	226	567	0
	IV	1	26	9	2

Table 4.15: Confusion Matrix for Patient 2 with the augmented dataset and Adam optimizer.

the training. The validation accuracy for all methods exhibits higher values compared to the training accuracy: this is because the sample size of the validation set, as explained previously for their jagged curves.
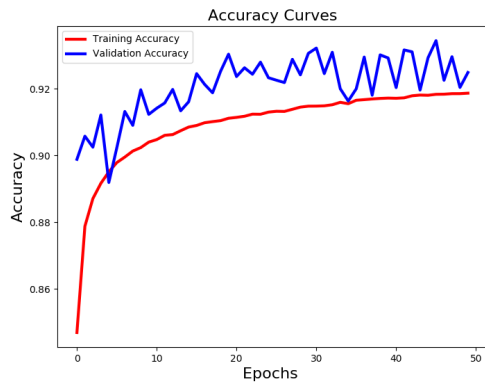
(a) Accuracy plot for normal dataset and SGD.
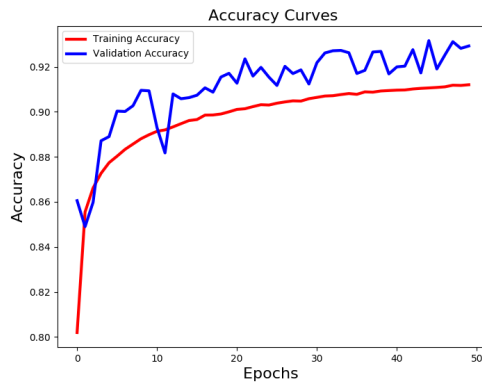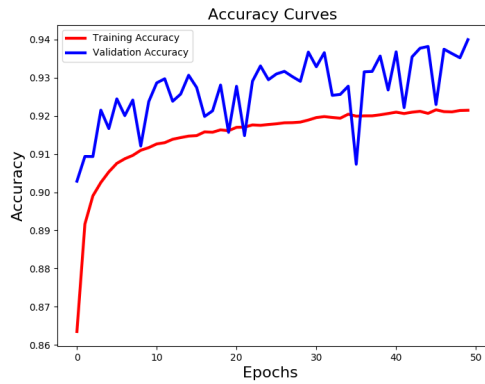
(b) Accuracy plot for augmented dataset and SGD.

(c) Accuracy plot for standard dataset and Adam.
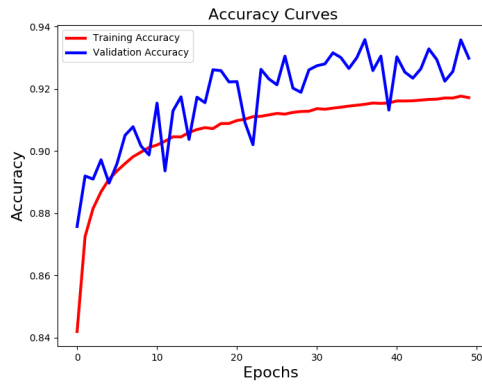
(d) Accuracy plot for augmented dataset and Adam.

Figure 4.8: Different accuracy plots for Patient 2 for the second architecture.

4.5.3 Partial results for patient 2 with Architecture 3

Visualization Comparison

Among the predicted images of the brain section in Fig. 4.16, the best representation of the ground truth image is the one generated by the Adam optimizer and the augmented dataset (experiment 4), even if some tiles are located in the incorrect position. The other predictions show a similar structure of the various regions as the ground truth image, although they present a significant number of tiles in the wrong area inside the brain region. Additionally, the core class is not shown, or it is displayed in a smaller shape, as it is possible to see in the predicted section with Adam function and the regular dataset.

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam
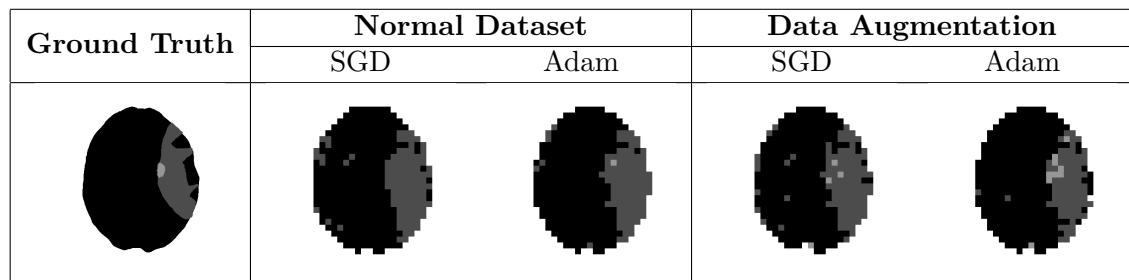
Table 4.16: Example of brain section comparison for patient 2 with different techniques of the third architecture.

Analysis with different Datasets

		Actual Class			
		I	II	III	IV
Pred. Class	I	8386	166	3	0
	II	332	3294	69	0
	III	138	345	575	2
	IV	0	0	2	0

Table 4.17: Confusion Matrix for Patient 2 with the normal dataset and SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8496	180	1	0
	II	275	3272	52	0
	III	81	356	593	2
	IV	0	1	3	0

Table 4.18: Confusion Matrix for Patient 2 with the normal dataset and Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8447	186	8	0
	II	302	3281	64	0
	III	103	325	563	2
	IV	0	17	14	0

Table 4.19: Confusion Matrix for Patient 2 with the augmented dataset and SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8450	182	11	0
	II	309	3363	85	0
	III	85	219	531	1
	IV	8	45	22	1

Table 4.20: Confusion Matrix for Patient 2 with the augmented dataset and Adam optimizer.
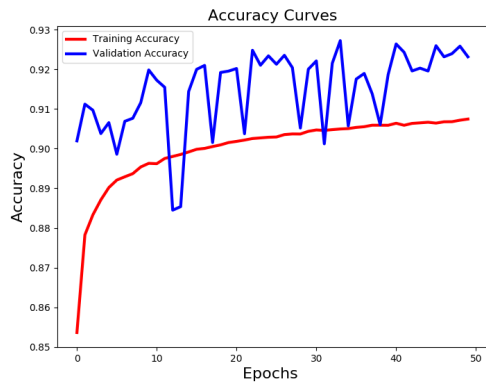
(a) Accuracy plot for normal dataset and SGD.
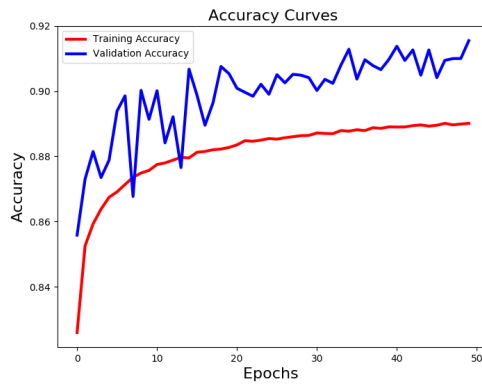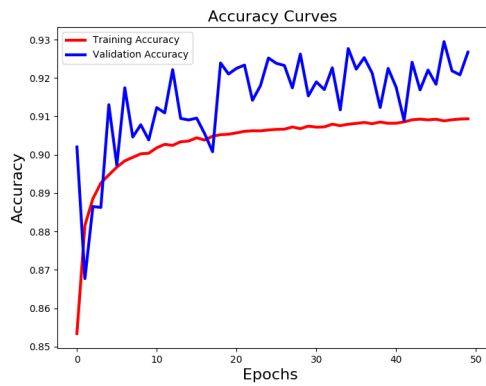
(b) Accuracy plot for augmented dataset and SGD.

(c) Accuracy plot for standard dataset and Adam.
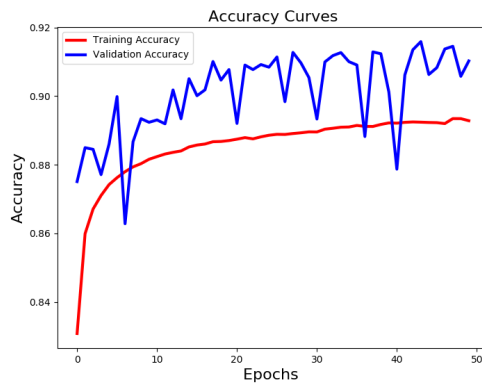
(d) Accuracy plot for augmented dataset and Adam.

Figure 4.9: Different accuracy plots for Patient 2 for the third architecture.

Results presented in Table 4.21 are akin among the four different methods: except for FDR and the precision in the last approach, each metric produces similar performances and results. Additionally, all the confusion matrix in the following tables are showing comparable results in the prediction; the only exceptions are given by Table 4.20, which contains the most significant number of false-negative for the last class, due to the usage of a dataset with an expanded sample size of the core class.

The learning curves in Fig. 4.9 are close to the other architectures' learning curves. They also present the same irregular curves for the validation set. Presented curves have a structure that is comparable with the different learning curves in Table 4.7 and 4.8 because of the irregular learning curves for the validation set and their higher accuracy compare to the training curves.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.952	0.963	0.053	0.02	0.038	0.099	0.962	0.947	0.98	0.901
		II	0.931	0.878	0.134	0.109	0.042	0.053	0.958	0.866	0.891	0.947
		III	0.958	0.673	0.114	0.458	0.038	0.006	0.962	0.886	0.542	0.994
		IV	1.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
		Average	**0.960**	**0.629**	**0.325**	**0.397**	**0.030**	**0.040**	**0.971**	**0.679**	**0.603**	**0.961**
	Adam	I	0.96	0.969	0.04	0.021	0.041	0.077	0.959	0.96	0.979	0.923
		II	0.935	0.883	0.141	0.091	0.034	0.055	0.966	0.859	0.909	0.945
		III	0.963	0.706	0.086	0.425	0.035	0.005	0.965	0.914	0.575	0.995
		IV	1.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0
		Average	**0.965**	**0.639**	**0.317**	**0.384**	**0.028**	**0.034**	**0.972**	**0.683**	**0.616**	**0.966**
Data Augment.	SGD	I	0.955	0.966	0.046	0.022	0.043	0.087	0.957	0.954	0.978	0.913
		II	0.933	0.88	0.139	0.1	0.039	0.055	0.961	0.861	0.9	0.945
		III	0.961	0.686	0.133	0.433	0.034	0.007	0.966	0.867	0.567	0.993
		IV	0.998	0.0	1.0	1.0	0.002	0.0	0.998	0.0	0.0	1.0
		Average	**0.962**	**0.633**	**0.33**	**0.389**	**0.029**	**0.037**	**0.97**	**0.67**	**0.611**	**0.963**
	Adam	I	0.955	0.966	0.045	0.022	0.043	0.086	0.957	0.955	0.978	0.914
		II	0.937	0.889	0.117	0.105	0.041	0.047	0.959	0.883	0.895	0.953
		III	0.968	0.715	0.182	0.365	0.024	0.009	0.976	0.818	0.635	0.991
		IV	0.994	0.026	0.5	0.987	0.006	0.0	0.994	0.5	0.013	1.0
		Average	**0.969**	**0.680**	**0.073**	**0.344**	**0.029**	**0.027**	**0.971**	**0.927**	**0.656**	**0.973**

Table 4.21: Comparison of the statistical information for prediction on Patient 2 based on two different datasets for the third architecture.

4.6 Experimental Results

The last section presents the overall experiment results for all the patients, with a focus on the average outcomes obtained for every single test and train. All architectures described in the previous sections achieved promising outcomes because they are successfully predicting, even with some minor mistakes and some false-positive results, the different regions of the ischemic strokes in the affected brains. They are all able to detect in which brain's hemisphere the stroke is located and, above all, they are capable of identifying with high accuracy all various shapes of the different areas of a stroke. A detailed overview of the many outcomes and the mixed results for all patients is presented in Appendix A.

4.6.1 Accuracy & Standard Deviation

Tab. 4.22 displays all the accuracies and losses for the K-Fold Cross-validation among all patients. It also shows the average of these results plus the standard deviation of all the outcomes. Except for the column for the patient with ID 11, it is possible to evince that average training, loss, and testing present analogous results among themselves.

| | | | | **Patient chosen with K-fold Cross Validation** | | | | | | | | | | **Avg.** | σ |
				P.2	P.3	P.4	P.5	P.6	P.7	P.8	P.9	P.10	P.11		
Normal	Train Acc. (%)	SGD	I	92.61	92.10	92.36	92.88	91.98	92.66	92.37	92.88	92.80	92.40	**92.50**	**0.299**
			II	91.41	91.34	91.32	91.02	91.35	92.01	91.77	91.67	91.92	91.87	**91.57**	**0.308**
			III	90.44	90.58	90.70	90.61	90.52	90.61	90.45	90.64	90.58	90.53	**90.57**	**0.078**
		Adam	I	93.00	93.06	93.12	93.23	93.28	92.65	93.12	93.23	93.00	93.20	**93.09**	**0.173**
			II	92.17	92.20	92.23	92.22	92.12	92.18	91.94	92.24	92.16	92.14	**92.16**	**0.082**
			III	90.94	91.00	90.83	91.00	90.97	90.91	90.61	90.95	90.95	90.92	**90.91**	**0.110**
	Train Loss	SGD	I	0.222	0.249	0.245	0.201	0.257	0.224	0.233	0.214	0.215	0.234	**0.229**	**0.017**
			II	0.275	0.283	0.290	0.304	0.287	0.252	0.259	0.264	0.255	0.259	**0.273**	**0.017**
			III	0.327	0.323	0.317	0.320	0.326	0.323	0.321	0.319	0.324	0.325	**0.323**	**0.003**
		Adam	I	0.204	0.201	0.197	0.200	0.191	0.191	0.196	0.192	0.202	0.196	**0.197**	**0.004**
			II	0.247	0.246	0.243	0.246	0.247	0.240	0.248	0.238	0.246	0.243	**0.244**	**0.003**
			III	0.311	0.309	0.317	0.308	0.310	0.314	0.316	0.309	0.311	0.312	**0.312**	**0.003**
	Test Acc(%)	SGD	I	90.39	92.45	91.78	94.36	93.88	92.46	95.59	95.12	94.24	80.22	**92.05**	**4.225**
			II	88.37	89.75	92.20	91.05	93.67	92.71	95.14	96.01	94.65	80.31	**91.39**	**4.340**
			III	88.33	91.50	91.45	90.04	92.82	90.01	94.02	95.40	92.63	80.44	**90.66**	**3.928**
		Adam	I	89.32	93.72	93.22	94.58	96.16	92.65	96.41	96.02	93.19	78.41	**92.37**	**5.063**
			II	90.55	92.43	94.83	94.10	93.73	92.96	95.48	95.92	93.81	76.04	**91.98**	**5.514**
			III	88.01	91.19	91.65	91.70	92.24	89.51	94.05	95.11	93.30	79.73	**90.65**	**4.134**
Data Augmentation	Train Acc. (%)	SGD	I	92.11	92.21	92.70	92.57	91.54	92.39	92.25	92.88	92.51	90.46	**92.16**	**0.666**
			II	90.64	90.85	90.75	90.26	90.76	90.45	90.35	90.81	90.61	91.26	**90.67**	**0.272**
			III	88.73	88.94	88.81	89.10	88.62	89.18	88.91	89.32	88.95	88.78	**88.93**	**0.204**
		Adam	I	92.79	92.78	92.98	93.01	92.85	92.95	92.66	93.07	92.88	92.73	**92.87**	**0.125**
			II	91.75	91.58	91.81	91.72	91.65	91.64	91.72	91.93	91.60	91.62	**91.70**	**0.103**
			III	89.28	89.38	89.36	89.46	89.22	89.39	89.13	89.52	89.14	89.38	**89.33**	**0.124**
	Train Loss	SGD	I	0.248	0.243	0.225	0.224	0.271	0.232	0.239	0.216	0.228	0.318	**0.244**	**0.029**
			II	0.304	0.298	0.298	0.315	0.295	0.307	0.308	0.293	0.304	0.276	**0.300**	**0.010**
			III	0.379	0.373	0.376	0.365	0.382	0.365	0.372	0.358	0.373	0.376	**0.372**	**0.007**
		Adam	I	0.218	0.216	0.210	0.207	0.214	0.210	0.218	0.203	0.214	0.220	**0.213**	**0.005**
			II	0.261	0.266	0.259	0.261	0.262	0.264	0.261	0.251	0.265	0.267	**0.262**	**0.004**
			III	0.359	0.357	0.355	0.354	0.362	0.357	0.363	0.352	0.366	0.356	**0.358**	**0.004**
	Test Acc(%)	SGD	I	90.84	92.84	93.45	93.30	92.83	91.40	95.10	95.78	93.52	80.21	**92.12**	**4.225**
			II	90.02	92.03	92.71	92.21	91.77	88.17	94.94	94.47	91.18	78.35	**90.59**	**4.482**
			III	87.49	90.99	90.08	90.99	93.12	88.41	92.35	95.46	90.97	78.32	**89.82**	**4.392**
		Adam	I	91.03	93.53	94.49	93.37	93.58	92.31	96.16	95.76	92.84	79.86	**92.56**	**4.475**
			II	88.01	92.33	94.01	93.66	95.16	92.41	95.69	96.27	93.68	78.71	**91.99**	**4.942**
			III	88.77	91.25	89.25	89.34	91.82	89.29	93.20	93.84	91.31	77.00	**89.51**	**4.485**

Table 4.22: Accuracy & Loss for all models.

4.6.2 Experimental Evaluation

Regarding the results obtained with the other patients, the three architectures proposed achieved similar results among training, testing, and loss function. All patients shows similar percentages during the training and testing, with the only exception of patient with ID 11, as shown in Table 4.22 and in the visualization results in Sec. A.1.10, A.2.10, A.3.10. Architecture 1 shows a training average among the four experiments of 92.66%,

a testing average of 92.28% and an average result in the two-loss functions of 0.237 for SGD and 0.205 for Adam. The second architecture produces an average training of 91.53%, an average test accuracy of 91.31%, 0.287 for the SGD loss function, and 0.253 regarding the Adam loss. Architecture 3 has an average accuracy during the train of 89.94% and the test of 90.09%; additionally, the results for the loss functions are 0.348 for SGD and 0.341 for Adam.

The first architecture obtains the best accuracy among the three architectures proposed. Furthermore, the training loss achieved the best result for this architecture, both for SGD and Adam functions. The worst architecture, among the three presented, is the last architecture because of its peculiar structure: since one of its first layers shrink the 3D volume in a one-dimensional volume, a relevant numbers of data is lost during the process, leading almost to a 3% less accurate results compared to the first architecture. Nevertheless, differences among the architectures are not too noticeable, as presented in detail in Appendix A. Plus, the average results are based on all the methods and the two datasets; thus, the process that achieved the highest accuracy is the first architecture using the standard dataset and Adam optimizer (experiment 3). However, architectures that show the best shapes for penumbra and core areas are the ones using the augmentation dataset due to its presence of more samples for the core area. Test results present a high standard deviation among all the architectures because the testing dataset does not have a proper sample size.

5

Pixel by Pixel Segmentation Approach
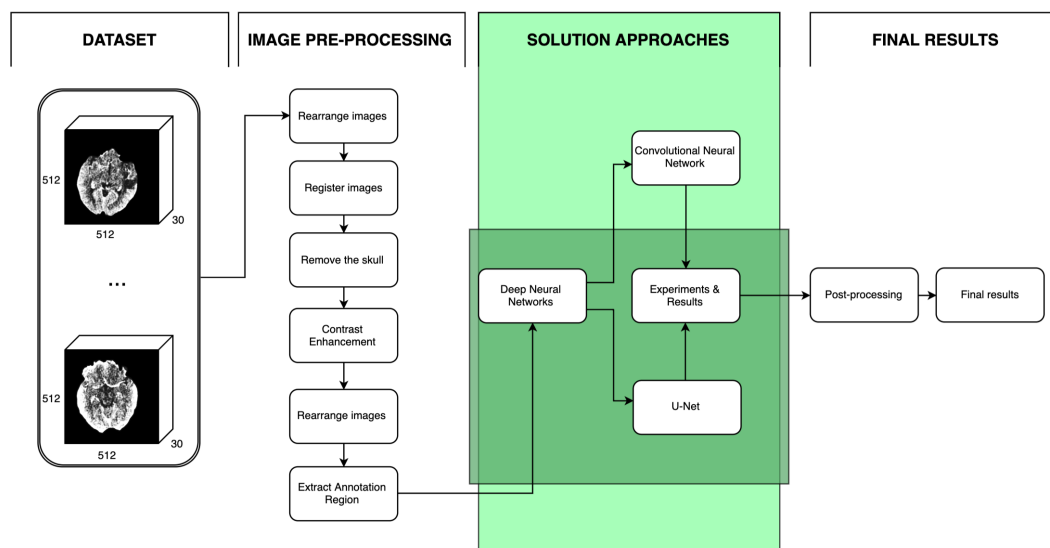
Figure 5.1: Focus of chapter five.

The chapter presents in detail the last architecture proposed, with a highlight of the different results achieved with the distinct experiments. Additionally, visual examples are presented to show a predicted brain section of the patient with ID 2. All the other predicted output for all the patients, with their corresponding confusion matrices and results, are displayed in Appendix B.

5.1 Introduction

The success of CNN is mainly based on managing datasets with thousands of annotated training images. The proposed approach, based on the U-Net approach, does not need a vast amount of images in the dataset because, as stated in the original article: *"the architecture consists of a contracting path to capture context and a symmetric expanding way that enables precise localization"* [60]. U-Net is trained with two different datasets containing hundreds of images which are decomposed in smaller sub-images. One of the two train datasets is also composed using data augmentation techniques for one of the classes involved. The primary consideration for using this approach is due to the good results achieved with the Tile Classification of 16x16 tiles in the previous chapter.

Fig. 5.2 displays the steps involved for the creation of predicted images using the proposed approach; as highlighted by the image, to produce the final result it is not necessary to use any post-processing steps since the output is directly given as a 2D vector, easily converted in an image.
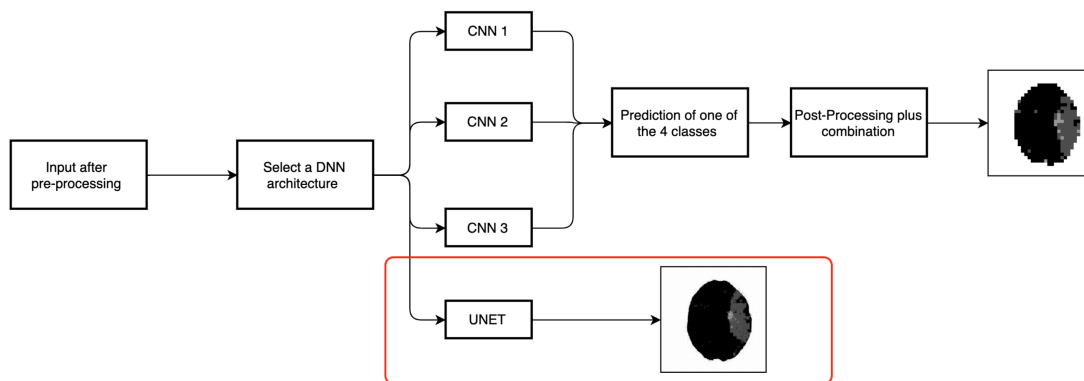
Figure 5.2: Overview of the input and output section for the U-net architecture.

5.2 Existing Approaches/Baselines

The approach implemented in this chapter, called Pixel by Pixel Segmentation approach, was described in detail in Sec. 2.2.7. It maps every input (a series of pixels) over the
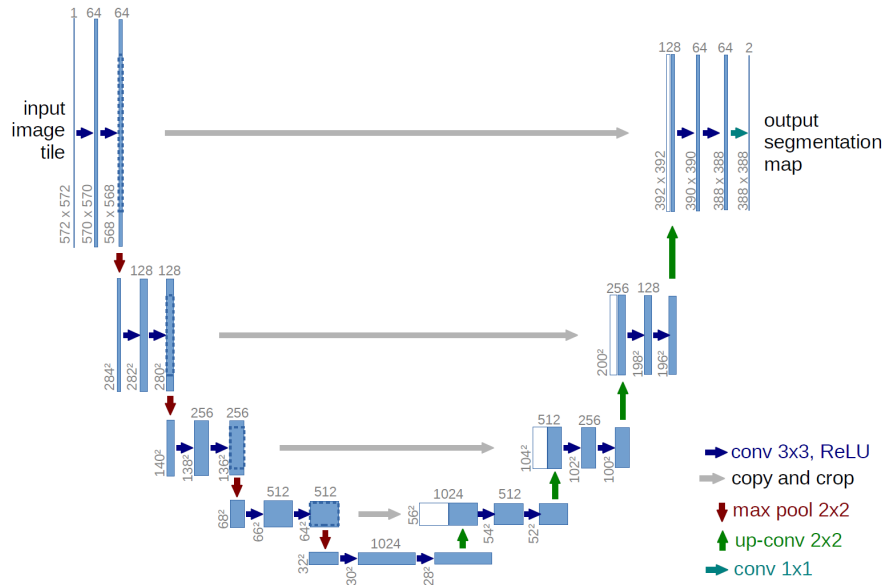
Figure 5.3: U-Net architecture. The figure is reprinted in unaltered form from [60].

same section of the brain during the passage of the contrast agent, with the output, a single pixel in the image target. This approach is based on a deep neural network, called U-Net, created by a group of researchers of the University of Freiburg [60] to segment in a fast way neuronal structures in electron microscopic stacks. Fig. 5.3 displays the architecture of the network implemented during their study. This thesis faces a different problem compared with the segmentation of neuronal structures; thus, the architecture was implemented with a different framework and modified from the original structure to allow it take in input a list of CTP images.

5.3 Analysis

The network implemented for the Pixel by Pixel Segmentation approach has a structure comparable to the one in Fig. 5.4. The proposed network contains 17 layers plus the output layer, which also executes a reshape operation. It has a larger number of layers, compared to the other architectures, because of its structure composition, composed of two main parts: a contracting part (from layer one until layer 10) and an extensive section (from layer 11 until the output layer). Additionally, the number of total parameters involved in the network is very high; which can lead to a possible disadvantage during the training. The network is slower than the other architectures, but the outcome that it produces is way more accurate compared to the Tile Classification approach, as it is possible to evince from Table 5.3. The training and testing evaluation, which is composed of 50 epochs like the other architectures, takes approximately 12 hours in total or 15 minutes on average per epoch, depending on the dataset involved.

5.4 Proposed Architecture

Table 5.1 describes in a more detailed way all layers for the Pixel by Pixel Segmentation approach. The first layer performs a convolutional operation with a ReLU activation function and a kernel size of $(3,3,N)$, where N is the number of time-series for the same brain section; it also executes a batch normalization. The second layer runs an average pooling function, over the output of the first layer, with a window shape of $(3,3,N)$; this layer is essential because it flats the input volume to a one-dimensional vector. The third and fourth layers represent two convolutional layers, using a ReLU activation function and a kernel size of $(3,3,1)$ and a batch normalization operation. Layer five executes a max-pooling process, with a window size of $(2,2,1)$, to halve the width and height of the input. The next three layers operate in the same way as the last three: two convolutional layers with a ReLU activation function and one max pooling layer. Subsequently, the other two convolutional layers and two batch normalization are executed in order to arrive on the eleventh layer: it represents the fundamental step in the architecture. The layer executes a transposition of a convolutional layer; the outcome of the previous transposition is concatenated with the output of the seventh layer, as it is possible to evince from Fig. 5.3. After this concatenation, which creates an input of dimension $(8,8,2)$, two convolutional operations are performed plus two batch normalization. Layer thirteen implements a max-pooling operation with a window size of $(1,1,2)$ to reduce the output to a one-dimensional volume. From layer fourteen to layer sixteen included, the same operation is operated to bring back the output to the initial dimension of $(16,16,1)$. The second-last layer executes a convolutional layer with a Sigmoid activation function to

U-Net Architecture		
#	**Layer**	Param.
1	conv3D	4336
	batch_norm	64
2	avg_pool3D	0
3	conv3D	4640
	batch_norm	128
4	conv3D	18496
	batch_norm	256
5	max_pool3D	0
6	conv3D	36928
	batch_norm	256
7	conv3D	73856
	batch_norm	512
8	max_pool3D	0
9	conv3D	147584
	batch_norm	512
10	conv3D	295168
	batch_norm	1024
11	Conv3DTranspose	131200
	concatenate	0
	conv3D	147584
	batch_norm	512
12	conv3D	73792
	batch_norm	256
13	max_pool3D	0
14	Conv3DTranspose	16448
	concatenate	0
	conv3D	18464
	batch_norm	128
15	conv3D	9248
	batch_norm	128
16	max_pool3D	0
17	conv3D	33
18	reshape	0

Table 5.1: Layers summary of Pixel by Pixel Segmentation approach.

transform the output into the interval $[0, 1]$. The number of total parameters involved in the network is 981553; the trainable parameters are 979665 while the non-trainable parameters are 1888.
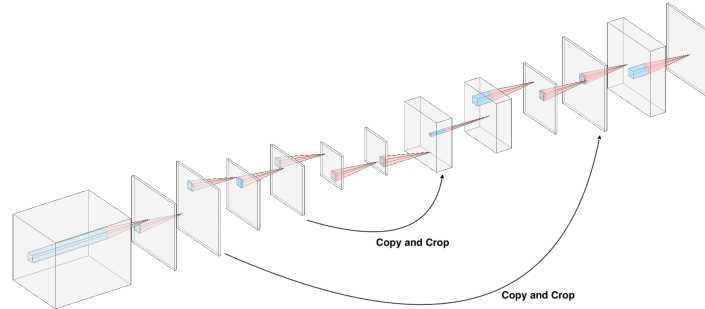
Figure 5.4: Pixel by Pixel segmentation network structure.

5.5   Experimental Setup and Data Set

"Dataset 1" and "Dataset 2" are involved in the same way as they were used for the other approach, as illustrated in Chap. 4. The training of the architecture bases its foundations on the K-Fold Cross-validation of the dataset. Thus, experiments are performed for ten different datasets due to a random exclusion of one patient for testing purposes. During the training and testing period, it is only evaluated the accuracy percentage and the output of the loss function. For this architecture, the Dice loss is used (Sec. 2.2.2) because of its property to gauge the similarity of two regions.

5.6   Post-processing

The Pixel-by-Pixel Segmentation approach does not follow the same post-processing technique illustrated in the previous chapter; conversely, since the output produced is a tile due to the property of a U-Net architecture, it just expands the RGB levels of the tile from 1 to 3. Additionally, since the integers inside the predicted tile are in the interval $[0, 1]$, they are all multiplied by 255 to produce the corresponding RGB values. The post-processing step converts every predicted pixel (X) to an RGB decimal code: (X, X, X).

5.7 Experimental Results

5.7.1 Visualization Results

The Brain sections created with the architecture described above, present almost the same structure of the ground truth image. The best result achieved in Table 5.2 is represented by the image realized with the data augmentation dataset and the SGD optimizer (experiment 2). However, also the other method, which uses the SGD function, aims for essential results. The shape of the final brain section generated with the data augmentation dataset and the Adam optimizer (experiment 4) looks almost perfect despite some small areas that are labeled with a background class inside the brain region. The generated image with Adam optimizer and standard dataset (experiment 1) yields the right shapes for all classes, but overall, the image looks indented.

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam

Table 5.2: Example of brain section comparison for patient 2 with different techniques on the U-net architecture.

5.7.2 Accuracy Results

Fig. 5.5 represents the different learning curves for this approach using particular datasets and distinct optimizer algorithms. Equivalently to the learning curves analyzed for the various architectures in Chap. 4, these plots reveal a high learning rate in the training curve along with a scattered validation accuracy curve due to the sample size of the validation set. Additionally, presented curves display an improvement in their learning results even around the 50th epoch, which can lead to better outcomes if these models are trained with a higher number of epochs and a larger dataset. The realization of the training with a more significant amount of epochs could be an interesting field of study for the future.

The network training strategy produces precise images, reaching a training accuracy of 98.27% and testing accuracy of 96.85% on average. The average result for the Dice loss is 0.017. Results presented in Fig. 5.3 are very promising: they lead to higher accuracy, both in training and testing, compared to the approaches proposed in Chap. 4,

(a) Accuracy plot for normal dataset and SGD.

(b) Accuracy plot for augmented dataset and SGD.

(c) Accuracy plot for standard dataset and Adam.

(d) Accuracy plot for augmented dataset and Adam.
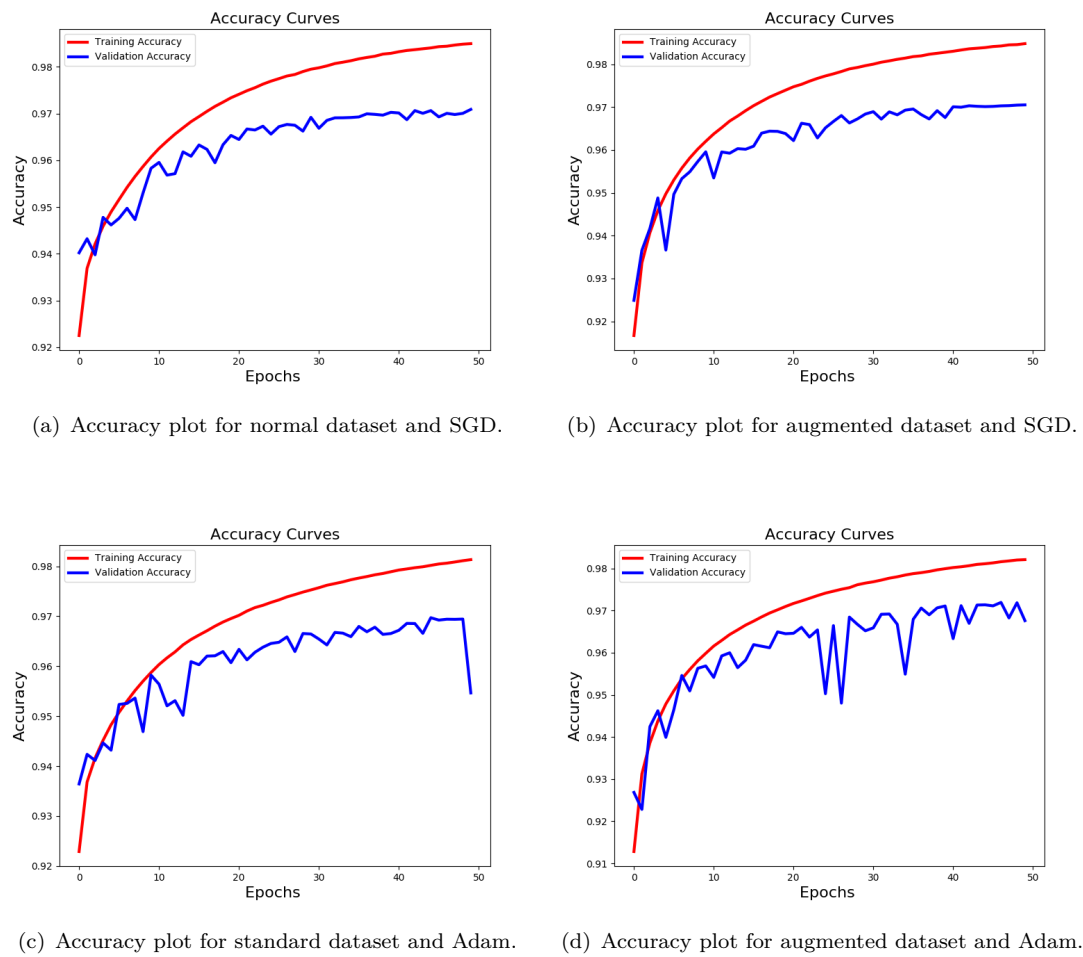
Figure 5.5: Different accuracy plots for Patient 2 for the U-Net architecture.

the produced shapes of the distinct areas are more accurate even if the execution time for each epoch takes more time than the other architectures. All the images predicted with the different methods are shown in Appendix B. In the same way as the other architectures, the standard deviation of the test results is high due to the not appropriate sample size of the dataset.

6

Results & Future Works

Figure 6.1: Focus of chapter six.

The chapter examines the outcomes achieved with the two approaches described in the previous chapters, Tile Classification, and Pixel by Pixel Segmentation, with a focus on the different performances and the comparison of the predicted outcomes of the various architectures involved in the thesis. Moreover, a section to address the possible improvements and future works for the thesis is presented.

6.1 Results & Discussion

6.1.1 Aggregate Confusion Matrices for Architecture 1

The aggregate confusion matrices for all the final images of the patients with "Dataset 1", SGD and ADAM optimizer and architecture 1 (Sec. 4.3.1) are display in Table 6.1 and Table 6.2:

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	95907	680	76	16
Class	*II*	2019	34550	972	26
	III	675	1952	6890	133
	IV	3	44	50	391

Table 6.1: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	**96092**	649	70	20
Class	*II*	1991	34251	699	8
	III	546	2233	**7186**	98
	IV	3	57	37	444

Table 6.2: Adam optimizer.

Furthermore, the aggregate confusion matrices for all the final images of the patients with "Dataset 2", SGD and ADAM optimizer are display in Table 6.3 and Table 6.4:

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	95824	677	110	6
Class	*II*	2203	**34832**	1115	10
	III	530	1536	6570	40
	IV	49	189	183	510

Table 6.3: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	96018	659	118	8
Class	*II*	2081	34817	951	5
	III	471	1593	6718	15
	IV	27	145	216	**542**

Table 6.4: Adam optimizer.

The distinct experiments, presented in the previous confusion matrices, show promising and similar results; the decision to select one of these experiment setups should be based on which class is more important to predict. Even if the aggregated outcomes are showing analogous values in the predicted classes on the various confusion matrices, Table 6.2 displays the best output for the prediction background and penumbra classes, while Table 6.3 presents the best result for the predicted brain class and finally Table 6.4 shows the best outcome for the predicted core class.

	Experiment			
	1	*2*	*3*	*4*
MSE	0.0079	**0.0068**	0.0082	0.0073

Table 6.5: Mean squared error for the predicted classes for architecture 1.

Table 6.5 presents the mean squared error for the predicted classes. The lowest MSE value is given by the experiment 2, which is expected since the corresponding confusion matrix (Table 6.2) shows excellent results for two classes among the total four.

6.1.2 Aggregate Confusion Matrices for Architecture 2

Table 6.6 and Table 6.7 display the aggregate confusion matrices for all the final images of the patients with "Dataset 1", SGD and ADAM optimizer and architecture 2 (Sec. 4.3.2):

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	95645	759	113	32
	II	2324	34717	1148	34
	III	641	1721	6688	233
	IV	2	31	25	271

Table 6.6: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	95575	673	98	19
	II	11033	31134	904	13
	III	686	1930	**6896**	101
	IV	5	31	83	436

Table 6.7: Adam optimizer.

Furthermore, the aggregate confusion matrices for all the final images of the patients with "Dataset 2", SGD and ADAM optimizer are displayed in Table 6.8 and Table 6.9:

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	95563	1170	292	19
	II	2421	34265	1393	12
	III	579	1455	6048	40
	IV	38	319	271	499

Table 6.8: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	**95739**	855	176	13
	II	2258	**34819**	1138	8
	III	576	1420	6487	34
	IV	33	140	173	**515**

Table 6.9: Adam optimizer.

Compare to the previous architecture, the confusion matrices presented in this section show similar results even if they give less accuracy than the first architecture. Nevertheless, among the four tables presented here, the one that displays the best results is Table 6.9 since it contains the three best outcomes per predicted class over the four total classes. The results presented in Table 6.10 confirm the previous statement because the lowest MSE discovered for this architecture is given by experiment 4.

	Experiment			
	1	*2*	*3*	*4*
MSE	0.0104	0.0091	0.0144	**0.0088**

Table 6.10: Mean squared error for the predicted classes for architecture 2.

6.1.3 Aggregate Confusion Matrices for Architecture 3

Table 6.11 and Table 6.12 present the aggregate confusion matrices for all the final images of the patients with "Dataset 1", SGD and ADAM optimizer and architecture 3 (Sec. 4.3.3):

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	**95493**	692	125	22
	II	2374	**34549**	1371	62
Class	*III*	753	1924	6403	218
	IV	3	37	93	265

Table 6.11: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	95252	671	125	33
	II	2688	34440	1284	56
Class	*III*	671	2028	**6482**	196
	IV	10	61	103	284

Table 6.12: Adam optimizer.

Furthermore, the aggregate confusion matrices for all the final images of the patients with "Dataset 2", SGD and ADAM optimizer are display in Table 6.13 and Table 6.14:

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	95209	862	205	25
	II	2716	34469	1523	37
Class	*III*	651	1650	5910	112
	IV	30	251	342	392

Table 6.13: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	95267	886	228	29
	II	2638	34197	1438	25
Class	*III*	639	1845	5882	69
	IV	55	303	439	**444**

Table 6.14: Adam optimizer.

The aggregate results of the last architecture are in line with its average accuracy, as shown in Table 4.22. The confusion matrices show total values that are lower than the other two architectures because of the peculiar structure of the model since it shrinks the input during one of the first layers. This structure, as proven by the analysis of the various results, loses some useful information during the training, leading to results that are not accurate as of the other architectures. The best experiment for this model is given by Table 6.11, which produces two of the four highest values for the predicted classes.

	Experiment			
	1	*2*	*3*	*4*
MSE	0.0130	**0.0124**	0.0159	0.0166

Table 6.15: Mean squared error for the predicted classes for architecture 3.

The lowest MSE is given by experiment 2, as shown in Table 6.15, despite the two of the highest predicted classes were produced by the first experiment. This could be caused by analogous results between these experiment 1 and experiment 2.

6.1.4 Aggregate Results for Architecture 4

Exp.	Train Acc. (%)	Train Loss	Test Acc. (%)	Test Loss
I	**98.37**	**0.014**	**97.06**	**0.027**
II	98.11	0.017	96.73	0.034
III	97.88	0.019	96.61	0.031
IV	98.24	0.016	97.00	0.030

Table 6.16: Average statistical information for the experiments on the different architectures.

Table 6.16 shows the aggregate statistical metrics for the training and testing across the four experiments. These statistics produce very high accuracy percentages, showing that the results are promising and a series of future works are necessary to validate the architecture proposed and their related outcomes.

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam

Table 6.17: Example of brain section comparison for patient 8 with different techniques on the U-net architecture.

Future works are necessary because the predicted image by the experiments are producing encouraging and, at the same time, ambiguous outputs. The predicted images, presented in Table 6.17, display a segmentation of the various classes comparable to the image ground truth; the images correspond to the brain section of a patient with ID 8. Differently, the predicted images related with the patient with ID 11, elaborated with the corresponding sample set, present a random segmentation over the example images in Table 6.18; the other outcomes are presented in Appendix B.1.10 and they display the same random segmentation for penumbra and core classes.

This phenomenon is notable only for this patient of the dataset. Additionally, all the other architectures present this type of event for the patient with ID 11. It is also possible to notice in Table 4.22 and Table 5.3 that the training and testing accuracy for the patient with ID 11 is radically inferior compared to the other patient's accuracies.

Ground Truth	Normal Dataset		Data Augmentation	
	SGD	Adam	SGD	Adam
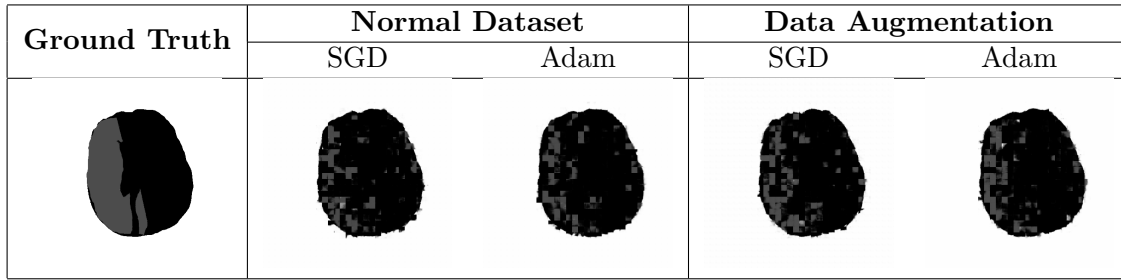
Table 6.18: Example of brain section comparison for patient 11 with different techniques on the U-net architecture.

Future researches with a larger dataset should be indispensable to check and validate the various predictions.

6.1.5 Overall Results

Arch.	Exp.	Test Acc. (%)	mAP (%)	Jaccard (%)	Comput. Time per epoch (s)
1	*I*	92.05	67.63	62.29	720
	II	92.37	66.92	61.93	720
	III	92.12	67.91	63.06	750
	IV	92.56	67.23	62.87	750
2	*I*	91.39	66.97	61.05	600
	II	91.98	65.62	60.37	600
	III	90.59	66.88	62.29	670
	IV	91.99	67.07	62.18	670
3	*I*	90.66	65.50	59.47	450
	II	90.65	64.58	58.72	450
	III	89.82	65.74	60.02	550
	IV	89.51	64.18	58.43	550
4	*I*	**97.06**	**70.92**	**67.64**	850
	II	96.73	69.57	65.79	850
	III	96.61	69.96	66.34	1000
	IV	97.00	70.36	67.06	1000

Table 6.19: Average statistical information for the experiments on the different architectures.

As it is possible to evince from Table 6.19, the four architectures achieved distinct results on testing calculated on the average of the K-Cross Fold Validation. Architecture 4 and the first experiment gives the best overall result. The overall Jaccard and mAP measurements show comparable results among all the architectures. Despite the similar results, architecture 4 presents slighter better outcomes for mAP and Jaccard than the other architectures due to its better accuracy in the experiments.

Architecture 4 displays, in general, the best results among all the other architectures, as it also possible to notice from the predicted images show in Appendix B. The main advantage of this architecture is the level of precision achieved to segment the distinct classes involved. The majority of brain sections predicted present a similar structure in the segmentation for all the classes compare to the linked ground truth images. Penumbra and core classes, the most critical areas to discover for medical doctors, display optimal region shapes for all the patients analyzed except for the one with ID 11; notwithstanding, the dataset does not contain a high number of these classes, just the 17% for "Dataset 1" and 21.5% for "Dataset 2". The results achieved based on these values are very encouraging for future works, as shown in Table 6.17. The disadvantage of the architecture 4 is the amount of time spent to train each different model, independently from the experiment setup: with a GPU Tesla V100-PCIE (32GB) every training epoch of a model with "Dataset 1" takes approximately 850 seconds while a model with "Dataset 2" takes 1000 seconds due to its larger sample size.

Architecture 1, architecture 2, and architecture 3 are achieving complementary results if they are compared only with themselves; the differences are more evident during the comparison of different experiments analyzed with the various architectures. The experiments trained with the augmented dataset ("Dataset 2") display better classification for the core and penumbra classes in contrast with the experiments trained with the "Dataset 1"; however, in some occasions, models are predicting regions of brains as core or penumbra class even if there is no evidence of that in the ground truth image, leading to false-positive classifications, as an example check Tables A.32, A.44, A.75, A.87, A.136, A.142.

One of the main advantages of these CNN architectures is the time spent on training the models. Each architecture created generally takes between 6 and 8 hours to train 50 epochs with a GPU Tesla V100-PCIE (32GB), depending on the dataset. Architecture 3, among all architectures, is the one that spends less time on the training due to its first max-pooling layer which shrinks the 3D input into a one-dimensional vector, reducing drastically the number of parameters to train. Nonetheless, this architecture shows the lowest score both in training and testing in association with the other architectures proposed. The predicted images created with these architectures are less accurate than the images created with architecture 4 because of their primary outcome (a number that indicates the corresponding class) and the post-processing technique (Sec. 4.2) which creates a pixelated image joining together the predicted tiles.

All the different architectures and hyperparameters that were tested performed similar results in general, except the core class prediction during the usage of "Dataset 1" and "Dataset 2". The application of SGD and Adam optimizer functions did not influence the

various outcomes significantly; even if the Adam optimizer generally achieved the same results as SGD in a lower number of epochs. All results achieved with the suggested approaches are promising since the average accuracy during training and testing exceeds 90% for almost every method. Even if all these architectures show essential results, a lot of possible future works and researches are likely within this field, and an essential number of improvements are achievable for the proposed structures. Plus, future work is fundamental due to the limitation of the involved dataset: the sample size was too small, the DICOM images are elaborated with the same scanner, and the patients were taken from the same hospital. Finally, the manually annotated images are produced in an approximately way, highlighting only the central regions of the infarcted areas, and by a single doctor, the external supervisor of the thesis, who is an expert in the field but there was no consensus over the various labels.

6.2 Future Works

This thesis presents four strategies to discover and predict the various regions of an ischemic stroke using a dataset of CTP images; however, it could be possible to improve the achieved results with future researches. An interesting study for the future could be to test all the various methods proposed with a dataset that contains a more significant number of patients to prove the validity of these architectures. Another interesting future work could be to combine the predicted section of the same brain and create a 3D model of the ischemic stroke. Additionally, since in one of the predicted patient (Appendix A.1.10, A.2.10, A.3.10, B.1.10) the approaches generate false-positive leading to random classification and segmentation in isolated areas of the brain, a possible future research could be to find an algorithm to improve the efficiency of the post-processing technique to avoid these false-positive discoveries. Furthermore, the algorithm proposed to extract the skull from CTP images is not perfect; thus, it can be improved to test if it is lead to better results.

7

Conclusion

The thesis presented four different architectures, all based on a CNN structure, to predict in which part of the brain a stroke can occur and to classify or to segment the various regions, penumbra, and core, of a possible ischemic stroke. The input of the distinct CNNs is based on a dataset of CPT images over time from a group of 11 patients. A list of vectors made of a series of 16x16 tiles is extrapolated with a sliding window technique from these CTP images of the various brain sections created over a period of time. Each vector contains the same portion of the pixels of a specific brain section during the injection of an iodinated contrast agent inside cubital veins to enhance contrast in the tissue.

The output of each model is the same section of the brain included as the input, with the addition of manual annotation of the possible areas of the penumbra and core made by a specialist. This approach is suggested to have the possibility to classify each vector of tile extracted with a specific class, corresponding to one of the possible outcomes to recreate, at the end of the training and testing of the model, a predicted brain section image. The results obtained in this thesis work are very encouraging: the best test result after K-Cross Fold Validation is found by using a Pixel by Pixel Segmentation approach (Chap. 5), learned with an SGD optimizer, the "Dataset 1", and the Dice loss function.

77

The best overall result is 97.06% accuracy over all the four classes (background, brain, penumbra, core). The accuracy achieved with one of the proposed architecture should be the foundation for future researches: a more extensive and heterogeneous dataset, an improved version of the algorithm to extract the skull, a new version of the model containing a more significant number of hidden layers and an increment of the training time.

List of Figures

List of Tables

A

Results for Tile Classification Approach

The discussion and comparison of the results of the Tile Classification approach are faced in Chap. 6. The chapter displays the different results for the different architectures that are based on this approach, described in Chap. 4. The final training and testing accuracy percentages plus result for the loss equations are presented in Table 4.22. A table with a comparison of all the brain section predicted with distinct methods is presented for all patients. Furthermore, the four confusion matrices are shown, and the final table containing useful statistical information regarding the predicted output. Tables in the "visualization results" sections are structured in the following way: the first row contains the ground truth images for the brain section; the second and fourth rows show the outcome after the training with the SGD optimizer function with respectively the standard dataset "Dataset 1" and the augmented dataset "Dataset 2". Row three displays the results for the model using the augmented dataset and the SGD optimizer (experiment 2), while the last row represents the outcomes for the Adam optimizer and the augmented dataset (experiment 4)augmented.

A.1   Results for architecture 1

This section contains all predicted brain sections for the first architecture, described in Sec. 4.3.1, with all various methods implemented. Among all patients outcome, results

for the patient with ID 2 does not contain the statistical table and the confusion matrices, since they were already displayed in Sec. 4.3.1.

A.1.1   Patient 2

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.1: Brain sections compared with different techniques of the first architecture on patient 2.

A.1.2 Patient 3

Visualization Results

Table A.2: Brain sections compared with different techniques of the first architecture on patient 3.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.3 and A.4) and with the augmented dataset (Tables A.5 and A.6). Table A.7 displays

the statistical information for different classes.

		Actual Class			
		I	II	III	IV
Pred.	I	8783	87	18	5
Class	II	227	3499	67	7
	III	75	163	331	11
	IV	0	3	4	32

Table A.3: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred.	I	8821	66	10	3
Class	II	230	3509	48	1
	III	35	167	360	6
	IV	1	6	3	46

Table A.4: Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred.	I	8738	58	9	0
Class	II	272	3528	71	3
	III	61	148	323	5
	IV	14	18	17	47

Table A.5: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred.	I	8745	59	10	3
Class	II	278	3549	59	1
	III	46	123	341	0
	IV	16	19	11	52

Table A.6: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.969	0.977	0.033	0.012	0.026	0.068	0.974	0.967	0.988	0.932
		II	0.958	0.927	0.067	0.079	0.031	0.027	0.969	0.933	0.921	0.973
		III	0.975	0.662	0.212	0.429	0.019	0.007	0.981	0.788	0.571	0.993
		IV	0.998	0.681	0.418	0.179	0.001	0.002	0.999	0.582	0.821	0.998
		Average	**0.975**	**0.812**	**0.182**	**0.175**	**0.019**	**0.026**	**0.981**	**0.817**	**0.825**	**0.974**
	Adam	I	0.974	0.981	0.029	0.009	0.019	0.06	0.981	0.971	0.991	0.94
		II	0.961	0.931	0.064	0.074	0.029	0.025	0.971	0.936	0.926	0.975
		III	0.98	0.728	0.145	0.366	0.016	0.005	0.984	0.855	0.634	0.995
		IV	0.998	0.821	0.179	0.179	0.001	0.001	0.999	0.821	0.821	0.999
		Average	**0.978**	**0.865**	**0.104**	**0.158**	**0.016**	**0.023**	**0.984**	**0.895**	**0.842**	**0.977**
Data Augment.	SGD	I	0.969	0.977	0.038	0.008	0.016	0.077	0.984	0.962	0.992	0.923
		II	0.957	0.925	0.06	0.089	0.036	0.024	0.964	0.94	0.911	0.976
		III	0.977	0.675	0.231	0.399	0.017	0.008	0.983	0.769	0.601	0.992
		IV	0.996	0.623	0.145	0.51	0.004	0.001	0.996	0.855	0.49	0.999
		Average	**0.975**	**0.799**	**0.122**	**0.252**	**0.018**	**0.028**	**0.982**	**0.878**	**0.748**	**0.973**
	Adam	I	0.969	0.977	0.037	0.008	0.017	0.076	0.983	0.963	0.992	0.924
		II	0.96	0.929	0.054	0.087	0.035	0.021	0.965	0.946	0.913	0.979
		III	0.981	0.733	0.19	0.331	0.013	0.006	0.987	0.81	0.669	0.994
		IV	0.996	0.675	0.071	0.469	0.003	0.0	0.997	0.929	0.531	1.0
		Average	**0.976**	**0.829**	**0.088**	**0.224**	**0.017**	**0.026**	**0.983**	**0.912**	**0.776**	**0.974**

Table A.7: Comparison of the statistical information for prediction on Patient 3 based on two different datasets.

A.1.3   Patient 4

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.8: Brain sections compared with different techniques of the first architecture on patient 4.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.9 and A.10) and with the augmented dataset (Tables A.11 and A.12). Table A.13

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8129	66	9	4
Class	*II*	25	3587	74	8
	III	70	224	886	71
	IV	0	3	15	141

Table A.9: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8129	50	3	6
Class	*II*	45	3558	21	0
	III	53	261	942	30
	IV	0	11	13	190

Table A.10: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8082	47	12	1
Class	*II*	66	3676	73	2
	III	76	132	842	16
	IV	3	25	54	205

Table A.11: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8099	34	23	2
Class	*II*	82	3740	86	2
	III	41	84	810	6
	IV	2	18	67	216

Table A.12: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.987	0.989	0.012	0.01	0.016	0.019	0.984	0.988	0.99	0.981
		II	0.97	0.947	0.076	0.029	0.011	0.03	0.989	0.924	0.971	0.97
		III	0.965	0.793	0.1	0.292	0.03	0.008	0.97	0.9	0.708	0.992
		IV	0.992	0.736	0.371	0.113	0.001	0.006	0.999	0.629	0.887	0.994
		Average	**0.978**	**0.866**	**0.14**	**0.111**	**0.014**	**0.016**	**0.985**	**0.86**	**0.889**	**0.984**
	Adam	*I*	0.988	0.99	0.012	0.007	0.012	0.019	0.988	0.988	0.993	0.981
		II	0.971	0.948	0.083	0.018	0.007	0.033	0.993	0.917	0.982	0.967
		III	0.971	0.832	0.038	0.267	0.028	0.003	0.972	0.962	0.733	0.997
		IV	0.995	0.864	0.159	0.112	0.002	0.003	0.998	0.841	0.888	0.997
		Average	**0.981**	**0.908**	**0.073**	**0.101**	**0.012**	**0.015**	**0.988**	**0.927**	**0.899**	**0.985**
Data Augment.	SGD	*I*	0.985	0.987	0.018	0.007	0.012	0.028	0.988	0.982	0.993	0.972
		II	0.974	0.955	0.053	0.037	0.015	0.021	0.985	0.947	0.963	0.979
		III	0.973	0.823	0.142	0.21	0.018	0.011	0.982	0.858	0.79	0.989
		IV	0.992	0.802	0.085	0.286	0.006	0.001	0.994	0.915	0.714	0.999
		Average	**0.981**	**0.892**	**0.074**	**0.135**	**0.013**	**0.015**	**0.987**	**0.925**	**0.865**	**0.985**
	Adam	*I*	0.986	0.989	0.015	0.007	0.012	0.024	0.988	0.985	0.993	0.976
		II	0.977	0.961	0.035	0.043	0.018	0.014	0.982	0.965	0.957	0.986
		III	0.977	0.841	0.178	0.139	0.011	0.014	0.989	0.822	0.861	0.986
		IV	0.993	0.817	0.044	0.287	0.007	0.001	0.993	0.956	0.713	0.999
		Average	**0.983**	**0.902**	**0.068**	**0.119**	**0.012**	**0.013**	**0.988**	**0.932**	**0.881**	**0.987**

Table A.13: Comparison of the statistical information for prediction on Patient 4 based on two different datasets.

A.1.4   Patient 5

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.14: Brain sections compared with different techniques of the first architecture on patient 5.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.15 and A.16) and with the augmented dataset (Tables A.17 and A.18). Table A.19

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8306	35	13	5
Class	*II*	443	3093	60	4
	III	43	197	946	16
	IV	1	7	10	133

Table A.15: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8411	42	8	6
Class	*II*	314	3152	57	4
	III	68	137	959	16
	IV	0	3	3	132

Table A.16: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8165	26	15	3
Class	*II*	576	3186	108	2
	III	48	90	870	3
	IV	4	32	34	150

Table A.17: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8492	45	31	0
Class	*II*	266	3181	94	2
	III	35	86	841	0
	IV	0	20	63	156

Table A.18: Adam optimizer.

			Metrics								
DS	**Opt.**	**Class**	Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.959	0.969	0.055	0.006	0.012	0.098	0.988	0.945	0.994	0.902
		II	0.944	0.892	0.072	0.141	0.051	0.025	0.949	0.928	0.859	0.975
		III	0.975	0.848	0.081	0.213	0.021	0.007	0.979	0.919	0.787	0.993
		IV	0.997	0.861	0.158	0.119	0.001	0.002	0.999	0.842	0.881	0.998
		Average	**0.969**	**0.893**	**0.091**	**0.12**	**0.021**	**0.033**	**0.979**	**0.908**	**0.88**	**0.967**
	Adam	*I*	0.967	0.975	0.043	0.007	0.012	0.079	0.988	0.957	0.993	0.921
		II	0.958	0.919	0.055	0.106	0.038	0.019	0.962	0.945	0.894	0.981
		III	0.978	0.869	0.066	0.187	0.018	0.006	0.982	0.934	0.813	0.994
		IV	0.998	0.892	0.165	0.043	0.0	0.002	1.0	0.835	0.957	0.998
		Average	**0.975**	**0.914**	**0.082**	**0.086**	**0.017**	**0.027**	**0.983**	**0.918**	**0.914**	**0.974**
Data Augment.	SGD	*I*	0.95	0.96	0.071	0.005	0.01	0.123	0.99	0.929	0.995	0.877
		II	0.937	0.884	0.044	0.177	0.069	0.016	0.931	0.956	0.823	0.984
		III	0.978	0.854	0.153	0.139	0.011	0.013	0.989	0.847	0.861	0.987
		IV	0.994	0.794	0.051	0.318	0.005	0.001	0.995	0.949	0.682	0.999
		Average	**0.965**	**0.873**	**0.08**	**0.16**	**0.024**	**0.038**	**0.976**	**0.92**	**0.84**	**0.962**
	Adam	*I*	0.972	0.978	0.034	0.009	0.017	0.063	0.983	0.966	0.991	0.937
		II	0.961	0.925	0.045	0.102	0.036	0.015	0.964	0.955	0.898	0.985
		III	0.977	0.845	0.183	0.126	0.01	0.015	0.99	0.817	0.874	0.985
		IV	0.994	0.786	0.013	0.347	0.006	0.0	0.994	0.987	0.653	1.0
		Average	**0.976**	**0.884**	**0.069**	**0.146**	**0.017**	**0.023**	**0.983**	**0.931**	**0.854**	**0.977**

Table A.19: Comparison of the statistical information for prediction on Patient 5 based on two different datasets.

A.1.5 Patient 6

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.20: Brain sections compared with different techniques of the first architecture on patient 6.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.21 and A.22) and with the augmented dataset (Tables A.23 and A.24). Table A.25

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8791	70	5	0
	II	109	3516	70	0
	III	53	139	553	0
	IV	1	4	1	0

Table A.21: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8793	49	1	0
	II	98	3447	18	0
	III	65	230	609	0
	IV	0	2	0	0

Table A.22: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8815	59	3	0
	II	85	3458	43	0
	III	52	200	581	0
	IV	2	13	1	0

Table A.23: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8837	85	1	0
	II	56	3478	30	0
	III	59	145	596	0
	IV	2	21	2	0

Table A.24: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.982	0.987	0.018	0.008	0.017	0.037	0.983	0.982	0.992	0.963
		II	0.971	0.947	0.057	0.048	0.019	0.022	0.981	0.943	0.952	0.978
		III	0.98	0.805	0.121	0.258	0.015	0.006	0.985	0.879	0.742	0.994
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.983**	**0.685**	**0.299**	**0.329**	**0.013**	**0.016**	**0.987**	**0.701**	**0.671**	**0.984**
	Adam	*I*	0.984	0.988	0.018	0.006	0.011	0.036	0.989	0.982	0.994	0.964
		II	0.97	0.946	0.075	0.033	0.012	0.029	0.988	0.925	0.967	0.971
		III	0.976	0.795	0.03	0.326	0.023	0.002	0.977	0.97	0.674	0.998
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.982**	**0.682**	**0.281**	**0.341**	**0.011**	**0.017**	**0.988**	**0.719**	**0.659**	**0.983**
Data Augment.	SGD	*I*	0.985	0.989	0.016	0.007	0.014	0.031	0.986	0.984	0.993	0.969
		II	0.97	0.945	0.073	0.036	0.013	0.028	0.987	0.927	0.964	0.972
		III	0.978	0.795	0.075	0.303	0.02	0.004	0.98	0.925	0.697	0.996
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
		Average	**0.983**	**0.682**	**0.291**	**0.337**	**0.012**	**0.016**	**0.988**	**0.709**	**0.663**	**0.984**
	Adam	*I*	0.985	0.989	0.013	0.01	0.02	0.027	0.98	0.987	0.99	0.973
		II	0.975	0.954	0.068	0.024	0.009	0.026	0.991	0.932	0.976	0.974
		III	0.982	0.834	0.052	0.255	0.016	0.003	0.984	0.948	0.745	0.997
		IV	0.998	0.0	1	1.0	0.002	0.0	0.998	0	0.0	1.0
		Average	**0.985**	**0.694**	**0.283**	**0.322**	**0.012**	**0.014**	**0.988**	**0.717**	**0.678**	**0.986**

Table A.25: Comparison of the statistical information for prediction on Patient 6 based on two different datasets.

A.1.6 Patient 7

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.26: Brain sections compared with different techniques of the first architecture on patient 7.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.27 and A.28) and with the augmented dataset (Tables A.29 and A.30). Table A.31

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8956	80	1	0
Pred.	*II*	31	3130	45	0
Class	*III*	64	196	792	8
	IV	0	2	0	7

Table A.27: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8970	81	7	1
Pred.	*II*	41	3099	22	0
Class	*III*	40	220	810	5
	IV	0	5	2	9

Table A.28: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8989	99	16	1
Pred.	*II*	17	3109	57	0
Class	*III*	46	171	738	1
	IV	0	29	26	13

Table A.29: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8984	88	12	0
Pred.	*II*	30	3106	51	0
Class	*III*	37	195	759	1
	IV	0	16	19	14

Table A.30: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.987	0.99	0.01	0.009	0.019	0.022	0.981	0.99	0.991	0.978
		II	0.973	0.946	0.082	0.024	0.008	0.028	0.992	0.918	0.976	0.972
		III	0.976	0.835	0.055	0.253	0.021	0.004	0.979	0.945	0.747	0.996
		IV	0.999	0.583	0.533	0.222	0.0	0.001	1.0	0.467	0.778	0.999
		Average	**0.984**	**0.839**	**0.17**	**0.127**	**0.012**	**0.014**	**0.988**	**0.83**	**0.873**	**0.986**
	Adam	*I*	0.987	0.991	0.009	0.01	0.021	0.019	0.979	0.991	0.99	0.981
		II	0.972	0.944	0.09	0.02	0.006	0.03	0.994	0.91	0.98	0.97
		III	0.978	0.846	0.037	0.247	0.021	0.003	0.979	0.963	0.753	0.997
		IV	0.999	0.581	0.4	0.438	0.001	0.0	0.999	0.6	0.562	1.0
		Average	**0.984**	**0.841**	**0.134**	**0.179**	**0.012**	**0.013**	**0.988**	**0.866**	**0.821**	**0.987**
Data Augment.	SGD	*I*	0.987	0.99	0.007	0.013	0.027	0.015	0.973	0.993	0.987	0.985
		II	0.972	0.944	0.086	0.023	0.007	0.029	0.993	0.914	0.977	0.971
		III	0.976	0.823	0.118	0.228	0.017	0.008	0.983	0.882	0.772	0.992
		IV	0.996	0.313	0.133	0.809	0.004	0.0	0.996	0.867	0.191	1.0
		Average	**0.983**	**0.767**	**0.086**	**0.268**	**0.014**	**0.013**	**0.986**	**0.913**	**0.732**	**0.987**
	Adam	*I*	0.987	0.991	0.008	0.011	0.023	0.016	0.977	0.992	0.989	0.984
		II	0.971	0.942	0.089	0.025	0.008	0.03	0.992	0.911	0.975	0.97
		III	0.976	0.828	0.098	0.235	0.019	0.007	0.981	0.902	0.765	0.993
		IV	0.997	0.438	0.067	0.714	0.003	0.0	0.997	0.933	0.286	1.0
		Average	**0.983**	**0.8**	**0.065**	**0.246**	**0.013**	**0.013**	**0.987**	**0.935**	**0.754**	**0.987**

Table A.31: Comparison of the statistical information for prediction on Patient 7 based on two different datasets.

A.1.7   Patient 8

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.32: Brain sections compared with different techniques of the first architecture on patient 8.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.33 and A.34) and with the augmented dataset (Tables A.35 and A.36). Table A.37

displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9853	14	2	1
	II	195	2308	38	3
	III	80	75	692	3
	IV	0	1	13	34

Table A.33: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9811	10	3	0
	II	267	2298	20	1
	III	56	82	712	1
	IV	0	3	8	40

Table A.34: Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9839	17	4	0
	II	242	2324	67	2
	III	46	48	651	0
	IV	3	9	21	39

Table A.35: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9781	10	7	1
	II	281	2336	56	0
	III	64	50	676	2
	IV	2	1	6	39

Table A.36: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.978	0.985	0.027	0.002	0.005	0.08	0.995	0.973	0.998	0.92
		II	0.976	0.934	0.038	0.093	0.022	0.008	0.978	0.962	0.907	0.992
		III	0.984	0.868	0.071	0.186	0.013	0.004	0.987	0.929	0.814	0.996
		IV	0.998	0.764	0.171	0.292	0.001	0.001	0.999	0.829	0.708	0.999
		Average	**0.984**	**0.888**	**0.077**	**0.143**	**0.01**	**0.023**	**0.99**	**0.923**	**0.857**	**0.977**
	Adam	I	0.975	0.983	0.032	0.001	0.004	0.093	0.996	0.968	0.999	0.907
		II	0.971	0.923	0.04	0.111	0.026	0.009	0.974	0.96	0.889	0.991
		III	0.987	0.893	0.042	0.163	0.011	0.002	0.989	0.958	0.837	0.998
		IV	0.999	0.86	0.048	0.216	0.001	0.0	0.999	0.952	0.784	1.0
		Average	**0.983**	**0.915**	**0.041**	**0.123**	**0.01**	**0.026**	**0.99**	**0.96**	**0.877**	**0.974**
Data Augment.	SGD	I	0.977	0.984	0.029	0.002	0.007	0.084	0.993	0.971	0.998	0.916
		II	0.971	0.924	0.031	0.118	0.028	0.007	0.972	0.969	0.882	0.993
		III	0.986	0.875	0.124	0.126	0.007	0.007	0.993	0.876	0.874	0.993
		IV	0.997	0.69	0.049	0.458	0.002	0.0	0.998	0.951	0.542	1.0
		Average	**0.983**	**0.868**	**0.058**	**0.176**	**0.011**	**0.025**	**0.989**	**0.942**	**0.824**	**0.976**
	Adam	I	0.972	0.982	0.034	0.002	0.006	0.099	0.994	0.966	0.998	0.901
		II	0.97	0.921	0.025	0.126	0.031	0.006	0.969	0.975	0.874	0.994
		III	0.986	0.88	0.093	0.146	0.009	0.006	0.991	0.907	0.854	0.994
		IV	0.999	0.867	0.071	0.188	0.001	0.0	0.999	0.929	0.812	1.0
		Average	**0.982**	**0.912**	**0.056**	**0.116**	**0.012**	**0.028**	**0.988**	**0.944**	**0.885**	**0.972**

Table A.37: Comparison of the statistical information for prediction on Patient 8 based on two different datasets.

A.1.8 Patient 9

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7	Sec. 8	Sec. 9	Sec. 10	Sec. 11
Ground Truth											
Normal Dataset SGD											
Normal Dataset Adam											
Data Augmen. SGD											
Data Augmen. Adam											

Sec. 12	Sec. 13	Sec. 14	Sec. 15	Sec. 16	Sec. 17	Sec. 18	Sec. 19	Sec. 20	Sec. 21	Sec. 22

Table A.38: Brain sections compared with different techniques of the first architecture on patient 9.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.39 and A.40) and with the augmented dataset (Tables A.41 and A.42). Table A.43 displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
	I	14343	92	6	0
Pred.	II	108	6597	51	0
Class	III	92	295	944	0
	IV	0	0	0	0

Table A.39: SGD optimizer.

		Actual Class			
		I	II	III	IV
	I	14235	71	3	0
Pred.	II	259	6654	40	0
Class	III	59	246	959	0
	IV	1	1	0	0

Table A.40: Adam optimizer.

		Actual Class			
		I	II	III	IV
	I	14259	68	8	0
Pred.	II	208	6714	98	0
Class	III	74	196	892	0
	IV	3	6	2	0

Table A.41: SGD optimizer.

		Actual Class			
		I	II	III	IV
	I	14289	72	7	0
Pred.	II	196	6682	72	0
Class	III	58	223	923	0
	IV	0	5	1	0

Table A.42: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.987	0.99	0.014	0.007	0.012	0.025	0.988	0.986	0.993	0.975
		II	0.976	0.96	0.055	0.024	0.01	0.025	0.99	0.945	0.976	0.975
		III	0.98	0.81	0.057	0.291	0.018	0.003	0.982	0.943	0.709	0.997
		IV	1.0	0	1	1	0.0	0.0	1.0	0	0	1.0
	Average		**0.986**	**0.69**	**0.281**	**0.331**	**0.01**	**0.013**	**0.99**	**0.719**	**0.669**	**0.987**
	Adam	I	0.983	0.986	0.022	0.005	0.009	0.039	0.991	0.978	0.995	0.961
		II	0.973	0.956	0.046	0.043	0.019	0.02	0.981	0.954	0.957	0.98
		III	0.985	0.846	0.043	0.241	0.014	0.002	0.986	0.957	0.759	0.998
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
	Average		**0.985**	**0.697**	**0.278**	**0.322**	**0.01**	**0.015**	**0.99**	**0.722**	**0.678**	**0.985**
Data Augment.	SGD	I	0.984	0.988	0.02	0.005	0.01	0.035	0.99	0.98	0.995	0.965
		II	0.974	0.959	0.039	0.044	0.02	0.017	0.98	0.961	0.956	0.983
		III	0.983	0.825	0.108	0.232	0.013	0.005	0.987	0.892	0.768	0.995
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
	Average		**0.985**	**0.693**	**0.292**	**0.32**	**0.011**	**0.014**	**0.989**	**0.708**	**0.68**	**0.986**
	Adam	I	0.985	0.988	0.018	0.005	0.01	0.031	0.99	0.982	0.995	0.969
		II	0.975	0.959	0.043	0.039	0.017	0.019	0.983	0.957	0.961	0.981
		III	0.984	0.836	0.08	0.233	0.013	0.004	0.987	0.92	0.767	0.996
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
	Average		**0.986**	**0.696**	**0.285**	**0.319**	**0.01**	**0.014**	**0.99**	**0.715**	**0.681**	**0.986**

Table A.43: Comparison of the statistical information for prediction on Patient 9 based on two different datasets.

A.1.9 Patient 10

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14

Table A.44: Brain sections compared with different techniques of the first architecture on patient 10.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.45 and A.46) and with the augmented dataset (Tables A.47 and A.48). Table A.49

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	10339	15	5	1
Pred.	*II*	64	2966	45	4
Class	*III*	61	151	615	23
	IV	0	4	0	43

Table A.45: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	10345	18	3	4
Pred.	*II*	64	2903	23	2
Class	*III*	55	210	642	39
	IV	0	2	0	26

Table A.46: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	10346	24	10	1
Pred.	*II*	79	2936	58	1
Class	*III*	34	168	588	13
	IV	1	12	9	56

Table A.47: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	10321	35	6	2
Pred.	*II*	87	2905	44	0
Class	*III*	48	176	582	6
	IV	2	21	38	63

Table A.48: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.99	0.993	0.012	0.002	0.005	0.031	0.995	0.988	0.998	0.969
		II	0.98	0.954	0.054	0.037	0.01	0.015	0.99	0.946	0.963	0.985
		III	0.98	0.812	0.075	0.276	0.017	0.004	0.983	0.925	0.724	0.996
		IV	0.998	0.729	0.394	0.085	0.0	0.002	1.0	0.606	0.915	0.998
		Average	**0.987**	**0.872**	**0.134**	**0.1**	**0.008**	**0.013**	**0.992**	**0.866**	**0.9**	**0.987**
	Adam	*I*	0.99	0.993	0.011	0.002	0.006	0.03	0.994	0.989	0.998	0.97
		II	0.978	0.948	0.073	0.03	0.008	0.02	0.992	0.927	0.97	0.98
		III	0.977	0.796	0.039	0.321	0.022	0.002	0.978	0.961	0.679	0.998
		IV	0.997	0.525	0.634	0.071	0.0	0.003	1.0	0.366	0.929	0.997
		Average	**0.985**	**0.816**	**0.189**	**0.106**	**0.009**	**0.014**	**0.991**	**0.811**	**0.894**	**0.986**
Data Augment.	SGD	*I*	0.99	0.993	0.011	0.003	0.009	0.029	0.991	0.989	0.997	0.971
		II	0.976	0.945	0.065	0.045	0.012	0.018	0.988	0.935	0.955	0.982
		III	0.98	0.801	0.116	0.268	0.016	0.006	0.984	0.884	0.732	0.994
		IV	0.997	0.752	0.211	0.282	0.002	0.001	0.998	0.789	0.718	0.999
		Average	**0.986**	**0.873**	**0.101**	**0.149**	**0.01**	**0.013**	**0.99**	**0.899**	**0.851**	**0.987**
	Adam	*I*	0.987	0.991	0.013	0.004	0.011	0.035	0.989	0.987	0.996	0.965
		II	0.975	0.941	0.074	0.043	0.012	0.021	0.988	0.926	0.957	0.979
		III	0.978	0.785	0.131	0.283	0.017	0.007	0.983	0.869	0.717	0.993
		IV	0.995	0.646	0.113	0.492	0.004	0.001	0.996	0.887	0.508	0.999
		Average	**0.984**	**0.841**	**0.083**	**0.205**	**0.011**	**0.016**	**0.989**	**0.917**	**0.794**	**0.984**

Table A.49: Comparison of the statistical information for prediction on Patient 10 based on two different datasets.

A.1.10 Patient 11

Visualization Results

Table A.50: Brain sections compared with different techniques of the first architecture on patient 11.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.51 and A.52) and with the augmented dataset (Tables A.53 and A.54). Table A.55

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	9952	67	15	0
	II	487	2487	487	0
	III	70	225	519	0
	IV	1	19	7	0

Table A.51: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	10024	85	29	0
	II	414	2348	425	0
	III	71	339	571	0
	IV	1	22	7	0

Table A.52: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	9994	89	28	0
	II	460	2454	452	0
	III	37	214	531	0
	IV	18	42	17	0

Table A.53: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	9945	62	17	0
	II	509	2400	392	0
	III	52	320	611	0
	IV	3	17	8	0

Table A.54: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.955	0.969	0.053	0.008	0.021	0.13	0.979	0.947	0.992	0.87
		II	0.91	0.795	0.111	0.281	0.084	0.029	0.916	0.889	0.719	0.971
		III	0.944	0.564	0.495	0.362	0.022	0.038	0.978	0.505	0.638	0.962
		IV	0.998	0.0	1	1.0	0.002	0.0	0.998	0	0.0	1.0
		Average	**0.952**	**0.582**	**0.415**	**0.413**	**0.032**	**0.049**	**0.968**	**0.585**	**0.587**	**0.951**
	Adam	*I*	0.958	0.971	0.046	0.011	0.03	0.116	0.97	0.954	0.989	0.884
		II	0.91	0.785	0.16	0.263	0.073	0.04	0.927	0.84	0.737	0.96
		III	0.939	0.567	0.447	0.418	0.031	0.035	0.969	0.553	0.582	0.965
		IV	0.998	0.0	1	1.0	0.002	0.0	0.998	0	0.0	1.0
		Average	**0.951**	**0.581**	**0.413**	**0.423**	**0.034**	**0.048**	**0.966**	**0.587**	**0.577**	**0.952**
Data Augment.	SGD	*I*	0.956	0.969	0.049	0.012	0.031	0.122	0.969	0.951	0.988	0.878
		II	0.912	0.796	0.123	0.271	0.079	0.031	0.921	0.877	0.729	0.969
		III	0.948	0.587	0.483	0.321	0.019	0.037	0.981	0.517	0.679	0.963
		IV	0.995	0.0	1	1.0	0.005	0.0	0.995	0	0.0	1.0
		Average	**0.953**	**0.588**	**0.414**	**0.401**	**0.034**	**0.048**	**0.967**	**0.586**	**0.599**	**0.953**
	Adam	*I*	0.955	0.969	0.054	0.008	0.021	0.131	0.979	0.946	0.992	0.869
		II	0.909	0.787	0.143	0.273	0.078	0.036	0.922	0.857	0.727	0.964
		III	0.945	0.608	0.406	0.378	0.028	0.031	0.972	0.594	0.622	0.969
		IV	0.998	0.0	1	1.0	0.002	0.0	0.998	0	0.0	1.0
		Average	**0.952**	**0.591**	**0.401**	**0.415**	**0.032**	**0.05**	**0.968**	**0.599**	**0.585**	**0.951**

Table A.55: Comparison of the statistical information for prediction on Patient 11 based on two different datasets.

A.2 Results for architecture 2

A.2.1 Patient 2

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.56: Brain sections compared with different techniques of the second architecture on patient 2.

A.2.2   Patient 3

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.57: Brain sections compared with different techniques of the second architecture on patient 3.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.58 and A.59) and with the augmented dataset (Tables A.60 and A.61). Table A.62

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8827	183	35	11
	II	206	3361	58	3
	III	53	204	324	11
	IV	1	2	2	31

Table A.58: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8680	61	6	5
	II	332	3477	46	2
	III	70	212	365	16
	IV	3	2	3	32

Table A.59: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8642	81	13	4
	II	365	3515	91	1
	III	69	131	302	4
	IV	11	22	14	47

Table A.60: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8668	84	16	3
	II	352	3493	68	1
	III	47	145	319	0
	IV	18	30	16	52

Table A.61: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.963	0.973	0.029	0.025	0.054	0.061	0.946	0.971	0.975	0.939
		II	0.951	0.911	0.104	0.074	0.028	0.04	0.972	0.896	0.926	0.96
		III	0.973	0.641	0.227	0.453	0.021	0.007	0.979	0.773	0.547	0.993
		IV	0.998	0.674	0.446	0.139	0.0	0.002	1.0	0.554	0.861	0.998
		Average	**0.971**	**0.8**	**0.202**	**0.173**	**0.026**	**0.028**	**0.974**	**0.798**	**0.827**	**0.972**
	Adam	*I*	0.964	0.973	0.045	0.008	0.017	0.089	0.983	0.955	0.992	0.911
		II	0.951	0.914	0.073	0.099	0.04	0.029	0.96	0.927	0.901	0.971
		III	0.973	0.674	0.131	0.449	0.023	0.004	0.977	0.869	0.551	0.996
		IV	0.998	0.674	0.418	0.2	0.001	0.002	0.999	0.582	0.8	0.998
		Average	**0.972**	**0.809**	**0.167**	**0.189**	**0.02**	**0.031**	**0.98**	**0.833**	**0.811**	**0.969**
Data Augment.	SGD	*I*	0.959	0.97	0.049	0.011	0.023	0.097	0.977	0.951	0.989	0.903
		II	0.948	0.91	0.063	0.115	0.048	0.025	0.952	0.937	0.885	0.975
		III	0.976	0.653	0.279	0.403	0.016	0.009	0.984	0.721	0.597	0.991
		IV	0.996	0.627	0.161	0.5	0.004	0.001	0.996	0.839	0.5	0.999
		Average	**0.97**	**0.79**	**0.138**	**0.257**	**0.023**	**0.033**	**0.977**	**0.862**	**0.743**	**0.967**
	Adam	*I*	0.961	0.971	0.046	0.012	0.024	0.092	0.976	0.954	0.988	0.908
		II	0.949	0.912	0.069	0.108	0.044	0.027	0.956	0.931	0.892	0.973
		III	0.978	0.686	0.239	0.376	0.015	0.008	0.985	0.761	0.624	0.992
		IV	0.995	0.605	0.071	0.552	0.005	0.0	0.995	0.929	0.448	1.0
		Average	**0.971**	**0.793**	**0.106**	**0.262**	**0.022**	**0.032**	**0.978**	**0.894**	**0.738**	**0.968**

Table A.62: Comparison of the statistical information for prediction on Patient 3 based on two different datasets.

A.2.3 Patient 4

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7

Table A.63: Brain sections compared with different techniques of the second architecture on patient 4.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.64 and A.65) and with the augmented dataset (Tables A.66 and A.67). Table A.68

displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8101	33	5	7
	II	63	3726	99	6
	III	63	117	862	87
	IV	0	4	13	126

Table A.64: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8102	34	9	5
	II	64	3734	71	2
	III	66	93	863	29
	IV	1	6	43	190

Table A.65: Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8105	77	51	3
	II	58	3703	99	3
	III	60	66	728	9
	IV	1	30	108	211

Table A.66: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	8077	57	29	3
	II	90	3723	94	1
	III	59	90	788	9
	IV	1	10	68	213

Table A.67: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.987	0.99	0.015	0.006	0.009	0.024	0.991	0.985	0.994	0.976
		II	0.976	0.959	0.04	0.043	0.018	0.016	0.982	0.96	0.957	0.984
		III	0.971	0.818	0.12	0.236	0.022	0.01	0.978	0.88	0.764	0.99
		IV	0.991	0.683	0.442	0.119	0.001	0.008	0.999	0.558	0.881	0.992
	Average		**0.981**	**0.863**	**0.154**	**0.101**	**0.012**	**0.015**	**0.987**	**0.846**	**0.899**	**0.986**
	Adam	I	0.987	0.989	0.016	0.006	0.009	0.025	0.991	0.984	0.994	0.975
		II	0.98	0.965	0.034	0.035	0.015	0.014	0.985	0.966	0.965	0.986
		III	0.977	0.847	0.125	0.179	0.015	0.01	0.985	0.875	0.821	0.99
		IV	0.994	0.815	0.159	0.208	0.004	0.003	0.996	0.841	0.792	0.997
	Average		**0.984**	**0.904**	**0.083**	**0.107**	**0.011**	**0.013**	**0.989**	**0.917**	**0.893**	**0.987**
Data Augment.	SGD	I	0.981	0.985	0.014	0.016	0.026	0.023	0.974	0.986	0.984	0.977
		II	0.975	0.957	0.045	0.041	0.017	0.018	0.983	0.955	0.959	0.982
		III	0.97	0.787	0.262	0.156	0.011	0.021	0.989	0.738	0.844	0.979
		IV	0.988	0.733	0.066	0.397	0.011	0.001	0.989	0.934	0.603	0.999
	Average		**0.979**	**0.866**	**0.097**	**0.152**	**0.016**	**0.016**	**0.984**	**0.903**	**0.847**	**0.984**
	Adam	I	0.982	0.985	0.018	0.011	0.018	0.029	0.982	0.982	0.989	0.971
		II	0.974	0.956	0.04	0.047	0.02	0.017	0.98	0.96	0.953	0.983
		III	0.974	0.819	0.195	0.167	0.013	0.015	0.987	0.805	0.833	0.985
		IV	0.993	0.822	0.058	0.271	0.006	0.001	0.994	0.942	0.729	0.999
	Average		**0.981**	**0.895**	**0.078**	**0.124**	**0.014**	**0.015**	**0.986**	**0.922**	**0.876**	**0.985**

Table A.68: Comparison of the statistical information for prediction on Patient 4 based on two different datasets.

A.2.4 Patient 5

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.69: Brain sections compared with different techniques of the second architecture on patient 5.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.70 and A.71) and with the augmented dataset (Tables A.72 and A.73). Table A.74

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8069	33	14	8
Pred.	*II*	670	3189	150	20
Class	*III*	54	111	863	92
	IV	0	1	0	38

Table A.70: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8079	21	21	5
Pred.	*II*	664	3229	116	5
Class	*III*	51	76	869	10
	IV	1	4	23	138

Table A.71: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8095	70	23	7
Pred.	*II*	658	3109	125	3
Class	*III*	42	136	849	11
	IV	0	17	30	137

Table A.72: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8131	24	13	3
Pred.	*II*	622	3192	110	3
Class	*III*	40	111	882	14
	IV	0	7	22	138

Table A.73: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.941	0.954	0.082	0.007	0.012	0.14	0.988	0.918	0.993	0.86
		II	0.926	0.866	0.043	0.208	0.084	0.016	0.916	0.957	0.792	0.984
		III	0.968	0.804	0.16	0.229	0.021	0.013	0.979	0.84	0.771	0.987
		IV	0.991	0.386	0.759	0.026	0.0	0.009	1.0	0.241	0.974	0.991
		Average	**0.957**	**0.752**	**0.261**	**0.118**	**0.029**	**0.045**	**0.971**	**0.739**	**0.883**	**0.956**
	Adam	*I*	0.943	0.955	0.081	0.006	0.01	0.138	0.99	0.919	0.994	0.862
		II	0.933	0.879	0.03	0.196	0.079	0.011	0.921	0.97	0.804	0.989
		III	0.978	0.854	0.155	0.136	0.011	0.013	0.989	0.845	0.864	0.987
		IV	0.996	0.852	0.127	0.169	0.002	0.002	0.998	0.873	0.831	0.998
		Average	**0.963**	**0.885**	**0.098**	**0.127**	**0.025**	**0.041**	**0.974**	**0.902**	**0.873**	**0.959**
Data Augment.	SGD	*I*	0.94	0.953	0.08	0.012	0.022	0.137	0.978	0.92	0.988	0.863
		II	0.924	0.86	0.067	0.202	0.079	0.024	0.921	0.933	0.798	0.976
		III	0.972	0.822	0.173	0.182	0.015	0.015	0.985	0.827	0.818	0.985
		IV	0.995	0.801	0.133	0.255	0.004	0.002	0.996	0.867	0.745	0.998
		Average	**0.958**	**0.859**	**0.113**	**0.163**	**0.03**	**0.044**	**0.97**	**0.887**	**0.837**	**0.956**
	Adam	*I*	0.947	0.959	0.075	0.005	0.009	0.129	0.991	0.925	0.995	0.871
		II	0.934	0.879	0.043	0.187	0.074	0.015	0.926	0.957	0.813	0.985
		III	0.977	0.851	0.141	0.158	0.013	0.012	0.987	0.859	0.842	0.988
		IV	0.996	0.849	0.127	0.174	0.002	0.002	0.998	0.873	0.826	0.998
		Average	**0.964**	**0.885**	**0.097**	**0.131**	**0.024**	**0.04**	**0.976**	**0.903**	**0.869**	**0.96**

Table A.74: Comparison of the statistical information for prediction on Patient 5 based on two different datasets.

A.2.5 Patient 6

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.75: Brain sections compared with different techniques of the second architecture on patient 6.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.76 and A.77) and with the augmented dataset (Tables A.78 and A.79). Table A.80

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8782	38	4	0
	II	113	3530	56	0
	III	59	162	568	0
	IV	0	0	0	0

Table A.76: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8781	61	4	0
	II	8781	61	4	0
	III	61	194	598	0
	IV	0	0	0	0

Table A.77: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8827	63	13	0
	II	98	3453	102	0
	III	27	82	499	0
	IV	2	131	15	0

Table A.78: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	8769	32	5	0
	II	146	3594	84	0
	III	38	101	539	0
	IV	1	3	0	0

Table A.79: Adam optimizer.

			Metrics									
DS	**Opt.**	**Class**	Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.984	0.988	0.019	0.005	0.01	0.038	0.99	0.981	0.995	0.962
		II	0.972	0.95	0.054	0.046	0.018	0.021	0.982	0.946	0.954	0.979
		III	0.979	0.802	0.096	0.28	0.017	0.005	0.983	0.904	0.72	0.995
		IV	1.0	0	1	1	0.0	0.0	1.0	0	0	1.0
		Average	**0.984**	**0.685**	**0.292**	**0.333**	**0.011**	**0.016**	**0.989**	**0.708**	**0.667**	**0.984**
	Adam	*I*	0.982	0.987	0.02	0.007	0.015	0.039	0.985	0.98	0.993	0.961
		II	0.97	0.946	0.068	0.039	0.015	0.026	0.985	0.932	0.961	0.974
		III	0.979	0.807	0.049	0.299	0.02	0.002	0.98	0.951	0.701	0.998
		IV	1.0	0	1	1	0.0	0.0	1.0	0	0	1.0
		Average	**0.983**	**0.685**	**0.284**	**0.336**	**0.013**	**0.017**	**0.988**	**0.716**	**0.664**	**0.983**
Data Augment.	SGD	*I*	0.985	0.989	0.014	0.009	0.017	0.029	0.983	0.986	0.991	0.971
		II	0.964	0.936	0.074	0.055	0.021	0.029	0.979	0.926	0.945	0.971
		III	0.982	0.807	0.207	0.179	0.009	0.01	0.991	0.793	0.821	0.99
		IV	0.989	0.0	1	1.0	0.011	0.0	0.989	0	0.0	1.0
		Average	**0.98**	**0.683**	**0.324**	**0.311**	**0.015**	**0.017**	**0.985**	**0.676**	**0.689**	**0.983**
	Adam	*I*	0.983	0.988	0.021	0.004	0.008	0.041	0.992	0.979	0.996	0.959
		II	0.973	0.952	0.036	0.06	0.024	0.014	0.976	0.964	0.94	0.986
		III	0.983	0.825	0.142	0.205	0.011	0.007	0.989	0.858	0.795	0.993
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.985**	**0.691**	**0.3**	**0.317**	**0.011**	**0.015**	**0.989**	**0.7**	**0.683**	**0.984**

Table A.80: Comparison of the statistical information for prediction on Patient 6 based on two different datasets.

A.2.6   Patient 7

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.81: Brain sections compared with different techniques of the second architecture on patient 7.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.82 and A.83) and with the augmented dataset (Tables A.84 and A.85). Table A.86

displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8974	96	9	0
Class	*II*	37	3156	64	0
	III	41	154	763	12
	IV	0	2	1	3

Table A.82: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8980	91	9	0
Class	*II*	28	3139	60	0
	III	46	166	767	8
	IV	0	6	5	7

Table A.83: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8979	296	99	1
Class	*II*	22	2929	109	0
	III	50	160	622	6
	IV	0	20	11	8

Table A.84: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8975	113	26	1
Class	*II*	32	3152	81	0
	III	45	127	714	2
	IV	0	16	16	12

Table A.85: Adam optimizer.

			Metrics									
DS	**Opt.**	**Class**	Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.986	0.99	0.009	0.012	0.025	0.018	0.975	0.991	0.988	0.982
		II	0.973	0.947	0.074	0.031	0.01	0.025	0.99	0.926	0.969	0.975
		III	0.979	0.844	0.088	0.213	0.017	0.006	0.983	0.912	0.787	0.994
		IV	0.999	0.286	0.8	0.5	0.0	0.001	1.0	0.2	0.5	0.999
		Average	**0.984**	**0.767**	**0.243**	**0.189**	**0.013**	**0.012**	**0.987**	**0.757**	**0.811**	**0.987**
	Adam	*I*	0.987	0.99	0.008	0.011	0.023	0.017	0.977	0.992	0.989	0.983
		II	0.974	0.947	0.077	0.027	0.009	0.026	0.991	0.923	0.973	0.974
		III	0.978	0.839	0.088	0.223	0.018	0.006	0.982	0.912	0.777	0.994
		IV	0.999	0.424	0.533	0.611	0.001	0.001	0.999	0.467	0.389	0.999
		Average	**0.985**	**0.8**	**0.176**	**0.218**	**0.013**	**0.012**	**0.987**	**0.824**	**0.782**	**0.987**
Data Augment.	SGD	*I*	0.965	0.975	0.008	0.042	0.093	0.018	0.907	0.992	0.958	0.982
		II	0.954	0.906	0.14	0.043	0.013	0.046	0.987	0.86	0.957	0.954
		III	0.967	0.741	0.26	0.258	0.017	0.018	0.983	0.74	0.742	0.982
		IV	0.997	0.296	0.467	0.795	0.002	0.001	0.998	0.533	0.205	0.999
		Average	**0.971**	**0.729**	**0.219**	**0.284**	**0.031**	**0.021**	**0.969**	**0.781**	**0.716**	**0.979**
	Adam	*I*	0.984	0.988	0.009	0.015	0.033	0.018	0.967	0.991	0.985	0.982
		II	0.972	0.945	0.075	0.035	0.011	0.025	0.989	0.925	0.965	0.975
		III	0.978	0.828	0.147	0.196	0.014	0.01	0.986	0.853	0.804	0.99
		IV	0.997	0.407	0.2	0.727	0.002	0.0	0.998	0.8	0.273	1.0
		Average	**0.983**	**0.792**	**0.108**	**0.243**	**0.015**	**0.013**	**0.985**	**0.892**	**0.757**	**0.987**

Table A.86: Comparison of the statistical information for prediction on Patient 7 based on two different datasets.

A.2.7   Patient 8

Visualization Results

Table A.87: Brain sections compared with different techniques of the second architecture on patient 8.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.88 and A.89) and with the augmented dataset (Tables A.90 and A.91). Table A.92

displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9787	12	0	4
	II	280	2283	39	1
	III	63	103	700	14
	IV	0	0	3	23

Table A.88: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9810	15	2	1
	II	238	2283	36	1
	III	84	94	704	13
	IV	0	1	3	27

Table A.89: Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9845	36	6	3
	II	238	2297	62	1
	III	44	50	659	1
	IV	1	14	18	37

Table A.90: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9856	21	4	1
	II	224	2316	56	1
	III	48	58	666	0
	IV	2	3	16	40

Table A.91: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.973	0.982	0.034	0.002	0.005	0.098	0.995	0.966	0.998	0.902
		II	0.967	0.913	0.048	0.123	0.029	0.011	0.971	0.952	0.877	0.989
		III	0.983	0.863	0.057	0.205	0.014	0.003	0.986	0.943	0.795	0.997
		IV	0.998	0.676	0.452	0.115	0.0	0.001	1.0	0.548	0.885	0.999
	Average		**0.98**	**0.859**	**0.148**	**0.111**	**0.012**	**0.028**	**0.988**	**0.852**	**0.889**	**0.972**
	Adam	I	0.974	0.983	0.032	0.002	0.006	0.092	0.994	0.968	0.998	0.908
		II	0.971	0.922	0.046	0.108	0.025	0.01	0.975	0.954	0.892	0.99
		III	0.983	0.859	0.055	0.213	0.015	0.003	0.985	0.945	0.787	0.997
		IV	0.999	0.74	0.357	0.129	0.0	0.001	1.0	0.643	0.871	0.999
	Average		**0.982**	**0.876**	**0.122**	**0.113**	**0.011**	**0.026**	**0.988**	**0.877**	**0.887**	**0.974**
Data Augment.	SGD	I	0.975	0.983	0.029	0.005	0.014	0.084	0.986	0.971	0.995	0.916
		II	0.97	0.92	0.042	0.116	0.028	0.009	0.972	0.958	0.884	0.991
		III	0.986	0.879	0.115	0.126	0.008	0.007	0.992	0.885	0.874	0.993
		IV	0.997	0.661	0.119	0.471	0.002	0.0	0.998	0.881	0.529	1.0
	Average		**0.982**	**0.861**	**0.076**	**0.179**	**0.013**	**0.025**	**0.987**	**0.924**	**0.821**	**0.975**
	Adam	I	0.977	0.985	0.027	0.003	0.008	0.08	0.992	0.973	0.997	0.92
		II	0.973	0.927	0.034	0.108	0.026	0.008	0.974	0.966	0.892	0.992
		III	0.986	0.879	0.104	0.137	0.008	0.006	0.992	0.896	0.863	0.994
		IV	0.998	0.777	0.048	0.344	0.002	0.0	0.998	0.952	0.656	1.0
	Average		**0.984**	**0.892**	**0.053**	**0.148**	**0.011**	**0.024**	**0.989**	**0.947**	**0.852**	**0.976**

Table A.92: Comparison of the statistical information for prediction on Patient 8 based on two different datasets.

A.2.8   Patient 9

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7	Sec. 8	Sec. 9	Sec. 10	Sec. 11
Ground Truth											
Normal Dataset SGD											
Normal Dataset Adam											
Data Augmen. SGD											
Data Augmen. Adam											

Sec. 12	Sec. 13	Sec. 14	Sec. 15	Sec. 16	Sec. 17	Sec. 18	Sec. 19	Sec. 20	Sec. 21	Sec. 22

Table A.93: Brain sections compared with different techniques of the second architecture on patient 9.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.94 and A.95) and with the augmented dataset (Tables A.96 and A.97). Table A.98 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	14262	67	7	0
Pred.	*II*	198	6769	125	0
Class	*III*	84	147	868	0
	IV	0	1	0	0

Table A.94: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	14309	93	7	0
Pred.	*II*	170	6683	75	0
Class	*III*	74	196	921	0
	IV	0	0	0	0

Table A.95: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	14316	248	21	0
Pred.	*II*	136	6509	110	0
Class	*III*	88	214	871	0
	IV	3	11	1	0

Table A.96: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	14253	63	5	0
Pred.	*II*	188	6763	101	0
Class	*III*	102	158	894	0
	IV	1	0	0	0

Table A.97: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.984	0.988	0.019	0.005	0.009	0.034	0.991	0.981	0.995	0.966
		II	0.976	0.962	0.031	0.046	0.021	0.014	0.979	0.969	0.954	0.986
		III	0.984	0.827	0.132	0.21	0.011	0.006	0.989	0.868	0.79	0.994
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
	Average		**0.986**	**0.694**	**0.295**	**0.315**	**0.01**	**0.013**	**0.99**	**0.705**	**0.685**	**0.986**
	Adam	*I*	0.985	0.988	0.017	0.007	0.013	0.03	0.987	0.983	0.993	0.97
		II	0.976	0.962	0.041	0.035	0.016	0.019	0.984	0.959	0.965	0.981
		III	0.984	0.84	0.082	0.227	0.013	0.004	0.987	0.918	0.773	0.996
		IV	1.0	0	1	1	0.0	0.0	1.0	0	0	1.0
	Average		**0.986**	**0.698**	**0.285**	**0.317**	**0.01**	**0.013**	**0.99**	**0.715**	**0.683**	**0.987**
Data Augment.	SGD	*I*	0.978	0.983	0.016	0.018	0.034	0.029	0.966	0.984	0.982	0.971
		II	0.968	0.948	0.068	0.036	0.016	0.03	0.984	0.932	0.964	0.97
		III	0.981	0.801	0.132	0.257	0.014	0.006	0.986	0.868	0.743	0.994
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
	Average		**0.982**	**0.683**	**0.304**	**0.328**	**0.016**	**0.016**	**0.984**	**0.696**	**0.672**	**0.984**
	Adam	*I*	0.984	0.988	0.02	0.005	0.009	0.035	0.991	0.98	0.995	0.965
		II	0.977	0.964	0.031	0.041	0.019	0.014	0.981	0.969	0.959	0.986
		III	0.984	0.83	0.106	0.225	0.012	0.005	0.988	0.894	0.775	0.995
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
	Average		**0.986**	**0.696**	**0.289**	**0.318**	**0.01**	**0.013**	**0.99**	**0.711**	**0.682**	**0.987**

Table A.98: Comparison of the statistical information for prediction on Patient 9 based on two different datasets.

A.2.9   Patient 10

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14

Table A.99: Brain sections compared with different techniques of the second architecture on patient 10.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.100 and A.101) and with the augmented dataset (Tables A.102 and A.103). Table

A.104 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10325	13	6	2
Class	*II*	71	3016	72	4
	III	64	105	585	15
	IV	0	6	2	50

Table A.100: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10342	23	11	3
Class	*II*	55	2937	42	3
	III	65	171	615	23
	IV	0	2	2	42

Table A.101: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10326	49	22	1
Class	*II*	73	2890	64	4
	III	58	169	530	9
	IV	1	29	54	57

Table A.102: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10359	51	18	2
Class	*II*	52	2969	75	2
	III	48	112	555	9
	IV	1	8	17	58

Table A.103: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.989	0.993	0.013	0.002	0.005	0.034	0.995	0.987	0.998	0.966
		II	0.981	0.957	0.039	0.046	0.013	0.011	0.987	0.961	0.954	0.989
		III	0.982	0.816	0.12	0.239	0.013	0.006	0.987	0.88	0.761	0.994
		IV	0.998	0.775	0.296	0.138	0.001	0.001	0.999	0.704	0.862	0.999
	Average		**0.988**	**0.885**	**0.117**	**0.106**	**0.008**	**0.013**	**0.992**	**0.883**	**0.894**	**0.987**
	Adam	*I*	0.989	0.992	0.011	0.004	0.01	0.03	0.99	0.989	0.996	0.97
		II	0.979	0.952	0.063	0.033	0.009	0.017	0.991	0.937	0.967	0.983
		III	0.978	0.797	0.082	0.296	0.019	0.004	0.981	0.918	0.704	0.996
		IV	0.998	0.718	0.408	0.087	0.0	0.002	1.0	0.592	0.913	0.998
	Average		**0.986**	**0.865**	**0.141**	**0.105**	**0.009**	**0.013**	**0.99**	**0.859**	**0.895**	**0.987**
Data Augment.	SGD	*I*	0.985	0.99	0.013	0.007	0.019	0.035	0.981	0.987	0.993	0.965
		II	0.973	0.937	0.079	0.047	0.013	0.022	0.987	0.921	0.953	0.978
		III	0.974	0.738	0.209	0.308	0.017	0.01	0.983	0.791	0.692	0.99
		IV	0.993	0.538	0.197	0.596	0.006	0.001	0.994	0.803	0.404	0.999
	Average		**0.981**	**0.801**	**0.124**	**0.239**	**0.014**	**0.017**	**0.986**	**0.875**	**0.76**	**0.983**
	Adam	*I*	0.988	0.992	0.01	0.007	0.018	0.026	0.982	0.99	0.993	0.974
		II	0.979	0.952	0.054	0.042	0.012	0.015	0.988	0.946	0.958	0.985
		III	0.981	0.799	0.165	0.233	0.012	0.008	0.988	0.835	0.767	0.992
		IV	0.997	0.748	0.183	0.31	0.002	0.001	0.998	0.817	0.69	0.999
	Average		**0.986**	**0.873**	**0.103**	**0.148**	**0.011**	**0.012**	**0.989**	**0.897**	**0.852**	**0.988**

Table A.104: Comparison of the statistical information for prediction on Patient 10 based on two different datasets.

A.2.10   Patient 11

Visualization Results

Table A.105: Brain sections compared with different techniques of the second architecture on patient 11.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.106 and A.107) and with the augmented dataset (Tables A.108 and A.109). Table

A.110 displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
Pred. Class	I	10022	85	24	0
	II	412	2409	425	0
	III	74	294	577	0
	IV	1	11	2	0

Table A.106: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	10011	105	28	0
	II	413	2208	413	0
	III	86	471	587	0
	IV	0	10	4	0

Table A.107: Adam optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	10005	80	34	0
	II	444	2470	557	0
	III	45	206	427	0
	IV	15	39	14	0

Table A.108: SGD optimizer.

		Actual Class			
		I	II	III	IV
Pred. Class	I	9992	108	38	0
	II	431	2362	418	0
	III	78	292	563	0
	IV	8	37	9	0

Table A.109: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.958	0.971	0.046	0.011	0.028	0.116	0.972	0.954	0.989	0.884
		II	0.914	0.797	0.139	0.258	0.073	0.035	0.927	0.861	0.742	0.965
		III	0.943	0.585	0.439	0.389	0.028	0.034	0.972	0.561	0.611	0.966
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
		Average	**0.954**	**0.588**	**0.406**	**0.414**	**0.033**	**0.046**	**0.968**	**0.594**	**0.585**	**0.954**
	Adam	I	0.956	0.969	0.047	0.013	0.035	0.119	0.965	0.953	0.987	0.881
		II	0.902	0.758	0.21	0.272	0.072	0.052	0.928	0.79	0.728	0.948
		III	0.93	0.54	0.431	0.487	0.042	0.034	0.958	0.569	0.513	0.966
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
		Average	**0.947**	**0.567**	**0.422**	**0.443**	**0.037**	**0.051**	**0.963**	**0.578**	**0.557**	**0.949**
Data Augment.	SGD	I	0.957	0.97	0.048	0.011	0.03	0.12	0.97	0.952	0.989	0.88
		II	0.908	0.788	0.116	0.288	0.087	0.03	0.913	0.884	0.712	0.97
		III	0.94	0.499	0.586	0.37	0.019	0.044	0.981	0.414	0.63	0.956
		IV	0.995	0.0	1	1.0	0.005	0.0	0.995	0	0.0	1.0
		Average	**0.95**	**0.564**	**0.438**	**0.417**	**0.035**	**0.049**	**0.965**	**0.562**	**0.583**	**0.952**
	Adam	I	0.954	0.968	0.049	0.014	0.038	0.123	0.962	0.951	0.986	0.877
		II	0.91	0.786	0.156	0.264	0.074	0.039	0.926	0.844	0.736	0.961
		III	0.942	0.574	0.452	0.397	0.028	0.035	0.972	0.548	0.603	0.965
		IV	0.996	0.0	1	1.0	0.004	0.0	0.996	0	0.0	1.0
		Average	**0.951**	**0.582**	**0.414**	**0.419**	**0.036**	**0.049**	**0.964**	**0.586**	**0.581**	**0.951**

Table A.110: Comparison of the statistical information for prediction on Patient 11 based on two different datasets.

A.3   Results for architecture 3

A.3.1   Patient 2

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.111: Brain sections compared with different techniques of the third architecture on patient 2.

A.3.2 Patient 3

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.112: Brain sections compared with different techniques of the third architecture on patient 3.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.113 and A.114) and with the augmented dataset (Tables A.115 and A.116). Table

A.117 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8609	62	11	7
Class	*II*	395	3534	112	12
	III	78	148	297	18
	IV	3	6	2	18

Table A.113: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8597	53	15	7
Class	*II*	391	3514	103	12
	III	99	177	300	22
	IV	0	4	3	15

Table A.114: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8592	87	16	4
Class	*II*	413	3500	107	10
	III	72	152	280	8
	IV	8	13	17	33

Table A.115: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8604	71	17	5
Class	*II*	395	3515	107	5
	III	71	124	275	4
	IV	17	39	21	42

Table A.116: Adam optimizer.

			Metrics									
DS	**Opt.**	**Class**	Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.958	0.969	0.052	0.009	0.019	0.103	0.981	0.948	0.991	0.897
		II	0.945	0.906	0.058	0.128	0.054	0.023	0.946	0.942	0.872	0.977
		III	0.972	0.617	0.296	0.451	0.019	0.01	0.981	0.704	0.549	0.99
		IV	0.996	0.429	0.673	0.379	0.001	0.003	0.999	0.327	0.621	0.997
		Average	**0.968**	**0.73**	**0.27**	**0.242**	**0.023**	**0.035**	**0.977**	**0.73**	**0.758**	**0.965**
	Adam	*I*	0.958	0.968	0.054	0.009	0.018	0.106	0.982	0.946	0.991	0.894
		II	0.944	0.905	0.062	0.126	0.053	0.025	0.947	0.938	0.874	0.975
		III	0.969	0.589	0.287	0.498	0.023	0.01	0.977	0.713	0.502	0.99
		IV	0.996	0.385	0.732	0.318	0.001	0.003	0.999	0.268	0.682	0.997
		Average	**0.967**	**0.712**	**0.284**	**0.238**	**0.024**	**0.036**	**0.976**	**0.716**	**0.762**	**0.964**
Data Augment.	SGD	*I*	0.955	0.966	0.054	0.012	0.025	0.107	0.975	0.946	0.988	0.893
		II	0.941	0.9	0.067	0.132	0.055	0.027	0.945	0.933	0.868	0.973
		III	0.972	0.601	0.333	0.453	0.018	0.011	0.982	0.667	0.547	0.989
		IV	0.995	0.524	0.4	0.535	0.003	0.002	0.997	0.6	0.465	0.998
		Average	**0.966**	**0.748**	**0.214**	**0.283**	**0.025**	**0.037**	**0.975**	**0.787**	**0.717**	**0.963**
	Adam	*I*	0.957	0.968	0.053	0.011	0.022	0.104	0.978	0.947	0.989	0.896
		II	0.944	0.905	0.062	0.126	0.053	0.025	0.947	0.938	0.874	0.975
		III	0.974	0.614	0.348	0.42	0.015	0.011	0.985	0.652	0.58	0.989
		IV	0.993	0.483	0.236	0.647	0.006	0.001	0.994	0.764	0.353	0.999
		Average	**0.967**	**0.743**	**0.175**	**0.301**	**0.024**	**0.036**	**0.976**	**0.825**	**0.699**	**0.965**

Table A.117: Comparison of the statistical information for prediction on Patient 3 based on two different datasets.

A.3.3 Patient 4

Visualization Results

Table A.118: Brain sections compared with different techniques of the third architecture on patient 4.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.119 and A.120) and with the augmented dataset (Tables A.121 and A.122). Table

A.123 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8069	42	8	6
Pred.	*II*	61	3670	117	14
Class	*III*	94	160	838	97
	IV	0	4	25	107

Table A.119: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8070	47	21	14
Pred.	*II*	91	3664	108	10
Class	*III*	72	150	828	85
	IV	0	6	29	117

Table A.120: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8001	82	32	9
Pred.	*II*	125	3636	124	10
Class	*III*	99	147	722	48
	IV	2	15	103	157

Table A.121: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8098	95	45	9
Pred.	*II*	64	3595	106	7
Class	*III*	61	158	671	20
	IV	1	32	162	188

Table A.122: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.984	0.987	0.019	0.007	0.011	0.03	0.989	0.981	0.993	0.97
		II	0.97	0.949	0.053	0.05	0.02	0.022	0.98	0.947	0.95	0.978
		III	0.962	0.77	0.152	0.295	0.028	0.012	0.972	0.848	0.705	0.988
		IV	0.989	0.594	0.522	0.213	0.002	0.009	0.998	0.478	0.787	0.991
		Average	**0.976**	**0.825**	**0.186**	**0.141**	**0.015**	**0.018**	**0.985**	**0.813**	**0.859**	**0.982**
	Adam	*I*	0.982	0.985	0.02	0.01	0.016	0.032	0.984	0.98	0.99	0.968
		II	0.969	0.947	0.052	0.054	0.022	0.022	0.978	0.948	0.946	0.978
		III	0.965	0.781	0.16	0.27	0.025	0.013	0.975	0.84	0.73	0.987
		IV	0.989	0.619	0.482	0.23	0.003	0.008	0.997	0.518	0.77	0.992
		Average	**0.976**	**0.833**	**0.178**	**0.141**	**0.017**	**0.019**	**0.983**	**0.821**	**0.859**	**0.981**
Data Augment.	SGD	*I*	0.974	0.979	0.027	0.015	0.024	0.044	0.976	0.973	0.985	0.956
		II	0.962	0.935	0.063	0.066	0.027	0.026	0.973	0.937	0.934	0.974
		III	0.958	0.723	0.264	0.289	0.024	0.021	0.976	0.736	0.711	0.979
		IV	0.986	0.627	0.299	0.433	0.009	0.005	0.991	0.701	0.567	0.995
		Average	**0.97**	**0.816**	**0.163**	**0.201**	**0.021**	**0.024**	**0.979**	**0.837**	**0.799**	**0.976**
	Adam	*I*	0.979	0.983	0.015	0.018	0.029	0.025	0.971	0.985	0.982	0.975
		II	0.965	0.94	0.073	0.047	0.019	0.03	0.981	0.927	0.953	0.97
		III	0.959	0.709	0.318	0.263	0.019	0.025	0.981	0.682	0.737	0.975
		IV	0.983	0.619	0.161	0.509	0.015	0.003	0.985	0.839	0.491	0.997
		Average	**0.972**	**0.813**	**0.142**	**0.209**	**0.021**	**0.021**	**0.979**	**0.858**	**0.791**	**0.979**

Table A.123: Comparison of the statistical information for prediction on Patient 4 based on two different datasets.

A.3.4 Patient 5

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.124: Brain sections compared with different techniques of the third architecture on patient 5.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.125 and A.126) and with the augmented dataset (Tables A.127 and A.128). Table

A.129 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8350	71	24	7
Class	*II*	401	3091	140	23
	III	44	168	847	52
	IV	0	0	18	76

Table A.125: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8017	38	28	7
Class	*II*	722	3181	159	23
	III	53	107	810	38
	IV	1	8	30	90

Table A.126: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8041	29	32	8
Class	*II*	695	3157	152	7
	III	53	127	782	18
	IV	4	19	63	125

Table A.127: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	8009	49	23	8
Class	*II*	727	3047	133	6
	III	57	203	823	24
	IV	0	35	48	120

Table A.128: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.959	0.968	0.051	0.012	0.023	0.092	0.977	0.949	0.988	0.908
		II	0.94	0.885	0.072	0.154	0.057	0.025	0.943	0.928	0.846	0.975
		III	0.966	0.792	0.177	0.238	0.021	0.015	0.979	0.823	0.762	0.985
		IV	0.992	0.603	0.519	0.191	0.001	0.006	0.999	0.481	0.809	0.994
		Average	**0.964**	**0.812**	**0.205**	**0.149**	**0.026**	**0.035**	**0.975**	**0.795**	**0.851**	**0.966**
	Adam	*I*	0.938	0.951	0.089	0.006	0.011	0.149	0.989	0.911	0.994	0.851
		II	0.923	0.861	0.041	0.219	0.089	0.015	0.911	0.959	0.781	0.985
		III	0.971	0.813	0.187	0.187	0.016	0.016	0.984	0.813	0.813	0.984
		IV	0.993	0.664	0.386	0.276	0.003	0.005	0.997	0.614	0.724	0.995
		Average	**0.956**	**0.822**	**0.176**	**0.172**	**0.03**	**0.046**	**0.97**	**0.824**	**0.828**	**0.954**
Data Augment.	SGD	*I*	0.936	0.95	0.088	0.009	0.016	0.149	0.984	0.912	0.991	0.851
		II	0.921	0.858	0.046	0.221	0.091	0.017	0.909	0.954	0.779	0.983
		III	0.969	0.796	0.211	0.196	0.016	0.018	0.984	0.789	0.804	0.982
		IV	0.992	0.627	0.43	0.302	0.003	0.005	0.997	0.57	0.698	0.995
		Average	**0.955**	**0.808**	**0.194**	**0.182**	**0.032**	**0.047**	**0.968**	**0.806**	**0.818**	**0.953**
	Adam	*I*	0.935	0.949	0.089	0.01	0.018	0.15	0.982	0.911	0.99	0.85
		II	0.913	0.841	0.086	0.221	0.087	0.031	0.913	0.914	0.779	0.969
		III	0.963	0.771	0.199	0.257	0.023	0.017	0.977	0.801	0.743	0.983
		IV	0.991	0.665	0.241	0.409	0.006	0.003	0.994	0.759	0.591	0.997
		Average	**0.951**	**0.806**	**0.154**	**0.224**	**0.034**	**0.05**	**0.966**	**0.846**	**0.776**	**0.95**

Table A.129: Comparison of the statistical information for prediction on Patient 5 based on two different datasets.

A.3.5 Patient 6

Visualization Results

Table A.130: Brain sections compared with different techniques of the third architecture on patient 6.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.131 and A.132) and with the augmented dataset (Tables A.133 and A.134). Table

A.135 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8775	56	1	0
Pred.	*II*	100	3489	86	0
Class	*III*	81	181	542	0
	IV	0	1	0	0

Table A.131: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8734	59	4	0
Pred.	*II*	144	3419	53	0
Class	*III*	78	248	572	0
	IV	0	1	0	0

Table A.132: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8724	60	2	0
Pred.	*II*	146	3524	101	0
Class	*III*	84	126	522	0
	IV	0	20	3	0

Table A.133: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8787	113	12	0
Pred.	*II*	119	3402	63	0
Class	*III*	48	206	553	0
	IV	0	9	0	0

Table A.134: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.982	0.987	0.02	0.006	0.013	0.04	0.987	0.98	0.994	0.96
		II	0.968	0.943	0.064	0.051	0.019	0.025	0.981	0.936	0.949	0.975
		III	0.974	0.756	0.138	0.326	0.021	0.007	0.979	0.862	0.674	0.993
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.981**	**0.671**	**0.305**	**0.346**	**0.013**	**0.018**	**0.987**	**0.695**	**0.654**	**0.982**
	Adam	*I*	0.979	0.984	0.025	0.007	0.014	0.049	0.986	0.975	0.993	0.951
		II	0.962	0.931	0.083	0.054	0.021	0.032	0.979	0.917	0.946	0.968
		III	0.971	0.749	0.091	0.363	0.026	0.005	0.974	0.909	0.637	0.995
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.978**	**0.666**	**0.3**	**0.356**	**0.015**	**0.022**	**0.985**	**0.7**	**0.644**	**0.979**
Data Augment.	SGD	*I*	0.978	0.984	0.026	0.007	0.014	0.051	0.986	0.974	0.993	0.949
		II	0.966	0.94	0.055	0.065	0.026	0.022	0.974	0.945	0.935	0.978
		III	0.976	0.768	0.169	0.287	0.017	0.008	0.983	0.831	0.713	0.992
		IV	0.998	0.0	1	1.0	0.002	0.0	0.998	0	0.0	1.0
		Average	**0.98**	**0.673**	**0.312**	**0.34**	**0.015**	**0.02**	**0.985**	**0.688**	**0.66**	**0.98**
	Adam	*I*	0.978	0.984	0.019	0.014	0.029	0.038	0.971	0.981	0.986	0.962
		II	0.962	0.93	0.088	0.051	0.019	0.034	0.981	0.912	0.949	0.966
		III	0.975	0.771	0.119	0.315	0.02	0.006	0.98	0.881	0.685	0.994
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
		Average	**0.979**	**0.671**	**0.306**	**0.345**	**0.017**	**0.02**	**0.983**	**0.694**	**0.655**	**0.98**

Table A.135: Comparison of the statistical information for prediction on Patient 6 based on two different datasets.

A.3.6 Patient 7

Visualization Results

Table A.136: Brain sections compared with different techniques of the third architecture on patient 7.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.137 and A.138) and with the augmented dataset (Tables A.139 and A.140). Table

A.141 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8938	92	7	0
Pred.	*II*	43	3041	99	0
Class	*III*	70	273	728	13
	IV	0	2	4	2

Table A.137: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8946	95	6	0
Pred.	*II*	50	3001	76	0
Class	*III*	55	304	757	11
	IV	0	5	2	4

Table A.138: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8965	136	17	0
Pred.	*II*	42	2992	100	0
Class	*III*	45	217	682	9
	IV	0	63	38	6

Table A.139: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	8973	120	18	0
Pred.	*II*	25	3032	97	0
Class	*III*	53	230	697	10
	IV	0	26	26	5

Table A.140: Adam optimizer.

			Metrics								
DS	**Opt.**	**Class**	Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.984	0.988	0.012	0.011	0.023	0.026	0.977	0.988	0.989	0.974
		II	0.962	0.923	0.108	0.045	0.014	0.036	0.986	0.892	0.955	0.964
		III	0.965	0.758	0.131	0.328	0.029	0.009	0.971	0.869	0.672	0.991
		IV	0.999	0.174	0.867	0.75	0.0	0.001	1.0	0.133	0.25	0.999
		Average	**0.978**	**0.711**	**0.279**	**0.283**	**0.017**	**0.018**	**0.984**	**0.72**	**0.717**	**0.982**
	Adam	*I*	0.985	0.989	0.012	0.011	0.024	0.025	0.976	0.988	0.989	0.975
		II	0.96	0.919	0.119	0.04	0.013	0.04	0.987	0.881	0.96	0.96
		III	0.966	0.769	0.1	0.328	0.03	0.007	0.97	0.9	0.672	0.993
		IV	0.999	0.308	0.733	0.636	0.001	0.001	0.999	0.267	0.364	0.999
		Average	**0.977**	**0.746**	**0.241**	**0.254**	**0.017**	**0.018**	**0.983**	**0.759**	**0.746**	**0.982**
Data Augment.	SGD	*I*	0.982	0.987	0.01	0.017	0.036	0.021	0.964	0.99	0.983	0.979
		II	0.958	0.915	0.122	0.045	0.014	0.041	0.986	0.878	0.955	0.959
		III	0.968	0.762	0.185	0.284	0.022	0.013	0.978	0.815	0.716	0.987
		IV	0.992	0.098	0.6	0.944	0.008	0.001	0.992	0.4	0.056	0.999
		Average	**0.975**	**0.691**	**0.229**	**0.323**	**0.02**	**0.019**	**0.98**	**0.771**	**0.677**	**0.981**
	Adam	*I*	0.984	0.988	0.009	0.015	0.032	0.019	0.968	0.991	0.985	0.981
		II	0.963	0.924	0.11	0.039	0.012	0.037	0.988	0.89	0.961	0.963
		III	0.967	0.763	0.168	0.296	0.023	0.011	0.977	0.832	0.704	0.989
		IV	0.995	0.139	0.667	0.912	0.004	0.001	0.996	0.333	0.088	0.999
		Average	**0.978**	**0.718**	**0.181**	**0.296**	**0.018**	**0.017**	**0.982**	**0.819**	**0.704**	**0.983**

Table A.141: Comparison of the statistical information for prediction on Patient 7 based on two different datasets.

A.3.7 Patient 8

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table A.142: Brain sections compared with different techniques of the third architecture on patient 8.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.143 and A.144) and with the augmented dataset (Tables A.145 and A.146). Table

A.147 displays the statistical information for different classes.

		Actual Class			
		I	II	III	IV
	I	9800	17	1	1
Pred.	II	270	2287	80	8
Class	III	58	92	657	15
	IV	0	2	6	18

Table A.143: SGD optimizer.

		Actual Class			
		I	II	III	IV
	I	9804	22	2	3
Pred.	II	267	2278	68	6
Class	III	57	97	672	16
	IV	0	0	4	16

Table A.144: Adam optimizer.

		Actual Class			
		I	II	III	IV
	I	9813	45	12	2
Pred.	II	265	2229	73	4
Class	III	46	100	630	6
	IV	6	24	28	29

Table A.145: SGD optimizer.

		Actual Class			
		I	II	III	IV
	I	9795	25	5	3
Pred.	II	258	2267	73	3
Class	III	71	95	637	6
	IV	4	11	30	29

Table A.146: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	I	0.974	0.983	0.032	0.002	0.006	0.094	0.994	0.968	0.998	0.906
		II	0.965	0.907	0.046	0.135	0.033	0.01	0.967	0.954	0.865	0.99
		III	0.981	0.839	0.117	0.201	0.013	0.007	0.987	0.883	0.799	0.993
		IV	0.998	0.529	0.571	0.308	0.001	0.002	0.999	0.429	0.692	0.998
		Average	**0.980**	**0.815**	**0.192**	**0.162**	**0.013**	**0.028**	**0.987**	**0.808**	**0.839**	**0.972**
	Adam	I	0.974	0.982	0.032	0.003	0.008	0.093	0.992	0.968	0.997	0.907
		II	0.965	0.908	0.05	0.13	0.031	0.011	0.969	0.95	0.87	0.989
		III	0.982	0.846	0.099	0.202	0.014	0.006	0.986	0.901	0.798	0.994
		IV	0.998	0.525	0.61	0.2	0.0	0.002	1.0	0.39	0.8	0.998
		Average	**0.980**	**0.815**	**0.198**	**0.134**	**0.013**	**0.028**	**0.987**	**0.802**	**0.866**	**0.972**
Data Augment.	SGD	I	0.972	0.981	0.031	0.006	0.019	0.092	0.981	0.969	0.994	0.908
		II	0.962	0.897	0.07	0.133	0.031	0.016	0.969	0.93	0.867	0.984
		III	0.98	0.826	0.152	0.194	0.012	0.009	0.988	0.848	0.806	0.991
		IV	0.995	0.453	0.293	0.667	0.004	0.001	0.996	0.707	0.333	0.999
		Average	**0.977**	**0.789**	**0.137**	**0.25**	**0.017**	**0.029**	**0.983**	**0.863**	**0.75**	**0.971**
	Adam	I	0.973	0.982	0.033	0.003	0.01	0.096	0.99	0.967	0.997	0.904
		II	0.965	0.907	0.055	0.128	0.031	0.012	0.969	0.945	0.872	0.988
		III	0.979	0.82	0.144	0.213	0.014	0.009	0.986	0.856	0.787	0.991
		IV	0.996	0.504	0.293	0.608	0.003	0.001	0.997	0.707	0.392	0.999
		Average	**0.978**	**0.803**	**0.132**	**0.238**	**0.015**	**0.029**	**0.986**	**0.868**	**0.762**	**0.971**

Table A.147: Comparison of the statistical information for prediction on Patient 8 based on two different datasets.

A.3.8   Patient 9

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7	Sec. 8	Sec. 9	Sec. 10	Sec. 11
Ground Truth											
Normal Dataset SGD											
Normal Dataset Adam											
Data Augmen. SGD											
Data Augmen. Adam											

Sec. 12	Sec. 13	Sec. 14	Sec. 15	Sec. 16	Sec. 17	Sec. 18	Sec. 19	Sec. 20	Sec. 21	Sec. 22

Table A.148: Brain sections compared with different techniques of the third architecture on patient 9.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.149 and A.150) and with the augmented dataset (Tables A.151 and A.152). Table A.153 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	14301	79	10	0
	II	179	6711	149	0
	III	74	182	841	0
	IV	0	2	0	0

Table A.149: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	14280	76	8	0
	II	214	6675	125	0
	III	60	223	867	0
	IV	0	0	0	0

Table A.150: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	14281	74	7	0
	II	193	6764	183	0
	III	68	144	809	0
	IV	2	2	1	0

Table A.151: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred. Class	*I*	14279	110	13	0
	II	191	6555	129	0
	III	69	313	858	0
	IV	4	6	1	0

Table A.152: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.985	0.988	0.017	0.006	0.011	0.031	0.989	0.983	0.994	0.969
		II	0.974	0.958	0.038	0.047	0.021	0.017	0.979	0.962	0.953	0.983
		III	0.982	0.802	0.159	0.233	0.012	0.007	0.988	0.841	0.767	0.993
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.985**	**0.687**	**0.303**	**0.322**	**0.011**	**0.014**	**0.989**	**0.696**	**0.678**	**0.986**
	Adam	*I*	0.984	0.988	0.019	0.006	0.011	0.034	0.989	0.981	0.994	0.966
		II	0.972	0.954	0.043	0.048	0.022	0.019	0.978	0.957	0.952	0.981
		III	0.982	0.807	0.133	0.246	0.013	0.006	0.987	0.867	0.754	0.994
		IV	1.0	0	1	1	0.0	0.0	1.0	0	0	1.0
		Average	**0.984**	**0.687**	**0.299**	**0.325**	**0.011**	**0.015**	**0.989**	**0.701**	**0.675**	**0.985**
Data Augment.	SGD	*I*	0.985	0.988	0.018	0.006	0.01	0.032	0.99	0.982	0.994	0.968
		II	0.974	0.958	0.032	0.053	0.024	0.014	0.976	0.968	0.947	0.986
		III	0.982	0.801	0.191	0.208	0.01	0.009	0.99	0.809	0.792	0.991
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.985**	**0.687**	**0.31**	**0.317**	**0.011**	**0.014**	**0.989**	**0.69**	**0.683**	**0.986**
	Adam	*I*	0.983	0.987	0.018	0.009	0.015	0.032	0.985	0.982	0.991	0.968
		II	0.967	0.946	0.061	0.047	0.021	0.027	0.979	0.939	0.953	0.973
		III	0.977	0.766	0.143	0.308	0.018	0.007	0.982	0.857	0.692	0.993
		IV	1.0	0.0	1	1.0	0.0	0.0	1.0	0	0.0	1.0
		Average	**0.982**	**0.675**	**0.305**	**0.341**	**0.014**	**0.017**	**0.986**	**0.694**	**0.659**	**0.983**

Table A.153: Comparison of the statistical information for prediction on Patient 9 based on two different datasets.

A.3.9 Patient 10

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14

Table A.154: Brain sections compared with different techniques of the third architecture on patient 10.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.155 and A.156) and with the augmented dataset (Tables A.157 and A.158). Table A.159 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10287	19	15	1
Class	*II*	122	2976	91	5
	III	55	136	529	21
	IV	0	5	30	44

Table A.155: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10330	25	8	2
Class	*II*	72	2990	76	5
	III	56	119	560	22
	IV	0	6	23	42

Table A.156: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10321	64	29	2
Class	*II*	101	2907	78	6
	III	38	133	509	21
	IV	0	36	49	42

Table A.157: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
Pred.	*I*	10314	35	24	4
Class	*II*	84	2955	78	4
	III	59	119	481	4
	IV	1	31	84	59

Table A.158: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.985	0.99	0.017	0.003	0.009	0.044	0.991	0.983	0.997	0.956
		II	0.974	0.94	0.051	0.068	0.019	0.014	0.981	0.949	0.932	0.986
		III	0.976	0.752	0.205	0.286	0.016	0.01	0.984	0.795	0.714	0.99
		IV	0.996	0.587	0.38	0.443	0.002	0.002	0.998	0.62	0.557	0.998
	Average		**0.983**	**0.817**	**0.163**	**0.2**	**0.011**	**0.017**	**0.988**	**0.837**	**0.8**	**0.982**
	Adam	*I*	0.989	0.992	0.012	0.003	0.009	0.032	0.991	0.988	0.997	0.968
		II	0.979	0.952	0.048	0.049	0.014	0.013	0.986	0.952	0.951	0.987
		III	0.979	0.787	0.16	0.26	0.014	0.008	0.986	0.84	0.74	0.992
		IV	0.996	0.592	0.408	0.408	0.002	0.002	0.998	0.592	0.592	0.998
	Average		**0.986**	**0.831**	**0.157**	**0.18**	**0.01**	**0.014**	**0.99**	**0.843**	**0.82**	**0.986**
Data Augment.	SGD	*I*	0.984	0.989	0.013	0.009	0.025	0.035	0.975	0.987	0.991	0.965
		II	0.971	0.933	0.074	0.06	0.017	0.021	0.983	0.926	0.94	0.979
		III	0.976	0.745	0.235	0.274	0.014	0.011	0.986	0.765	0.726	0.989
		IV	0.992	0.424	0.408	0.669	0.006	0.002	0.994	0.592	0.331	0.998
	Average		**0.981**	**0.773**	**0.182**	**0.253**	**0.015**	**0.017**	**0.984**	**0.818**	**0.747**	**0.983**
	Adam	*I*	0.986	0.99	0.014	0.006	0.016	0.036	0.984	0.986	0.994	0.964
		II	0.976	0.944	0.059	0.053	0.015	0.016	0.985	0.941	0.947	0.984
		III	0.974	0.723	0.279	0.275	0.013	0.014	0.987	0.721	0.725	0.986
		IV	0.991	0.48	0.169	0.663	0.008	0.001	0.992	0.831	0.337	0.999
	Average		**0.982**	**0.784**	**0.13**	**0.249**	**0.013**	**0.017**	**0.987**	**0.87**	**0.751**	**0.983**

Table A.159: Comparison of the statistical information for prediction on Patient 10 based on two different datasets.

A.3.10   Patient 11

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14

Table A.160: Brain sections compared with different techniques of the third architecture on patient 11.

Analysis of different Datasets

The following tables represent the confusion matrices with the normal dataset (Tables A.161 and A.162) and with the augmented dataset (Tables A.163 and A.164). Table

A.165 displays the statistical information for different classes.

		Actual Class			
		I	*II*	*III*	*IV*
	I	9978	88	45	0
Pred.	*II*	471	2456	428	0
Class	*III*	61	239	549	0
	IV	0	15	6	0

Table A.161: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	9978	76	32	0
Pred.	*II*	462	2446	464	0
Class	*III*	60	247	523	0
	IV	9	30	9	0

Table A.162: Adam optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	10024	99	50	0
Pred.	*II*	434	2479	541	0
Class	*III*	43	179	411	0
	IV	8	42	26	0

Table A.163: SGD optimizer.

		Actual Class			
		I	*II*	*III*	*IV*
	I	9958	86	60	0
Pred.	*II*	466	2466	567	0
Class	*III*	65	178	356	0
	IV	20	69	45	0

Table A.164: Adam optimizer.

DS	Opt.	Class	Metrics									
			Acc.	F1 score	FDR	FNR	FOR	FPR	NPV	Prec.	Recall	Selec.
Normal	SGD	*I*	0.954	0.968	0.051	0.013	0.035	0.126	0.965	0.949	0.987	0.874
		II	0.913	0.798	0.122	0.268	0.078	0.031	0.922	0.878	0.732	0.969
		III	0.946	0.585	0.466	0.353	0.023	0.036	0.977	0.534	0.647	0.964
		IV	0.999	0.0	1	1.0	0.001	0.0	0.999	0	0.0	1.0
		Average	**0.953**	**0.588**	**0.41**	**0.408**	**0.034**	**0.048**	**0.966**	**0.59**	**0.591**	**0.952**
	Adam	*I*	0.956	0.969	0.046	0.015	0.039	0.117	0.961	0.954	0.985	0.883
		II	0.91	0.793	0.114	0.282	0.085	0.029	0.915	0.886	0.718	0.971
		III	0.941	0.495	0.6	0.351	0.017	0.045	0.983	0.4	0.649	0.955
		IV	0.995	0.0	1	1.0	0.005	0.0	0.995	0	0.0	1.0
		Average	**0.951**	**0.564**	**0.44**	**0.412**	**0.037**	**0.048**	**0.964**	**0.56**	**0.588**	**0.952**
Data Augment.	SGD	*I*	0.948	0.963	0.06	0.012	0.03	0.146	0.97	0.94	0.988	0.854
		II	0.901	0.781	0.089	0.316	0.102	0.024	0.898	0.911	0.684	0.976
		III	0.943	0.475	0.639	0.305	0.012	0.048	0.988	0.361	0.695	0.952
		IV	0.994	0.0	1	1.0	0.006	0.0	0.994	0	0.0	1.0
		Average	**0.946**	**0.555**	**0.447**	**0.408**	**0.038**	**0.054**	**0.962**	**0.553**	**0.592**	**0.946**
	Adam	*I*	0.951	0.966	0.052	0.014	0.038	0.13	0.962	0.948	0.986	0.87
		II	0.905	0.783	0.119	0.295	0.09	0.031	0.91	0.881	0.705	0.969
		III	0.936	0.438	0.654	0.406	0.018	0.049	0.982	0.346	0.594	0.951
		IV	0.991	0.0	1	1.0	0.009	0.0	0.991	0	0.0	1.0
		Average	**0.946**	**0.547**	**0.456**	**0.429**	**0.039**	**0.053**	**0.961**	**0.544**	**0.571**	**0.948**

Table A.165: Comparison of the statistical information for prediction on Patient 11 based on two different datasets.

B

Results for Pixel by Pixel Segmentation Approach

The chapter presents the final results for the predicted images realized with the Pixel by Pixel Segmentation approach described and analyzed in detail in Chap. 5. The final training and testing accuracy percentages plus results for loss equations are presented in Table 5.3.

B.1   Results for U-net

This section displays predicting results for all brain sections for all patients using the K-Fold cross-validation, through the U-net architecture, described in detail in Sec. 5.4. The first row of each table represents the ground truth of a brain section, extrapolated from the manual annotation with a pre-processing step described in Sec. 3.3.2. Second and third rows show the predicted brain section resulting from the normal dataset and two different optimizer functions, SGD and Adam. The last two rows display the same brain sections with an augmented dataset and the optimizer functions.

B.1.1   Patient 2

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.1: Brain sections compared with different techniques on patient 2.

B.1.2 Patient 3

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.2: Brain sections compared with different techniques on patient 3.

B.1.3 Patient 4

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.3: Brain sections compared with different techniques on patient 4.

B.1.4 Patient 5

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.4: Brain sections compared with different techniques on patient 5.

B.1.5   Patient 6

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.5: Brain sections compared with different techniques on patient 6.

B.1.6   Patient 7

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.6: Brain sections compared with different techniques on patient 7.

B.1.7 Patient 8

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13

Table B.7: Brain sections compared with different techniques on patient 8.

B.1.8 Patient 9

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14	Sec. 15

Sec. 16	Sec. 17	Sec. 18	Sec. 19	Sec. 20	Sec. 21	Sec. 22

Table B.8: Brain sections compared with different techniques on patient 9.

B.1.9 Patient 10

Visualization Results

	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7
Ground Truth							
Normal Dataset SGD							
Normal Dataset Adam							
Data Augmen. SGD							
Data Augmen. Adam							

Sec. 8	Sec. 9	Sec. 10	Sec. 11	Sec. 12	Sec. 13	Sec. 14

Table B.9: Brain sections compared with different techniques on patient 10.

B.1.10 Patient 11

Visualization Results

Table B.10: Brain sections compared with different techniques on patient 11.

C

Thesis Code

The following Appendix briefly describes the files included in this thesis's attachment. The attachment contains two different compressed folders: *pythoncode.zip* and *matlabcode.zip*.

The *pythoncode.zip* contains the building core of the various CNNs:

- **extract_annotation.py**: the script consists of function to extract the manual annotations defined in Sec. 3.3.2.

- **get_complete_training_data.py**: the file contains all the methods to retrieve and parse the new dataset for training, as described in Sec. 3.3.3 and Sec. 3.3.3.

- **run_training.py**: this is the main script which contains the implementations of the four different architectures proposed with the various experiment setting. The script refers to Chap. 4 and Chap. 5.

The *matlabcode.zip* contains the scripts to start the pre-processing steps (to use together with the code created in [63, 66]):

- **anotherSkullRemovalTechnique.m, generalSkullRemoval.m**: two similar scripts to compute the skull removal in the DICOM images.

- **improveImagesSkullRemoved.m**: function that enhance the contrast in the images without the skull in order to augment the different values in the pixels, as described in 3.3.1.

- **MAIN_PREPROCESSING.m**: main script that consists of the call of the various function involved in the pre-processing steps.

- **pre_processing.py**: python script to call the **MAIN_PREPROCESSING.m** script.

- **registerAnnotated.m**: script to register the images with the corresponding manual annotation image.

- **start_pre_processing.m**: MATLAB script to call the **MAIN_PREPROCESSING.m** script.

Bibliography

[1] CP Warlow. Epidemiology of stroke. *The Lancet*, 352:S1–S4, 1998.

[2] Bent Indredavik, R Salvesen, H Næss, and D Thorsvik. Nasjonal retningslinje for behandling og rehabilitering ved hjerneslag. *Oslo: Helsedirektoratet*, 4, 2010.

[3] Haidong Wang, Mohsen Naghavi, Christine Allen, Ryan M Barber, Zulfiqar A Bhutta, Austin Carter, Daniel C Casey, Fiona J Charlson, Alan Zian Chen, Matthew M Coates, et al. Global, regional, and national life expectancy, all-cause mortality, and cause-specific mortality for 249 causes of death, 1980–2015: a systematic analysis for the global burden of disease study 2015. *The lancet*, 388(10053):1459–1544, 2016.

[4] Rupali Rajendra Akerkar, Grace M Egeland, Janne Dyngeland, Rune Kvåle, Marta Ebbing, Truc Trung Nguyen, Inger Johanne Bakken, and Gunhild Forland. Hjerte-og karregisteret: Rapport for 2012–2016. 2018.

[5] The top 10 causes of death. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death, 2018. [Online].

[6] Luciano A Sposato, Lauren E Cipriano, Gustavo Saposnik, Estefanía Ruíz Vargas, Patricia M Riccio, and Vladimir Hachinski. Diagnosis of atrial fibrillation after stroke and transient ischaemic attack: a systematic review and meta-analysis. *The Lancet Neurology*, 14(4):377–387, 2015.

[7] Norhealth. Norwegian Institute of Public Health: Norway's health statistic database. http://www.norgeshelsa.no/norgeshelsa/, 2017. [Online].

[8] World Health Organization et al. Cerebrovascular disorders: a clinical and research classification. 1978.

[9] Gabor T Herman. *Fundamentals of computerized tomography: image reconstruction from projections.* Springer Science & Business Media, 2009.

[10] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach.* Malaysia; Pearson Education Limited, 2016.

[11] David H Hubel and Torsten N Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1):215–243, 1968.

[12] Samer Hijazi, Rishi Kumar, and Chris Rowen. Using convolutional neural networks for image recognition. *Cadence Design Systems Inc.: San Jose, CA, USA*, 2015.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] KD Kurz, G Ringstad, A Odland, R Advani, E Farbu, and MW Kurz. Radiological imaging in acute ischaemic stroke. *European journal of neurology*, 23:8–17, 2016.

[15] Werner Hacke, Markku Kaste, Erich Bluhmki, Miroslav Brozman, Antoni Dávalos, Donata Guidetti, Vincent Larrue, Kennedy R Lees, Zakaria Medeghri, Thomas Machnig, et al. Thrombolysis with alteplase 3 to 4.5 hours after acute ischemic stroke. *New England journal of medicine*, 359(13):1317–1329, 2008.

[16] Jeffrey L Saver. Time is brain-quantified. *Stroke*, 37(1):263–266, 2006.

[17] Klaus Kaae Andersen, Tom Skyhøj Olsen, Christian Dehlendorff, and Lars Peter Kammersgaard. Hemorrhagic and ischemic strokes compared: stroke severity, mortality, and risk factors. *Stroke*, 40(6):2068–2072, 2009.

[18] J Donald Easton, Jeffrey L Saver, Gregory W Albers, Mark J Alberts, Seemant Chaturvedi, Edward Feldmann, Thomas S Hatsukami, Randall T Higashida, S Claiborne Johnston, Chelsea S Kidwell, et al. Definition and evaluation of transient ischemic attack: a scientific statement for healthcare professionals from the american heart association/american stroke association stroke council; council on cardiovascular surgery and anesthesia; council on cardiovascular radiology and intervention; council on cardiovascular nursing; and the interdisciplinary council on peripheral vascular disease: the american academy of neurology affirms the value of this statement as an educational tool for neurologists. *Stroke*, 40(6):2276–2293, 2009.

[19] Thrombus. MedlinePlus. U.S. National Library of Medicine. https://medlineplus.gov/ency/imagepages/18120.htm, 2016. [Online].

[20] Vinay Kumar, Abul K Abbas, Nelson Fausto, and Jon C Aster. *Robbins and Cotran pathologic basis of disease, professional edition e-book*. Elsevier health sciences, 2014.

[21] Ashfaq Shuaib and Vladimir C Hachinski. Mechanisms and management of stroke in the elderly. *CMAJ: Canadian Medical Association Journal*, 145(5):433, 1991.

[22] Jan Stam. Thrombosis of the cerebral veins and sinuses. *New England Journal of Medicine*, 352(17):1791–1798, 2005.

[23] Mayank Goyal, Andrew M Demchuk, Bijoy K Menon, Muneer Eesa, Jeremy L Rempel, John Thornton, Daniel Roy, Tudor G Jovin, Robert A Willinsky, Biggya L Sapkota, et al. Randomized assessment of rapid endovascular treatment of ischemic stroke. *New England Journal of Medicine*, 372(11):1019–1030, 2015.

[24] Marc Fisher and Myron Ginsberg. Current concepts of the ischemic penumbra: introduction. *Stroke*, 35(11_suppl_1):2657–2658, 2004.

[25] European Stroke Organisation (ESO) Executive Committee, ESO Writing Committee, et al. Guidelines for management of ischaemic stroke and transient ischaemic attack 2008. *Cerebrovascular diseases*, 25(5):457–507, 2008.

[26] Avinash C Kak, Malcolm Slaney, and Ge Wang. Principles of computerized tomographic imaging. *Medical Physics*, 29(1):107–107, 2002.

[27] David J Brenner and Eric J Hall. Computed tomography an increasing source of radiation exposure. *New England Journal of Medicine*, 357(22):2277–2284, 2007.

[28] American College of Radiology et al. Acr–asnr–spr practice parameter for the performance of computed tomography (ct) of the extracranial head and neck, 2016.

[29] M Wintermark and MH Lev. Fda investigates the safety of brain perfusion ct, 2010.

[30] Andrew Bivard, Neil Spratt, Christopher R Levi, and Mark W Parsons. Acute stroke thrombolysis: time to dispense with the clock and move to tissue-based decision making? *Expert review of cardiovascular therapy*, 9(4):451–461, 2011.

[31] Lois Romans. *Computed Tomography for Technologists: A comprehensive text*. Lippincott Williams & Wilkins, 2018.

[32] Tomasz Hachaj and Marek R Ogiela. Cad system for automatic analysis of ct perfusion maps. *Opto-Electronics Review*, 19(1):95–103, 2011.

[33] Max Wintermark, Marc Reichhart, Jean-Philippe Thiran, Philippe Maeder, Marc Chalaron, Pierre Schnyder, Julien Bogousslavsky, and Reto Meuli. Prognostic accuracy of cerebral blood flow measurement by perfusion computed tomography, at the time of emergency room admission, in acute stroke patients. *Annals of Neurology: Official Journal of the American Neurological Association and the Child Neurology Society*, 51(4):417–432, 2002.

[34] Self-Learning Packet. Overview of adult traumatic brain injuries. *Orlando Regional Healthcare, Education & Development, Orlando, FL*, 2004.

[35] Niranjan Khandelwal. Ct perfusion in acute stroke. *The Indian journal of radiology & imaging*, 18(4):281, 2008.

[36] Dominik Deniffel, Timothé Boutelier, Aissam Labani, Mickael Ohana, Daniela Pfeiffer, and Catherine Roy. Computed tomography perfusion measurements in renal lesions obtained by bayesian estimation, advanced singular-value decomposition deconvolution, maximum slope, and patlak models: Intermodel agreement and diagnostic accuracy of tumor classification. *Investigative radiology*, 53(8):477–485, 2018.

[37] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[38] Jim X Chen. The evolution of computing: Alphago. *Computing in Science & Engineering*, 18(4):4, 2016.

[39] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[40] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

[41] Reuven Y Rubinstein and Dirk P Kroese. *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning.* Springer Science & Business Media, 2013.

[42] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

[43] Alex P Zijdenbos, Benoit M Dawant, Richard A Margolin, and Andrew C Palmer. Morphometric analysis of white matter lesions in mr images: method and validation. *IEEE transactions on medical imaging*, 13(4):716–724, 1994.

[44] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553):436, 2015.

[45] David E Rumelhart, Geoffrey E Hinton, Ronald J Williams, et al. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[46] Benjamin Recht, Christopher Re, Stephen Wright, and Feng Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in neural information processing systems*, pages 693–701, 2011.

[47] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[48] Geoffrey E Hinton, Terrence Joseph Sejnowski, and Tomaso A Poggio. *Unsupervised learning: foundations of neural computation.* MIT press, 1999.

[49] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[50] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2013.

[51] A Krizhevsky, I Sutskever, and G Hinton. //proc. advances in neural inform. *Proces. Systems. 2012. V. 25.*, 25:1090, 2012.

[52] Tara N Sainath, Abdel-rahman Mohamed, Brian Kingsbury, and Bhuvana Ramabhadran. Deep convolutional neural networks for lvcsr. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 8614–8618. IEEE, 2013.

[53] Michael KK Leung, Hui Yuan Xiong, Leo J Lee, and Brendan J Frey. Deep learning of the tissue-regulated splicing code. *Bioinformatics*, 30(12):i121–i129, 2014.

[54] Hui Y Xiong, Babak Alipanahi, Leo J Lee, Hannes Bretschneider, Daniele Merico, Ryan KC Yuen, Yimin Hua, Serge Gueroussov, Hamed S Najafabadi, Timothy R Hughes, et al. The human splicing code reveals new insights into the genetic determinants of disease. *Science*, 347(6218):1254806, 2015.

[55] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[56] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[57] CS231n Convolutional Neural Networks for Visual Recognition. `http://cs231n.github.io/`, 2019. [Online].

[58] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285*, 2016.

[59] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[60] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[61] Mu Zhu. Recall, precision and average precision. *Department of Statistics and Actuarial Science, University of Waterloo, Waterloo*, 2:30, 2004.

[62] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.

[63] Eivind Hovland. Feature extraction for exploring infarcted regions in perfusion ct images of the brain. Master's thesis, University of Stavanger, Norway, 2018.

[64] DICOM Standard. https://www.dicomstandard.org/, 2019. [Online].

[65] Mario Mustra, Kresimir Delac, and Mislav Grgic. Overview of the dicom standard. In *2008 50th International Symposium ELMAR*, volume 1, pages 39–44. IEEE, 2008.

[66] Sigurd Myklebust. Cerebral vessel segmentation in contrast ct images. Master's thesis, University of Stavanger, Norway, 2018.

[67] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM computing surveys (CSUR)*, 24(4):325–376, 1992.

[68] Sudipta Roy, Sanjay Nag, Indra Kanta Maitra, and Samir Kumar Bandyopadhyay. Artefact removal and skull elimination from mri of brain image. *International Journal of Scientific and Engineering Research*, 4(6):163–170, 2013.

[69] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle. Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31, 2017.

[70] Xiaomei Zhao, Yihong Wu, Guidong Song, Zhenye Li, Yazhuo Zhang, and Yong Fan. A deep learning model integrating fcnns and crfs for brain tumor segmentation. *Medical image analysis*, 43:98–111, 2018.

[71] Konstantinos Kamnitsas, Christian Ledig, Virginia FJ Newcombe, Joanna P Simpson, Andrew D Kane, David K Menon, Daniel Rueckert, and Ben Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical image analysis*, 36:61–78, 2017.

[72] Bora Erden, Noah Gamboa, and Sam Wood. 3d convolutional neural network for brain tumor segmentation, 2018.

[73] Adrien Payan and Giovanni Montana. Predicting alzheimer's disease: a neuroimaging study with 3d convolutional neural networks. *arXiv preprint arXiv:1502.02506*, 2015.