TÉCNICO LISBOA

Using a Siamese Network to Accurately Detect Ischemic Stroke in Computed Tomography Scans

Ana Beatriz Gil Vieira

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. Arlindo Manuel Limede de Oliveira
Prof. Ana Catarina Gaspar Fonseca

Examination Committee

Chairperson: Prof. João Manuel de Freitas Xavier
Supervisor: Prof. Arlindo Manuel Limede de Oliveira
Member of the Committee: Prof. Maria Margarida Campos da Silveira

June 2022

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

To my grandmother,
who died 2 years ago from an ischemic stroke.

Acknowledgments

First, I would like to thank my supervisors, professor Catarina Fonseca for the welcome at *Hospital de Santa Maria*, for her availability and help in this huge and complex medical area, and professor Arlindo Oliveira for his availability, guidance, knowledge and motivation during this last year.

I would also like to thank my family for their constant support, for believing in me and never letting me give up in the face of numerous difficulties and, specially for giving me the opportunity to learn in such a distinguished institute as *Instituto Superior Técnico*.

Last but not least, I would like to thank all my friends and colleagues for believing in me, for all the help they gave me with their knowledge, for all the hours spent finding solutions to the problems that kept appearing, and for the motivation to continue this adventure.

To each and every one of you – Thank you.

Abstract

The diagnosis procedure of stroke, a leading cause of death in the world, involves the acquisition of images using computed tomography scans, making possible the assessment of the severity of the incident and the type and location of the lesion. The fact that the brain has two hemispheres with a high level of anatomical similarity, exhibiting significant symmetry, has led to extensive research based on the assumption that a decrease in symmetry is directly related to the presence of pathologies. This work is focused on the analysis of the symmetry (or lack of it) of the two brain hemispheres, and on the use of this information for the classification of computed tomography brain scans of stroke patients. The objective is to contribute to the process of automatic identification of brain lesions caused by stroke events. To perform this task, we used the Siamese Network architecture, which uses two parallel neural networks that share the same weights. The composed network receives a double image (the original image and the mirrored one) and a label that reflects the existence or not of stroke. The network then extracts the relevant features and classifies the images taking into account their similarity. The resulting network can be used to classify unseen scans, depending on the perceived level of symmetry into one of two existing classes: evidence of stroke or absence of stroke. The accuracy of the proposed method is approximately 72%, significantly outperforming a standard convolutional network architecture, which was used as a baseline.

Keywords

Stroke, Computed Tomography, Symmetry Detection, Image Classification, Convolutional Neural Network, Siamese Network.

Resumo

O procedimento de diagnóstico do acidente vascular cerebral (AVC), uma das principais causas de morte no mundo, envolve a aquisição de imagens de tomografias computadorizadas cerebrais, tornando possível avaliar a gravidade do AVC e o tipo e a localização da lesão. O facto de o cérebro ter dois hemisférios com elevado nível de semelhança anatómica e, portanto, apresentando simetria significativa, levou a que fosse desenvolvida uma extensa atividade de investigação baseada no pressuposto de que a diminuição da simetria está diretamente relacionada com a presença de patologias. Este trabalho foca-se na análise da simetria (ou falta dela) dos dois hemisférios cerebrais e na utilização desta informação para a classificação das tomografias computadorizadas de doentes com AVC. O objetivo é contribuir para o processo de identificação automática de lesões cerebrais causadas por episódios de AVC. Para realizar esta tarefa, utilizou-se a Rede Siamesa, que é composta por duas redes neuronais paralelas que partilham os mesmos pesos. A rede recebe uma dupla imagem (a imagem original e a imagem espelhada) e um rótulo que identifica a existência ou não de AVC. Posteriormente, a rede extrai as caraterísticas mais relevantes e classifica as imagens tendo em conta a sua semelhança. Deste modo, esta rede pode ser utilizada para classificar novas imagens de tomografias computadorizadas, de acordo com o nível de simetria observado numa das duas classes existentes: existência de AVC ou ausência de AVC. A precisão do método proposto é de, aproximadamente, 72%, o que permite superar, significativamente, a arquitetura padrão da rede convolucional, que foi utilizada como base de comparação.

Palavras Chave

AVC, Tomografia Computadorizada, Deteção de Simetria, Classificação de Imagens, Rede Neuronal Convolucional, Rede Siamesa.

Contents

List of Figures

List of Tables

Acronyms

ASPECTS	Alberta Stroke Program Early Computer Tomography Score
AUC ROC	Area Under the ROC Curve
CNN	Convolutional Neural Network
CSF	Cerebro-Spinal Fluid
CT	Computed Tomography
CTA	Computed Tomography Angiography
CTP	Computed Tomography Perfusion
DICOM	Digital Imaging and Communications in Medicine
FLAIR	Fluid-Attenuated Inversion Recovery
GAP	Global Average Pooling
Grad-CAM	Gradient-Weighted Class Activation Mapping
HU	Hounsfield Units
HSM	Hospital de Santa Maria
MCA	Middle Cerebral Artery
MSP	Mid-Sagittal Plane
MRI	Magnetic Resonance Imaging
NCCT	Non-Contrast Computed Tomography
ReLU	Rectified Linear Unit
ResNet	Residual Network
ROC	Receiver Operating Characteristic
SiameseNet	Siamese Network
WL	Windows Level
WW	Windows Width

1

Introduction

Contents

1.1 Motivation

A stroke occurs due to damage to brain cells resulting either from the absence of oxygen and nutrients caused by a blockage of blood flow (an ischemic stroke) or from the flooding of blood from a ruptured artery (a hemorrhagic stroke) [1].

According to the World Stroke Organization, stroke is the second leading cause of death and disability in the world, with about 13 million cases annually [2]. In this work, only ischemic strokes were considered since they represent the majority of all strokes.

Increasingly, stroke detection techniques have been the subject of research, since the extent/severity of the lesion is directly related to the duration of the stroke, and therefore the earlier it is flagged and diagnosed, the earlier treatments can be administered and the greater the possibility of recovery (total or partial) of the affected areas.

The diagnosis is made based on the evaluation of the symptoms and images from Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) scans since this type of exam allows the precise identification of lesion location. It is widely known that the brain has two hemispheres with a high level of anatomical similarity. However, over the years, symmetry has been the subject of extensive research and many researchers believe that its decrease is directly related to the presence of pathologies [3].

Therefore, it is in this task that this work is focused, the analysis of symmetry (or lack thereof) of CT scan images, contributing to the process of identification of lesions and, consequently, improving the diagnosis made by neurologists, making it less complex and less susceptible to different opinions and interpretations.

1.2 Thesis Proposal

The purpose of this work is to detect the probable presence or absence of stroke in CT images, based on the level of symmetry between the two brain hemispheres. In this way, we developed a complementary diagnostic method that, although still rudimentary, will help doctors in the identification of strokes, making the process faster, which is fundamental in this type of disease.

To do this, we use the Siamese Network (SiameseNet), a particular neural network architecture developed to identify similarities between images, in this case, symmetric ones. This model consists of two equal neural networks that share the same weights and parameters. Each of these networks receives a double image (the original image and the mirrored one), extracts its features, and, classifies the images into one of the two existing classes according to their similarity.

It is hoped that this method can provide a good contribution to the identification of strokes and that it can be the starting point for other studies focusing on symmetry and how it is related to the existence of pathologies. The early perception that there is a considerable degree of asymmetry can prevent the

evolution of diseases or even avoid them.

1.3 Thesis Outline

This document is composed of six chapters. The purpose of this chapter is to present the work at hand, as well as its objective. The second chapter refers to the problem itself, i.e., the occurrence of strokes, as well as the entire process from patient admission to the hospital, to medical images acquisition. The third chapter presents the literature review, where the fundamental concepts behind convolutional neural networks are presented, as well as the architectures used in this work to identify symmetry. The fourth chapter presents the entire process involved in this project, including the construction of the dataset, through image preprocessing, and culminating with symmetry detection. The fifth chapter is intended to present and discuss the results obtained with different architectures for the proposed dataset. The sixth and final chapter presents the conclusions and proposes possible directions for future work that can fill the gaps found.

2

Ischemic Stroke

Contents

2.1 Characterization

Ischemic stroke occurs when a vessel supplying blood to the brain is obstructed, leading to damage or death of brain cells. It is usually caused by blood clots in a brain vessel or by narrowing of the blood vessels that irrigate the brain due to a process of atherosclerosis [1]. Figure 2.1 illustrates an ischemic stroke.

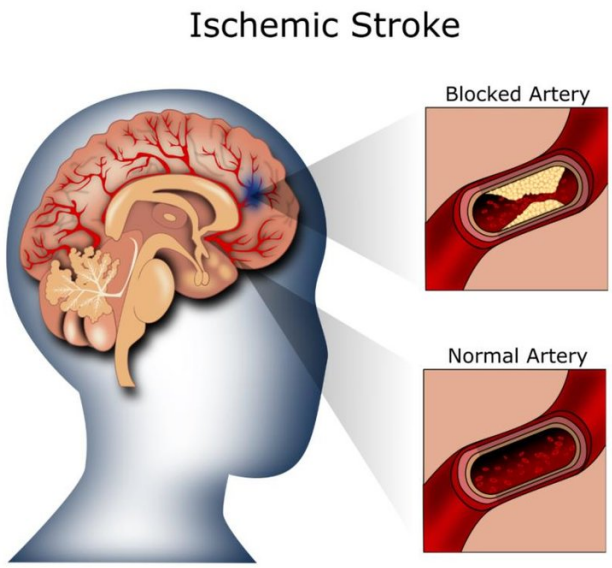


Figure 2.1: Ischemic stroke representation. Source: Image taken from [4].

The diagnosis is made based on the evaluation of the symptoms and images resulting from CT or MRI exams. In this particular case, only the results of the CT scans are considered.

The exams should be done immediately, in order to be able to identify and evaluate the existing lesions. The time elapsed since the onset of the stroke is crucial for the treatment and recovery of the patient [5]. The ischemic core consists of the region where the damage is irreversible, while the penumbra is the region around the ischemic region that can be recovered [1]. Recovery depends on the severity of the stroke, and the faster the diagnosis and treatment, the greater the chance of recovering the penumbra.

After the diagnosis is made, the appropriate treatment is administered, and then the CT scans are repeated to verify the existence or not of the changes that caused the stroke. The CT scans look for the hypodense regions of the brain, i.e, dark regions, as there is a loss of blood flow.

In this project, only the images of the exams were collected after the beginning of the stroke symptoms in patients at *Hospital de Santa Maria (HSM)*.

2.2 Brain symmetry

A human brain has two mostly symmetrical hemispheres, separated by the longitudinal fissure, which is a membrane filled with Cerebro-Spinal Fluid (CSF) [6]. This is usually known as the Mid-Sagittal Plane (MSP), because it is a virtual plane perpendicular to the brain, which divides it into left and right parts [6, 7].

Although we consider that the MSP may be a straight plane and on CT images may appear to be a straight line, it is actually a curved surface. However, sometimes it could be very difficult to see this surface on CT images, because, in pathological cases that involve the presence of tumors, it may happen that one side can compress the other side of the brain, making the fissure very curved to adjust a good plane [6].

It is widely acknowledged that hemispheres display both anatomical and functional asymmetries and this has been a topic of research over the years [8]. Previous work has shown that the asymmetry of hemispheres indicates the presence of brain injuries [3, 9], tumors [10, 11] or mental illness [12].

2.3 ASPECTS

The Alberta Stroke Program Early Computer Tomography Score (ASPECTS) is a quantitative score used to assess the severity of ischemic stroke on a Non-Contrast Computed Tomography (NCCT) scans. It is a quick and easy rating system that assesses the extent of Middle Cerebral Artery (MCA) involvement [13].

In order to make the evaluation, the MCA is divided into 10 regions, in order to facilitate the analysis of possible changes in each of the zones, scoring each affected area. Figure 2.2 illustrates the 10 anatomically defined regions, 4 for subcortical structures [caudate (C); lentiform (L); internal capsule (IC); insular ribbon (I)] and 6 for cortical structures in the MCA territory, labeled M1–M6 [13].

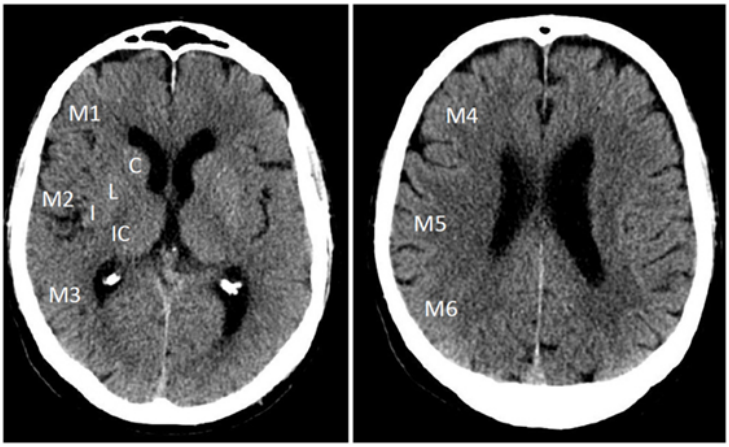


Figure 2.2: Representation of the 10 defined regions of the MCA on a NCCT for the calculation of ASPECTS. Source: Image taken from [13].

The scale ranges from 0 to 10, and the lower the ASPECTS value, the greater the severity of the patient's condition. For each area showing early ischemic changes on at least two consecutive slices, one point is subtracted from the overall score. If the score is 10 it means that the CT scan is normal [13].

2.4 Computed Tomography

After the patient is admitted to the emergency room at the *Hospital de Santa Maria*, he is referred to the imaging area, where the exams that identify and evaluate the lesions caused by the stroke will be performed.

During the first few hours after the onset of stroke, CT scans should be used, since they can detect the presence of ischemic or hemorrhagic lesions, pinpoint the location and extent of the lesion, and exclude situations that can be confused with strokes, known as stroke mimics [14].

A CT scan is a first-line diagnostic test that uses X-rays together with complex algorithms to obtain images of a portion of the human body, in this case the skull. The images obtained are produced based on the radiation emitted after passing through the body.

Afterwards, through a process that solves the so called inverse problem, it is possible to make a very detailed reconstruction of the image of the brain, which will be analyzed by the doctor. The inverse problem in CT scan is the process of reconstructing cranial images based on the two dimensional images that are acquired during the examination, i.e. with noise and/or few observations [15].

The devices used to perform a CT scan obtain a series of images of the brain to be analyzed. To perform the CT scan, the patient is placed in the supine position, i.e., 'face up' on the motorized table that will slide into the CT scanner. Subsequently, a beam of X-rays will cross an axial plane due to the continuous movement of the X-ray ampule around the patient, as illustrated in figure 2.3. Due to the helical technology, image production is achieved during the continuous movement of the table and the continuous movement of the detectors, installed in the gantry [16]. In this way, CT achieves a volumetric acquisition.

Thus, based on the images acquired in the axial plane, it is possible, using computational techniques, to reproduce images in the sagittal and coronal planes, allowing axial, sagittal and coronal sections to be obtained, as illustrated in figure 2.4.

There are several types of CT scans, each of which can detect different changes. The first exam to be performed to the patient is the Computed Tomography without contrast, since it allows to quickly identify if the stroke is ischemic or hemorrhagic [17] and inform what should be the following exams. In this case, only patients with ischemic stroke are considered and, therefore, the exams to be performed later are: Computed Tomography Perfusion (CTP) and Computed Tomography Angiography (CTA).

Computed Tomography Perfusion, which requires the administration of contrast intravenously, allows

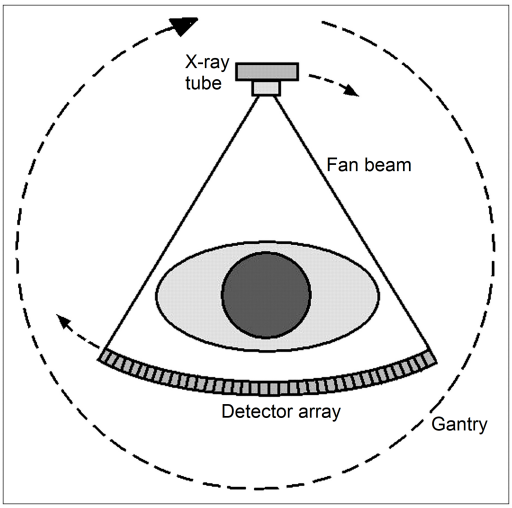
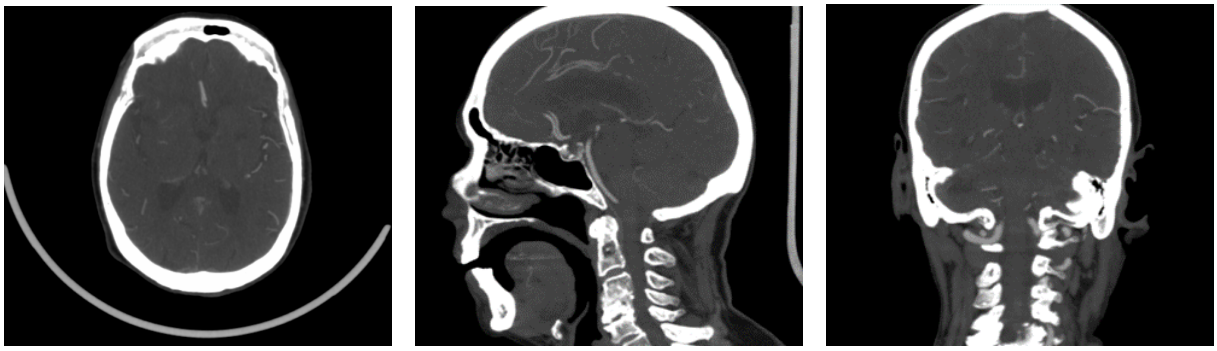


Figure 2.3: Representation of the computed tomography equipment. Source: Image taken from [16].



(a) Axial section **(b)** Sagittal section **(c)** Coronal section

Figure 2.4: Representation of the 3 sections obtained by reformatting the CT scan. Source: Images taken and adapted from HSM database.

the viability of brain tissue to be assessed and perfusion maps to be obtained [17].

After CTP, Computed Tomography Angiography is performed to map the anatomy of the aortic arch, carotid and cerebral arteries and the presence of stenosis or thrombus of the carotid or cerebral arteries [17].CTP provides detailed images of the blood vessels, allowing us to determine the exact location of the blocked blood vessel.

2.5 Data Acquisition

Convolutional neural networks use images as inputs, so it is very important to select images that contain the characteristics needed to assess the severity of an ischemic stroke. Thus, we decided to use only images resulting from NCCT scans, because it is easier to identify the symmetric and asymmetric regions.

The images are acquired in Digital Imaging and Communications in Medicine (DICOM) format, an international standard format, which aims to facilitate the storage, processing, and transmission of all medical images resulting from any type of exam. Since all images are stored in this type of format, all equipment can recognize and display the images in their original quality [18].

Each DICOM image is accompanied by a header, where fixed information such as date of birth, identification, or gender is described. In addition, other information can be included in this header according to the type of diagnostic modality that is used [18]. For example, in a diagnostic CT scan, information about the slice used in the acquisition or the characteristics of the X-ray beam used such as voltage, tube current, and exposure time are displayed.

A CT image consists of a set of points called voxels. The voxels represent the three-dimensional tissue volumes, while the pixels are the two-dimensional representation of the slice, as represented in figure 2.5. The resolution of the image is given by the total number of voxels along with the slice thickness. The 'color' of each voxel is conventionally displayed in a shade of grey, corresponding to the density of brain tissue at that location, and is called the attenuation value, which is given in Hounsfield Units (HU) [16, 19].

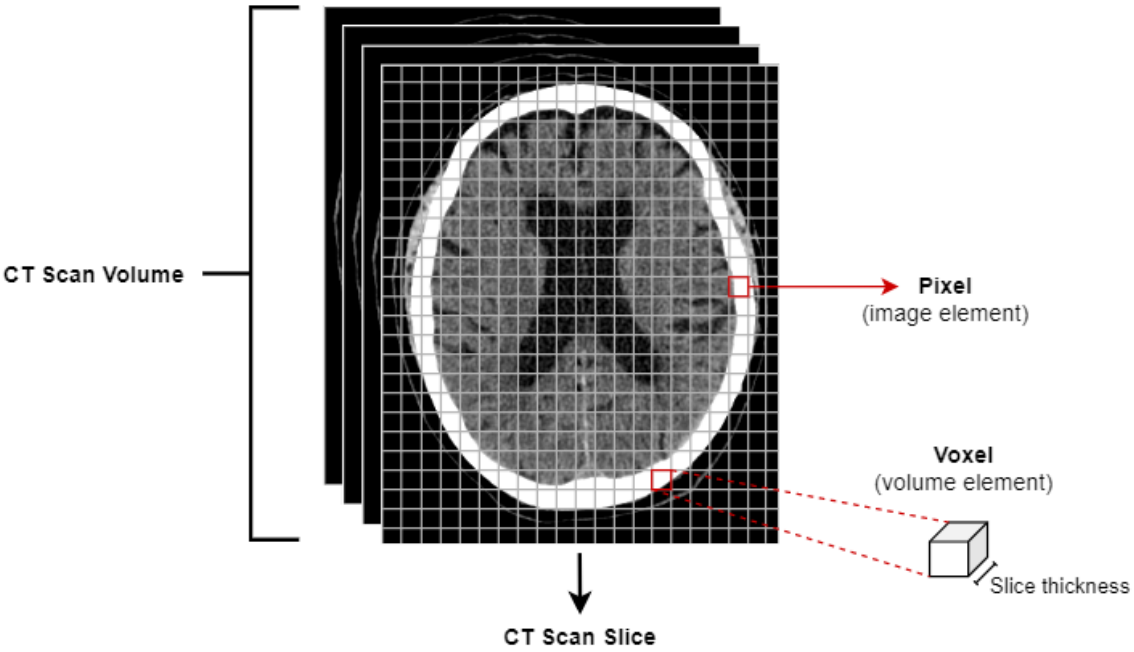


Figure 2.5: Pixels and voxels representation.

The Hounsfield unit also referred to as the CT unit, is a relative quantitative measurement of radio density used by radiologists in the interpretation of CT images. This unit was named after Sir Godfrey Hounsfield for his part in the invention of CT [19]. The HU is calculated as follows:

$$HU = \frac{(\mu_{voxel} - \mu_{water})}{\mu_{water}} \times K \tag{2.1}$$

In this equation, μ_{voxel} is the calculated voxel attenuation coefficient, μ_{water} is the attenuation coefficient of water and K is an integer constant, standardized as 1000 [16].

The most common range of CT unit is 2000HU, where the distilled water (under standard conditions) has an attenuation value of 0HU and air is defined as -1000HU. This scale starts in -1000HU (white) and finishes in +1000HU (black) [19,20]. Figure 2.6 show the HU scale and another values of measurements.

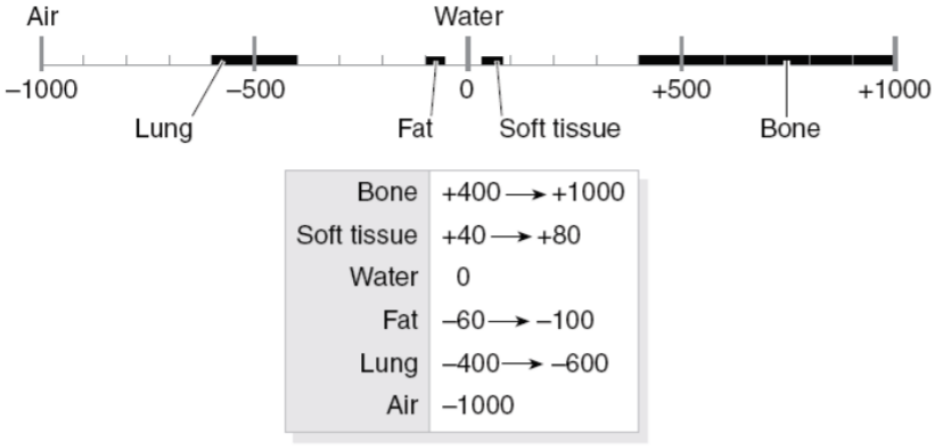


Figure 2.6: Hounsfield unit scale. Source: Image taken from [21].

As we can see in the figure above, we should choose the best window for brain images. So, to set this window, we have two parameters: Windows Level (WL), also known as Windows Center, and Windows Width (WW). The windows level is the midpoint of the range of the CT numbers and it is responsible for the image brightness, while windows width is the range of the CT numbers that an image contains and is responsible for the image contrast.

This process of manipulating CT values and, thereby enhancing the contrast for a particular tissue is known as windowing and it is a technique frequently used in the evaluation of CT scans [20]. The upper and lower grey levels in the window are given by:

$$UpperLevel = WL + (WW/2) \tag{2.2}$$

$$LowerLevel = WL - (WW/2) \tag{2.3}$$

This means that all values above the upper level will be white and all values below the lower level will be black [20].

3

Background

Contents

3.1 Convolutional Neural Networks

A perceptron is the simplest neural network currently used to classify a set of continuous-values inputs into one of two classes. A perceptron consists of a single neuron/node with adjustable weights. The neuron is the fundamental processor of a neural network and it consists of three basic elements [22]:

- a set of connecting links, with each linkage having in associated weight;
- a summation unit that sums the input signals after they have been multiplied by their respective weights;
- an activation function that adds nonlinearity to the neural network.

Figure 3.1 presents the three basic elements mentioned above.

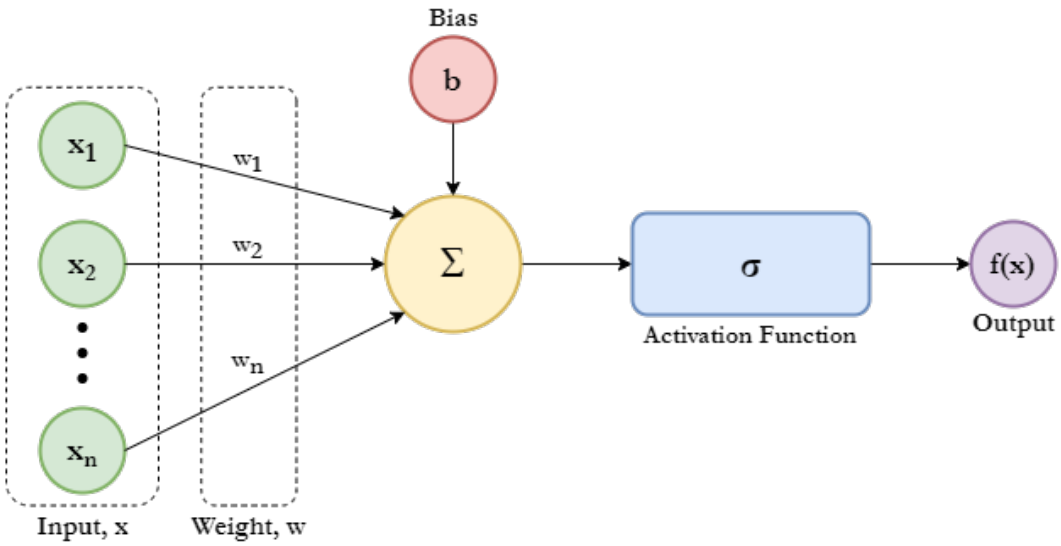


Figure 3.1: Composition of a perceptron.

A one-layer perceptron can only classify linearly separable data, so when it is necessary to classify data that is not linearly separable it is necessary to use more layers [22]. A multi-layer network contains one or more hidden layers, and consequently its neurons are called hidden neurons. A multi-layer network is fully connected if the nodes in the current layer are connected to all nodes in the next layer [22], as depicted in the figure 3.2.

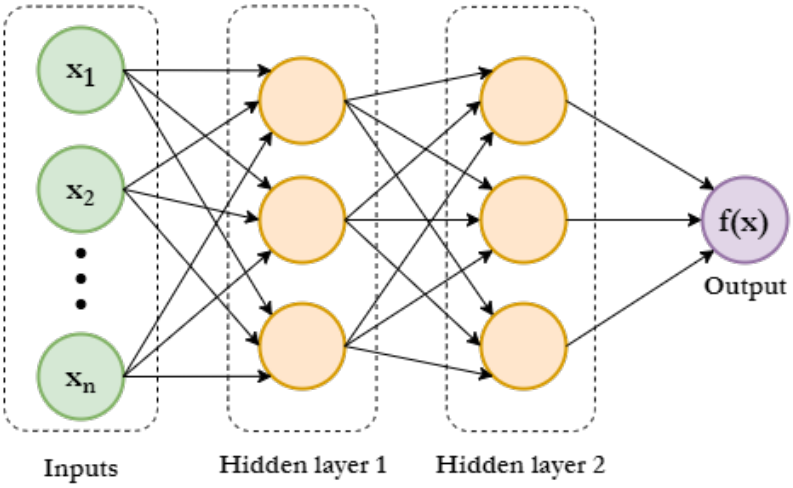


Figure 3.2: Fully connected.

Convolutional Neural Network (CNN) are multi-layer networks similar to traditional neural networks suited to treat inputs that are images. Thus, convolutional neural networks are used for image recognition and classification tasks.

A convolutional neural network is essentially composed by the following parts:

- Convolution
- Activation function
- Pooling
- Fully connected layer

3.1.1 Convolution

The convolution is the first layer of the network and it is responsible for extracting the features of the input image, through a kernel that moves over the input data. Mathematically, a convolution is a linear operation, which uses two functions to generate a third, named a feature map as illustrated in figure 3.3.

This process is performed several times to different regions of the image, and at each application the region is changed by a parameter known as stride. The purpose of this parameter is to define the size of the 'jump', when the kernel is applied. Thereby, when stride is equal to 1, it means that the transformation is applied to all pixels in the image [23].

One of the convolution disadvantages is related to the loss of information at the edges of the image. Hence, it is common to use one padding layer, in order to add pixels (usually with value 0) around the image before the convolution operation [23]. Moreover, this process avoids the decreasing of the size

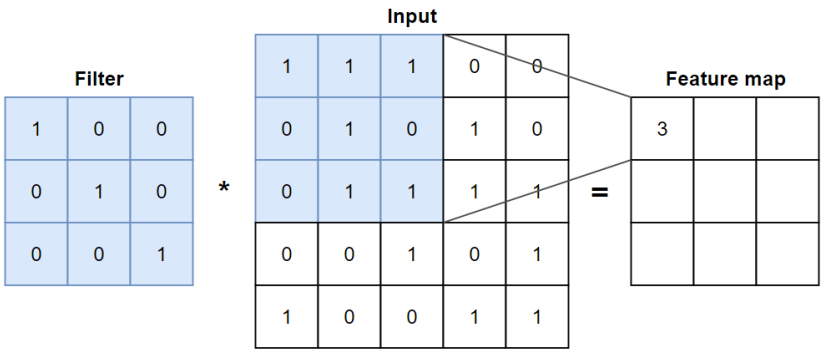


Figure 3.3: Convolution operation on a 5x5 input with a 3x3 kernel of binary values.

of the output network with increasing depth. Typically, this concept is used in the initial layers of the convolutional neural network [23].

3.1.2 Activation Functions

An activation function is just a function that transforms an input signal into an output signal, which will be used in the next layer as input signal.

The purpose of this process is to help the network learn complex patterns in the data. If we don't use an activation function in our network, it means that the output signal will simply be a linear function, i.e. a polynomial of degree one. Since, their complexity and performance are limited, these prevent neural networks from being able to learn more complicated tasks that require high dimensional and nonlinear functions and also complex architectures [24].

There are several activation functions and their choice depends on the type of data, model and task. However, it is common to use nonlinearity activation functions because adding nonlinearity to the neural network, allow us to be able to achieve nonlinear mappings from inputs to outputs [24].

The two main nonlinear activation functions used, are: Rectified Linear Unit (ReLU) and Softmax.

- ReLU is usually applied after a convolutional layer, because its main goal is to saturate or limit the output signal from the previous layer [23, 25, 26]. Mathematically it is defined as:

$$y = max(0, x) \tag{3.1}$$

  The function ReLU is used because it has a simple definition of the function itself and of its gradient. Moreover, there is no problem of gradient vanishing, which occurs in the case of other activation functions, because it has a constant gradient for positive inputs [23, 25, 26].

- Softmax is typically used in the last layer of the neural network in order to convert the results

coming from the previous layers into values between 0 and 1. These values can be treated as the probability that an object corresponds to a certain class [24]. Mathematically it is defined as:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{i=1}^{K} e^{z_k}}, \qquad for j = 1, ..., K \tag{3.2}$$

Softmax function is a combination of multiple sigmoid functions, that return values between 0 and 1. This function, unlike the sigmoid function, can be used for multiclass classification problems, where the output layer of the network will have the same number of neurons as the number of classes in the target [24].

3.1.3 Pooling

The main idea of the pooling layer is to decrease the samples in order to decrease the complexity for the following layers. Bearing in mind the image processing, pooling can be defined as the decreasing of image resolution. Furthermore, this process enables a decrease in the number of parameters to be trained by the neural network [23, 25–27].

There are different pooling approaches, where the most common are the max pooling and the average pooling. The main difference between these two methods is the mathematical technique applied. While the max pooling layer selects the maximum pixel value in the receptive field, the average pooling layer calculates the average of the pixels values in the receptive field, as illustrated in the figure 3.4.

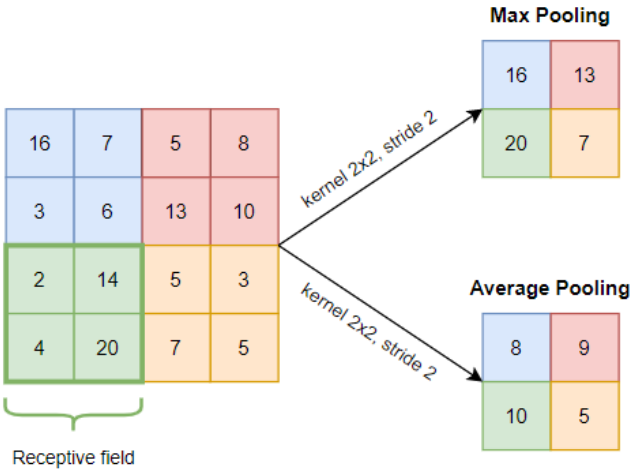


Figure 3.4: Pooling techniques: max pooling and average pooling.

In contrast to the previous two layers that are applied between the convolutional layers, the Global Average Pooling (GAP) layer is designed to replace fully connected layers in classical CNNs. The idea is to generate one feature map for each corresponding category of the classification task in the

last convolutional layer [28]. The GAP computes the average of each feature map until each spatial dimension is one. Figure 3.5 illustrates the configuration of these two processes.

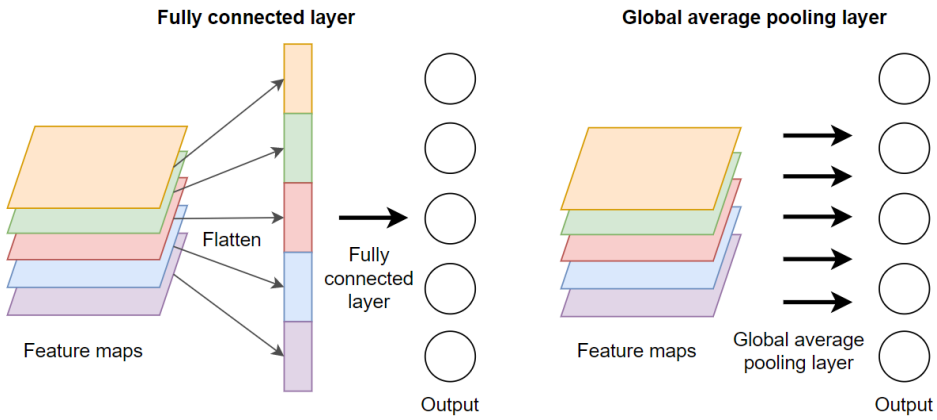


Figure 3.5: Comparison of the fully connected layer and global average pooling layer configurations.

3.1.4 Fully Connected Layer

The goal of the fully connected layer is to use the outputs from the previous layer, i.e., the pooling layer and, classify the image into one of the existing classes. However, before moving to the fully connected layer, the data that leaves the pooling layer needs to undergo a change, using the flatten layer.

The purpose of the flatten layer is to convert the matrix with the data from the pooling layer into a vector containing the probability of each feature belonging to a certain class.

Finally, it is usual to use the softmax function to convert probability values between 0 and 1. Figure 3.6 illustrates the typical constitution of a CNN.

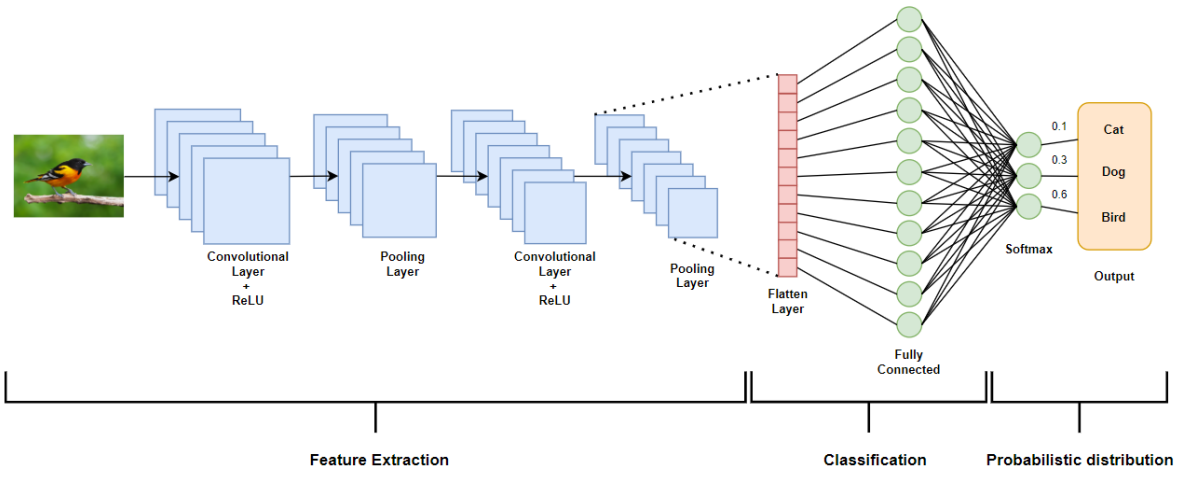


Figure 3.6: Convolutional Neural Network.

3.2 Deep Learning Architectures

3.2.1 Residual Neural Network

Convolutional neural networks exhibit a high performance on image classification tasks but can be difficult to train due to the complexity and the vanishing gradient problem - caused by the derivative/gradient of the activation function that decreases exponentially as it propagates down to the initial layers, which means that the weights and bias of the initial layers will not be updated properly - and, also to the difficulty in optimizing the training process as a result of the large number of parameters [29, 30].

Thereby, to fight these problems the Residual Network (ResNet) was proposed, which presents (residual) connections in parallel with the convolutional networks [30], as represented in figure 3.7.

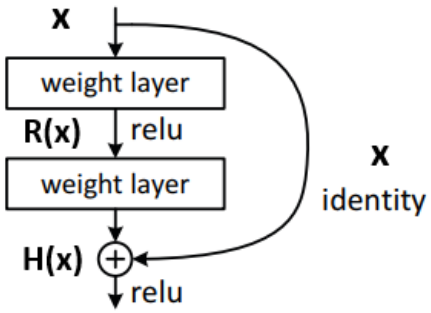


Figure 3.7: Example of a residual block. Source: Image taken and adapted from [30].

The name residual is based on the fact that the layers learn not the true result, $H(x)$, as in traditional networks, but the residual, $R(x)$. Consider a neural network block, where the input is denoted by x and is intended to learn the true output $H(x)$. Then, the difference (residual) between the two is given by:

$$R(x) = Output - Input = H(x) - x \Leftrightarrow H(x) = R(x) + x \tag{3.3}$$

The residual block is trying to learn the true output H(x), but since the connection is an identity matrix, then the layers are actually learning the residual R(x).

In contrast to convolutional layers, these connections are always active and therefore allow gradients to propagate through them, leading to a faster training process [29].

If the input and the output of the basic block (figure 3.7) are equal, then the connection is just an identity matrix. Otherwise, the average pooling and zero padding techniques are used to adjust the size, depending on whether it needs to be decreased or increased, respectively. It is common to add a ReLU or a dropout layer after the add operator since these parameterized layers do not interfere with the advantages of ResNet [29].

Inspired by the VGG network philosophy, the researchers created the ResNet, which consists of

convolutional layers, usually with 3x3 filters. The downsampling process is performed directly through the convolutional layers with stride 2. The network ends with a global average pooling layer and a fully connected layer with softmax [30].

3.2.2 Siamese Network

Siamese Networks were first introduced in the 1990s by Bromley and LeCun for signature verification [31]. This network is composed of two twin neural networks, which share the same weights and parameters, but are joined by a loss function on top [32]. Unlike traditional machine learning classifiers that receive one input and predict its class, Siamese networks receive two distinct inputs and predict whether or not they belong to the same class. Therefore, instead of learning a classifier, this network learns a similarity function between inputs and is particularly useful in few-shot learning, where the examples of each class are scarce [31].

It is widely known that cross-entropy loss is very popular in deep neural networks, especially in classification tasks. However, since we are learning the similarity between images, the most suitable loss function is the contrastive loss, which is given by [32]:

$$L(x_1, x_2, y) = \alpha(1-y)D^2 + \beta y\{max(0, margin - D)\}^2 \tag{3.4}$$

where y is our label, and it will be 1 if the images are of the same class, or 0 if the images are of a different class, and α and β are two constants usually assigned as 0.5. D is the Euclidean distance, which consists of the similarity function, and measures the distance between the outputs of the twin networks. Finally, the max function takes the largest value of 0 and the margin, which is defined by us, minus the distance (D). The Euclidean distance is given by:

$$D(x_1, x_2) = \|F(x_1) - F(x_2)\|_2 \tag{3.5}$$

where $F(x_1)$ and $F(x_2)$ correspond to the output vector of each subnetwork.

3.3 Related Work

Stroke segmentation and classification is a topic with many years of research, and therefore many machine learning models have been proposed.

Chin *et al.* [5] proposed a method to assist neurologists in the diagnosis, since the sooner the patient is treated, the better the chances of recovery. This work developed a CNN for the early detection of ischemic stroke, which was composed by five layers: two convolutional layers, one pooling layer, one fully connected layer and a softmax layer. The dataset is composed by CT images of the brain, with

size 512x512, enriched with MRI data. However, since only the stroke area is considered, the authors captured a size 32x32 patch image, which is used as model input. Skull bone and other structures, such as, CSF that might mislead the model were removed, since these structures exhibited pixel intensities as the ischemic area. The authors report an accuracy of 93% in the classification task, where 128 patch images were used for training and 128 patch images were used for test, for a total of 256. Since this approach uses also MRI data, the results are not directly comparable with ours.

Recently, Herzog and Magoulas [33] proposed a method based on brain asymmetry to identify early dementia and its diverse stages, such as amnestic early mild cognitive impairment and Alzheimer's Disease. They analyzed the structural and functional cerebral changes in both hemispheres using supervised machine learning algorithms (Naive Bayes, Linear Discriminant, Support Vector Machine, K-Nearest Neighbor) and a convolutional neural network, AlexNet. The dataset was composed of brain asymmetries features, extracted from 750 MRI scans from the Alzheimer's Disease Neuroimaging Initiative database, which was normalized and resized to 256x256x3. For running the supervised machine learning algorithms, the 10-fold cross-validation was used, while for running the CNN, the images were resized to 227x227x3 (a requirement of the AlexNet architecture) and divided into training, testing and validation sets. The proposed pipeline achieved an accuracy that ranged from 75% to 93% for the mentioned diseases. Furthermore, this method offers a promising low-cost alternative for the classification of dementia and could potentially be useful in other brain degenerative disorders that are accompanied by visible changes in the brain symmetry. Again, since it uses MRI data it cannot be directly compared with ours.

To perform brain symmetry studies on large neuroimaging archives, reliable and automatic detection of the MSP is required to extract the brain hemispheres first. However, traditional planar estimation techniques fail when the MSP presents a curvature caused by existing pathology or a natural phenomenon known as brain torque. As a result, midline estimates can be inaccurate. To address this issue, Gibicar *et al.* [34] suggested an unsupervised midline estimation method that consisted of three main stages: head angle correction, control point estimation and midline generation. The technique was applied on a slice-to-slice basis for more accurate results and also provides accurate delineation of the midline in the septum pellucidum (exactly in the middle of the brain), which is a source of failure for traditional approaches. A midline validation dataset was created by sampling 75 Fluid-Attenuated Inversion Recovery (FLAIR) MRI volumes (around 3000 images) from three different databases. To validate the performance of the proposed midline estimation algorithm, the work was evaluated over a set of ground truth images and compared with to two other methods for midline estimation [35, 36]. Performance is quantified with three different validation metrics: mean Hausdorff distance, mean absolute distance, and mean volume-difference, which all compare the automated midline to the ground truth delineations.

Inspired by the characteristics of the Siamese Network, Barman *et al.* [37] proposed a Siamese

Neural Network (DeepSymNet) for the detection of ischemic stroke from CTA volumes. This method enabled them to detect the changes in symmetry of vascular and brain tissue texture of the two brain hemispheres in parallel. The model was tested on a clinical dataset of 217 patients with two different approaches: original CTA and brain tissue only. With the first alternative, the authors tested the network's ability to recognize strokes with the original 3D images, which contain asymmetries in both vascular structures and brain tissue. Then, removing the vasculature, they evaluated the network's ability to recognize strokes by analyzing only the brain tissue, i.e., detecting the darker textures in the brain tissue. In these two approaches, the data was resized to 29x73x20 using bilinear interpolation. The proposed model achieved a Area Under the ROC Curve (AUC ROC) of 0.89 for CTA volumes and 0.91 using only the brain tissue.

Tummala [38] proposed a deep learning framework that uses Siamese neural networks for computer aided diagnosis of autism spectrum disorder (ASD) using T1-weighted MRI. The dataset is composed of 102 control and 112 ASD patients. Preprocessing of the images involved reorienting them to a standard space, followed by applying the cropping technique to remove the neck regions, where FSL interfaces were used. After that, affine registration was applied, and finally, the images were reduced and resized to 224x224x3 to match the ResNet-50 input size. It was implemented a Siamese network with pre-trained ResNet-50 model to train and test 1070 positive and negative images pairs. Afterwards, the L1-norm was computed between the embeddings of the two inputs which is further used to backpropagate the error computed using the contrastive loss function. The model was trained using 5-fold stratified cross-validation, achieving an accuracy, recall, precision and f1-score of 0.99 within 50 epochs in the validation set.

Our approach differs from these and other related works in that we use only CT data obtained at the time of hospital admission. This makes the problem harder, since the changes in the brain images are more subtle, but also potentially more relevant in a clinical setting.

4

Methodology

Contents

4.1 Dataset

As we mentioned in section 2.5, the data used in this work was collected from ischemic stroke patients at *Hospital de Santa Maria*. The dataset is composed of non-contrast computed tomography images in DICOM format, which is the most commonly used format for storing medical images. The data was properly anonymized to preserve participant privacy and fulfill all ethical requirements.

Since we decided to use only the most relevant slices of each NCCT, i.e. those in which the ischemic stroke can be detected, we converted the 3D data to 2D slices, each with the size of 512x512 pixels. The final dataset used in the tests consists of slices chosen from different brain regions of different patients, in order to make the model more robust.

The dataset is balanced and includes 340 images of each class, where one class corresponds to the absence of stroke (and therefore symmetry between both hemispheres) and the other class corresponds to the visible effects of a stroke event (and lack of symmetry).

Since all CT scan images correspond to patients with ischemic stroke, the selection of slices for each class was made based on the ASPECTS score assigned to each of the patients at HSM. Taking into account that lower scores correspond to more severe strokes and, consequently, with more visible lesions, the respective slices were assigned to the ”presence of stroke” class. On the other hand, slices corresponding to patients with an ASPECTS score close to 10, and who did not present readily identifiable lesions were assigned to the ”absence of stroke” class. The images from the class where stroke effects were visible have been annotated by Dr. Catarina Fonseca, an expert neurologist at *Hospital de Santa Maria*.

Given that the dataset is limited in size and in order to avoid model overfit, we applied data augmentation techniques, namely Gaussian blur, to the training set. The idea of using this type of data augmentation was due to the fact that we want to ”force” the model to learn to identify the images that contain asymmetries. In addition, we decided to use stratified 5-fold cross-validation.

Besides making sure that all the folds contained the same number of images for each class, we also made sure that all the slices corresponding to the same patient were only in one of the folds, avoiding overfitting and data leakage.

4.1.1 Cross Validation

Cross-validation is a technique used to evaluate the performance of the model by dividing the data into two segments: one used to train a model and the other used to validate the model [39].

In K-fold cross-validation, the whole dataset is equally (or nearly equally) divided into K-folds, where K-1 folds are used for training and the remaining one is used for testing [39]. This process is repeated K times until all folds have been used for testing. Figure 4.1 illustrates the K-fold cross-validation, for K=5.

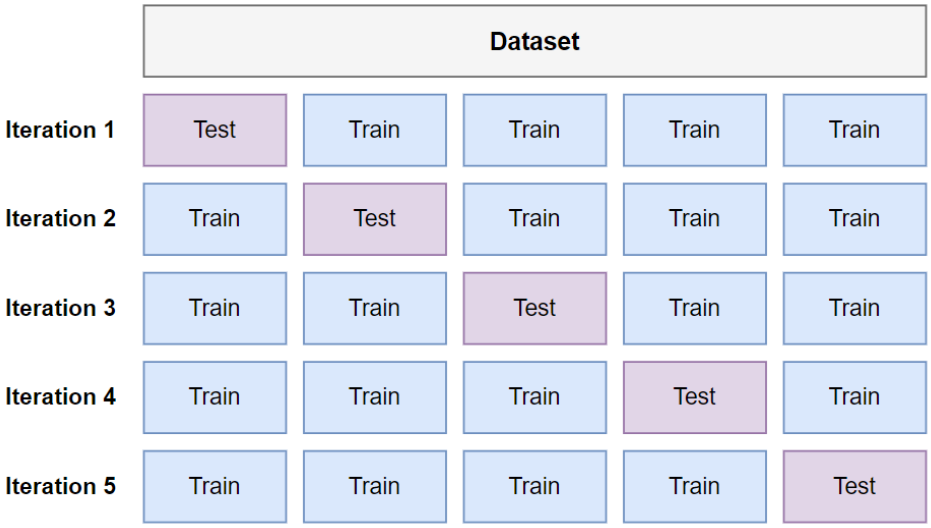


Figure 4.1: 5-fold cross-validation technique.

Stratified K-fold cross-validation is an extension of K-fold cross-validation where the dataset is divided into K folds ensuring that samples from each class are selected in the same proportion [40]. This technique guarantees that the data in both training and testing is balanced.

In this case, since K=5 was chosen and each class consists of 340 samples, this means that each fold contains 68 CT scan images from each class, for a total of 136 samples in each fold.

In this paper, we have two distinct approaches when dividing the data. The first approach is to split the data randomly, while the second approach is to split the samples by group, in this case by subject, i.e. the data is split in a way that ensures that all samples from the same patient are in the same fold.

The performance of the cross-validation is obtained by calculating the mean and standard deviation of the model performance on the test set in each iteration.

4.2 Image Preprocessing

Image preprocessing is a method to transform raw image data into clean image data, as most of the raw image data contain noise, redundant or false values, etc [41]. This process allows us to select the most relevant regions of the image and improve them, to later be used as model inputs.

In Machine Learning, this step is extremely important and one of the most time-consuming [41]. Therefore, this is the first step of this project, in which we submit the images to a set of preprocessing techniques, as illustrated in figure 4.2. The next subsections will present the most relevant ones. All the applied techniques were developed with Python 3.7 using the PyTorch framework, which is one of the preferred platforms for deep learning research.

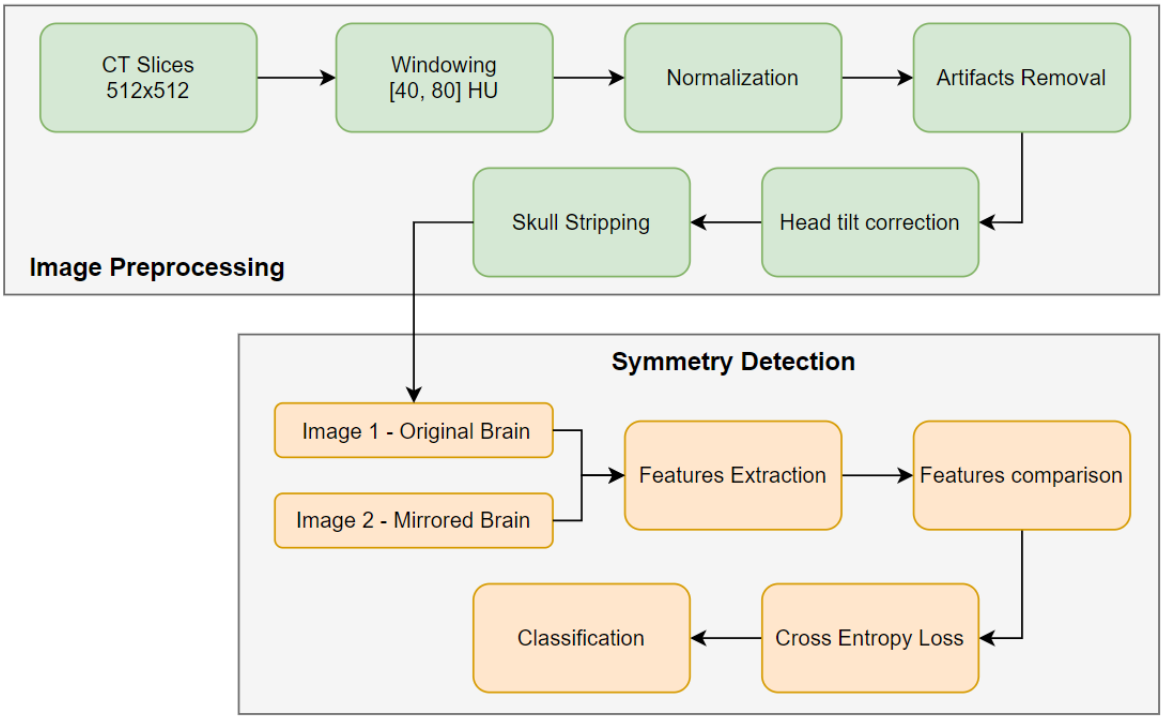


Figure 4.2: Project tasks.

4.2.1 Windowing

In our case we want to see the details of the brain, so we need to choose a window that matches well the nature of brain tissue, otherwise, some little details will be not visible [20]. Therefore, we decided to choose the range between 0HU and 80HU, i.e., WL: 40 and WW: 80, which is typically the window selected for strokes.

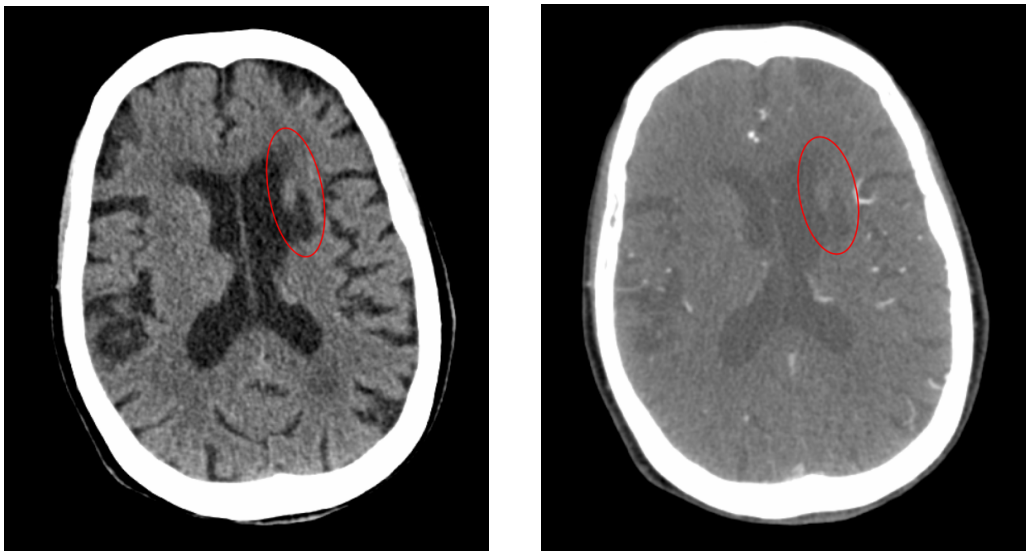
Figure 4.3 represents two images from a NCCT (WL: 40, WW: 80) and CTA (WL: 60, WW: 360), respectively, where a lesion caused by ischemic stroke can be seen near the left ventricle. The lesion is marked by a red circle.

As we can see in the figure, the correct window allows us to see more details and, consequently, it is easier to find the asymmetry between the two hemispheres.

4.2.2 Head Tilt Correction

A common phenomenon in medical imaging devices is that they produce distorted brain images, which can mislead visual inspection and lead to false clinical interpretation. The main reasons for this are the mobility of the patients, the inexperience of the technicians, and the imprecision of calibration systems [3].

In order to correct the head tilt, it was necessary to find the rotation angle, to align the MSP with the



(a) Non-Contrast Computed Tomography **(b)** Computed Tomography Angiography

Figure 4.3: Ischemic stroke lesions. Source: Images taken and adapted from HSM database.

y-axis [6, 33]. For that, the external contours of the brain were found. After that, the ellipse that best fits these contours was found. Finally, the angle between the ellipse and the vertical axis was calculated.

Figure 4.5 illustrates the same brain slice with and without head tilt correction.

4.2.3 Skull Stripping

Skull stripping is a fairly common process when using CT imaging because we can focus only on the brain tissue, which is where the lesions are, and we can perform a better segmentation of the different brain areas [5].

To assess the parts of the input that were more strongly influencing the decision, we used Gradient-Weighted Class Activation Mapping (Grad-CAM) to identify the image regions that weight more on the output result. Grad-CAM produces a localization map highlighting the important regions in the image to predict the target [42]. Figure 4.4 illustrates some examples after applying the Grad-CAM to the test set using the ResNet-50 architecture.

In fact, the network attention was mainly focused on the eye region or the bones around the brain. In order to get a better system performance and following expert advice, we decided to remove the bone leaving only the brain tissue.

The process of skull stripping consisted of doing a subtraction of the original image with an image where only the bone was visible. Using a windowing process, a mask was created with the bone and this image was subtracted from the original one, creating an image with only the brain tissue. Figure 4.5 illustrates a slice before and after skull stripping.

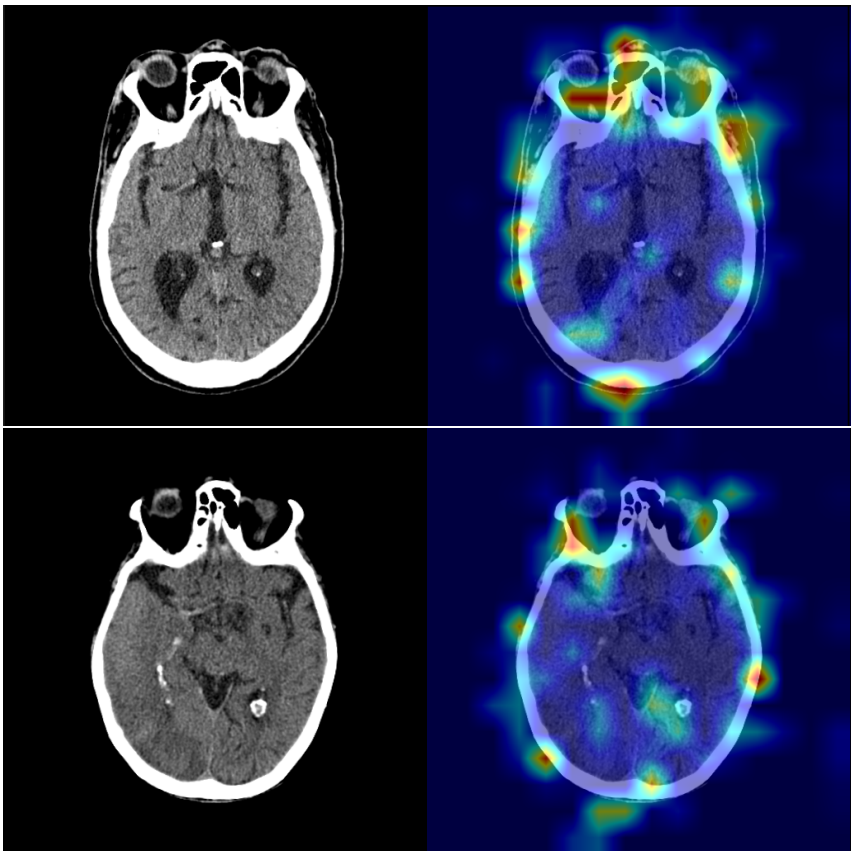
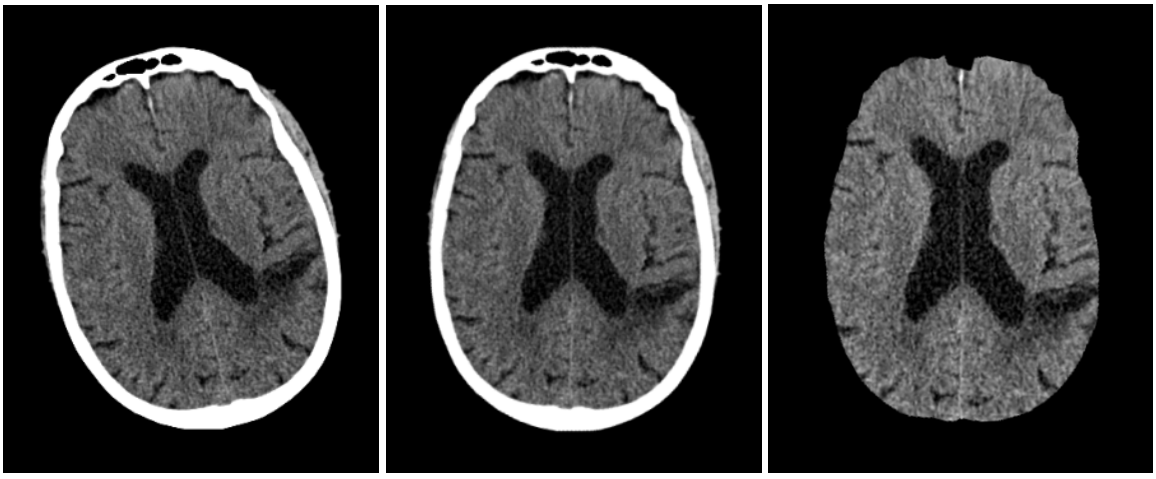


Figure 4.4: Attention maps obtained using Grad-CAM on the test set. Source: Images taken and adapted from HSM database.



(a) Original slice **(b)** Head tilt correction **(c)** Skull stripping

Figure 4.5: Main steps in CT images processing. Source: Images taken and adapted from HSM database.

4.3 Symmetry Detection

Originally proposed for signature verification, similarity detection has been used widely for various computer vision applications [32, 37, 43–45]. Similarity detection can be used to perform brain symmetry detection by using as inputs the original image and a mirrored image. Figure 4.6 shows the schematic of our Siamese Network.

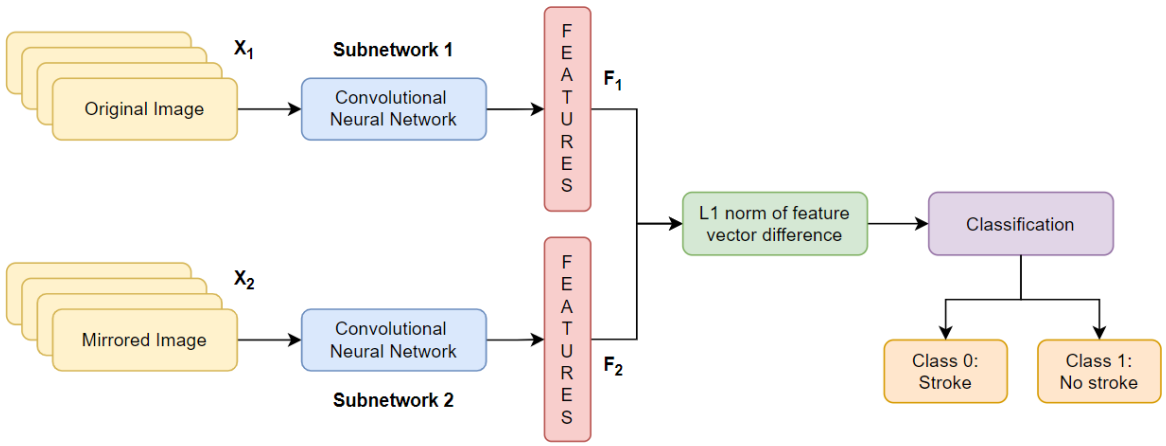


Figure 4.6: Siamese Network schematic.

The first step is the feature extraction, where we used two CNNs (subnetwork 1 and subnetwork 2) to extract the features of each image. They are trained with a label that reflects whether the two input images correspond to a visible stroke event (and are therefore approximately symmetric) or not. The output of each subnetwork was subsequently flattened (F_1 and F_2).

The second step is the comparison between the two feature vectors. The difference of each subnetwork output is computed, and the L1 norm is used to generate a new feature vector given by:

$$D = \|F_1 - F_2\|_1 \tag{4.1}$$

Finally, the network output is the probability of the two input images being symmetric or not, reflecting the probability of stroke. The smaller the value of D, the more likely the images are to be symmetric and, therefore, to not exhibit evidence of stroke.

4.3.1 Siamese Network

Since the Siamese network is made up of two equal networks, each of them has a set of images as input. One of the networks, in figure 4.6, subnetwork 1, takes as input an (original) set of slices with the size 512x512, while subnetwork 2 takes as input the same set of mirrored slices. The purpose of this is to make it possible to compare the two hemispheres (right and left) simultaneously.

Table 4.1 shows in detail the characteristics of the different layers. The feature extraction process is composed of 2D convolutional layers, 2D batch normalization layers and max pooling layers and a flatten layer, generating a feature vector with a size of 8192. The entire network uses ReLU activations, except for the prediction layer that uses softmax activation.

After feature extraction and subtraction, two fully connected layers were used, the first generating a vector of size 1024 and the second generating a vector of size 2, one output unit for each class.

Usually, when working with SiameseNet the preferred loss function is the contrastive loss since it gives a similarity score by calculating the Euclidean distance between the feature vectors [32]. However, once this is a classification task, we choose to use the cross-entropy loss and, as we have only two classes, is given by:

$$L(y_i, \hat{y_i}) = -(y_i \log \hat{y_i} + (1 - y_i) \log(1 - \hat{y_i})) \tag{4.2}$$

where y_i and $\hat{y_i}$ are the actual class and the predicted class, respectively.

The model was trained and evaluated using 5-fold cross-validation, for 200 epochs with a batch size of 32, using the ADAM optimizer with an initial learning rate $\gamma = 0.00001$. Since high weight values increase the chances of overfitting, L2 regularization with a weight factor $\lambda = 0.0005$ was used, applying a penalty for high weight values. The model required 10M trainable parameters.

Table 4.1: Siamese Network architecture.

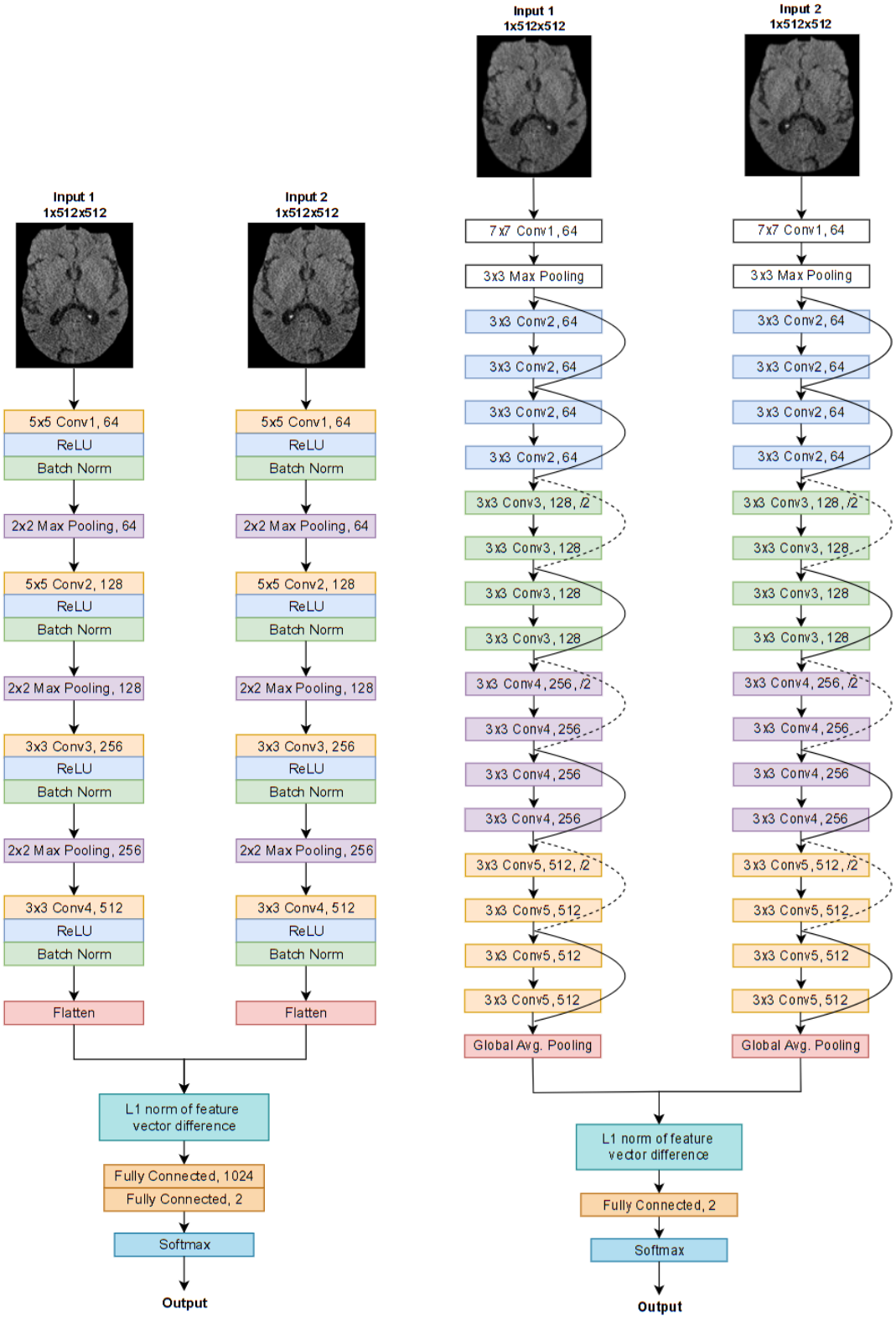
Operation Layer	Number of Filters	Filter Size	Stride Value	Padding Value	Output Shape
Input Layer					$1 \times 512 \times 512$
2D Convolution	64	5×5	2	2×2	$64 \times 256 \times 256$
ReLU					$64 \times 256 \times 256$
Batch Normalization 2D					$64 \times 256 \times 256$
Max Pooling	64	2×2	2	0	$64 \times 128 \times 128$
2D Convolution	128	5×5	2	2×2	$128 \times 64 \times 64$
ReLU					$128 \times 64 \times 64$
Batch Normalization 2D					$128 \times 64 \times 64$
Max Pooling	128	2×2	2	0	$128 \times 32 \times 32$
2D Convolution	256	3×3	2	1×1	$256 \times 16 \times 16$
ReLU					$256 \times 16 \times 16$
Batch Normalization 2D					$256 \times 16 \times 16$
Max Pooling	256	2×2	2	0	$256 \times 8 \times 8$
2D Convolution	512	3×3	2	1×1	$512 \times 4 \times 4$
ReLU					$512 \times 4 \times 4$
Batch Normalization 2D					$512 \times 4 \times 4$
Flatten					8192
Fully Connected					1024
Fully Connected					2
Softmax					

4.3.2 SimResNet-18

Initially motivated by the idea of combining ResNet and the Siamese Network architecture [45], we also tested an alternative approach where the ResNet is used only for the feature extraction step. As we have chosen the ResNet, the main difference between the previous architecture and this one is the residual connections presented in the ResNet model.

The Siamese Network consists of two ResNet, where one of them accepts the original brain image and the other accepts the mirrored one and extracts its features. Since ResNet is composed of a GAP layer after the last convolution layer, the resulting output is a feature vector with a size of 512 which is used to calculate the absolute value.

The remaining architecture is similar to that described for the SiameseNet. The feature vectors are subtracted, the absolute value of the difference is computed, and the images are classified into one of the two existing classes using a fully connected layer. In this architecture, we tested different ResNet models, and after analyzing the number of parameters, the training time, and the results obtained, we opted for the 18-layer model. The hyperparameters used were the same as the ones for the Siamese Network. Figure 4.7 illustrates the two architectures developed, SiameseNet and SimResNet-18 models.



(a) SiameseNet architecture.

(b) SimResNet-18 architecture.

Figure 4.7: Comparison between the two developed architectures.

5

Results

Contents

5.1 Evaluation Metrics

Evaluation metrics are used to measure the quality of a machine learning model. There are many types of metrics to evaluate the performance of a model. To evaluate our method, we decided to use the accuracy, precision, recall, F1-score, and AUC ROC. All of these evaluation metrics are based on the confusion matrix that displays the number of cases for each combination of the current classification and the predicted classification [46, 47].

The confusion matrix is a cross-table with size N x N, where N is the number of classes. The elements that make up the diagonal represent the number of correct samples (the predicted label is equal to the true label), while the elements represented outside the diagonal are those that are mislabeled by the classifier. Figure 5.1 shows the configuration of the confusion matrix.

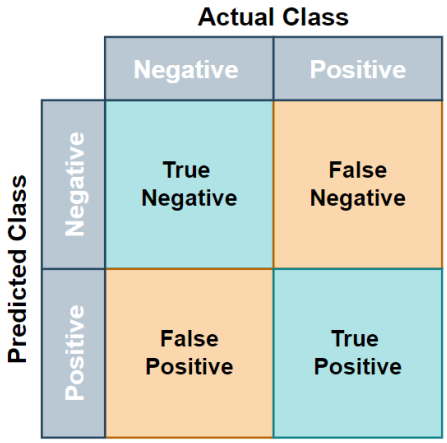


Figure 5.1: Confusion Matrix.

- **True Positive:** elements that are labelled as positive by the model and are truly positive.
- **True Negative:** elements that are labelled as negative by the model and are actually negative.
- **False Positive:** elements that are labelled as positive by the model, but that are actually negative.
- **False Negative:** elements that are labelled as negative by the model, but that are actually positive.

The accuracy is the ratio of the number of correct predictions (true positives and true negatives) to the total number of predictions/input samples, which is given by [46]:

$$\textbf{Accuracy} = \frac{\text{True Positives + True Negatives}}{\text{Total of predictions}} \tag{5.1}$$

The accuracy value tells us how well the model is predicting correctly across the entire data set, with each unit having equal weight and contributing equally to the accuracy value [46].

The precision corresponds to the portion of correct predictions that are truly positives/correct. Basically, this performance metric tells us how much we can trust the model when it predicts a sample as positive [46], and is given by:

$$\textbf{Precision} = \frac{\text{True Positives}}{\text{True Positives + False Positives}} \tag{5.2}$$

Recall, also called sensitivity, measures the portion of positive patterns that are correctly classified. The best value is 1 and worst is 0 [46], and is given by:

$$\textbf{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}} \tag{5.3}$$

The F1-score can be interpreted as a harmonic mean of the precision and recall, where an F1-score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1-score are equal [46]. The formula for the F1-score is given by:

$$\textbf{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision + Recall}} \tag{5.4}$$

The Receiver Operating Characteristic (ROC) curve shows how well the proposed model can distinguish between the two classes. For the construction of the ROC curve, there are two essential parameters:

$$\textbf{True Positive Rate} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}} \tag{5.5}$$

$$\textbf{False Positive Rate} = \frac{\text{False Positives}}{\text{False Positives + True Negatives}} \tag{5.6}$$

In this way, the ROC curve represents the relationship between the true positive rate and the false positive rate at different thresholds applied to the loss function of the model, as is shown in figure 5.2. The diagonal of the ROC graph can be defined as random guessing, and the classification models that fall below the diagonal are considered to be worse than random guessing [47]. On the other hand, the closer to 1 is the true positive rate and the closer to 0 is the false positive rate, the better the performance of the model.

The AUC ROC summarizes the data from the whole ROC curve into a single value, and characterizes the performance of the classification model [47, 48]. The AUC ROC has a range between 0 and 1, the greater the value, the better is the performance of the model. The AUC ROC is given by [48]:

$$\textbf{AUC ROC} = \int_0^1 ROC(u)du \tag{5.7}$$

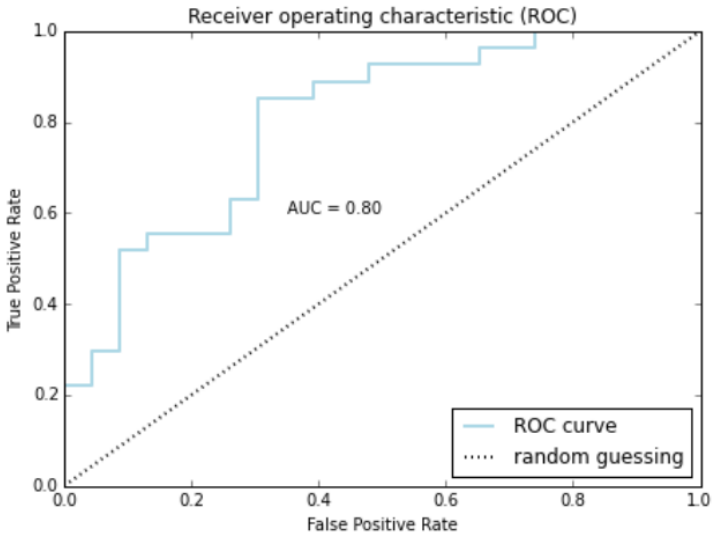


Figure 5.2: Example of a ROC curve. Source: Image taken from [47].

5.2 Results and Discussion

To assess the performance of the Siamese Network architecture on this dataset, we performed an empirical comparison with the alternative architecture, the SimResNet-18, and, a baseline architecture, the ResNet-50. All models were trained using a PowerEdge C41402 server with an NVIDIA 32GB Tesla V100S.

5.2.1 ResNet-50

As explained in subsection 3.2.1, ResNet is a commonly used deep convolutional neural network architecture that normally exhibits high performance in image classification tasks. While other CNNs have difficulty optimizing the network weights as a result of a large number of parameters and vanishing gradients, ResNet overcomes these problems by using residual connections in parallel with the convolutional layers, as described in section 3.2.1.

We tested ResNet with 18 layers, 34 layers, and 50 layers (three common variations of ResNet) and found that the best performance was obtained using ResNet-50. In addition, the choice of ResNet-50 over the other two models tested is due to the fact that it has more layers and therefore can learn more features, which can make a difference considering the complexity of the images. With this model, we required 174 minutes of training time using a network with 23M parameters

5.2.2 SimResNet-18

SimResNet-18, which is a combination of a Siamese Network and a ResNet-18, explained in detail in subsection 4.3.2, required 154 minutes to train and 11M parameters, approximately the same as SiameseNet architecture.

5.2.3 Experimental Results

Table 5.1 summarizes the classification performance of the two developed architectures, SiameseNet and SimResNet-18, and the baseline architecture, ResNet-50.

Table 5.1: Model classification performance.

Model	Accuracy	Precision	Recall	F1-Score	AUC ROC
ResNet-50	0.62 ± 0.086	0.71 ± 0.140	0.47 ± 0.268	0.57 ± 0.166	0.62 ± 0.086
SimResNet-18	0.64 ± 0.040	0.73 ± 0.124	0.48 ± 0.123	0.58 ± 0.070	0.64 ± 0.040
SiameseNet	$\mathbf{0.72 \pm 0.127}$	$\mathbf{0.75 \pm 0.165}$	$\mathbf{0.69 \pm 0.252}$	$\mathbf{0.72 \pm 0.144}$	$\mathbf{0.72 \pm 0.127}$

The results presented in the table show that, although the accuracy achieved is not very high (72%), the Siamese network outperforms the other two models. Moreover, it is the best result known to us for this specific problem, ischemic stroke classification from CT scans. The runtime for training the SiameseNet in the proposed dataset was 144 minutes.

The results obtained with ResNet-50 are due to the fact that CT images, although labeled, have several features that make model learning difficult. In fact, the model looks for changes in the brain hemispheres, but most of the time it cannot distinguish them from anatomical asymmetries, which means that the extracted features are not the most relevant for this specific problem.

On the other hand, when adding ResNet to the Siamese network (SimResNet-18), we see an improvement in performance, although not significant. Finally, when using the Siamese network, we see that there was a considerable improvement in model performance. However, these low accuracy results are mainly due to the fact that the dataset is small, considering the complexity of the problem.

The high value of the standard deviation of the performance measures is essentially due to two points. First, since the dataset is small, with few labeled images, it cannot be guaranteed that all patients have the same number of samples. In fact there are patients with about 45 slices and others with about 3 slices. Also, the guarantee that all slices from the same patient were in only one of the two sets (training or testing), led the model to learn better the cases where there is a high number of sequential slices, such as the case of 45 slices from the same patient. In this way, we can say that the model contains a sampling bias that affects the performance of the model. This is one of the most common types of dataset bias and occurs when randomization is not properly achieved during data collection, resulting in

poor generalization of the learning algorithm.

On the other hand, if it is not guaranteed that all samples from the same patient are in only one of the two sets, then there is data leakage, leading to the model not learning the features relevant to the problem but learning features to relate/sequence the slices (of the same patient).

In this case, splitting the dataset randomly, and training the same three models, we obtained the results shown in the table 5.2.

Table 5.2: Model classification performance with data leakage.

Model	Accuracy	Precision	Recall	F1-Score	AUC ROC
ResNet-50	0.89 ± 0.037	0.87 ± 0.082	0.92 ± 0.069	0.89 ± 0.045	0.88 ± 0.037
SimResNet-18	0.91 ± 0.033	0.89 ± 0.039	0.93 ± 0.040	0.91 ± 0.034	0.91 ± 0.033
SiameseNet	$\mathbf{0.97 \pm 0.017}$	$\mathbf{0.96 \pm 0.015}$	$\mathbf{0.97 \pm 0.028}$	$\mathbf{0.97 \pm 0.017}$	$\mathbf{0.97 \pm 0.017}$

The results show that not only SiameseNet achieved very high accuracy results (97%) but also SimResNet-18 (91%) and ResNet-50 (89%).

Thus, we can state that in this case, the results presented are not reliable, and that the network is using features that are not relevant to the problem, such as the dimension of the slices, or some other features that are not correlated with the stroke effects.

Chronologically, the first results obtained were those in which the dataset was randomly split, i.e., the results represented in the table 5.2. After observing the results, there were some doubts that they were correct and it was finally concluded that there was data leakage. Only then the 5-fold cross-validation by subject was implemented leading to the results in table 5.1.

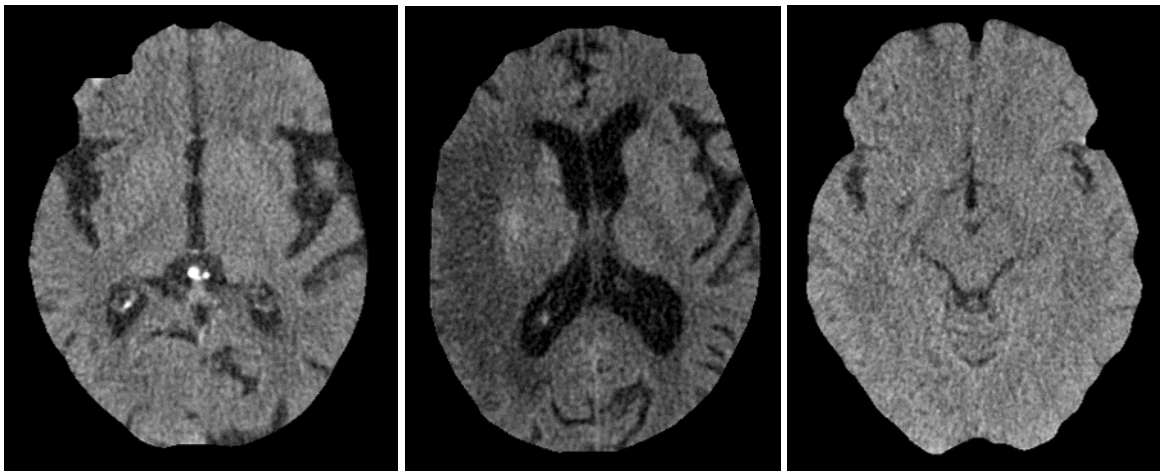
5.2.4 Examples of predictions

Figure 5.3 illustrates some examples of good predictions by the SiameseNet architecture, using the stratified 5-fold cross-validation by subject.

In the case of class 0 images, 5.3a and 5.3b, these ones contain strong visible changes in one of the hemispheres that help the model learn and correctly classify the samples. In the case of the image corresponding to class 1, i.e. the image present in 5.3c, the absence of variations in the hemispheres and the high anatomical similarity enabled the model to learn and correctly classify the image.

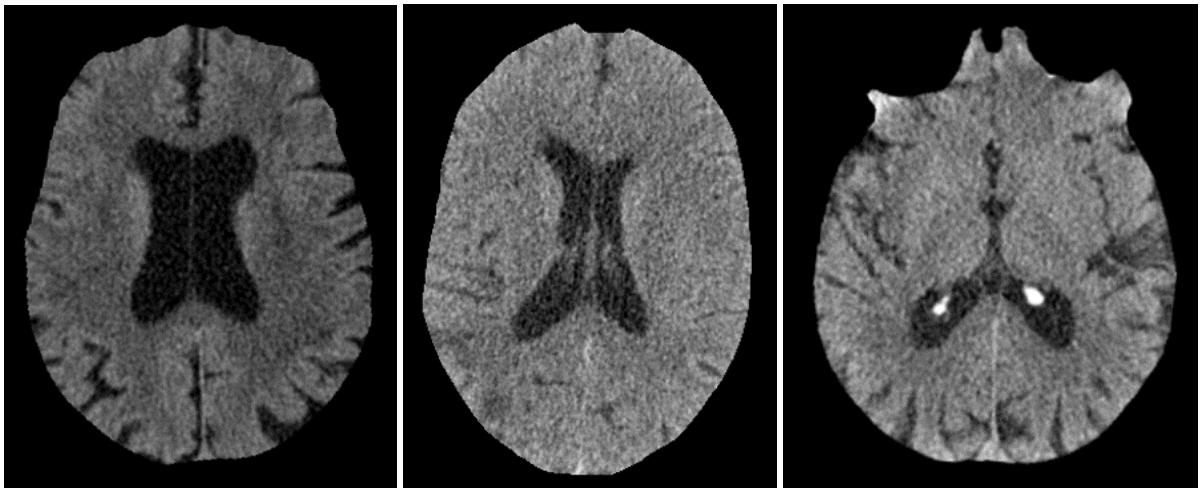
On the other hand, figure 5.4 illustrates some examples of wrong predictions by the same model and in the same conditions.

Considering images 5.4a and 5.4c belonging to class 1, i.e, absence of stroke, the model misclassified these images due to the high asymmetry present in the image, as a result of the brain anatomy itself. In this case, it can be stated that the model failed to learn to distinguish between stroke injuries and brain anatomy.



(a) (b) (c)

Figure 5.3: Examples of good predictions by the SiameseNet model. The images shown in a) and b) are examples of slices belonging to class 0 - presence of stroke and, the image shown in c) is an example of slices belonging to class 1 - absence of stroke. Source: Images taken and adapted from HSM database.



(a) (b) (c)

Figure 5.4: Examples of wrong predictions by the SiameseNet model. The images shown in a) and c) are examples of slices belonging to class 1 - absence of stroke and, the image shown in b) is an example of slices belonging to class 0 - presence of stroke. Source: Images taken and adapted from HSM database.

In the case of image 5.4b, which is an example of class 0, the presence of stroke (in the right hemisphere) is so subtle that the model has not learned to classify these less visible lesions.

6

Conclusions

Contents

6.1 Conclusions and Future Work

We proposed to use a Siamese Network architecture for the detection of ischemic stroke, based on the fact that detected asymmetry of brain hemispheres is strongly correlated with evidence of stroke. The architecture obtained 72% accuracy, on average, on an independent test set. Although the model exhibits an accuracy insufficient for medical applications, this architecture may become the starting point for several other studies focusing on the symmetry of the brain hemispheres and how it is related to the existence of pathologies in the brain.

Although the results were not as positive as expected, a large part of this work was on the preprocessing of the images. Therefore, a labeled and preprocessed dataset is now available and can be used for future research.

During the development of the work, one of the concerns was the choice of images, since the model could learn unwanted features. By achieving very high performance, we realized that the model was learning something else since there are sequences of similar images in both sets (training and test).

Another reason that explains the results obtained is related to the complexity of the images. The hemispheres present a certain anatomical similarity that can mislead the learning of the features associated with stroke.

There are some aspects of the presented work that can be improved. One of the aspects that can be improved is image preprocessing. Although the skull stripping process is mostly successful there are slices where small portions of non-brain tissue are mistakenly removed. This happens because all the white pixels in the image are removed when only the white pixels around the brain (bone) should be removed.

Another possible improvement is the removal of the CSF area since the intensity of the pixels is the same as the ischemic zone and can mislead the model.

Finally, since the dataset is not large and extensive enough, the performance of the model falls short. In fact, only slices from 60 patients were used, with 35 patients associated with class 1 and 25 patients assigned to class 0. So, one possibility of improving the performance of the model would be to label more data and to select the same number of slices from each patient, avoiding overfitting, data leakage, and selection bias.

Bibliography

[1] M. S. Phipps and C. A. Cronin, "Management of acute ischemic stroke," *BMJ*, vol. 368, p. l6983, feb 2020. [Online]. Available: https://www.bmj.com/lookup/doi/10.1136/bmj.l6983

[2] M. P. Lindsay, B. Norrving, R. L. Sacco, M. Brainin, W. Hacke, S. Martins, J. Pandian, and V. Feigin, "World Stroke Organization (WSO): Global Stroke Fact Sheet 2019," *International Journal of Stroke*, vol. 14, no. 8, pp. 806–817, oct 2019. [Online]. Available: http://journals.sagepub.com/doi/10.1177/1747493019881353

[3] S. X. Liu, "Symmetry and asymmetry analysis and its implications to computer-aided diagnosis: A review of the literature," *Journal of Biomedical Informatics*, vol. 42, no. 6, pp. 1056–1064, dec 2009. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1532046409000975

[4] U. Rajendra Acharya, K. M. Meiburger, O. Faust, J. En Wei Koh, S. Lih Oh, E. J. Ciaccio, A. Subudhi, V. Jahmunah, and S. Sabut, "Automatic detection of ischemic stroke using higher order spectra features in brain MRI images," *Cognitive Systems Research*, vol. 58, pp. 134–142, dec 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1389041719300993

[5] C.-L. Chin, B.-J. Lin, G.-R. Wu, T.-C. Weng, C.-S. Yang, R.-C. Su, and Y.-J. Pan, "An automated early ischemic stroke detection system using CNN deep learning algorithm," in *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*. IEEE, nov 2017, pp. 368–372. [Online]. Available: http://ieeexplore.ieee.org/document/8256481/

[6] G. C. S. Ruppert, L. Teverovskiy, C.-P. Yu, A. X. Falcao, and Y. Liu, "A new symmetry-based method for mid-sagittal plane extraction in neuroimages," in *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. IEEE, mar 2011, pp. 285–288. [Online]. Available: http://ieeexplore.ieee.org/document/5872407/

[7] H. Wu, X. Chen, P. Li, and Z. Wen, "Automatic Symmetry Detection From Brain MRI Based on a 2-Channel Convolutional Neural Network," *IEEE Transactions on Cybernetics*, vol. 51, no. 9, pp. 4464–4475, sep 2021. [Online]. Available: https://ieeexplore.ieee.org/document/8917648/

[8] M. E. Miletto Petrazzini, V. A. Sovrano, G. Vallortigara, and A. Messina, “Brain and Behavioral Asymmetry: A Lesson From Fish,” *Frontiers in Neuroanatomy*, vol. 14, p. 11, mar 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnana.2020.00011/full

[9] A. Vupputuri, S. Dighade, P. S. Prasanth, and N. Ghosh, “Symmetry determined superpixels for efficient lesion segmentation of ischemic stroke from MRI,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, jul 2018, pp. 742–745. [Online]. Available: https://ieeexplore.ieee.org/document/8512283/

[10] Z. Barzegar and M. Jamzad, “Fully automated glioma tumour segmentation using anatomical symmetry plane detection in multimodal brain MRI,” *IET Computer Vision*, vol. 15, no. 7, pp. 463–473, oct 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1049/cvi2.12035

[11] C.-P. Yu, G. Ruppert, D. Nguyen, A. Falcão, and Y. Liu, “Statistical Asymmetry-based Brain Tumor Segmentation from 3D MR Images,” in *Biosignals*, 2012.

[12] M. Ribolsi, Z. J. Daskalakis, A. Siracusano, and G. Koch, “Abnormal Asymmetry of Brain Connectivity in Schizophrenia,” *Frontiers in Human Neuroscience*, vol. 8, p. 1010, dec 2014. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnhum.2014.01010/abstract

[13] J. Schröder and G. Thomalla, “A Critical Review of Alberta Stroke Program Early CT Score for Evaluation of Acute Stroke Imaging,” *Frontiers in Neurology*, vol. 7, p. 245, jan 2017. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fneur.2016.00245/full

[14] M. Pohl, D. Hesszenberger, K. Kapus, J. Meszaros, A. Feher, I. Varadi, G. Pusch, E. Fejes, A. Tibold, and G. Feher, “Ischemic stroke mimics: A comprehensive review,” *Journal of Clinical Neuroscience*, vol. 93, pp. 174–182, nov 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0967586821004811

[15] O. Senouf, S. Vedula, T. Weiss, A. Bronstein, O. Michailovich, and M. Zibulevsky, “Self-supervised learning of inverse problem solvers in medical imaging,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11795 LNCS. Springer, 2019, pp. 111–119.

[16] L. W. Goldman, “Principles of CT and CT Technology,” *Journal of Nuclear Medicine Technology*, vol. 35, no. 3, pp. 115–128, sep 2007. [Online]. Available: http://tech.snmjournals.org/cgi/doi/10.2967/jnmt.107.042978

[17] O. Öman, T. Mäkelä, E. Salli, S. Savolainen, and M. Kangasniemi, “3D convolutional neural networks applied to CT angiography in the detection of acute ischemic stroke,”

European Radiology Experimental, vol. 3, no. 1, p. 8, dec 2019. [Online]. Available: https://eurradiolexp.springeropen.com/articles/10.1186/s41747-019-0085-6

[18] D. R. Varma, "Managing DICOM images: Tips and tricks for the radiologist," *The Indian journal of radiology & imaging*, vol. 22, no. 1, pp. 4–13, jan 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/22623808https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3354356/

[19] T. D. DenOtter and J. Schubert, *Hounsfield Unit*, jan 2022. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/31613501

[20] Z. Xue, S. Antani, L. R. Long, D. Demner-Fushman, and G. R. Thoma, "Window classification of brain CT images in biomedical articles," *AMIA Annual Symposium proceedings. AMIA Symposium*, vol. 2012, pp. 1023–1029, 2012. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3540547/

[21] O. Elsayed, K. Mahar, M. Kholief, and H. A. Khater, "Automatic detection of the pulmonary nodules from CT images," in *2015 SAI Intelligent Systems Conference (IntelliSys)*. IEEE, nov 2015, pp. 742–746. [Online]. Available: http://ieeexplore.ieee.org/document/7361223/

[22] J. Jantzen, "Introduction to perceptron networks," *Neural Networks*, vol. 873, no. 98, pp. 1–32, 1998.

[23] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*. IEEE, aug 2017, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/8308186/

[24] S. Sharma and S. Sharma, "Activation functions in neural networks," *Towards Data Science*, vol. 6, no. 12, pp. 310–316, 2017.

[25] A. A. M. Al-Saffar, H. Tao, and M. A. Talab, "Review of deep convolution neural network in image classification," in *2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*. IEEE, oct 2017, pp. 26–31. [Online]. Available: http://ieeexplore.ieee.org/document/8253139/

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, may 2017. [Online]. Available: https://dl.acm.org/doi/10.1145/3065386

[27] J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology. Nanjing University. China*, vol. 5, p. 23, 2017.

[28] M. Lin, Q. Chen, and S. Yan, "Network In Network," *arXiv preprint arXiv:1312.4400*, dec 2013. [Online]. Available: http://arxiv.org/abs/1312.4400

[29] M. S. Ebrahimi and H. K. Abadi, “Study of Residual Networks for Image Recognition,” *CoRR*, vol. abs/1805.0, 2018. [Online]. Available: http://arxiv.org/abs/1805.00325

[30] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] G. R. Koch, “Siamese Neural Networks for One-Shot Image Recognition,” 2015.

[32] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, “SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification,” *CoRR*, vol. abs/1707.0, 2017. [Online]. Available: http://arxiv.org/abs/1707.02131

[33] N. J. Herzog and G. D. Magoulas, “Brain Asymmetry Detection and Machine Learning Classification for Diagnosis of Early Dementia,” *Sensors*, vol. 21, no. 3, p. 778, jan 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/3/778

[34] A. Gibicar, A. R. Moody, and A. Khademi, “Automated Midline Estimation for Symmetry Analysis of Cerebral Hemispheres in FLAIR MRI,” *Frontiers in Aging Neuroscience*, vol. 13, p. 644137, apr 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnagi.2021.644137/full

[35] F. P. G. Bergo, A. X. Falcão, C. L. Yasuda, and G. Ruppert, “Fast, accurate and precise mid-sagittal plane location in 3D MR images of the brain,” in *International Joint Conference on Biomedical Engineering Systems and Technologies*. Springer, 2008, pp. 278–290.

[36] H. J. Kuijf, S. J. van Veluw, M. I. Geerlings, M. A. Viergever, G. J. Biessels, and K. L. Vincken, “Automatic extraction of the midsagittal surface from brain MR images using the Kullback–Leibler measure,” *Neuroinformatics*, vol. 12, no. 3, pp. 395–403, 2014.

[37] A. Barman, M. E. Inam, S. Lee, S. Savitz, S. Sheth, and L. Giancardo, “Determining Ischemic Stroke From CT-Angiography Imaging Using Symmetry-Sensitive Convolutional Networks,” in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE, apr 2019, pp. 1873–1877. [Online]. Available: https://ieeexplore.ieee.org/document/8759475/

[38] S. Tummala, “Deep Learning Framework using Siamese Neural Network for Diagnosis of Autism from Brain Magnetic Resonance Imaging,” in *2021 6th International Conference for Convergence in Technology (I2CT)*, 2021, pp. 1–5.

[39] P. Refaeilzadeh, L. Tang, and H. Liu, *Cross-Validation*. Boston, MA: Springer US, 2009, pp. 532–538. [Online]. Available: https://doi.org/10.1007/978-0-387-39940-9{_}565

[40] X. Zeng and T. R. Martinez, “Distribution-balanced stratified cross-validation for accuracy estimation,” *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 12, no. 1, pp. 1–12, jan

2000. [Online]. Available: https://doi.org/10.1080/095281300146272http://www.tandfonline.com/doi/abs/10.1080/095281300146272

[41] J. Chaki and N. Dey, *A beginner's guide to image preprocessing techniques*. CRC Press, 2018.

[42] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization," *CoRR*, vol. abs/1610.0, 2016. [Online]. Available: http://arxiv.org/abs/1610.02391

[43] X. Liu, Y. Zhou, J. Zhao, R. Yao, B. Liu, and Y. Zheng, "Siamese Convolutional Neural Networks for Remote Sensing Scene Classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 8, pp. 1200–1204, aug 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8642394/

[44] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd International Conference on Pattern Recognition (ICPR)*. IEEE, dec 2016, pp. 378–383. [Online]. Available: http://ieeexplore.ieee.org/document/7899663/

[45] K. Qiu, Y. Ai, B. Tian, B. Wang, and D. Cao, "Siamese-ResNet: Implementing Loop Closure Detection based on Siamese Network," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2018, pp. 716–721. [Online]. Available: https://ieeexplore.ieee.org/document/8500465/

[46] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," *ArXiv*, vol. abs/2008.0, 2020.

[47] S. Raschka, "An Overview of General Performance Metrics of Binary Classifier Systems," *ArXiv*, vol. abs/1410.5, 2014.

[48] L. Gonçalves, A. Subtil, M. R. Oliveira, and P. de Zea Bermudez, "ROC Curve Estimation: An Overview," *REVSTAT-Statistical Journal*, vol. 12, no. 1 SE - Article, pp. 1–20, apr 2014. [Online]. Available: https://revstat.ine.pt/index.php/REVSTAT/article/view/141