Automatic AI-Driven segmentation of Acute Ischemic Stroke Regions with CT Perfusion Images

by

Luca Tomasetti

Thesis submitted in fulfillment of
the requirements for the degree of

PHILOSOPHIAE DOCTOR
(PhD)

University of
Stavanger

Faculty of Science and Technology
Department of Electrical Engineering and Computer Science
2023

University of Stavanger
N-4036 Stavanger
NORWAY
www.uis.no

ISBN 978-82-8439-201-1
ISSN 1890-1387

PhD Thesis UiS no. 733

Preface

This thesis is submitted as partial fulfilment of the requirements for the degree of *Philosophiae Doctor* at the University of Stavanger, Norway. The research has been carried out at the Department of Electrical Engineering and Computer Science, University of Stavanger in the period of August 2019 to August 2023. The compulsory courses were taken at the University of Stavanger.

The thesis is based on a collection of five papers - four published and one currently under review. For increased readability, the papers have been reformatted for alignment with the format of the thesis and are included as chapters.

Luca Tomasetti, November 2023

Abstract

This thesis investigates artificial intelligence (AI) methodologies to automatically delineate ischemic areas of brain Computed Tomography Perfusion (CTP) scans acquired at hospital admission in patients suspected of acute ischemic stroke (AIS). Stroke, a critical neurological disorder, has a substantial socio-economic impact and a tremendous effect on the quality of life of afflicted subjects. Time is essential for dealing with this neurological disorder: every minute millions of brain cells die during a cerebral stroke. Consequently, developing accurate and rapid automatic prediction techniques for identifying the location and size of ischemic regions, including tissue with an extremely high probability of infarction (core) and potentially recoverable tissue (penumbra), are of serious clinical interest.

CTP is a fast and widely used 4D imaging modality employed upon hospital admission for evaluating stroke severity and aiding in treatment planning. Automated segmentation methods for CTP need to be perform fast and within the golden hour for identifying tissue-at-risk and prepare treatments. Current methods primarily rely on clinically interpretable 3D parametric maps derived from these scans. Parametric maps are also generally adopted by neuroradiologists for assessing this neurological disorder. However, few segmentation analyses have investigated the usage of AI pipelines with 4D CTP as input. These segmentation methods only focus on segmenting already infarcted areas or core regions, neglecting the penumbra. Nonetheless, predicting penumbra areas can hold significant importance in treatment planning.

This thesis explores conventional supervised and semi-supervised AI approaches to segment both penumbra and core areas. Parametric maps and CTP scans have been adopted as input for Machine Learning (ML) and Deep Learning (DL) algorithms to determine the most suitable input data for the automatic segmentation task. Scans from subjects of different

ages and severity groups have been leveraged for training the ML and DL models, thus simulating real-world scenarios. Exploiting 4D CTP scans as input provided promising segmentation results on this dataset, regardless of the severity group, but it produced over-segmentation on large ischemic areas. Few-shot learning approaches returned promising outcomes; however, the results are still distant from supervised architectures. The thesis demonstrated the feasibility of employing CTP studies as input modality for segmenting both ischemic regions (penumbra and core) at hospital admission.

Acknowledgements

At the end of this four-year PhD project, I would like to express my profound gratitude to the following people who have played a crucial role in my academic journey and in completing this PhD thesis.

First and foremost, I would like to express how blessed I have been to be on this project with my supervisor, Associate Professor Mahdieh Khanmohammadi. Thank you for all your support, help, and guidance during this project. Many things have happened during these years; I know it's been a lot of work. I sincerely appreciate it. I would also like to extend my gratitude to my co-supervisor, Professor Kjersti Engan. Thank you for all the support and feedback provided along the way. I am honestly thankful that you challenged me every time and made me rethink every step. Working with all of you has been an honor, and I will profoundly miss our weekly meeting in Kjersti's office.

My deepest thanks to co-supervisor Dr. Kathinka Dæhli Kurz. It was always nice knowing that, along with your hectic schedule, you always had time for me. I would also extend my gratitude to Dr. Liv Jorunn Høllesli (my PhD twin) for her immense help during this period. Thank you for all the time invested in the project and for explaining to me a lot of medical details. This research would have been impossible without the help and support from the two of you.

Additionally, I want to express my gratitude to my co-authors for their invaluable contributions to this project and sharing your knowledge with me over these years. I would also like to thank the committee members for taking the time to read my dissertation and attending the defense.

To all the beautiful people in the UiS PhD Family group, thank you for being who you are, for the amazing time spent together, and for making the department a fantastic place.

To everyone in the UiT Machine Learning group, thank you for your very kind and warm welcome in that cold weather, I really appreciate it. It felt like being at home.

And finally a special thanks to my friends and family for the support I received during these four years. Thank you for being there for me. You made this journey look like an easy walk in the park.

Thank you all from the depths of my heart.

Luca Tomasetti, November 2023

List of Notations

(Some symbols have different meanings in different chapters. Hence, they appear in multiple chapters of this list.)

Chapter 2 - Technical Background

$\mathcal{M}$	Machine Learning model
$\mathcal{C}$	Number of Classes
$\mathcal{D}$	Dataset
N	Dataset size
N_T	Training set size
M_T	Unlabeled data size
$\mathcal{D}_T$	Training Set
$\mathcal{D}_{\mathbf{Test}}$	Test set
$\mathcal{D}_V$	Validation Set
x	General Input
$\mathcal{I}$	Image
(x_i, y_i)	Input-label pairs of dataset N for $i \in N$
y	Output
$\hat{y}$	Predicted Output
$\hat{y}_{i,c}$	Probability of a pixel i to be as class c in the predicted image $\hat{y}$

$\hat{y}_{i,\hat{c}}$	Probability of a pixel i to NOT be as class c
$\mathcal{L}$	Loss Function
$\mathcal{H}$	Kernel
$\widetilde{h}$	Half-height of the kernel $\mathcal{H}$
$\widetilde{w}$	Half-width of the kernel $\mathcal{H}$
θ	Activation Function
w_i	Vector of weights
$\mathbf{b}$	Biased value

Chapter 4 - Data Material

$\mathcal{D}$	Dataset
$\mathcal{D}_T$	Training Set
$\mathcal{D}_{\mathbf{Test}}$	Test set
$\mathcal{D}_V$	Validation Set

Chapter 5 - Ground Truth and Pre-Processing

x	First Input dimension (width)
y	Second Input dimension (height)
z	Third Input dimension (depth)
t	Forth Input dimension (time)
$V(x, y, z, t)$	Input Vector
$\dot{V}(x, y, z, t)$	Vector after the Brain Extraction step
$\ddot{V}(x, y, z, t)$	Vector the Gamma Correction step
$z_{\mathbf{high}}$	Slice with the highest intensity value
$\widetilde{V}(x, y, z, t)$	Output Vector

Chapter 6 - Supervised Ischemic Stroke Segmentation

z_i	ith brain slice
$z_{\mathcal{I}}$	List of $\{z_{i-1}, z_i, z_{i+1}\}$ slices
$P_{z_i}(x, y)$	Output Image of slice z_i
$\widehat{V}_{z_i}$	Input Vector of slice z_i (2D+time Tensor)
$\bar{V}^t_{z_{\mathcal{I}}}(x, y)$	List of 2D+time Inputs (3D+time Tensor)
$\widehat{V}^t_{z_{\mathcal{I}}}$	Input Vector, concatenation of a 2D+time Tensor (4D Tensor)

Paper I - Machine learning algorithms versus thresholding to segment ischemic regions in patients with acute ischemic stroke

p	Patient index
T	Training set $T = \{(x_1, y_1), \ldots, (x_T, y_T)\}$
(x_i, y_i)	Input-label pairs of dataset T for $i \in T$
$\mathbf{CBV}^{\ p}$	Parametric map (in this case CBV) for patient p
$\mathbf{CBV}^{\ p}_{\mathbf{SLIC}}$	The superpixel version of the parametric map CBV (in this case) for patient p
$\mathbf{BT}^p$	The set of pixels belonging to the brain tissue for patients p
$\mathbf{BT}$	The set of pixels belonging to the brain tissue for all patients p: $\mathrm{BT} = \bigcup \mathrm{BT}^p$
L^p	The number of pixels in BT^p
$x^p_{\mathbf{CBV}}$	Input feature vector for a parametric map (in this case CBV) for patient p; $x^p_{\mathrm{CBV}} = \mathrm{stack}(\mathrm{CBV}^p(i,j))_{\forall (i,j) \in \mathrm{BT}^p}$. It is a vector of size L^p.
X^p	Matrix containing the entire input features of the parametric maps for a patient p
$[\mathbf{1}]$	All-ones vector of length L^p
$x_{\mathbf{NIHSS}}$	Input feature vector for the NIHSS score

$X_{\mathbf{SLIC}}$	Input features totality for the superpixel version of the parametric maps: $X_{\text{SLIC}} = [x_{\text{CBV}_s}\ x_{\text{CBF}_s}\ x_{\text{TTP}_s}\ x_{\text{T}_{\text{Max}s}}]$
$x_{\mathbf{CBV}_s}$	Input vector for the superpixel version of a parametric map (in this case CBV): $x_{\text{CBV}_s} = \text{stack}(\text{CBV}^p_{\text{SLIC}}(i,j))_{\forall(i,j)\in\text{BT}^p}$
X_T	Total training matrix $X_T = [X\ X_{\text{SLIC}}]$
$\mathbf{C}$	Set of classes: $\{\text{core}, \text{penumbra}, \text{healthy brain}\}$
$\mathbf{TP}_c$	True Positive values for class c
$\mathbf{FP}_c$	False Positive values for class c
$\mathbf{FN}_c$	False Negative values for class c
$\mathbf{TN}_c$	True Negative values for class c
rec_c	Recall metric for class c, calculated as $\frac{\text{TP}_c}{\text{TP}_c+\text{FN}_c}$
prec_c	Precision metric for class c, calculated as $\frac{\text{TP}_c}{\text{TP}_c+\text{FP}_c}$
Dice_c	Dice Coefficient for class c, calculated as $\frac{2\cdot\text{prec}_c\cdot\text{red}_c}{\text{prec}_c+\text{rec}_c}$
V_p	Predicted volume
V_g	Ground Truth volume
ΔV	Absolute difference in the volume among the predictions and the ground truth, expressed as $\lvert V_g - V_p\rvert$

Paper II - Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection

γ	Hyper-parameter of the Focal Tversky Loss function
α	Hyper-parameter of the Focal Tversky Loss function
β	Hyper-parameter of the Focal Tversky Loss function
V_p	Predicted volume
V_g	Ground Truth volume
ΔV	Absolute difference in the volume among the predictions and the ground truth, expressed as $\lvert V_g - V_p\rvert$

Paper III - CNN Based Segmentation of Infarcted Regions in Acute Cerebral Stroke Patients From Computed Tomography Perfusion Imaging

t	Time Index
j	Tile Index
A_j	Set of image coordinate pairs corresponding to a tile number j
(m, n)	Spatial Image Coordinates
S	Total Number of Brain Slices per Patient
T	Total Number of Timepoints
s_i	ith Brain Slice
V	4D CTP data ($V \in Z^4$), 4D signal of dimension ($T \times M \times N \times S$)
$\hat{V}_{s_i}$	Input Volume of slice s_i after brain extraction $\in Z^4$
$\tilde{V}_{s_i}$	Input Volume of slice s_i after all the pre-processing steps (2D+time Tensor) $\in Z^4$
Y_{s_i}	2D Predicted Output of slice s_i
Y_{GT_i}	2D Ground Truth Image of slice i
y_k	kth element of the probability vector generated by a NN
$\mathbf{loss}_j$	Loss function for the jth tile
$\mathbf{cost_{batch}}$	Cost function for each batch, defined as $\sum_{j \in \text{batch}} \text{loss}_j$

Paper IV - CT Perfusion is All We Need: 4D CNN Segmentation of Penumbra and Core in Patients With Suspected Ischemic Stroke

X	Width dimension of a CTP scan
Y	Height dimension of a CTP scan
Z	Depth dimension of a CTP scan
T	Time dimension of a CTP scan
V	4D raw CTP scan, $V \in \mathbb{R}^{(X \times Y \times Z \times T)}$
t	List of all the time points, defined as $[t_j \vert \forall j \in \{1, 2, \cdots, t_{\max}\}]$
$t_{\mathbf{max}}$	Last time point in the time dimension
z	List of all the slices, represented as $[z_i \vert \forall i \in \{1, 2, \cdots, z_{\max}\}]$
$z_{\mathbf{max}}$	Last slice in the depth dimension
$\mathcal{I}$	Set of indexes i, plus its neighbours $i-1$ and $i+1$, expressed as $\{i-1, i, i+1\}$
z_I	Set of slices: z_{i-1}, z_i, z_{i+1}
$\mathcal{C}$	Set of classes: {healthy brain, penumbra, core}
$c \in \mathcal{C}$	Single class
$I_{z_i}^{t_j}$	2D brain slice z_i at time point t_j, $I \in \mathbb{R}^{(X \times Y)}$
P_{z_i}	2D probability output of brain slice z_i; $P \in \mathbb{R}^{(X \times Y)}$
$\widetilde{\cdot}$	Input after pre-processing steps
$\bar{\cdot}$	List of inputs
$\varphi(\cdot)$	Concatenation function
$\widehat{\cdot}$	Concatenated inputs after passing through $\varphi(\cdot)$
$\bar{V}_{z_i}^{t}$	List of 2D images $\widetilde{I}_{z_i}$ for all the time points t; defined as $\bar{V}_{z_i}^{t} = [\widetilde{I}_{z_i}^{t_j} \vert \forall t_j \in t] \in \mathbb{R}^{(X \times Y)}$
$\widehat{V}_{z_i}^{t}$	2D+time volume of slice z_i. Expressed as: $\widehat{V}_{z_i}^{t} = \varphi(\widetilde{I}_{z_i}^{t_j} \vert \forall t_j \in t) \in \mathbb{R}^{(X \times Y \times T)}$

$\widehat{V}_{z_\mathcal{I}}^{t_j}$	3D volume of slices $z_\mathcal{I}$ for a time point t_j; defined as: $\widehat{V}_{z_\mathcal{I}}^{t_j} = \varphi(\widetilde{V}_{z_{i-1}}^{t_j}, \widetilde{V}_{z_i}^{t_j}, \widetilde{V}_{z_{i+1}}^{t_j}) \in \mathbb{R}^{(X \times Y \times Z)}$
$\bar{V}_{z_\mathcal{I}}^{t}$	List of 3D volumes of slices $z_\mathcal{I}$ for all the time points t. Represented as: $\bar{V}_{z_\mathcal{I}}^{t} = [\widehat{V}_{z_\mathcal{I}}^{t_j} \vert \forall t_j \in t] \in \mathbb{R}^{(X \times Y \times Z)}$
$\bar{V}_{z_\mathcal{I}}^{t}$	List of 2D+time volumes for slices $z_\mathcal{I}$; defined as: $\bar{V}_{z_\mathcal{I}}^{t} = [\widehat{V}_{z_{i-1}}^{t}, \widehat{V}_{z_i}^{t}, \widehat{V}_{z_{i+1}}^{t}] \in \mathbb{R}^{(X \times Y \times Z \times T)}$
$\widehat{V}_{z_\mathcal{I}}^{t}$	4D Tensor of slices $z_\mathcal{I}$ over all the time points t; expressed as: $\widehat{V}_{z_\mathcal{I}}^{t} = \varphi(\widehat{V}_{z_{i-1}}^{t}, \widehat{V}_{z_i}^{t}, \widehat{V}_{z_{i+1}}^{t}) \in \mathbb{R}^{(X \times Y \times Z \times T)}$
γ	Gamma correction value
$I(x, y, z, t) \in \mathbb{R}^4$	4D Tensor
$\mathcal{H}(w, h, d, p) \in \mathbb{R}^4$	4D Kernel
$g''(x, y)$	2D Convolution function
$g'''(x, y, z)$	3D Convolution function
$g''''(x, y, z, t)$	4D Convolution function
$\widetilde{h} \equiv \lfloor \frac{1}{2}(h-1) \rfloor$	Half-width of kernel $\mathcal{H}$
$\widetilde{w} \equiv \lfloor \frac{1}{2}(w-1) \rfloor$	Half-height of kernel $\mathcal{H}$
$\widetilde{d} \equiv \lfloor \frac{1}{2}(d-1) \rfloor$	Half-depth of kernel $\mathcal{H}$
$\widetilde{p} \equiv \lfloor \frac{1}{2}(p-1) \rfloor$	Half-time of kernel $\mathcal{H}$
$\mathbf{FTL}($	Focal Tversky loss
$\mathbf{TI}_c$	Tversky index for a class c
SDCL	Soft Dice Coefficient loss
DCL	Dice Coefficient loss
WCC	Weighted categorical cross-entropy loss
V_x	Predicted volume
V_y	Ground Truth volume
ΔV	Absolute difference in the volume among the predictions and the ground truth, expressed as $\lvert V_y - V_x \rvert$

Paper V - Self-Supervised Few-Shot Learning for Ischemic Stroke Lesion Segmentation

I	4D Input tensor $I \in \mathbb{R}^{(W \times H \times Z \times M)}$
W	Width dimension of the input I
H	Height dimension of the input I
Z	Depth dimension of the input I
M	Additional dimension of the input I, , corresponding to the number of modalities
$O \in \mathbb{R}^{(W \times H \times Z)}$	3D Volume
$I_{\mathbf{PMs}}$	4D Input with the additional modality dimension given by the PMs
$O_{\mathbf{PMs}}$	Corresponding output of the input I_{PMs}
$I_{\mathbf{CTP}}$	4D Input with the additional modality dimension given by the CTP
ρ	hyper-parameter that control the supervoxel size in Felzenszwalb' algorithm
$\mathcal{S} = \{X_s, L_s\}$	Support pair, where X_s is a support 2D+time slice and L_s is the corresponding 2D binary label image
$\mathcal{Q} = \{X_q, L_q\}$	Query pair, where X_q is a query 2D+time slice and L_q is the corresponding 2D binary label image
F^s	High-level support features
F^q	High-level query features
$p \in \mathbb{R}^d$	Foreground prototype, where d is the dimension of the embedding space
CS	Negative Cosine Similarity
$\hat{y}_q$	Predicted foreground mask
Th	Learned parameter
$V_{\hat{y}}$	Predicted volume
V_y	Ground Truth volume
ΔV	Absolute difference in the volume among the predictions and the ground truth, expressed as $\lvert V_y - V_{\hat{y}} \rvert$

List of Abbreviations

AI	Artificial Intelligence
AIS	Acute Ischemic Stroke
BT	Brain Tissue
CBF	Cerebral Blood Flow
CBV	Cerebral Blood Volume
CNN	Convolutional Neural Network
CT	Computed Tomography
CTA	Computed Tomography Angiography
CTP	Computed Tomography Perfusion
DALY	Disability-Adjusted Life Year
DL	Deep Learning
DNN	Deep Neural Network
DT	Decision Tree
DWI	Diffusion-Weighted Imaging
FIA	Final Infarct Area
FSS	Few-Shot Segmentation
HE	Histogram Equalization
HU	Hounsfield Unit
IS	Ischemic Stroke

LVO	Large Vessel Occlusion
ML	Machine Learning
MRI	Magnetic Resonance Imaging
MIP	Maximum Intensity Projection
MLP	Multi-Layer Perceptron
MTT	Mean of capillary Transit Time
NIHSS	National Institutes of Health Stroke Scale
NCCT	Non-Contrast Computed Tomography
NN	Neural Network
Non-LVO	Non-Large Vessel Occlusion
PM	Parametric Map
RF	Random Forest
SLIC	Simple Linear Iterative Clustering
SMOTE	Synthetic Minority Over-sampling Technique
SVM	Support Vector Machine
$\mathbf{T_{Max}}$	Time-To-Maximum
TTP	Time-To-Peak
WIS	Without Ischemic Stroke

List of publications

The main part of this dissertation is made up of the following published scientific papers:

- **Paper 1**

  Machine learning algorithms versus thresholding to segment ischemic regions in patients with acute ischemic stroke
  L. Tomasetti, L. J. Høllesli, K. Engan, K. D. Kurz, M. W. Kurz, M. Khanmohammadi
  Published by the IEEE Journal of Biomedical and Health Informatics, 2021

- **Paper 2**

  Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection
  L. Tomasetti, M. Khanmohammadi, K. Engan, L. J. Høllesli, K. D. Kurz
  Published in the Proceedings of the Northern Lights Deep Learning Workshop (NLDL), 2022

- **Paper 3**

  CNN Based Segmentation of Infarcted Regions in Acute Cerebral Stroke Patients From Computed Tomography Perfusion Imaging
  L. Tomasetti, K. Engan, M. Khanmohammadi, K. D. Kurz
  Published in the Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2020

- **Paper 4**

  CT Perfusion is All We Need: 4D CNN Segmentation of

Penumbra and Core in Patients With Suspected Acute Ischemic Stroke
L. Tomasetti, K. Engan, L. J. Høllesli, K. D. Kurz, M. Khanmohammadi
Published by the IEEE Access, 2023

- **Paper 5**

Self-Supervised Few-Shot Learning for Ischemic Stroke Lesion Segmentation
L. Tomasetti, S. Hansen, M. Khanmohammadi, K. Engan, L. J. Høllesli, K. D. Kurz, M. Kampffmeyer
Published by IEEE, 20th IEEE International Symposium on Biomedical Imaging (ISBI), 2023

Contents

Chapter 1

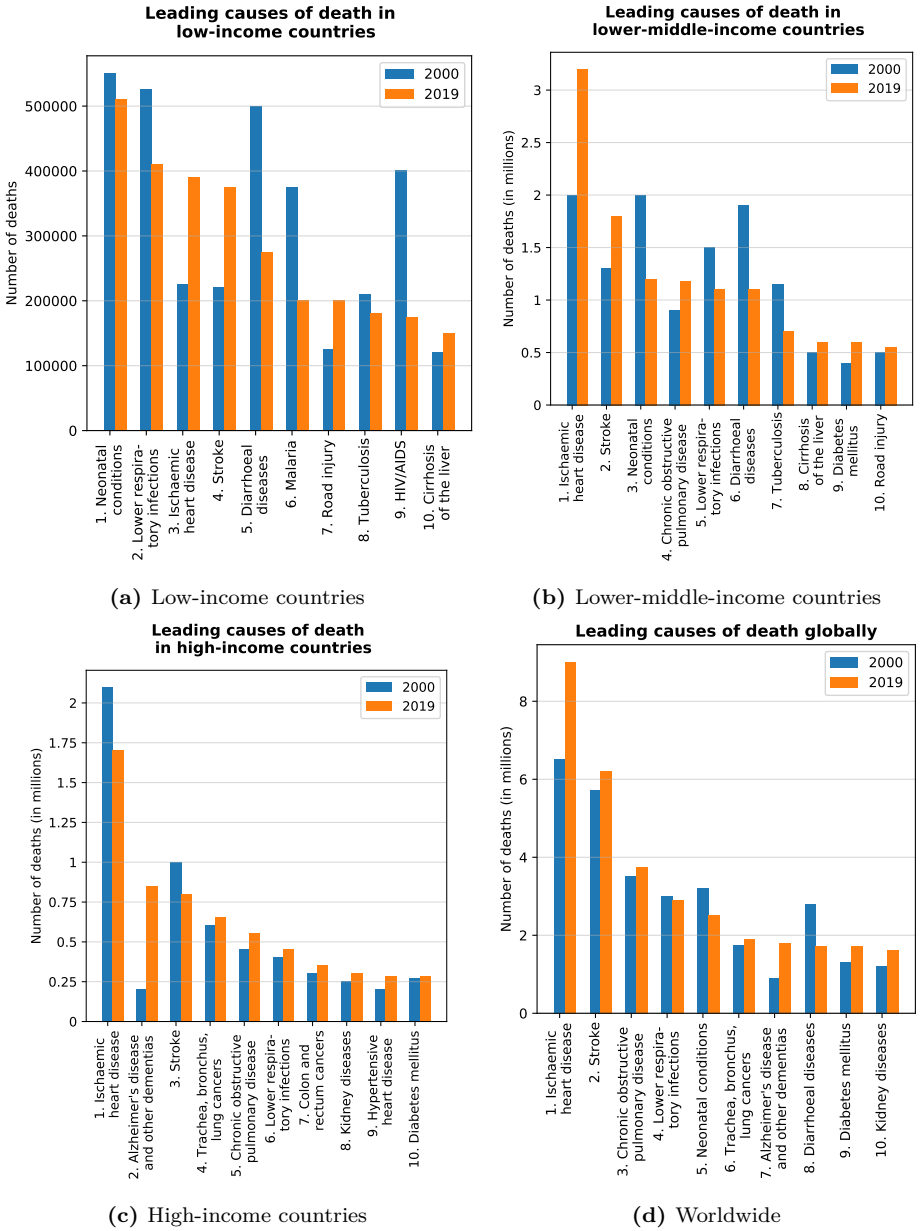
Introduction

Neurological disorders are the first cause of disability-adjusted life years (DALYs) and the second leading cause of death worldwide [1–3]. Cerebral stroke is the major contributor to these disorders; Fig. 1.1 shows the global trend of the most common causes of death in the last decade, divided by the country's income. In this thesis, we concentrate on ischemic stroke (IS) since they comprise the majority of the cerebral strokes [4, 5]. More specifically, we focus on the early stage of an IS, typically within the first day after symptoms onset. This stage is defined as acute IS (AIS) [6, 7].

A stroke has a huge socio-economic impact on society and a tremendous effect on the quality of life for every patient [8, 9]. According to the World Health Organization (WHO)[1], the trend for stroke deaths is increased in the past decades: the number of fatalities risen significantly in low and middle-income countries (LMICs) [1] (Fig. 1.1a and Fig. 1.1b). High-income countries (HICs) are the only category of income group in which there has been a slight decrease in deaths from this neurological disorder (Fig. 1.1c). Worldwide, deaths have increased since the last decade (Fig. 1.1d). There is an urgent need to improve access to stroke units and services globally, especially in LMICs, to reduce the global stroke burden [9–13]. Neuroradiologists face the challenge of quickly evaluating medical imaging of the brain to initiate treatment as soon as possible to minimize this disorder's immense burden.

The European Stroke Organization[2] recommend Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) as standard modalities for

[1] https://www.who.int/
[2] https://eso-stroke.org/



(a) Low-income countries

(b) Lower-middle-income countries

(c) High-income countries

(d) Worldwide

Figure 1.1: Leading causes of death worldwide, divided by countries income, based on the World Health Organization (WHO) data [14].

diagnostic imaging in patients suspected of IS [15]. CT is the preferred imaging modality because of its rapid scan time and common availability [16, 17]. CT Perfusion (CTP) is widely used in case of suspected AIS to positively identify ischemia and to make treatment decisions. In today's clinical practice, to rapidly assess a patient suspected of IS, radiologists examine Parametric Maps (PMs) derived from raw CTP scans [18]; these PMs emphasize spatio-temporal information from the passage of a contrast agent within the brain tissue.

The thesis's objective is to properly define two areas in the brain tissue that can appear during the early stage of an IS (a.k.a. AIS): the ischemic *core* and *penumbra*. The ischemic core is "a probabilistic estimate of tissue that is highly likely to become infarcted, if fast reperfusion does not occur" [19]. The ischemic penumbra was introduced by Astrup *et al.* as "a region of hypoperfused, electrically silent, and functionally impaired but viable tissue" [20]. These regions have been the long-standing therapeutic target of IS treatment [21]; thus, a precise and fast automatic evaluation of these regions can significantly help medical doctors during treatment decisions [9].

In the past decades, thresholding [22–25], machine learning (ML) [26–28], and deep learning (DL) [29–31] approaches have been broadly explored to make systems to aid the radiologists in critical decision-making regarding treatment. Algorithms using thresholding values on PMs are fast and straightforward solutions to implement for this task. Nevertheless, they are inaccurate, and there is no firm consensus on which threshold value to use to define the ischemic regions given the differences in patient and tissue characteristics [19, 32, 33]. Lately, ML and DL architectures have been implemented to overcome the abovementioned issues resulting from thresholding. These approaches are far more complex than thresholding algorithms, they are data-driven and need a large amount of training data to make reliable models. ML and DL have been demonstrated to give state-of-the-art performance in various domains, including medical applications. Some examples from medical applications include: image classification [34–37], image segmentation [38–42], wearable solutions [43, 44], and video analysis [45–47] are just some examples of their potential. These algorithms can analyze a large amount of data quickly and accurately. Hence, they can be deployed in various healthcare settings, including LMICs, to guide medical doctors in their diagnosis, making their workflow more efficient. A lower burden on medical doctors can help increase access to stroke diagnosis and treatment, which is essential in LMICs [9, 11–13].

Thesis focus The focus of this thesis is on the development of novel ML and DL architectures for IS semantic segmentation (Sec. 2.3), especially during the first stages of the neurological disorder. A quick and accurate delineation of AIS areas and their severity can be crucial for medical doctors to properly assess the condition and suitability for treatment within the first few hours after stroke onset. The task is highly challenging for several reasons: first, IS lesions vary considerably over time, even within the same clinical stage of stroke [7, 19]; and second, IS areas have a high variation in shape and location in the brain tissue, the size of a large lesion can cover an entire hemisphere, while the size of a small lesion can be extremely small [48]. We mainly concentrate on raw CTP scans and generated PMs derived from CTP as input for our ML and DL methods because CT scans are always available in hospital facilities and have a very fast response [16, 17, 49–51]. We exploited different inputs and input dimensions in our models to solve research challenges related to AIS segmentation. These research challenges and opportunities are discussed further in the following section.

1.1 Research Challenges and Opportunities

Medical images are essential for diagnostic and prognosis, especially in an AIS scenario. Despite the vast amount of medical images and information collected daily in all the hospitals worldwide, there is still a need for more manual annotation of the given data. Manual annotation refers to labeling images to create context and group data. These annotations are fundamental from a ML/DL perspective: annotations are commonly used to train models and evaluate their performances in a supervised or semi-supervised approach. Nevertheless, producing annotations on medical images can be highly time-consuming and costly as it usually requires expert input from medical doctors [51, 52].

For the research challenges of this thesis, manual annotations of the two ischemic areas (penumbra and core) were created from CTP scans acquired at hospital admission using an in-house built software (Sec. 5.1.1). Most of the research effort in the literature focuses on segmenting the final infarct area (FIA), the regions defined at follow-up of the dead tissue due to ischemia. However, precise and fast delineations of the *hospital admission ischemic areas* are also of fundamental clinical interest and might be essential in improving diagnosis and assessing the proper treatment. There is a lack of research for automatically segmenting ischemic penumbra

and core areas, which can partly be explained by the absence of consensus for defining these regions [33, 53, 54]. In addition, to the best of our knowledge, there are no publicly available datasets with annotations of core and penumbra on CTP scans. The public ISLES18 dataset [55] is the closest one found to our research challenges: the dataset contains CTP scans and PMs in combination with FIAs annotations.

In alternative, semi-supervised few-shot segmentation (FSS) approaches have gained momentum in the past years for image segmentation [41, 56, 57]. These models only require a few labeled training images, making them highly versatile and particularly suitable when labeled data is scarce or expensive to acquire, like in medical applications [57]. FSS models adopt, during training, a large set of unlabeled images together with a small portion of labeled images to provide a learning method that only relies on few labeled images. This approach drastically reduces the need for annotated data and leverages the vast amount of unlabeled data available in a medical setting. Moreover, these methods have shown higher flexibility when encountering new unseen semantic classes than supervised networks [41, 57]. However, these promising architectures only sometimes achieve state-of-the-art performances compared with fully supervised algorithms [41, 58, 59].

1.2 Objectives

The thesis objectives are divided into one main objective and three sub-objectives as follows:

- O_1: Development of a fast and automatic diagnostic system for segmenting ischemic penumbra and core areas in patients suspected of AIS with the data acquired at admission.
  - SO_1: Use PMs (derived from CTP) as input for ML and DL architectures, mimicking what medical doctors are commonly using for assessing neurological disorders.
  - SO_2: Leverage of CTP scans as input for the segmentation task, exploiting the totality of the raw scans in DL architectures.
  - SO_3: Investigate if self-supervised and/or unsupervised DL methods are suitable for the AIS segmentation.

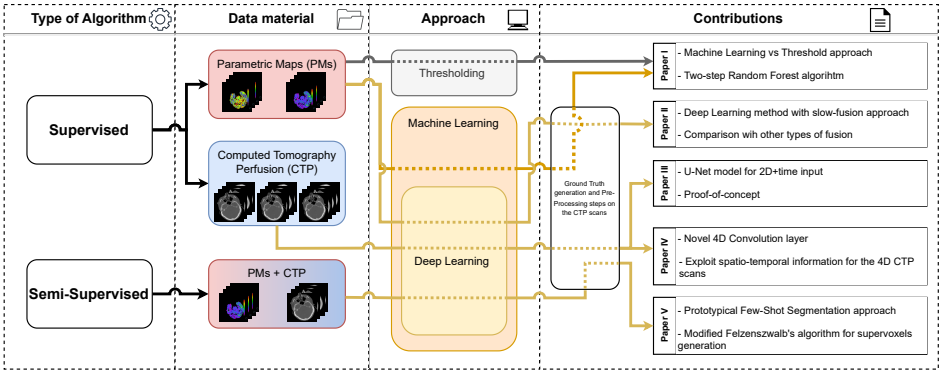


Figure 1.2: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy.

1.3 Proposed Approaches

Different ML and DL algorithms were proposed in five academic papers to meet each part of the thesis's objectives. In addition, various input combinations were studied for these algorithms to accurately determine the most suitable for the AIS segmentation. Figure 1.2 displays a general overview of the topics analyzed in the included papers and how they are connected.

The five papers of the thesis focus on the main objective (O_1). The methodology developed in this thesis addresses the first research sub-objective (SO_1) in two different ways: in Paper I, standard ML models have mainly been explored to assess the ischemic areas and compared with classic thresholding methods. In Paper II, a custom DL architecture has been proposed to combine patient's PMs and meta-information to segment the ischemic regions quickly. The second research sub-objective SO_2 is mainly addressed in Paper III, where a novel architecture is used as a proof-of-concept in a small cohort study. Paper IV further builds on Paper III's architecture by creating a more robust model, using a larger dataset and different input sizes, and comparing its accuracy with various approaches. The last research sub-objective SO_3 is addressed in Paper V. Inspired by the semi-supervised state-of-the-art method in medical segmentation [41],

a novel approach is implemented for the AIS segmentation task.

1.4 Project Contributions

The primary objectives of this thesis are to develop an automatic tool for AIS segmentation and to facilitate medical analysis by exploring non-conventional parameters. The thesis's main contributions are the five included papers summarized in the following.

I. Luca Tomasetti, Liv Jorunn Høllesli, Kjersti Engan, Kathinka Dæhli Kurz, Martin Wilhelm Kurz, and Mahdieh Khanmohammadi, **"Machine learning algorithms versus thresholding to segment ischemic regions in patients with acute ischemic stroke"**, IEEE Journal Of Biomedical and Health Informatics, 2021.

II. Luca Tomasetti, Mahdieh Khanmohammadi, Kjersti Engan, Liv Jorunn Høllesli, and Kathinka Dæhli Kurz, **"Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection"**, Proceedings of the Northern Lights Deep Learning Workshop, 2022.

III. Luca Tomasetti, Kjersti Engan, Mahdieh Khanmohammadi, and Kathinka Dæhli Kurz, **"CNN Based Segmentation of Infarcted Regions in Acute Cerebral Stroke Patients From Computed Tomography Perfusion Imaging"**, Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2020.

IV. Luca Tomasetti, Kjersti Engan, Liv Jorunn Høllesli, Kathinka Dæhli Kurz, and Mahdieh Khanmohammadi, **"CT Perfusion is All We Need: 4D CNN Segmentation of Penumbra and Core in Patients With Suspected Acute Ischemic Stroke"**, 2023.

V. Luca Tomasetti, Stine Hansen, Mahdieh Khanmohammadi, Kjersti Engan, Liv Jorunn Høllesli, Kathinka Dæhli Kurz, and Michael Kampffmeyer, **"Self-Supervised Few-Shot Learning for Ischemic Stroke Lesion Segmentation"**, IEEE International Symposium on Biomedical Imaging, 2023.

Paper I This Paper investigates the implementation of classical ML methods for the segmentations of the ischemic regions (penumbra and core). The proposed model utilizes the PMs as input. The study was performed to explicitly understand if PMs are suitable for a fully automatic network to solve the segmentation task.

Paper II This Paper proposes a novel DL architecture using the PMs as input. The model relies on a slow fusion approach to ensure high performances in segmenting the ischemic areas (penumbra and core). The research analyzes the strengths, limitations, and potentials of various approaches of input fusion and how they can influence the model's performance.

Paper III This Paper explores a convolutional neural network-based architecture for segmenting penumbra and core areas. Instead of relying on PMs, the network uses the full four-dimensional CTP dataset as input. The model is a proof-of-concept with a small sample of patients with promising outcomes.

Paper IV This Paper builds on top of the first three papers to create a more robust and reliable network for the segmentation task. The Paper explores how the fourth dimension for the input (CTP scans) can be exploited in order to generate more accurate segmentation regions compared to other input dimensions. The proposed methodology achieved promising results for both the areas analyzed.

Paper V This Paper illustrates using a semi-supervision method tailored for the IS segmentation. The model leverages domain knowledge by using as input CTP scans and pseudolabel areas generated by PMs using a modified supervoxel algorithm.

1.5 List of Other Works

VI. Luca Tomasetti, Mahdieh Khanmohammadi, Kjersti Engan, Liv Jorunn Høllesli, and Kathinka Dæhli Kurz, **"Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection"**, Norwegian Society for Image Processing and Machine Learning Conference (NOBIM), 2021.

VII. Liv Jorunn Høllesli, Luca Tomasetti, Kim Mouridsen, Jörn Schulz, Martin Wilhelm Kurz, and Kathinka Dæhli Kurz, **"The combination of CBV and transit time coefficient variation seems promising in predicting tissue outcome in large vessel occlusion AIS with subsequent recanalization"**, European Stroke Organisation Conference 2023.

VIII. Liv Jorunn Høllesli, Luca Tomasetti, Kim Mouridsen, Martin Wilhelm Kurz, Kjersti Engan, Mahdieh Khanmohammadi, Lars Fjetland, and Kathinka Dæhli Kurz, **"In patients with large vessel occlusion acute ischemic stroke, the distribution of CTP capillary transit time heterogeneity voxel values differs in recanalized versus non-recanalized patients"**, European Stroke Organisation Conference 2023.

IX. Liv Jorunn Høllesli, Luca Tomasetti, Kjersti Engan, Jörn Schulz, Kim Mouridsen, and Kathinka Dæhli Kurz, **"The combination of CBF and transit time coefficient variation seems promising in predicting tissue outcome in patients with acute ischemic stroke and large vessel occlusions"**, European Stroke Organisation Conference 2023.

X. Liv Jorunn Høllesli, Luca Tomasetti, Kim Mouridsen, Jörn Schulz, Kjersti Engan, Mahdieh Khanmohammadi, Martin Wilhelm Kurz, and Kathinka Dæhli Kurz, **"Using transit time coefficient variation in CT perfusion for prediction of tissue outcome in acute ischemic LVO stroke"**, in preparation.

1.6 Outlines

The thesis is structured in four parts: I) *Methodology*, II) *Clinical Applications*, III) *Contributions*, and IV) *Included Papers*.

Methodology describes basics ML and DL concepts relevant for the thesis (Chapter 2). *Clinical Applications* introduces knowledge about CTP and AIS and provides a brief overview of existing approaches to solve the segmentation tasks considered in this thesis (Chapter 3). It also provides an overview of the dataset used in the various papers included in the thesis (Chapter 4). *Contributions* summarize the pre-processing steps adopted (Chapter 5), the methods proposed with their scientific

contributions concerning the IS segmentation task (Chapter 6-7). Finally, it provides remarks on this work (Chapter 8). *Included Papers* lists the included papers in this thesis.

Part I

Methodology

Chapter 2

Technical Background

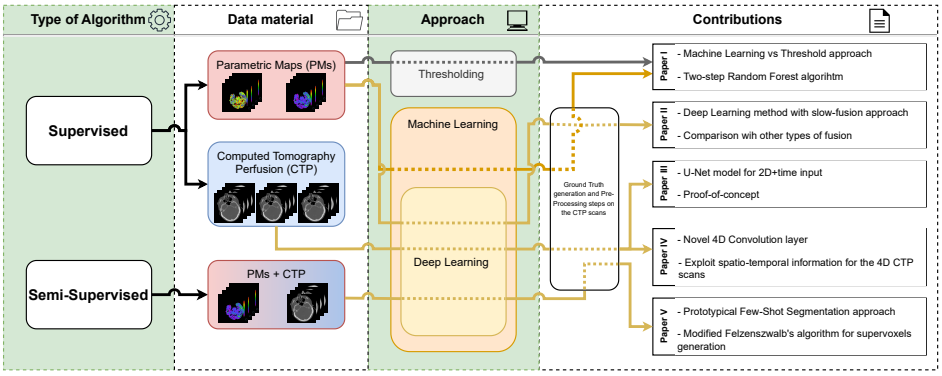


Figure 2.1: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy. The topics of this chapter (type of algorithms and approach) are highlighted in green.

This chapter serves to define and explain the theory behind this thesis. Fig. 2.1 highlights the topic of this chapter (in green) in relation to the proposed pipeline. The reason is to provide a basic understanding of the technical concepts discussed further in the thesis, as the overall topic is far too wide to cover in the scope of this thesis. Firstly, an introduction to Machine Learning is presented, discussing the various learning approaches and the input dimensionality adopted in the project. Secondly, the concept

of Deep Learning is introduced, focusing on Convolutional Neural Networks, Activation Functions, Loss Functions, and Optimizers. Finally, a brief description of the Image Segmentation problem is introduced.

2.1 Machine Learning

In the past decades, Artificial Intelligence (AI) has been a subject of intense media hype: from Google's DeepMind AlphaGo [60], a computer program that in 2016 defeated the world's number one Go player Ke Jie, to OpenAI's ChatGPT [61], a powerful and versatile language model used as a chatbot.

AI is a broad discipline where a machine (software/hardware) can demonstrate how to perform intelligent tasks. Machine learning (ML) is a sub-field of AI (Fig. 2.2) that involves developing algorithms and models that can be trained to *learn* from input data and make predictions or decisions without being explicitly programmed. Tom Mitchell, a pioneer in the field, formally defined ML as follows:

> "A computer program is said to learn from experience E for some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E". [62]

The term "Machine Learning" was coined in 1959 by Arthur Samuel, who investigates self-learning algorithms for the game of checkers [63]. Since then, ML has been proven to be efficient and a rapidly growing area of research. It has applications in a wide range of fields, including computer vision [39, 64–66], natural language processing [67–69], privacy preservation [70, 71], robotics [72–74], and healthcare [43, 46, 75–78].

2.1.1 Learning Approaches

The main goal of ML is to develop a self-learning architecture that can optimize its predictions by making decisions with a high probability of confidence. Moreover, an essential feature for an ML model $\mathcal{M}$ is generalizing well on new, unseen data; this is typically achieved by training $\mathcal{M}$ on a large dataset $\mathcal{D}$ of labeled examples. A dataset $\mathcal{D}$ contains a finite number of samples N, each defined by a d dimensional column vector $x = [x_1, \ldots, x_d]^T$. Typically, the dataset $\mathcal{D}$ is divided in three groups: training $\mathcal{D}_T$, validation

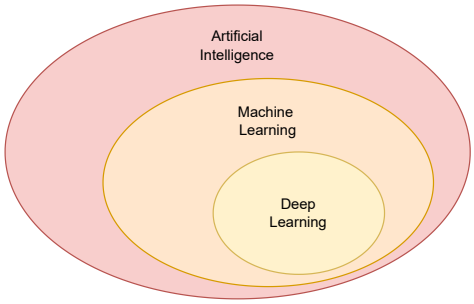


Figure 2.2: Artificial Intelligence, Machine Learning, and Deep Learning interdependence.

$\mathcal{D}_V$, and test set $\mathcal{D}_{\text{Test}}$. During training, the algorithm learns to identify patterns and relationships between the input features and the output labels extracted from the $\mathcal{D}_T$ set. The model's performances are measured on the $\mathcal{D}_V$ set, and if they increase over time, the model is said to be learning. Once the model is trained, it can be used to make predictions or decisions on new, unseen data, hence samples from the $\mathcal{D}_{\text{Test}}$.

ML algorithms build models that can solve tasks by exploiting specific patterns in the data using a large set of training samples. Fig. 2.3 illustrates this concept's visual representation. During training, a model $\mathcal{M}_\theta$ updates its parameters θ through an optimizer to solve its specific task. The training process typically involves optimizing a loss function $\mathcal{L}$. Sec. 2.2.3 discusses the loss functions implemented in this work, while Sec. 2.2.4 describes the optimizers.

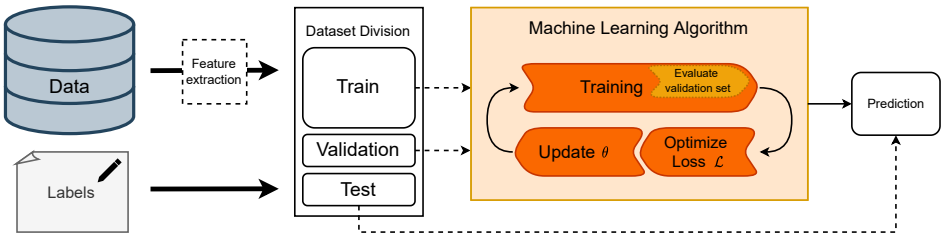


Figure 2.3: General pipeline of architecture in a Machine Learning setting.

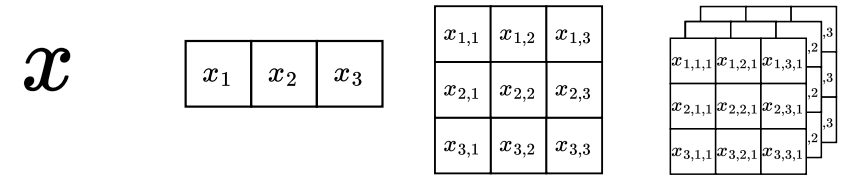
There are several types of ML algorithms, depending on the availability of labeled input data used during a model's training, including supervised, semi-supervised, and unsupervised learning.

Supervised Supervised learning involves training a model on labeled examples, where the input features and the corresponding output labels are provided during training and validation. In this approach, a sample $x_i, i \in N$ is associated with a label y_i, representing the actual class of the ith sample. Thus, a training dataset of size N_T is given as a number of input-label pairs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N_T}$.

Unsupervised On the other hand, unsupervised learning involves training a model on unlabeled examples, where the goal is to identify patterns and structure in the data without having labeled samples to rely on. The training set of size N_T is given as many input elements without any labels: $\mathcal{D} = \{(x_i)\}_{i=1}^{N_T}$.

Semi-Supervised A semi-supervised algorithm is a hybrid between a supervised and an unsupervised approach. In the semi-supervised approach, the algorithm receives both pairs of input-label data (of size N_T) and several M_T unlabeled data: $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N_T} \cup \{(x_i)\}_{i=1}^{M_T}$. Typically, the number of unlabeled data provided in the dataset is preponderantly compared to the number of input-label pairs.

2.1.2 Dimensionality



(a) Rank-0 **(b)** Rank-1 **(c)** Rank-2 **(d)** Rank-3

Figure 2.4: Example of various tensors dimensions utilized in this thesis.

All current ML systems use tensors as their primary data structure. A tensor is a multidimensional container for data; usually, numerical data: vectors and matrices are all examples of a tensor with different dimensions [79]. Fig. 2.4 displays an example of the various tensors used in this work. The number of dimensions of a tensor is also called *rank*. Typical tensors implemented in this work are:

- *Scalar* (rank-0): A tensor that contains only a number is called a scalar (Fig. 2.4a);
- *Vector* (rank-1): A list of number is a vector; this data structure has one dimension (Fig. 2.4b);
- *Matrix* (rank-2): A list of vectors is called a matrix. A matrix has two dimensions, usually defined as *rows* and *columns* (Fig. 2.4c);
- *Rank-3 and higher-rank tensor*: A list of matrices is defined as a rank-3 tensor, where it can be visualized as a cube of numbers. If multiple rank-3 tensors are stacked together, the result is a rank-4 tensor (Fig. 2.4d).

2.2 Deep Learning

Deep Learning (DL) is a subset of ML (Fig. 2.2). The main difference is the emphasis on learning successive layers of increasingly meaningful representations [80]. The "deep" in DL represents a model's extensive amount of successive layers. The number of layers defines the *depth* of the model.

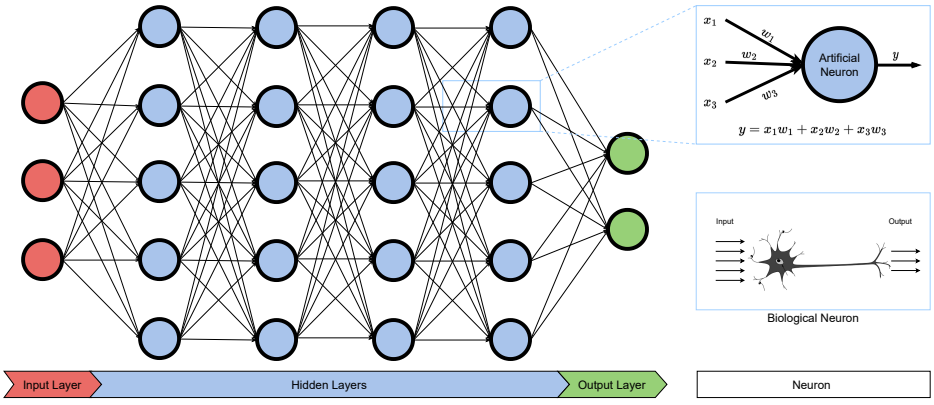


Figure 2.5: Overview of a Multi-Layer Perceptron (left side), and a comparison between an artificial and biological neuron (right side).

DL utilizes neural networks (NNs) with multiple layers to learn complex representations of data. A NN is a type of architecture that imitates biological neurons found in animal brains. One example of a NN is a

Multi-Layer Perceptron (MLP), also called a fully connected feed-forward network. On the left side of Fig. 2.5, a general MLP structure is illustrated. The networks comprise node layers, including an input layer, several hidden layers, and a final output layer. Each node in a layer is connected to all the other nodes in a successive layer. Every node has an associated weight and threshold. The weights are adjusted during the learning process to improve the performance of the NN. The threshold is used to decide whether or not the node's output will be sent to the next layer.

The key idea behind DL is to use a NN, where each layer learns increasingly abstract representations of the input data. These networks are trained using a large dataset, typically through a process known as backpropagation, which involves computing the gradient of the network's output for its parameters and adjusting those parameters and the nodes' weights using an optimizer (Sec. 2.2.4), to minimize a loss function (Sec. 2.2.3).

One of the main advantages of DL, compared to an ML algorithm, is the ability to automatically learn features from raw data without requiring explicit feature engineering, making it particularly useful for tasks where human experts, such as in computer vision or speech recognition, may need help identifying the relevant features.

DL has been applied to a wide range of applications, including image and video classification [38, 65, 81, 82], object detection [83, 84], speech recognition [68, 85], natural language processing [61, 69, 86, 87], and drug discovery [88, 89]. In many cases, DL has achieved state-of-the-art performance on these tasks, surpassing previous approaches that relied on hand-engineered features or less sophisticated ML methods.

2.2.1 Artificial Neuron

The single components of a NN present a similar design with the structure of a biological neuron (right side of Fig. 2.5). A biological neuron is an electrically excitable nerve cell in the brain that receives many inputs from other neurons via dedicated connections called synapses. A biological neuron is composed of various parts:

- *Soma*: the body of the neuron, which contains the cell nucleus,
- *Dentrites*: branches of the soma in charge of receiving input signals from other neurons,

- *Axons*: a long projection that transmits nerve signals from the soma to the dendrites of another neuron cell via synapses through some terminals.

This nerve cell has been described as a simple logic gate with binary output; signals arrive at the dendrites, are processed into the soma, and eventually, an output signal is generated that will be passed on by the axon [90]. Similarly, an artificial neuron receives a set of inputs x_i and weights w_i from a previous neuron plus a biased value $\mathbf{b}$; processes them through a nonlinear function called activation function θ (Sec. 2.2.2), and generates an output y that will be forwarded to another artificial neuron. The output of a neuron can be expressed with the following equation:

$$y = \theta(\sum_i x_i w_i + \mathbf{b}) \quad \forall i \quad \text{in the previous neuron}$$

2.2.2 Activation Function

An activation function plays an essential role in a NN by introducing non-linearity into the output of a neuron or a layer of neurons. The nonlinearity is essential because it enables NNs to learn complex relationships between inputs and outputs. Common activation functions used in this thesis are: Rectified Linear Unit (ReLU) [91], Leaky ReLU [92], Exponential Linear Unit (ELU) [93], Softmax [94], and Sigmoid [95]:

- ReLU provides a straightforward nonlinear transformation; it is one of the most common choices for an activation function due to its speed during training and performances [96]. The function is the maximum between zero and a given element x. With this setting, sometimes the neurons can be pushed to be set all to zero, creating a vanishing-gradient problem [97].

$$\text{ReLU}(x) = max(0, x), \quad [0, \infty]$$

- Leaky ReLU is an activation function almost identical to ReLU; there is only a small difference between them: if x is less than zero, the function allows a small positive constant (α), which helps mitigate the vanishing-gradient problem but decreases the performances [92].

$$\text{Leaky ReLU}(x) = \begin{cases} \alpha x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}, \quad [-\infty, \infty]$$

- ELU is a modified version of the ReLU activation function that allows negative values; the hyperparameter α controls the output for negative x. The ELU function avoids the vanishing-gradient problem. It has been shown that ELU improves the model's accuracy and reduces the training time. However, it is more computationally expensive than other activation functions [93].

$$\text{ELU}(x) = \begin{cases} \alpha\left(e^x - 1\right) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}, \quad [-\alpha, \infty]$$

- The Softmax function converts a vector of $\mathcal{C}$ real number (where $\mathcal{C}$ is the number of classes) into a probability distribution of $\mathcal{C}$ possible outcomes; thus the outputs are always between zero and one, and will sum to one. This function is normally used as the last activation function in a NN for normalizing the output into a probability distribution of classes $\mathcal{C}$.

$$\text{Softmax}(x) = \frac{e^{x_i}}{\sum_{c=1}^{\mathcal{C}} e^{x_c}}, \quad \mathcal{C} \geq 1 = \text{number of classes} \quad [0, 1]$$

- The Sigmoid, as the Softmax, converts any input value into a single value between zero and one; however, the sum of the values is not one. The Sigmoid adds nonlinearity to the output of a node. This activation function suffers, as the ReLU, of the vanishing-gradient problem [97].

$$\text{Sigmoid}(x) = \sigma(x) = \frac{1}{1 + e^{-x}}, \quad [0, 1]$$

2.2.3 Loss Function

A NN aims to minimize the output of the loss function $\mathcal{L}$ during the training phase to generate optimal predictions during the inference phase. A function $\mathcal{L}$ measures how well a NN performs on a given task. To tackle the main objective of this thesis (O_1), we implemented and experimented with the following loss functions: Dice Coefficient loss (DCL) [98], soft DCL (SDCL) [99], Focal Tversky Loss (FTL) [100], and weighted categorical cross-entropy loss (WCCL) [101]

In the remaining of the thesis, we denote $\hat{y}_{i,c} \in [0, 1]$ as the probability of the ith pixel to be as class c in the predicted image $\hat{y}$; $y_{i,c} \in \{0, 1\}$ as

the pixel i with class c in a ground truth image y, and $M \times N$ is the total number of pixel in an image.

The DCL is a widely used measurement to calculate the similarity of two images; it is based on the Dice Coefficient (DC) [102]:

$$\mathrm{DCL}(\hat{y}, y) = \sum_{c}^{\mathcal{C}} \left(1 - \frac{2\sum_{i}^{M\times N} \hat{y}_{i,c} y_{i,c} + \epsilon}{\sum_{i}^{M\times N} \hat{y}_{i,c} + \sum_{i}^{M\times N} y_{x_i,c} + \epsilon}\right)$$

where ϵ is a small number that provides numerical stability to prevent division by zero.

The SDCL is a modified version of the DC mainly used in medical domains where the classes to predict are highly unbalanced due to a small region of interest compared to the background of the scans [99]. It can be written as:

$$\mathrm{SDCL}(\hat{y}, y) = \sum_{c}^{\mathcal{C}} \left(1 - \frac{2\sum_{i}^{M\times N} \hat{y}_{i,c} y_{i,c} + \epsilon}{\sum_{i}^{M\times N} \hat{y}_{i,c}^2 + \sum_{i}^{M\times N} y_{i,c}^2 + \epsilon}\right)$$

A limitation of the DCL and SDCL is that they equally weigh false positive (FP) and false negative (FN) values. In practice, it might results in segmentation outcomes with high precision but low recall [100]. The FTL function was created to overcome this issue. The FTL is a generalization of the DCL, which allows balancing FP and FN values flexibly. The FTL function for a specific class c is defined as:

$$\mathrm{TCL}_c(\hat{y}, y) = \sum_{c} (1 - \mathrm{TI}_c(\hat{y}, y))^{1/\gamma}$$

Where $\gamma \geq 1$ is a hyperparameter value that forces the loss function to have more focus on less accurate predictions that have been misclassified [100], while TI_c is the Tversky index for a class c defined as:

$$\mathrm{TI}_c(\hat{y}, y) = \frac{1 + \sum_{i=1}^{M\times N} \hat{y}_{i,c} y_{i,c} + \epsilon}{1 + \sum_{i=1}^{M\times N} \hat{y}_{i,c} y_{i,c} + \alpha \sum_{i=1}^{M\times N} \hat{y}_{i,\hat{c}} y_{i,c} + \beta \sum_{i=1}^{M\times N} \hat{y}_{i,c} y_{i,\hat{c}} + \epsilon}$$

where $\hat{y}_{i,\hat{c}} = 1 - \hat{y}_{i,c}$ and $y_{i,\hat{c}} = 1 - y_{i,c}$ are the probability that the ith pixel is not of class $c \in \mathcal{C}$. The hyperparameters α and β control the trade-off between precision and recall. It is important to note that in the case of $\alpha = \beta = 0.5$, the TI simplifies to the DCL [100].

The WCCL is a variation of the categorical cross-entropy loss function, and it is used for classify datasets with imbalanced classes assigning different

weights to each class based on their frequency in the training data, giving more importance to the underrated classes [103]. The WCCL loss can be written as follows:

$$\text{WCCL}(\hat{y}, y) = \sum_{c}^{\mathcal{C}} \sum_{i}^{M \times N} (y_{i,c} \log \hat{y}_{i,c}) \cdot (w_{i,c} y_{i,c})$$

where $w_{i,c}$ corresponds to the weight of the ith pixel for a class $c \in \mathcal{C}$.

2.2.4 Optimizer

The optimizer is a crucial component of a NN: it has the responsibility to update the weights and biases $(w_i, \mathbf{b})$ of every layer in the network in a way that minimizes a loss function $\mathcal{L}$ (Fig. 2.3). In this thesis, we have utilized two optimizers for the proposed architectures:

- Stochastic Gradient Descent (SGD): It is an algorithm that updates the weights and biases $(w_i, \mathbf{b})$ using backpropagation utilizing the gradient of a loss function $\nabla\mathcal{L}$; the following equation shows how to update a weight w_i using the previous weight w_{i-1} and the gradient of the loss function $\nabla\mathcal{L}$ for the weight w_{i-1}; η is the learning rate of the SGD:
$$w_i = w_{i-1} - \eta \nabla \mathcal{L}(w_{i-1})$$
- Adaptive Moment Estimation (Adam) [104]: It is an optimization algorithm, commonly used to train NNs. Adam calculates an exponential moving average of the gradient and the squared gradient to update the weights and biases $(w_i, \mathbf{b})$. This optimizer has been demonstrated to achieve good results faster than other optimizers [104].

2.2.5 Regularization

Regularization is a common technique used in NN to prevent overfitting, a well-known problem in ML [105]. Overfitting occurs when a NN can fit and learn the training data well, but it cannot generalize unseen data (a.k.a data in a test set). Regularization techniques, like L1 and L2, help prevent this issue by adding a penalty term to the loss function to encourage the model to learn more generalized representations [106–108]. Dropout is another technique where some neurons are randomly dropped (a.k.a. made inactive) in each training iteration. We used various regularization techniques in the thesis to create more generalized models: L1, L2, Dropout, and Early stopping regularization.

L1 regularization The L1 regularization [106], also known as the Lasso (Least Absolute Shrinkage and Selection Operator) regression, adds a penalty to the loss function proportional to the absolute value of the weights of the NN, encouraging learning sparse representations: $\text{L1} = \lambda \sum |w|$, where λ is a regularization strength and w is the weights vector. The minimization of a loss function $\mathcal{L}$, together with the L1 regularization, encourages some of the coefficients to be precisely zero, effectively performing feature selection and reducing the complexity of the model.

L2 regularization The L2 regularization [107, 108], also known as the Ridge regression, adds to the loss function a penalty term proportional to the square of the weights of the NN: $\text{L2} = \lambda \sum w^2$. The L2 regularization encourages the model to learn smaller weight values. The objective of this function is to prevent overfitting and improve the generalization of the model by shrinking the magnitude of the coefficients.

Dropout The Dropout regularization [109, 110] is a technique implemented to prevent overfitting and reduce the complexity of the network. During training, this technique "drops" some of the neurons in the model to create a more robust architecture that does not rely too heavily on a single neuron. A dropout layer is usually placed between other layers in a NN; they have no associated weights.

Early stopping The overfitting problem can arise when training a model for too long. The Early stopping regularization [111] is a technique used to improve the generalization performances of the network. This technique stops the training process of a NN before it overfits by monitoring the validation loss for a predefined amount of epochs. The process stops the training phase if the validation loss does not decrease after a certain time.

2.2.6 Skip Connections

Skip connection refers to a path that connects the output of some layers in a NN to the input of a subsequent layer in the network, which might not be directly adjacent. It has been validated that these additional paths benefit the model's training [82, 112]. The most common ways to use skip connections in non-sequential layers are:

- **Concatenation**: The output of a layer is directly connected with the input of a (non-)sequential layer to pass information across the NN. Famous examples of this approach are U-Net [112], where the

skip connections helped recover fine-grained details in the predictions, and DenseNet [113], where each layer of the network is connected to every other layer, alleviating the vanishing-gradient problem and reduced the number of parameters to be trained.

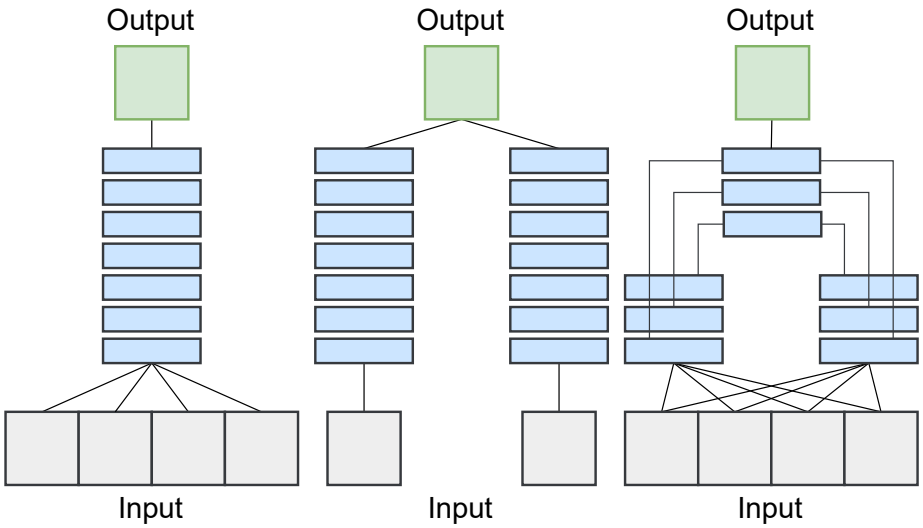
- **Addition**: The output of a layer is added to the output of a successive layer in the NN. This summation serves to maintain the gradient and mitigate the vanishing problem. It is the main idea behind the famous ResNet architecture [82].

2.2.7 Fusion of inputs

A fusion approach refers to integrating multiple input sources, or features, to improve the robustness of a model's predictions [114, 115]. This approach can be performed in several ways and with different input dimensions. One common approach is called "Feature-level fusion", where multiple inputs extracted from different sources can be combined into a single feature representation. The feature-level fusion can be divided into early, late, and slow. Fig. 2.6 visually represents the various feature-level fusion approaches.

Early Fusion In this approach (Fig. 2.6a), all the information are immediately concatenated before being fed to a NN; then, the connected input is analyzed and reduced during the first layer of the NN. Early fusion is practical when the features from different sources are complementary and can be combined. Couprie *et al.* [116] firstly introduced this concept for a DL model: they concatenate RGB and depth channels from images before providing them to a segmentation architecture.

Late Fusion For this fusion approach (Fig. 2.6b), the various inputs are processed independently and concatenated in the high-dimensional, or just before the output layer; this approach allows the feature to be processed separately by distinct models, and later merge with a fused algorithm. Late fusion can be feasible when the inputs are not complementary and cannot be combined at the input level. A disadvantage of this approach is its high computational cost since each input is processed in a separate architecture. The late fusion approach was first introduced by Gupta *et al.* [117] where two CNNs extract RGB and depth features which are combined by a classifier in a later stage.



(a) Early Fusion **(b)** Late Fusion **(c)** Slow Fusion

Figure 2.6: The feature-level fusion approaches: (a) Early Fusion, (b) Late Fusion, and (c) Slow Fusion. A gray box represents the input, green boxes are the outputs of a NN, while a blue box indicates a general intermediate layer in the NN.

Slow Fusion The slow (or gradual) fusion approach creates a balanced mix between the early and late fusion (Fig. 2.6c): the input is slowly combined throughout the entire network such that higher layers can access progressively more global information in both spatial and temporal dimensions. In other terms, the input is processed by different models, and the output of these models is combined at various stages of the architecture. Karpathy *et al.* [114] introduced a slow fusion model for video classification and compared it with early and late fusion architectures; they concluded that a slow fusion approach performs better than the alternatives.

2.2.8 Convolution Neural Network

Adopting a MLP structure can be problematic for image processing and computer vision tasks because its architecture does not consider spatial information in its layers, which is fundamental for images and videos. A structure called Convolution Neural Network (CNN) solved this issue. CNNs are a class of NN.

Creating a CNN was inspired by how the human brain's visual cortex works when recognizing objects [118]. The CNN was first developed by LeCun *et al.* to classify handwritten digits from images [119]. This type of network is primarily used for analyzing and processing images and other types of multidimensional data, such as audio and video. CNNs are designed to learn hierarchical feature representations from input data automatically. Typically, a CNN comprises a series of convolutional and pooling layers.

Convolutional Layer

A Convolutional layer performs a discrete convolution operation on the input I using a learnable kernel $\mathcal{H}$. Let define $I(x, y) \in \mathbb{R}^2, \mathcal{H}(w, h) \in \mathbb{R}^2$ as a 2D image and a 2D kernel. The (x, y) and (w, h) indicate the width and height of the image and the kernel, respectively. The following equation defines the mathematical definition of a 2D convolution operation between a I and $\mathcal{H}$:

$$g''(x, y) = \mathcal{H}(w, h) \circledast I(x, y) = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \mathcal{H}(i, j) I(x + \widetilde{w} - i, y + \widetilde{h} - j)$$

where $\circledast$ is the convolution operation. $\widetilde{h} \equiv \lfloor \frac{1}{2}(h-1) \rfloor, \widetilde{w} \equiv \lfloor \frac{1}{2}(w-1) \rfloor$ correspond to the half-width, and half-height of the kernel $\mathcal{H}$. Convolutional operations can be performed with higher dimensional input and/or kernel (3D, 4D, etc., ...). Paper IV describes in detail these higher dimensional operations.

During a convolutional layer, the network applies a set of kernels to the input data, followed by a nonlinear activation function (Sec. 2.2.2). It generates a series of feature maps that capture local patterns and features in the data. The number of pixels shifted during each kernel application $\mathcal{H}$ in an input image I is defined as strides. A stride>1 implies smaller feature maps as output due to larger movements when applying the kernel $\mathcal{H}$.

Pooling Layer

A pooling layer reduces the size of the feature maps in the spatial dimension, i.e., it provides a downsampled version of the feature maps using an operation on each input data channel. Additionally, this step helps make the model more robust to variations in the position of features in the input image The operation is performed to reduce the dimensionality and spatial

complexity of the input. The spatial dimension can also be reduced by doing convolution layers (Sec. 2.2.8) with stride>1.

Two types of pooling layers (Fig. 2.7) were utilized in the NNs described in this thesis: *average-pooling* and *max-pooling*. An *average-pooling* calculates the average value from a region of the input data and generates a new downsampled feature map. In contrast, the *max-pooling* operation selects the maximum value from the input data's region and creates a new downsampled feature map.

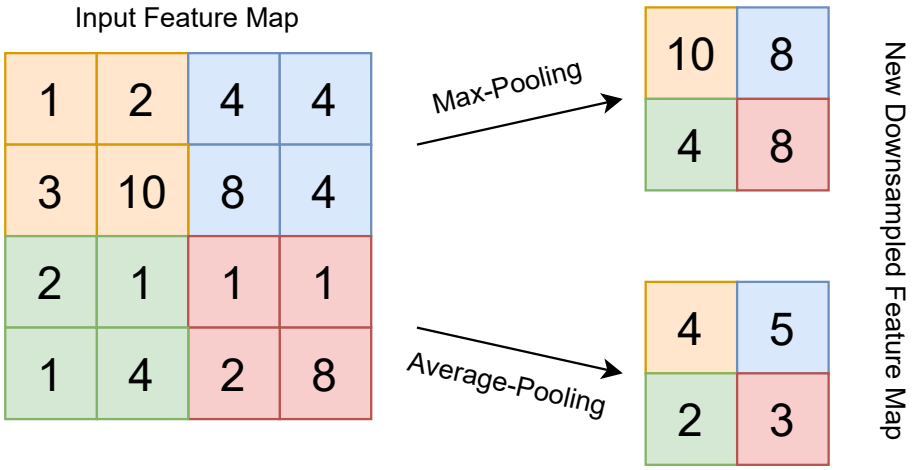


Figure 2.7: Graphical representation of the *max-pooling* and *average-pooling* operations.

2.3 Image Segmentation

In image processing and computer vision, the image segmentation problem involves dividing an input image I into an arbitrary number of non-overlapping sub-regions based on their inherent characteristics, such as color, texture, shape, or brightness [120].

Several algorithms have been proposed to solve the image segmentation task: thresholding is one of the simplest and most commonly used techniques. It involves dividing the image based on a threshold value: pixels with intensity above or equal the threshold value are set to one region. In contrast, the other pixels (the ones with values below the threshold) are assigned to a different region.

More complex algorithms are based on a clustering approach, where pixels are grouped based on predefined characteristics. Simple Linear Iterative Clustering (SLIC) [121] and Felzenszwalb's algorithm [122] are two standard methods used in the thesis for performing image segmentation. The SLIC algorithm divides the image into a set of superpixel regions (small areas of pixels with similar intensity and color); subsequently, the algorithm refines the borders of the superpixels to better align with the edges of objects in the image. Felzenszwalb's algorithm is a graph-based method that generates superpixel regions based on the dissimilarity measures of the image's pixels. The pixels serve as the nodes in the graph, and the connections between these nodes represent the degree of similarity between the pixels. Furthermore, ML and DL architectures have been proven to be very effective for image segmentation problems [82, 112, 123–126].

Depending on the specific task to solve, the image segmentation process can be divided into three types: semantic, instance, and panoptic segmentation.

Semantic Segmentation: The task involves labeling each pixel in the image I with a corresponding class label. The semantic segmentation can be resolved via a pixel-by-pixel classification where each pixel in the image I is classified into one of the predefined classes. Many semantic segmentation DL frameworks were proposed, from the CNN first utilized by Long *et al.* [127] in 2014, to more classical and known architectures, like U-Net [112], a U-shaped encoder-decoder structure developed for accurate semantic segmentation for biomedical images, SegNet [123], an encoder-decoder architecture that uses max pooling for upsampling the encoder and produces semantic segmentation, and DeconvNet [124], a semantic segmentation algorithm that uses a network composed of deconvolution and unpooling layers.

Instance segmentation: The problem aims to identify and differentiate each unique instance of an object in an image. The idea is to classify each object in an image I. Thus, it not only assigns a class for each pixel but also creates separate segmentation for every class instance. This type of segmentation is beneficial for object detection tasks, where a bounding box is created around the detected object to indicate the object's location [128]. Examples of DL architectures for the instance segmentation tasks are VGGNet [125], a simple network for object detection that uses small 3x3 filters, pooling layers, and fully connected layers, ResNet [82] a deep NN architecture that implements skip connection to address the vanishing-gradient

problem, achieving state-of-the-art performances, and Inception [126] a deep CNN that utilizes a combination of convolutional layers with different kernel sizes and pooling layers to achieve good performances for the instance segmentation task.

Panoptic segmentation: It combines both segmentations previously described. This segmentation assigns two labels to every pixel in an image: a semantic label and an instance identifier. The semantic label defines the pixel's class, while the instance identifier distinguishes between different class instances. Architectures that tackle this type of segmentation are UPSNet [129], a network that combines two architectures to solve the instance and semantic problems separately, EPSNet [130], an efficient panoptic segmentation architecture that unifies instance and semantic segmentation in one task using an early fusion approach, and VPSNet [131], an architecture that performs panoptic segmentation on videos; the NN jointly predicts object classes, bounding boxes, masks, and semantic segmentation.

Part II

Clinical Application

Chapter 3

Medical Application

The following chapter introduces what a cerebral stroke is and the current clinical routine for this medical condition. Additionally, an understanding of the various diagnostic imaging techniques used for assessing a stroke is presented with an overview of the related work for segmenting ischemic areas based on the described imaging techniques.

3.1 Cerebral Stroke

A cerebral stroke occurs when the blood flow to an area of the brain is interrupted or reduced due to a blockage (a.k.a. ischemic) or a ruptured blood vessel (a.k.a. hemorrhagic). Brain cells start to die within minutes when the blood flow is occluded. The focus of the thesis will only be on ischemic stroke (IS). The majority of the cerebral strokes are due to ischemia (around 80%), whereas the remaining are due to hemorrhage [4, 5, 132]. According to the National Institute of Neurological Disorders and Stroke (NINDS) [1], blockages that can cause an IS stem from three conditions:

- *Thrombosis*: a clot develops within a brain's blood vessel and grows large enough to impair blood flow.
- *Embolism*: a clot moves from another part of the body (such as the heart or a diseased artery in the chest or neck) into a narrower artery in the neck or brain.

[1] https://www.ninds.nih.gov/

- *Stenosis*: an artery in the brain or neck narrows.

The longer the brain is deprived of blood flow, the greater the damage and the more severe the long-term effects can be. Neuroradiologists must quickly diagnose a patient's condition and propose the proper treatment. Early stroke detection is fundamental: the earlier the diagnosis, the better the outcome since the recommended treatment window for intravenous thrombolysis is 4.5 hours [133] from symptom onset and 12h for endovascular thrombectomy [13, 134, 135]. Under the guidance of advanced imaging, the treatment windows have been extended during the past few years. Therefore, fast and accurate perfusion imaging analysis gains importance in hospitals treating acute stroke patients [13, 136, 137].

3.2 Clinical routine

A cerebral stroke is the second most common cause of death among adults [138]. This severe medical condition can cause one or more symptoms in a patient affected by it: speech impediment, confusion, paralysis, vision loss, and walking [139]. Time is essential in dealing with this neurological disorder: the phrase "time is brain" emphasizes the need for fast and rapid treatment [137, 140]. Every minute counts when treating a stroke because millions of brain cells die quickly when deprived of oxygen and nutrients [9]. A patient can lose almost 2 million neurons, 14 billion synapses, and approximately 12 km of nerve fibers every minute from when the stroke happens [141].

According to the European Stroke Organization, standard clinical routines for diagnostic imaging in AIS patients, as of today, are CT and MRI [15]. In an AIS stage, CT and CTP have demonstrated to be a fast, affordable, and reliable tool for evaluating diagnosis and prognosis [33, 53, 142–144]. MRI with Diffusion-Weighted imaging (DWI) is more sensitive for detecting stroke; however, it is more prone to motion artifacts [145, 146]. Some hospitals worldwide use MRI as the first imaging modality tool for AIS patients.

When a call from a suspect of a cerebral stroke is received at Stavanger University Hospital, a precise and fast treatment protocol is performed:

I. Firstly, a pre-notification of the in-hospital stroke treatment team through a dedicated "stroke thrombolysis" alarm is executed [147, 148].

II. Secondly, during the transport of the patient to the hospital with an evaluation of the medical history, a measurement of the blood pressure, and the insertion of a peripheral venous cannulation [147, 148].

III. Then, as soon as the patient arrives at the facility, a thorough examination of the subject is completed; the analysis included the assessment of the National Institutes of Health Stroke Scale (NIHSS), a quantitative analysis of the stroke severity by measuring neurological impairment caused by the stroke [149–151]. NIHSS is used for clinical evaluation and as an outcome measure in clinical trials.

IV. Subsequently, diagnostic imaging of the brain, such as CT and MRI, are taken to properly assess if the patient has a cerebral stroke, the severity of the stroke, and to evaluate the ischemic areas (penumbra and core). CT is the most commonly used diagnostic tool, however MRI can also be used as a first-line diagnostic tool, especially in patients with suspected wake-up stroke or stroke with unknown symptom onset time [147, 152, 153].

V. After that, a (neuro)radiologist evaluates the images while a neurologist executes the clinical evaluation. A possible treatment solution is decided (intravenous thrombolysis or mechanical thrombectomy), depending on the level of stroke severity and/or vessel occlusion.

VI. Finally, follow-up images are acquired during the first days after hospital admission (in most patients within 24h) [147]. The follow-up images commonly determine the exact FIA(s) distribution and evaluate potential differential diagnoses. Supplementary CT or MRI examinations are also performed in patients with clinical deterioration [147].

3.2.1 Computed Tomography (CT)

The etymology of the term tomography derived from the Greek words *tome* and *graphein*, where the first word can be translated with “slice”, while *graphein* is a direct translation of the verb “to write”. This fundamental machine credits its foundation to a vast number of innovations in various fields, some of them not even related to each other, but combined, they gave the possibility to create the first CT scanner, contributing enormously to the improvement in the medical field [154].

The first CT scanner was developed by Godfrey Hounsfield, an English electrical engineer, during his work at EMI (Electric and Musical Industries Ltd) in 1967. The first brain scan of a human patient was performed on the 1st of October 1971 [155]. The selected patient was a woman suspected of a brain tumor; the images generated with the new machine highlighted the area where the cancer was spreading (Fig. 3.1).

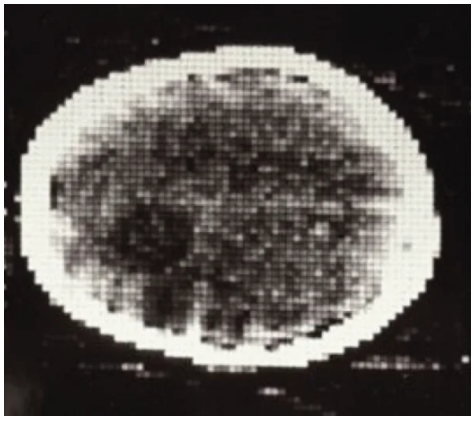


Figure 3.1: First image of a brain tumor using a CT scanner. Source: [156], CC BY.

A CT scan is a rapid medical procedure that uses a combination of many X-ray measurements taken at different angles concerning the source that is analyzed to produce a series of cross-sectional images of the inside of a particular area of study in the subject's body. Modern CT scans combine the latest innovation in mathematical algorithms with the power of x-rays. Nowadays, a CT scan is currently one of the basic techniques in medical diagnosis due to its importance, availability, easy usage, and fast response [49–51].

A CT study is saved and stored in a DICOM (Digital Imaging and Communications in Medicine) format, a standard commonly used for storing and transmitting medical images from multiple manufacturers, managed by the National Electrical Manufacturers Association (NEMA)[2]. DICOM files consist of a header containing meta information on the subject, the imaging protocol, and one or more images stored in a 16-bit format. The DICOM header contains information to encode the image intensity pixels in Hounsfield Unit (HU), a linear transformation to describe radiodensity efficiently in CT scans [157].

[2] https://www.nema.org/

It is possible to divide CT examinations in Non-contrast CT (NCCT) and Contrast CT. The difference relies on the usage or not of a contrast-enhanced agent injected into the body during image acquisition.

3.2.2 Non-contrast Computed Tomography (NCCT)

NCCT is the first diagnostic imaging tool used in patients suspected of a stroke, with Contrast CT adopted right after. The NCCT can also be implemented as a follow-up imaging tool; however, it is less sensitive than MRI for FIA delineations [158, 159].

3.2.3 Computed Tomography Angiography (CTA)

CT Angiography (CTA) is a standard Contrast CT examination used for clear visualization of arteries and veins. The images are acquired after injecting a radio-contrast material that enhances blood vessels in the human body. CTA can be adopted for assessing the blockage in the arteries in AIS patients [160].

3.2.4 Computed Tomography Perfusion (CTP)

A CTP study is a 4D spatio-temporal examination of an area of the brain to assess the blood flow in the region [18, 161]. The CTP study collects a series of 3D scans over the same portion of the brain during intravenous administration of 1 or 2 boluses of an iodinated contrast agent over time [142]. The agent changes the brain tissue's density, making it easier for analysis. This 4D spatio-temporal examination contains a vast amount of data. Detecting any neurological condition can be challenging and time-consuming for neuroradiologists. It has been demonstrated that a CTP scan adds relevant clinical information in patients suspected of AIS compared with NCCT and CTA scans [142–144].

3.2.5 Parametric Maps (PMs)

It is impractical for neuroradiologists to study directly CTP scans as the relevant information lies in the differences and the development over time. Therefore, it would be necessary for them to look at many images simultaneously, which can be challenging and time-consuming. Instead,

they rely on a set of Parametric Maps (PMs) summarizing the development over time, generated from the 4D CTP scan as a diagnostic tool.

PMs are generally calculated in two steps: the first consists of acquiring time density curves, and the second involves extracting specific information from these curves. Time density curves are calculated during the passage of the contrast agent in the brain tissue. Fig. 3.2 displays an example of a time density curve, highlighting the various measurements for calculating the PMs. These curves map the intensity of every image's pixel over the injection of the contrast agent [18]. The x-axis shows the elapsed time from the start of the CTP scan (in seconds); the y-axis displays the intensity level measured in HU.

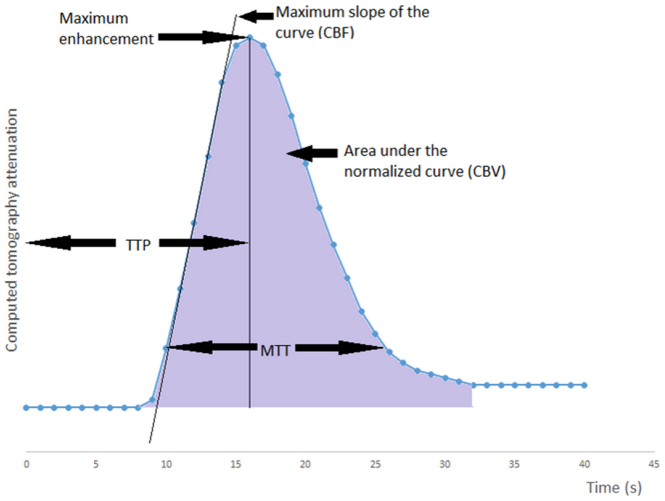


Figure 3.2: An example of time density curve; it shows the different measurements for creating the PMs. The figure is reprinted in unaltered form from [18] under the consensus of the author.

These PMs highlight the spatio-temporal variations in the brain tissue, allowing neuroradiologists to diagnose and plan treatments rapidly [18]. Fig. 3.3 shows a set of PMs, emphasizing different aspects of a time density curve.

Cerebral Blood Flow (CBF) Cerebral Blood Flow (CBF) represents the volume of blood flow per unit of brain tissue per minute. It is measured in milliliters of blood per minute per 100g of brain tissue (ml/100g/min).

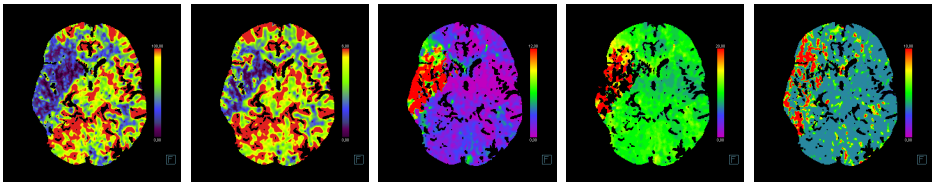
In a patient without any ischemic problem, the value of CBF is around 50-60ml/100g/min [162]. Fig. 3.3a gives a general overview of this PM.

Cerebral Blood Volume (CBV) Cerebral Blood Volume (CBV) is defined as the blood volume 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 around 4-5ml/100g; a blood volume below 2.5 ml/100g might indicate infarcted tissue [163]. Fig. 3.3b shows a brain slice's CBV. Both CBF and CBV are helpful maps to assess ischemic core [22, 164].

Time-to-Maximum (TMax) The Time-to-Maximum of the residue function (T_{Max}) displays the time a contrast agent takes to reach and traverse areas of the brain. T_{Max} is measured in seconds (s), and it is acquired with deconvolution-based post-processing steps from CTP scans [23]. An example of T_{Max} map is given in Fig. 3.3c.

Time-to-Peak (TTP) Time-to-Peak (TTP) represents the time from the start of the contrast injection to the peak of enhancement in the tissue. It is measured in seconds (s). Together with T_{Max}, this map helps define ischemic penumbra [25, 164]. Fig. 3.3d displays an example of this PM.

Mean of capillary Transit Time (MTT) Mean of capillary Transit Time (MTT) (example in Fig. 3.3e) measures the average time that blood spends within a determinate volume of capillary circulation. It is measured in seconds (s). MTT can be also calculated as the division between CBV and CBF [165, 166].



(a) CBF **(b)** CBV **(c)** T_{Max} **(d)** TTP **(e)** MTT

Figure 3.3: Parametric maps of a single brain slice from a patient in our dataset: Cerebral Blood Flow (a), Cerebral Blood Volume (b), Time-to-Maximum (c), Time-to-Peak (d), and Mean of capillary Transit Time (e).

3.2.6 Diffusion-Weighted imaging (DWI)

DWI is a specific MRI sequence that estimates the rate of movement of water molecules in the extracellular space in the tissue. DWI has been considered the gold standard for IS estimation [22, 23, 167, 168]. However, MRI machines are not always available, and some patients might have contraindications for this image modality [169]. For these reasons, MRI is usually performed within the first days after an AIS, for patients without contraindications, as a follow-up procedure [16, 23, 170]. Fig. 3.4 displays a DWI scan for a patient with a FIA located on the left hemisphere.

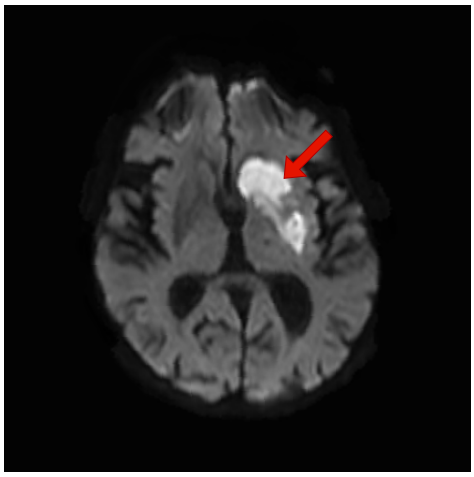


Figure 3.4: Example of a DWI scan from a subject in our dataset. In this patient, the FIA is located on the left side in the vascular territory of the middle cerebral artery (pointed by a red arrow).

Lately, various researchers have contested DWI as the gold standard for IS in clinical trials [19, 24, 171]. It has been proven that DWI cannot accurately differentiate FIA from salvageable tissue [19, 171]. Moreover, some estimation of the FIA can be reversed if a DWI scan is completed in the early stage of an IS [19, 159, 172, 173].

3.3 Related work

Over the last decades, different approaches (semi and fully-automatic) and imaging modalities (PMs or CT scans) were tested to find the most suitable

method to segment the FIAs and/or both the ischemic regions, core and penumbra.

3.3.1 Semi-automatic Approaches

A semi-automatic approach combines manual and automatic steps to achieve a specific goal. In semi-automatic methods, static human input is commonly introduced (an initial seed in a region-growing algorithm or a threshold value), causing the approach to be heavily dependent on the input. Several researchers have used region-growing techniques algorithms for delineating stroke lesions in CT scans [174] and for calculating volume infarction from MRI [175]. Their main objective was to delineate FIAs, not to segment the ischemic core and penumbra regions. Both research works [174, 175] adopted post-processing steps to improve the labeled images generated with their semi-automatic approach. Their results are promising, but a semi-automatic approach is deprecated. Contin *et al.* proposed a method to delineate both ischemic regions using PMs derived from CTP scans as input [176]. The method is based on a region-growing algorithm where a user manually inputs an initial seed. The manual input of the seed point limits the usability of their approach.

Semi-automatic thresholding approaches have been explored and implemented in clinical routines. Several studies have experimented with various measurements on a set of the derived PMs to establish the best combination of values for FIA estimation [22–25, 33, 168, 177, 178]. Most of these research works evaluated their method by testing the mismatch between areas from CTP images and the corresponding follow-up DWI as the FIA ground truth. We only found one research [25] that used NCCT images, acquired 5 to 7 days after the initial study, as a ground truth base while evaluating the ischemic lesions.

Table 3.1 displays some of these studies with their discoveries. There is no consensus on which threshold value to use for the penumbra, core, and FIA or which parametric map is more suitable [22, 33, 53, 54, 178]. As it is possible to exemplify from Table 3.1, the absence of consensus is mainly due to the different CT vendors utilized to generate the PMs and how each study defines the ground truth for the FIAs [24, 169]. Another possible reason for the lack of consensus is the ethnicity's influence in the patient's cohort during the threshold(s) selection [179]. Sex and age, on the contrary, showed no significant association with radiological predictors for IS. Thus, they cannot be used to refute the consensus's absence [180].

Furthermore, threshold values derived from follow-up images present some limitations in adequately defining the ischemic regions during the initial stages of an IS [19, 171]. Since the threshold values were compared with FIAs (delineated after treating the patient), they do not present a valid estimation of the ischemic regions before the treatment decision. Thus, they might not be the best alternative to help medical doctors with treatment decisions. Moreover, simple thresholding approaches might over-simplify the ischemic regions in a complex scenario as an AIS [181].

Table 3.1: Information about the dataset and the various research methods' threshold value(s) found in the literature.

Study	Year	Vendor for CT and PMs	# Patients	NIHSS (mean)	Stroke onset	Follow-up Images	Threshold	
							Penumbra	Core
Bathla *et al.* [177]	2019	Siemens	39	7	N.A.	$\leq$ 24h	T_{Max}>6s	CBF<20%
Wintermark *et al.* [22]	2006	Philips	130	15.3	$\leq$ 12h	$\leq$7d	MTT > 145%	CBV $\leq$ 2.0ml/100g
Campbell *et al.* [23]	2012	Philips	49	16.5	$\leq$ 6h	$\leq$1h	T_{Max}>6s	CBF<31% (with TTP >4s)
Cereda *et al.* [24]	2015	In-house	103	16	$\leq$ 8h	$\leq$3h	N.A.	CBF<38% (with T_{Max}>4s)
Bivard *et al.* [167]	2014	Toshiba	180	12	$\leq$ 6h	$\leq$24h	TTP>+5s	CBF<50%
Murphy *et al.* [25]	2006	General Electric	25	15.1	$\leq$ 7h	N.A.	CBF$\leq$ 25ml/100g CBV$\leq$ 2.15ml/100g	CBF $\leq$ 13.3ml/100g CBV$\leq$ 1.12ml/100g
Schaefer *et al.* [168]	2014	General Electric	55	14	$\leq$ 9h	$\leq$3h	N.A.	CBF$\leq$15% + CBV$\leq$30%
Lin *et al.* [16]	2014	Philips Toshiba	45	17	$\leq$ 9h	$\leq$ 2h	T_{Max}>6s	CBF$\leq$30%
Campbell *et al.* [17]	2011	Philips	98	N.A.	$\leq$ 6h	$\leq$ 7h	N.A	CBF<31%
Yu *et al.* [33]	2016	Siemens	82	12	$\leq$ 6h	24h	Delay time $\geq$ 3s	CBF$\leq$30%

3.3.2 Fully-automatic Approaches

In recent years, ML and DL architectures have been primarily investigated in medical applications with promising results [99, 112]. They have been proven to provide more suitable solutions for AIS than semi-automatic approaches, such as thresholding or region-growing algorithms [27, 29, 182].

Supervised Approaches Several researchers have implemented and analyzed ML approaches to predict the FIAs or the ischemic areas from various image modalities.

Qiu *et al.* used a ML-based algorithm for detecting infarction in patients with AIS using NCCT as input and compared it with follow-up DWI [27]. In their study, a random forest classifier is trained (using follow-up DWI as reference) for a FIA segmentation task. The model was finally refined with a post-processing step, which remove isolated components exploiting

morphological operations, showing promising results despite some limitations. Kemmling *et al.* proposed a multivariate generalized linear model with clinical data and PMs as input to quantify changes in FIAs [26]. The study aimed to determine the effect of the interval of treatment and degree of recanalization of FIAs, exploring the variables contributing to the infarct probability of each brain voxel. Qiu *et al.* developed two ML models, using DWI follow-up images as ground truth, to predict ischemic core and penumbra regions from multiphase CTA scans [28]. They developed two random forest classifiers to predict the ischemic areas separately. The prediction maps of the penumbra architecture were used as input for the primary model. Despite some limitations, they concluded that CTA images could be a valuable tool for predicting ischemic areas. Werdiger *et al.* [183] introduced a ML segmentation method, based on XGBoost, to delineate penumbra and core tissues, demonstrating the capabilities of this methodology over classic thresholding approaches. Four PMs are used as input for the model, establishing a moderate agreement in segmenting the ischemic areas on test images.

Due to the recent and promising results achieved by DL algorithms, a large number of studies have tried to use NNs, CNNs, and other approaches, to tackle the segmentation of ischemic areas in brain tissue using numerous image modalities. Kasasbeh *et al.* implemented a CNN-based approach using the PMs as input to classify the FIA using follow-up DWI as ground truth images [29]: this was the first study that used CTP on human subjects for the prediction of FIA, showing valuable results in the AIS setting, where CT is the primary imaging modality. Abulnaga *et al.* proposed a modified CNN with pyramid pooling and a focal loss, providing global and local contextual information stacking CT slices and PMs as input [184], segmenting FIAs showing better outcomes than classic U-Net [112] and V-Net [99] architectures. Clèrigues *et al.* [182] adopted a 2D asymmetric residual encoder-decoder to segment FIAs from CT scans and PMs. Their architecture is trained with 64×64 patches, limiting the spatial information given to the network. Soltanpour *et al.* developed two models using modified U-Nets [112], inputting a set of PMs derived from CTP for FIAs segmentation [185, 186]. Their models presented promising results but also stated that they needed to be more precise and could not be implemented in a real-case scenario. Rava *et al.* investigated a modified U-Net [112] with multiple PMs as input to segment FIAs showing encouraging results with a small dataset of patients affected by an IS [187, 188].

Recently, Amador *et al.* designed a framework based on Temporal

Convolution Network for predicting FIA from 4D CTP studies [30], using a dataset composed of AIS patients from multiple centers. However, they independently processed each 2D dataset slice due to memory constraints. In their recent work, Amador *et al.* proposed an extension of their previous model [30], where 3D+time tensors of the ipsilateral stroke hemisphere are used as input to predict only the FIAs [31]. Both models demonstrated the feasibility of using the entire 4D CTP sequence to improve segmentation's precision. Soltanpour *et al.* [189] utilized 4D CTP images to create 2D matrices in which each row is a voxel, and each column is a time point. These matrices are used as input for a model that shows encouraging results in differentiating healthy tissue from FIAs. In [190], Bertels *et al.* introduced an automatic DL-based architecture for FIA voxel-wise segmentation from CTP imaging. Their network extrapolates spatial information from the surrounding voxels and the contra-lateral side of the brain. They displayed promising outcomes during the validation stage but could have presented better results for a public test dataset (ISLES2018 dataset [55]). Zoetmulder *et al.* promoted a DL strategy to segment FIAs in patients affected by AIS evaluating the performances between posterior and anterior circulation strokes [191]. Robben *et al.* proposed a NN that predicts the FIAs directly from 4D raw CTP plus treatment parameters [158]. They showed supportive results in segmenting the FIA. However, a significant limitation in both their work [158, 191] is the quality of their ground truth since they relied on NCCT follow-up scans, which have been proven to be less reliable than other techniques [159]. Wang *et al.* used NCCT and CTA for identifying FIAs in AIS patients using a 3D U-Net architecture [192]. Their results demonstrated that adopting NCCT and CTA produced similar accuracy results to CTP scans.

All the studies above propose methods to segment the FIAs from healthy brain tissue. Although a precise understanding of the FIA is vital for decision treatment, an accurate knowledge of the extension of both the ischemic regions (penumbra and core) during the first stages of an IS is still needed and is crucial for selecting the best treatment strategy [25].

Semi-supervised Approaches Recently, semi-supervised models (Sec. 2.1.1) have gained momentum in the literature [41, 57]. A semi-supervised approach overcomes the limitation of fully-supervised methods relying on many labeled samples for training purposes. These architectures utilize a small number of labeled samples during their training phase. Due to

its recent momentum, few researchers have explored this technique for segmenting AIS patients.

Cao *et al.* evaluated the effectiveness of weakly supervised methods in detecting the location of AIS and hemorrhagic infarction lesions in DWI scans, showing the feasibility of this learning approach to avoid misdiagnosis [193]. Zhao *et al.* investigated a combination of K-means clustering and CNN model with a mixture of many weakly-labeled and few fully-labeled subjects for FIA segmentation using DWI scans [194]. They achieved auspicious segmentation results using a small set of fully-labeled patients for training. Zhao *et al.* proposed a method for segmenting AIS using a CNN [195]. Their strategy used many weakly-labeled subjects for training and few fully-labeled patients. The weakly-labeled subjects were used to extract semantic features from the CNN, while the fully-labeled patients were adopted for fine-tuning the segmentation results. Both their proposed methods [194, 195] drastically reduced the expense of obtaining many fully-labeled subjects in a supervised setting. Cao *et al.* analyzed the use of weakly supervised learning to identify FIAs in AIS patients directly from DWI scans [196]. They proposed a method that spatially locates small stroke lesions and hemorrhagic infarction lesions using residual neural and visual geometry group networks. Their approach generated probability maps for both stroke lesions and hemorrhagic infarctions, demonstrating the practicability of their method.

Chapter 4

Data Material

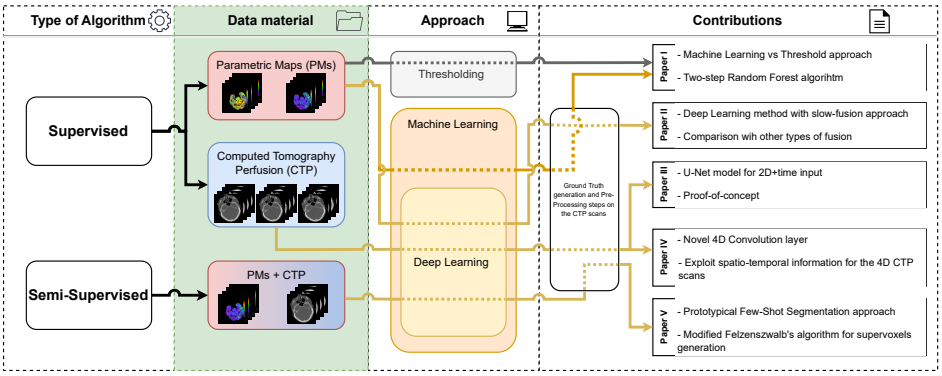


Figure 4.1: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy. The topic of this chapter (data material) is highlighted.

The following chapter describes the data material used in this thesis; CTP images and PMs act as the input to the proposed methods. Fig. 4.1 displays how the chapter's topic (the green highlighted section) is connected to the rest of the thesis. A focus on how the data were collected, the patient characteristics, how to divide the various groups, and ethical approval is shown.

4.1 Data Collection

Stavanger University Hospital annually admits approximately 450 patients with AIS. These patients, whose symptoms began within 4.5 hours before hospital admission, undergo routine examination using NCCT of the head (Sec. 3.2.2). Patients with a suspected acute stroke routinely undergo an examination with NCCT of the head. In patients deemed potentially suitable for treatment with intravenous thrombolysis and/or mechanical thrombectomy, this is followed by CTA (Sec. 3.2.3) and CTP (Sec. 3.2.4). In eligible patients with a suspected AIS, NCCT is followed by the administration of intravenous thrombolysis in the CT lab after contraindications are excluded [147]. This standardized approach, known as the CT stroke protocol, involves the use of NCCT, CTP, and CTA to image patients presenting with acute neurological symptoms that may indicate cerebral infarction [197]. A detailed description of the CT stroke protocol for the Stavanger University Hospital can be found in Sec. 3.2.

Table 4.1: Imaging protocol for three different CT-based dataset for AIS patients; NCCT, CTA, and CTP

	NCCT of the head	**CTA of the cerebral arteries**	**CTP**
Patient position	Head first, supine	Head first, supine	Head first, supine
Spiral/sequence	Spiral	Spiral	Spiral
kV	120	100	80
mAs	280	160	200
Rotation time (s)	1	0.28	0.28
Slice collimation	3 mm c 20 x 0.6 mm	0.6 mm c 128 x 0.6 mm	5 mm c 32 x 1.2 mm
Pitch	0.55	1.0	-
X-care	Yes	No	No
IV contrast	No	60 ml Omnipaque 350 mg I/ml + 40 ml NaCl	40 ml Omnipaque 350 mg I/ml + 40 ml NaCl
Flow rate	-	5 ml/second	6 ml/second
Start delay	-	4 seconds	4 seconds, ≥60 seconds after CTA
Scan direction	Caudocranial		

For the dataset used in the study, image acquisition was performed using two Siemens Somatom Definition scanners: the Somatom Definition Flash (installed in 2012) and the Somatom Definition Edge (installed in 2014). The CTP images were analyzed using the "syngo.via"[1] software from

[1] https://www.siemens-healthineers.com/digital-health-solutions/syngovia

Siemens Healthineers with default settings provided by the manufacturer. The software generated PMs, such as CBF, CBV, TTP, MTT, and T_{Max} (details in Sec. 3.2.5). Table 4.1 shows technical details about the acquisition protocols.

4.1.1 Group Division

The dataset $\mathcal{D}$ for this thesis consisted of CTP scans from 152 patients admitted to Stavanger University Hospital between January 2014 and August 2020. The patients were divided into three groups: 77 patients with large vessel occlusion (LVO), 60 patients with non-large vessel occlusion (Non-LVO), and 15 patients without IS (WIS) who were admitted with suspicion of stroke but were later diagnosed with a different condition. Including all three groups in the dataset helped simulate a real-life scenario, as the three groups encompass the entire range of possible patients admitted to a hospital suspected of stroke. Non-LVO patients comprised the majority of the IS patients, while LVO patients contributed to approximately 30% of the total IS patients [198]. However, LVO patients are of clinical significance due to the extension of the ischemic area and the grim natural course of the condition [132]. LVO patients contribute approximately 60% of all the post-ischemic stroke disability and almost 95% of post-ischemic stroke mortality [12]. LVO was defined using CTA, where occlusion of the internal carotid artery, M1 and proximal M2 segment of the middle cerebral artery, A1 segment of the anterior cerebral artery, P1 segment of the posterior cerebral artery, basilar artery, and vertebral artery occlusion were regarded LVO. Non-LVO was defined as patients with perfusion deficits and affection of more distal arteries or with perfusion deficits without visible proximal artery occlusion.

Table 4.2: Numerical and percentage division of the patient set in training $\mathcal{D}_T$, validation $\mathcal{D}_V$, and testing $\mathcal{D}_{\text{Test}}$ dataset, illustrated for each group (LVO, Non-LVO, and WIS).

	$\mathcal{D}_T$ (#; %)	$\mathcal{D}_V$ (#; %)	$\mathcal{D}_{\textbf{Test}}$ (#; %)	**Tot.** (#; %)
LVO	29; 37.7	29; 37.7	19; 24.6	77; 50.6
Non-LVO	24; 40	25; 41.7	11; 18.3	60; 30.5
WIS	6; 40	6; 40	3; 20	15; 9.8
Total	59; 38.8	60; 39.5	33; 21.7	152; 100

With the sole exception of Paper III, all 152 patients were utilized for training, validating, and testing our proposed approaches. Thirty-three

patients were kept aside for testing, while the remaining 119 were used during training. The training and testing $\mathcal{D}_{\text{Test}}$ set was divided on a standard 80/20 split. Subsequently, the 119 training patients were divided in a 50/50 split between training $\mathcal{D}_T$ and validation $\mathcal{D}_V$ sets, keeping a balanced distribution among the groups. Table 4.2 summarizes the patients' division

Table 4.3: Patient characteristics are categorized and segregated based on specific groupings (LVO, Non-LVO, and WIS), highlighting the age, gender, and NIHSS score at hospital admission and discharge.

		LVO	**Non-LVO**	**WIS**
Age (average/range)		72 (39-94) years	75 (41-94) years	60 (27-85) years
Gender	Male	49 (64%)	37 (62%)	8 (53%)
	Female	28 (36%)	23 (38%)	7 (47%)
NIHSS score (maximum /minimum /average)	On hospital admission	38/0/13	19/0/6	14/1/3
	On hospital discharge	25/0/5	10/0/2	1/0/0

Table 4.3 contains information on the age, gender, and NIHSS score for different groups. The long-term functional outcome measured by the modified Rankin scale (mRS) at 90 days is also registered, with mRS on hospital admission. Overall, the information in Table 4.3 provides essential data on stroke severity in different groups and their long-term functional outcomes.

4.2 Ethical Approval

The Regional Ethic Committee, Norway granted ethical approval, ref.no: 2012/1499. The project is also approved by the Norwegian Agency for Shared Services in Education and Research (Sikt) Ref number 953392.

Part III

Contributions

Chapter 5

Ground Truth and Pre-Processing

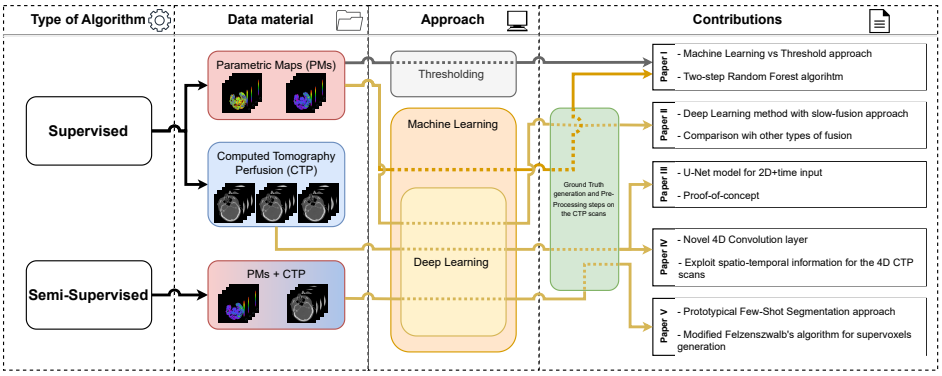


Figure 5.1: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy. The topics of this chapter (ground truth image generations and pre-processing steps) are highlighted in green.

The chapter is dedicated to the contributions made to the original dataset as a part of this PhD work. Fig. 5.1 gives an overview of the topic described in this chapter, highlighted in green, concerning the entire thesis. Moreover, the ground truth image generation is presented by examining the in-house

graphical user interface implemented for creating manual annotations. Finally, the pre-processing algorithm for the CT scans is described.

5.1 Ground Truth Generation

We got access to a dataset of CTP scans from Stavanger University Hospital, and initially, only weak labels were associated with the CTP scans. The original weak labels consisted of clinical labels as diagnose (LVO, Non-LVO or WIS), plus patient information, such as gender, age, and NIHSS score. As a part of this PhD work, an in-house Graphical User Interface (GUI) (Sec. 5.1.1) was developed to facilitate manual annotations of ischemic regions. Using the GUI, two expert neuroradiologists manually created the ground truth images containing the ischemic areas (core and penumbra) for all the patients in the dataset. The manual annotations were generated using the entire set of the CT examination, including the PMs from the CTP (Sec. 3.3), such as CBV, CBF, TTP, and T_{Max}, together with the maximum intensity projection (MIP) images. MIP is a method calculated as the maximum HU value over the time sequence of the CTP, providing a 3D volume from the 4D acquisition of CTP. Furthermore, the DWI examination performed within 1 to 3 days after the CT at hospital admission (Sec. 3.4) was used in assistance to generate the ground truth images.

5.1.1 In-house Graphical User Interface (GUI)

An in-house software was developed to enable neuroradiologists to generate the labeled masks to delineate the AIS areas using the various scans and PMs. The GUI software was developed using the App Designer from MATLAB R2022a [199]. The decision to select Matlab's App Designer was based on its simplicity and integrated environment to design and build GUIs; plus, all the neuroradiologists involved in generating the labeled masks are proficient in Matlab, which made the decision obvious.

Fig. 5.2 shows a screenshot of the GUI. The GUI was developed before the submission of Paper I to create the new labeled masks for the new patients in the study. The GUI was mainly built to solve the following matters in a unique place: 1) extraction and conversion of the images, 2) manually drawing the ischemic regions, and 3) generation of the predicted regions based on the proposed models in Paper I.

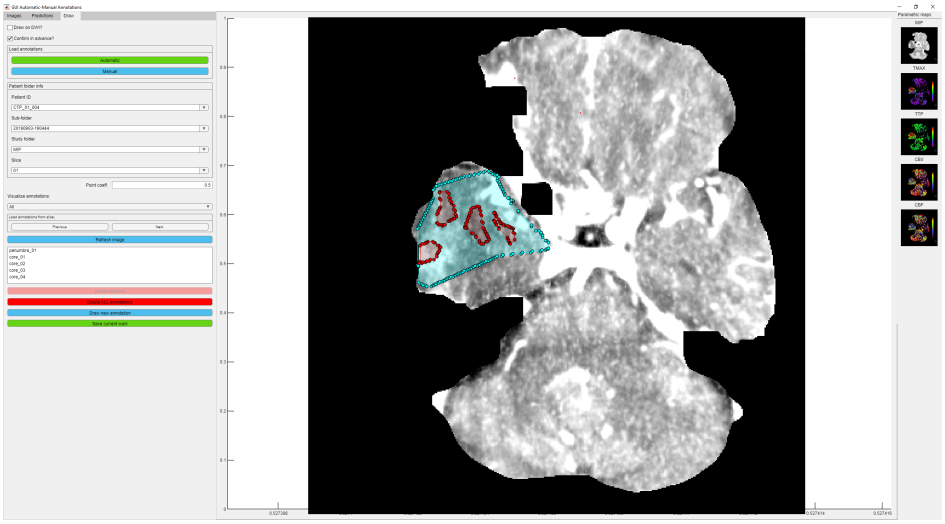


Figure 5.2: A visual representation captured from the GUI software meticulously crafted for manual annotations. A prominent and easily selectable PM occupies the central focus of this graphical interface. In the left panel, users can draw, edit, and save the annotations, move across patients and slices, and filter out annotations. The right panel collected the other selectable PMs for visualization and switched on the central panel. Penumbra and core regions are defined by light blue and red colors, respectively.

Images extraction and conversion Anonymized data received from Stavanger University Hospital were stored in DICOM. Hence, all the images (CT, CTP, PMs, and DWI) needed to be converted in a more suitable format to be used for ML and DL architectures. The decision fell on the TIFF format [200] due to its wide usability and popular choice for high color (16-bit) and deep color (30-bit) images.

Ischemic regions labeling The primary purpose of the GUI was to generate the labeled mask to be later used during training in ML and DL networks. As seen from Fig. 5.2, the center of the application is dominated by a selectable PM, where a user can freely draw an unlimited number of ischemic regions (core and penumbra). Specific color coding is applied to distinguish and delineate the penumbra and core regions. A blue color defines the penumbra area, while the core region is delineated in a red coloration. This color scheme aids users in precisely identifying and annotating these critical regions within the medical imagery or data, facilitating a more intuitive and accurate analysis.

These regions can be edited, deleted, saved, and copied to produce a product that is easy to use. The left panel (Fig. 5.2) is used to draw, edit, save, and copy the annotations in the central image. An optional adjustable interpolation function was implemented to smooth the manual annotations and change the amount of coordinates used to draw each region. The right panel (Fig. 5.2) contains the other PMs, which can be selectable for visualization purposes and inserted in the central panel of the GUI.

Prediction using ML methods Additionally, automatic prediction of the ischemic regions can be generated using the proposed model described in Paper I. The automated predictions share the same properties as the manual draws; thus, they can be edited, deleted, copied, and saved.

5.2 Pre-Processing steps

A series of pre-processing steps were performed to extract brain tissue from the raw CTP scans and improve the visual interpretation by enhancing the contrast. Algorithm 1 describes in detail the various steps for pre-processing all the data in a patient study.

In alternative, the pre-processing steps can also be summarized as follows:

(i) Co-registration of all the images in the 4D CTP scan using the first time point image as the frame of reference to correct possible motion artifacts. An intensity-based image registration with similarity transformation was used in this step.

(ii) All the registered CTP scans were encoded into HU values to have a known quantitative scale to describe radiodensity efficiently.

(iii) Brain extraction of CT studies plays an essential role in stroke imaging research [201, 202]. An automatic brain extraction method designed by Najm *et al.* [201] was selected due to its proven efficiency with CT datasets and public availability.

(iv) Gamma correction ($\gamma = 0.5$) and histogram equalization were performed to increase contrast and visual interpretability.

(v) Finally, z-score (z) on the enhanced 4D tensor is applied to normally distribute the data.

Algorithm 1 Pre-processing steps for one single patient study

Input: 4D CTP scan: $V(x, y, z, t)$
ref $\leftarrow V(x, y, z, t_1)_{\text{HU}}$ {Get the 1st time point image as frame of reference}

for $j = 2$ to $t_{\max}$ **do**
Co-register $V(x, y, z, t_j)$ using ref as frame of reference.
end for
$V(x, y, z, t)_{\text{HU}} \leftarrow \text{ConvertToHU}(V(x, y, z, t))$
$\dot{V}(x, y, z, t) \leftarrow \text{BrainExtraction}(V(x, y, z, t)_{\text{HU}})$ {The brain extraction function is designed by Najm *et al.*[201]}
$\ddot{V}(x, y, z, t) \leftarrow \text{GammaCorrection}(\dot{V}(x, y, z, t))$
$z_{\text{high}} \leftarrow \text{GetSliceWithHighestIntensityValue}(\ddot{V}(x, y, z, t))$
bins $\leftarrow 2^{16}$
$\widetilde{V}(x, y, z_{\text{high}}, t), \mathcal{T}_{z_{\text{high}}} \leftarrow \text{HistEq}(\ddot{V}(x, y, z_{\text{high}}, t), \text{bins})$ {$\mathcal{T}_{z_{\text{high}}}$ is the grayscale transformation for z_{high}}
$\widetilde{V}(x, y, z_{\text{high}}, t) \leftarrow \frac{\widetilde{V}(x,y,z_{\text{high}},t) - \text{mean}(\widetilde{V}(x,y,z_{\text{high}},t))}{\sigma(\widetilde{V}(x,y,z_{\text{high}},t))}$ {Standardization of the data}
for $i = 1$ to $z_{\max}$ **do**
if $i \neq z_{\text{high}}$ **then**
$\widetilde{V}(x, y, z_i, t) \leftarrow \text{HistEq}(\ddot{V}(x, y, z_i, t), \mathcal{T}_{z_{\text{high}}})$
$\widetilde{V}(x, y, z_i, t) \leftarrow \frac{\widetilde{V}(x,y,z_i,t) - \text{mean}(\widetilde{V}(x,y,z_i,t))}{\sigma(\widetilde{V}(x,y,z_i,t))}$
end if
end for
return Processed 4D CTP scan: $\widetilde{V}(x, y, z, t)$

Chapter 6

Supervised Ischemic Stroke Segmentation

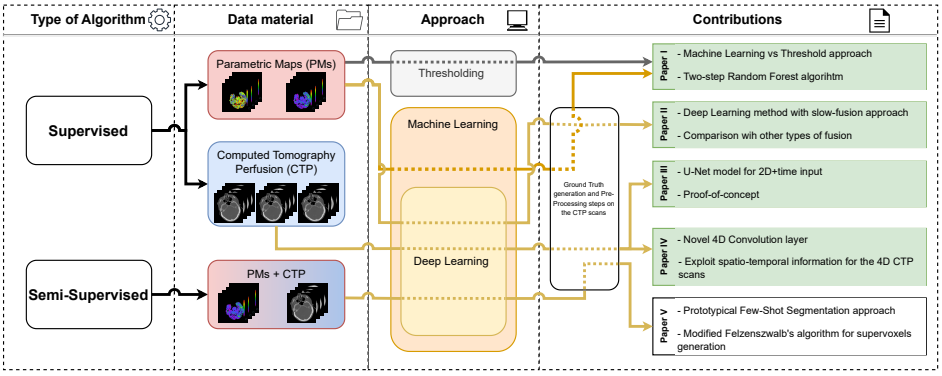


Figure 6.1: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy. The green color highlights the topic of this chapter (the four papers that utilize a supervised approach).

The following chapter is dedicated to supervised IS segmentation. The green highlighted sections in Fig. 6.1 present an overview of the contents described in this chapter. Papers I-IV are dedicated to the topic, and the main methods, results, and contributions will be presented. In the thesis, Papers I-IV are part of the sub-objectives SO_1 and SO_2:

- SO_1: Use PMs (derived from CTP) as input for ML and DL architectures, mimicking what medical doctors are commonly using for assessing neurological disorders.
- SO_2: Leverage of CTP scans as input for the segmentation task, exploiting the totality of the raw scans in DL architectures.

6.1 Contributions overview

The thesis aims to properly define two areas in the brain tissue, the ischemic core and penumbra, that can appear during an AIS. Penumbra is the hypoperfused but still salvageable tissue [20], while core can be defined as infarcted tissue if fast reperfusion doesn't occur [19]. Neuroradiologists can significantly benefit from evaluating these two ischemic regions automatically and precisely, especially in the first few hours after stroke onset. A system capable of identifying these regions can be used to lessen the burden on medical doctors, which can help increase diagnosis and treatment, a crucial need worldwide, from LMICs to HICs [1, 9, 11–13].

The segmentation task of penumbra and core is notably challenging: first, IS lesions (core and penumbra) are not irreversible, and they can change over time [13]; second, IS areas have an extremely high variation in shape and location in the brain tissue, making them difficult to locate in some cases [48]. To tackle the first challenge, we generated ground truth masks based on CTP, PMs, MIP, and DWI images (Sec. 5.1). For the second challenge, we have used raw CTP scans and/or PMs as input for our ML and DL methods and studied which input combinations give the best results.

6.2 Segmentation task using parametric maps

PMs have been widely implemented in the literature to segment the ischemic areas (and FIAs), both using thresholding approaches [22–25, 33, 168, 177, 178], ML models [26, 183], and DL architectures [29, 185–188]. Temporal information of the CTP scans are exploited during the production of the PMs, as described in Sec. 3.2.5, hence a 4D CTP scan is compressed to Y $\times$ 3D, where Y is the number of PMs. This filtering makes it possible for a human to interpret the data studying directly at the 3D PMs together, whereas examining the 4D CTP scans is very hard for a person. However,

an AI model could utilize the 4D information better. Currently, PMs are visually examined by neuroradiologists for diagnosis and treatment planning. Therefore, it is natural to consider whether an ML architecture can segment core and penumbra regions efficiently, using the same PMs as a neuroradiologist would use. This architecture is later used as a reference when using the raw 4D CTP data as input for an AI method.

Papers I and II are directly related to the thesis sub-objective SO_1.

6.2.1 Paper I - ML algorithms vs. Thresholding

The objective of this paper was to answer the following question: "Can a ML model segment the ischemic areas using PMs as input?". In clinical practice today, the radiologist use these PMs, together with the MIP and NIHSS score, as input for their manual assessment. Can an automated method with the same input be helpful? This objective fits under the thesis sub-objective SO_1.

The paper proposes a ML-approach for automatically segmenting penumbra and core regions. The approach is shown in Fig. 6.2. CT scans from 152 patients suspected of AIS were used for this work. The dataset was divided into three groups: LVO, Non-LVO, and WIS (Sec. 4.1.1).

Several well-known ML algorithms were trained and tested, such as Support Vector Machine [203], Decision Tree [204], and Random Forest [205]. The proposed multi-classification methods take a set of PMs, the MIP map, and the NIHSS score as input data for a ML algorithm (Fig. 6.2). A series of pre-processing steps were performed in order to create input vectors for the ML architectures: firstly, only the pixels inside the brain tissue were taken into consideration; hence a brain extraction step was completed; secondly, an optional step for augmenting the number of input was performed using the Simple Linear Iterative Clustering (SLIC) [121] algorithm for extracting superpixel regions (Sec. 2.3). The resulting superpixel regions were used as additional input to provide spatial information to the ML models. During the training phase, the Synthetic Minority Over-sampling Technique (SMOTE) [206] was implemented to over-sample the classes with minor occurrences and augment the dataset.

Two architectures were studied: a multi-class *Single-Step* approach and a *Two-Step* binary-class approach. In the *Single-Step* approach, the PMs (CBV, CBF, TTP, T_{Max}), the MIP map, and the NIHSS score are used as input for predicting the ischemic areas. For the *Two-Step* approach, two

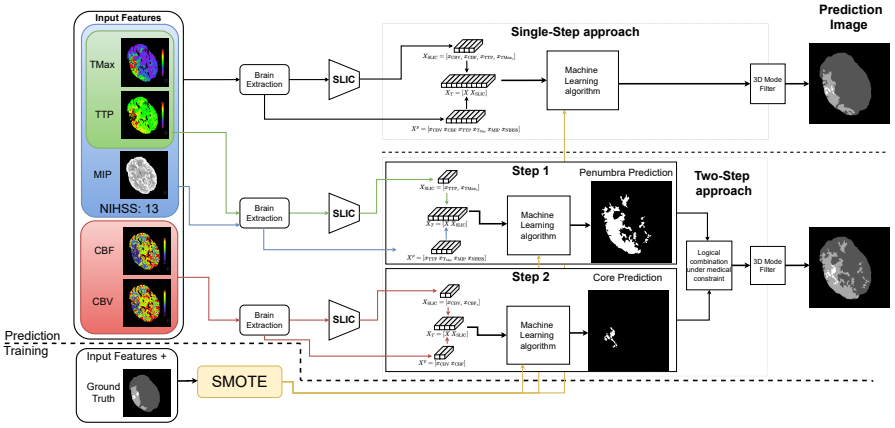


Figure 6.2: Visual description of the multi-classification methods for Paper I. *Single-Step* approach: all PMs (CBV, CBF, TTP, T_{Max}), the MIP, and the NIHSS scare are adopted as input for the ML algorithm. *Two-Step* approach: divided into two stages. *Step1* predicts penumbra regions using six input (two PMs, T_{Max} and TTP, plus their SLIC version, the MIP, and the NIHSS score), and *Step 2* predicts core regions using four input (two PMs (CBV and CBF) and their corresponding SLIC version). The final prediction combines these two binary maps, including only the core regions inside the penumbra regions. A final post-processing step using a 3D mode filter is implemented for both *Single-Step* and *Two-Step* approach. SLIC refers to the algorithm to extract superpixel regions.

binary classification methods were combined to predict penumbra areas (if any) and core. For the penumbra class, six inputs were implemented: TTP (and its SLIC version), T_{Max} (and its SLIC version), MIP, and NIHSS score as input. For the core regions, the CBV, its SLIC version, and the CBF with its SLIC version were combined as input (Fig. 6.2). The selection of the PMs for the *Two-Step* approach is in line with methods found in the literature [23, 167, 168]: TTP and T_{Max} are typically used to detect penumbra, while CBF and CBV are utilized to find the core. After the final predictions, a 3D mode filter is implemented as a post-processing step to reduce unwanted noise, allowing the predictions to rely on adjacent voxels in the z-axis.

Many experiments were completed to evaluate the pipelines and select the best prediction parameters. The first experiment was implemented to select the best combination of pre-processing steps and ML model for the segmentation (named *Mdl-5.2*). The second experiment was completed based on the previous results to choose the most suitable number of super-pixel regions for the *Mdl-5.2*. The third experiment validated the proposed

Table 6.1: The results are presented for penumbra (core) regions on the holdout set. Comparison is made between our best pipeline (*Mdl-5.2*) and methods from the literature using thresholding values with the same vendor "syngo.via" (default setting and [177]). Predictions from [22, 23, 25, 167] are presented but can not be directly compared due to the usage of different vendor and/or post-processing steps when generating the PMs. ‡ marks the results for the *Mdl-5.2* method without using any post-processing step. Note that for the Dice coefficient, higher values are better (⇑), while for Hausdorff Distance and ΔV, lower values are preferable (⇓).

Method	Vendor	Dice Coefficient ⇑			Hausdorff Distance ⇓			ΔV (ml) ⇓			
		LVO	Non-LVO	All	LVO	Non-LVO	All	LVO	Non-LVO	WIS	All
		Penumbra (Core)									
Best Method (*Mdl-5.2*) ‡	Siemens "syngo.via"	0.66 (**0.26**)	0.51 (0.03)	0.66 (0.26)	6.9 (4.8)	3.5 (0.9)	5.2 (3.1)	44.2 (16.2)	6.9 (**0.8**)	1.4 (**0.0**)	27.9 (9.6)
Best Method (*Mdl-5.2*)		**0.69** (**0.27**)	**0.56** (0.03)	**0.68** (**0.26**)	**6.5** (**4.3**)	**3.0** (**0.7**)	**4.8** (**2.7**)	**40.7** (**12.9**)	**4.9** (1.0)	**0.9** (**0.0**)	**25.1** (**7.8**)
Default Setting		0.31 (0.25)	0.11 (**0.04**)	0.27 (0.20)	7.8 (6.2)	5.6 (4.4)	6.6 (5.2)	67.5 (48.2)	51.8 (37.4)	3.7 (12.1)	58.2 (40.8)
Bathla *et al.* [177]		0.47 (0.17)	0.22 (0.03)	0.45 (0.14)	6.9 (6.9)	4.5 (4.7)	5.6 (5.7)	65.2 (65.3)	16.5 (44.5)	22.5 (6.6)	43.3 (53.5)
Other thresholding methods presented but not used for comparison											
Bivard *et al.* [167]	Toshiba	0.42 (0.19)	0.16 (0.03)	0.39 (0.15)	7.3 (6.5)	4.6 (4.4)	5.8 (5.4)	70.6 (52.9)	30.2 (36.0)	1.5 (9.1)	50.8 (43.3)
Cambell *et al.* [23]	Philips	N.A. (0.22)	N.A. (0.04)	N.A. (0.18)	N.A. (5.9)	N.A. (3.9)	N.A. (4.9)	N.A. (35.2)	N.A. (24.9)	N.A. (5.6)	N.A. (29.1)
Murphy *et al.* [25]	General Electric	0.17 (0.27)	0.08 (0.05)	0.16 (0.23)	7.5 (5.0)	4.8 (3.1)	6.1 (4.0)	96.7 (13.4)	21.1 (13.3)	8.6 (2.1)	63.5 (12.3)
Wintermark *et al.* [22]	Philips	N.A. (0.19)	N.A. (0.02)	N.A. (0.14)	N.A. (7.5)	N.A. (5.5)	N.A. (6.4)	N.A. (90.8)	N.A. (71.4)	N.A. (20.1)	N.A. (77.9)

pipeline with the chosen number of superpixel regions against all the combinations evaluated in the first experiment. Finally, a hyper-parameter optimization was performed on the selected best pipeline.

Mdl-5.2 was used as the proposed pipeline and compared with a thresholding approach that used the same vendor for generating the PMs [177]. Moreover, the proposed method was also compared with the vendor's default settings ("syngo.via") for detecting ischemic regions. The results are introduced in Table 6.1 demonstrating how the proposed method performed better than the thresholding approaches for penumbra and core segmentation, showing promising results for the LVO patients but meager segmentation results for the Non-LVO group.

Contributions of the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

6.2.2 Paper II - Multi-input segmentation

From the thesis sub-objective SO_1 we can formulate the question "Since ML algorithms can segment the ischemic areas with PMs as input, can a DL architecture perform better?", which leads to the work of Paper II.

In Paper I (Sec. 6.2.1), promising results were obtained using ML methods to automatically predict core and penumbra based on four PMs (CBV, CBF, TTP, T_{Max}), combined with MIP and NIHSS score. Based on this success, a DL architecture was implemented and tested in this work using a multi-input combination adopting a slow fusion approach. The proposed network is displayed in Fig. 6.3. The same dataset of patients and inputs to the pipeline, as used in Paper I, was utilized in Paper II. The input to the pipeline is the PMs (CBV, CBF, TTP, T_{Max}) together with MIP images and the patient's NIHSS score.

The proposed method adopted a slow fusion technique (Sec. 2.2.7) to handle the information extracted from the multi-input. Features are extracted from four to five VGG-16 models [125] used as backbone during the convolutional part of the architecture (left side of Fig. 6.3) and fused in the deconvolutional part (right side of Fig. 6.3) using concatenation

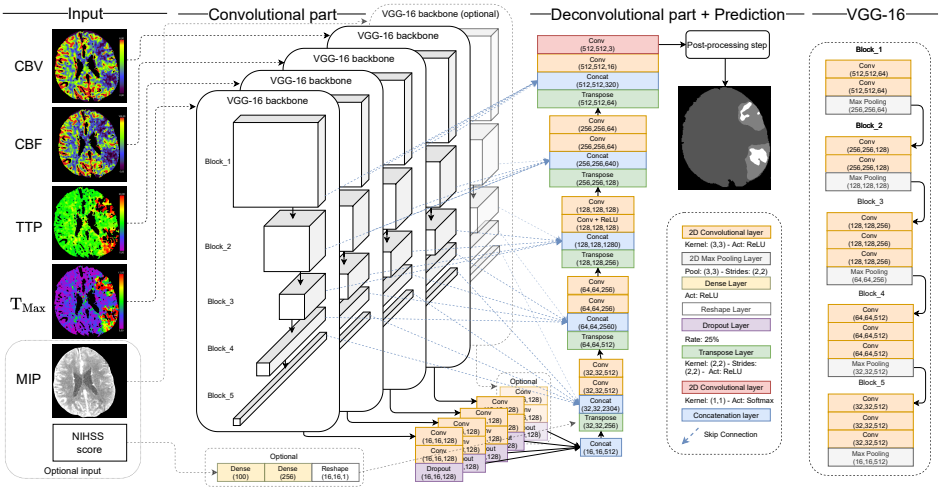


Figure 6.3: Proposed architecture with multi-input and slow fusion for Paper II. Depending on the experiment, the network has a convolutional part composed of four to five VGG-16 models [125]. Skip connections combined with feature concatenation from VGG-16 models ensure slow data fusion. On the right is an overview of the feature extraction part of the VGG-16 architecture.

layers and skip connections at different levels of the convolutional part (Sec. 2.2.6). Like many other medical datasets, the dataset presented an imbalance among the classes involved. Pixels labeled as core and penumbra represent, collectively, the 6.9% of the total number of pixels; the imbalance was even more pronounced in the Non-LVO group, where core and penumbra pixels denote only the 2.4% of the totality. To deal with this imbalance, a specific loss function (Sec. 2.2.3) was implemented for the proposed model, the FTL, a function specifically developed and demonstrated to address the data imbalance problem in medical image segmentation [100].

Several experiments have been performed in this paper: firstly, a hyper-parameter search is executed to find the best combination of parameters for the FTL. The best parameters found were in line with the ones suggested by the original paper of the FTL [100]. Secondly, the proposed method was trained with different combinations of inputs to properly understand how the diversity in the feeding input affected the model's performance. A combination of PMs and NIHSS score as input, with a gradual fine-tuning approach, was the one that yielded the best performances overall. We named this network SF_G(PMs,N). Lastly, the proposed method was compared with other models adopting the same input combination but

Table 6.2: Statistical results over the validation set for the models divided for the second and third experiments. The last two rows contain results for the third experiment. Each value exhibits the patients' average in the validation set and the standard deviation (SD) for the distinct groups. For the metrics: ⇑ indicates that higher values are better, while with ⇓, lower values are preferable. **Bold** values represent the best result for a specific class and metric.

Model	Input			Dice Coeff. (Avg.) ± SD ⇑		Hausdorff Dist. (Avg.) ± SD ⇓		ΔV (Avg.) ± SD (ml) ⇓			
				LVO	Non-LVO	LVO	Non-LVO	LVO	Non-LVO		WIS
	PMs	MIP	NIHSS	Penumbra				Core			
Exp-2 - Layer weights: **Frozen**											
SF_F(PMs)	X			0.71±0.1 / **0.37±0.3**	0.27±0.3 / 0.22±0.3	5.9±1.0 / 2.7±1.8	3.2±1.5 / 0.9±0.8	27.0±28 / 9.4±20	10.0±15 / 0.8±1.3		9.8±8.0 / 0.5±0.5
SF_F(PMs,M)	X	X		0.69±0.2 / 0.36±0.3	0.29±0.3 / 0.20±0.3	5.9±0.9 / 3.0±1.7	3.1±1.4 / 1.0±0.8	25.9±27 / 10.0±21	5.5±6.0 / 0.8±1.0		8.1±7.0 / 0.5±0.3
SF_F(PMs,N)	X		X	0.70±0.2 / 0.36±0.3	0.29±0.3 / 0.16±0.2	5.6±1.2 / **2.3±1.9**	2.3±1.3 / 0.7±0.6	27.0±37 / 5.0±9.0	4.1±4.8 / 0.6±1.2		4.0±3.4 / 0.1±0.1
SF_F(PMs,M,N)	X	X	X	0.68±0.2 / 0.34±0.3	0.3±0.3 / 0.18±0.3	5.7±1.4 / 2.5±1.8	2.4±1.2 / 0.8±0.7	29.9±37 / 6.3±12	**2.8±3.0** / 0.5±1.0		3.9±4.0 / **0.0±0.0**
Exp-2 - Layer weights: **Unfrozen**											
SF_U(PMs)	X			0.70±0.2 / 0.34±0.3	0.29±0.4 / 0.24±0.3	5.4±1.3 / 2.7±1.7	2.1±1.5 / 0.7±0.7	29.8±36 / 6.5±14	4.2±6.9 / 0.5±0.7		**0.1±0.1** / **0.0±0.0**
SF_U(PMs,M)	X	X		0.70±0.2 / 0.36±0.3	0.34±0.3 / **0.24±0.3**	5.6±1.4 / 2.5±1.8	2.3±1.6 / 0.6±0.6	28.1±40 / 6.5±8.1	4.9±9.8 / 0.4±0.8		0.9±1.5 / **0.0±0.0**
SF_U(PMs,N)	X		X	**0.72±0.2** / 0.36±0.3	0.29±0.3 / 0.23±0.3	5.6±1.1 / 2.6±1.8	2.7±1.4 / 0.9±0.7	28.8±28 / 5.2±10	7.8±12 / 0.5±0.7		3.4±2.7 / 0.1±0.1
SF_U(PMs,M,N)	X	X	X	0.71±0.2 / 0.36±0.3	0.32±0.3 / 0.22±0.3	5.7±1.0 / 2.8±1.7	2.4±1.6 / 1.0±0.8	29.0±25 / 7.0±14	9.0±18 / 0.8±0.9		5.5±4.5 / 0.1±0.1
Exp-2 - Layer weights: **Gradual Fine-tuning**											
SF_G(PMs)	X			0.71±0.2 / 0.35±0.3	0.30±0.3 / 0.19±0.3	5.8±1.0 / 2.4±1.8	3.0±1.4 / 0.8±0.7	29.3±25 / 4.7±7.6	11.8±22 / 0.6±0.8		6.2±5.5 / **0.0±0.0**
SF_G(PMs,M)	X	X		0.69±0.2 / 0.35±0.3	**0.34±0.6** / 0.22±0.4	5.4±1.4 / 2.4±1.8	**1.9±1.2** / **0.6±0.5**	27.8±38 / 5.2±9.4	3.3±3.7 / **0.3±0.3**		2.4±3.8 / **0.0±0.0**
SF_G(PMs,N)	X		X	**0.72±0.2** / **0.37±0.3**	0.31±0.3 / 0.21±0.3	**5.3±1.4** / 2.4±1.7	2.3±1.4 / 0.7±0.6	26.0±35 / **4.4±7.0**	6.5±14 / 0.5±0.7		1.0±1.7 / **0.0±0.0**
SF_G(PMs,M,N)	X	X	X	0.68±0.3 / 0.34±0.3	0.31±0.3 / 0.18±0.3	5.5±1.3 / 2.5±1.8	2.3±1.3 / 0.7±0.6	26.9±35 / 5.1±9.6	5.5±9.2 / 0.5±0.8		0.9±1.1 / **0.0±0.0**
Exp-3 - Layer weights: **Gradual Fine-tuning**											
EF_G(PMs,N)	X		X	0.68±0.2 / 0.26±0.3	0.32±0.3 / 0.19±0.3	5.6±1.4 / 2.4±1.9	2.3±1.5 / 0.7±0.6	31.8±47 / 8.0±21	5.3±10 / 0.7±0.9		1.8±0.3 / **0.0±0.0**
EFI_G(PMs,N)	X		X	0.67±0.2 / 0.29±0.3	0.29±0.3 / 0.16±0.2	6.0±0.8 / 2.5±1.9	3.3±1.5 / 0.7±0.6	**25.5±28** / 5.3±10	6.3±7.1 / 0.6±1.2		6.7±4.7 / **0.0±0.0**

using distinct fusion approaches to verify the validity of the slow fusion approach. The architecture was compared with an early fusion approach (Sec. 2.2.7) and with an inflated early fusion approach inspired by the model proposed in [83]. Table 6.2 presents the results achieved by the various networks during the second (*Exp-2*) and third experiments (*Exp-3*). The proposed system was presented as a first step to being a supportive tool to help neuroradiologists make treatment decisions.

Contributions of the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

6.3 Segmentation task using Computed Tomography Perfusion scans

According to the European Stroke Organization[1], CT and MRI are recommended imaging modalities for patients admitted in a hospital with a suspicion of IS. However, CT is a faster and cheaper diagnostic tool compared to MRI, with less contraindications [169]. A CTP scan is a 4D spatio-temporal examination containing a vast amount of data (details in Sec. 3.2.4). In comparison to NCCT and CTA scans, CTP combines more pertinent clinical data for patients suspected of AIS, as demonstrated by various studies [142–144].

Few methods have explored the four dimensions of the CTP scans for segmenting FIAs in patients affected by an IS [30, 31, 158, 189]. These methods exhibit encouraging results for the FIA segmentation; nevertheless, to the best of our knowledge, there is no research on the segmentation of the ischemic regions developing during the first stages of an IS (penumbra and core) using CTP scans. Using the entire 4D CTP scans as input to a DL architecture can be beneficial for the IS segmentation of the two areas.

Paper III is directly related to the thesis sub-objective SO_2. Paper IV is linked to the main objective O_1, combining sub-objectives SO_1 and SO_2.

[1] https://eso-stroke.org/

6.3.1 Paper III - CNN Architecture

The content of this paper aligns with the thesis sub-objective SO_2. The paper tries to answer the the following question: "Can a DNN take 4D CTP scans as input to segment penumbra and core areas?"

This paper aims to solve the SO_2 problem and answer the abovementioned question by presenting a proof-of-concept model for using CTP scans to segment penumbra and core regions. The dataset used in this study comprises 10 patients with LVO collected at Stavanger University Hospital. The proposed architecture, called *mJ-Net*, follows a classic U-Net structure [112] with convolution and deconvolution parts. Fig. 6.4 gives a general overview of the proposed network. The convolution section (left side of Fig. 6.4) first extracts temporal information from the input and then extracts spatial features.

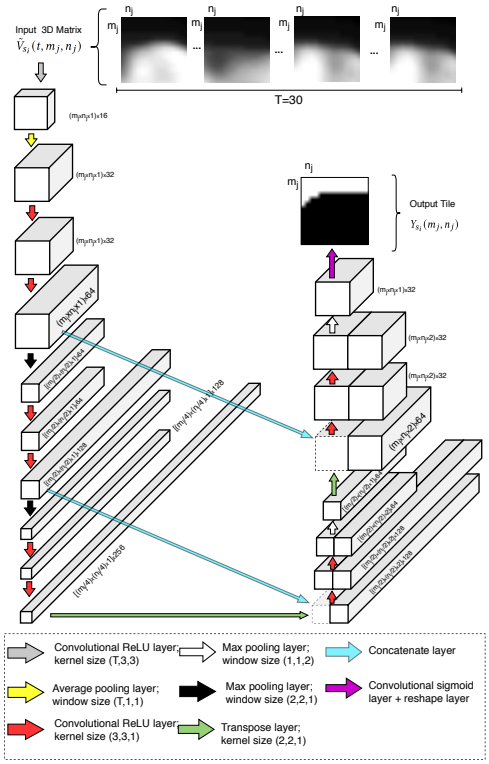


Figure 6.4: Detailed overview of the mirror J-Net (*mJ-Net*). The top side highlights the 2D+time tensor used as input for the architecture.

The model is a CNN (Sec. 2.2.8) that takes in input a 2D+time tensor (Sec. 2.1.2) and returns a 2D image as output, where the first two input dimensions are the height and width of a CTP image, and the third dimension is the time dimension. Due to a limited number of patients involved in this study, the input is generated in 16×16 tiles using a slicing window technique over the entire CTP images; this is accomplished to increase the data fed to the architecture. Moreover, leave-one-patient-out cross-validation was performed to strengthen the achieved results. Two different optimizers (Sec. 2.2.4), SGD and Adam, were tested in this study together with an SDCL loss function (Sec. 2.2.3). The proposed *mJ-Net* was verified against two architectures to extract features from the inputs, showing the possibilities of this network in solving the research objective SO_2. Due to the dimensionality of the dataset and the usage of a single patient group, the LVO group, further work was needed to validate the proposed pipeline.

Contributions of the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

6.3.2 Paper IV - Exploiting 4D CTP scans

From the results of the previous works and the thesis main objective O_1, we formulate the following question: "Which input (CTP or PMs) is more suitable for a DNN for the segmentation task of ischemic regions in patients suspected of AIS?" Answering that question brings us to the work of Paper IV, which is connected with the main objective O_1, plus it combined sub-objectives SO_1 and SO_2.

After the promising results achieved in Paper III, a study on what type of input to feed on a DNN was necessary. The amount of information provided in a 4D CTP scan granted a vast spectrum of possibilities: the four dimensions of the input were exploited and tested to assess which one achieved more accurate segmentations. Raw CTP scans undergo a series of pre-processing steps (Sec. 5.2). This paper is built on top of Paper II (Sec. 6.2.2) and Paper III (Sec. 6.3.1): two extensions of the network proposed

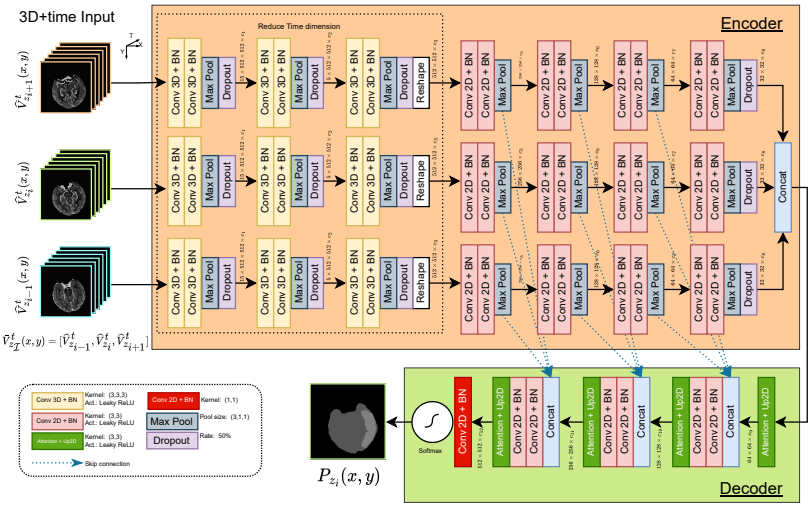


Figure 6.5: Illustration of the *3D+time mJ-Net* model. The list of 2D+time inputs $\bar{V}^t_{z_\mathcal{I}}(x,y) = [\widehat{V}_{z_{i-1}}, \widehat{V}_{z_i}, \widehat{V}_{z_{i+1}}]$ is trained in parallel, where $z_\mathcal{I} = \{z_{i-1}, z_i, z_{i+1}\}$. The output is a 2D image $P_{z_i}(x,y)$. The orange section defines the encoder part of the network, while the green rectangle highlights the decoder section.

in Paper III (*mJ-Net*) were explored and compared with the architecture presented in Paper II, here renamed "*Multi-Input PMs*". The dataset used for this work is the same as that adapted for Papers I and II.

The first proposed system, called *3D+time mJ-Net*, used as input a list of 2D+time tensors (Sec. 2.1.2), adding an extra dimension to the *mJ-Net*'s input. The architecture (a visual overview is given in Fig. 6.5) implemented three *mJ-Net* to train in parallel three neighbouring brain slices, named z_{i-1}, z_i, z_{i+1}. In the *3D+time mJ-Net*, the encoder was used to reduce the time dimension and extract spatial features. The outputs of the three parallel networks were concatenated before being fed to the decoder, which was implemented to upsample the features with concatenation layers and skip connections in order to generate a final 2D prediction image corresponding to the brain slice z_i.

Fig. 6.6 displays the second proposed architecture. The network, named *4D mJ-Net*, adopted a 4D input $\widehat{V}^t_{z_\mathcal{I}}$, which is the concatenation of a 2D+time tensor of a brain slice z_i at index i over all the time points t (a.k.a. the *mJ-Net* input) plus its neighbouring brain slice tensors. The proposed model is built with a single encoder that reduces the input's time dimension with a customized 4D Convolutional layer; afterward, the depth dimension

Table 6.3: Experiment results for the validation set. Values in **bold** exhibit the best results for each column and each class. Mean results plus standard deviation for Dice Coefficient (DC), Hausdorff Distance (HD), and ΔV are presented. Results are for the penumbra and core areas divided by the distinct patient groups (LVO, Non-LVO, WIS, and All). Note that for the DC, higher values are better (⇑), while for HD and ΔV, lower values are preferable (⇓).

Method	DC ⇑			HD (mm) ⇓			ΔV (ml) ⇓			
	LVO	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**WIS**	**All**
	Penumbra									
Multi-input PMs [207]	**0.70±0.1**	0.27±0.3	0.47±0.3	2.9±0.4	1.4±0.7	2.0±0.8	27.0±28.6	10.0±15.5	9.8±8.1	**19.0±24.2**
mJ-Net [208]	0.66±0.2	0.39±0.3	0.50±0.3	2.9±0.5	2.6±0.6	2.7±0.7	**25.5±20.0**	24.7±29.2	45.5±39.1	27.2±26.0
2D-TCN [30]	0.12±0.1	0.02±0.0	0.07±0.1	4.1±0.5	3.8±0.6	4.0±0.6	81.3±65.6	80.6±57.8	131.6±93.1	86.0±66.5
3D-TCN-SE [31]	0.25±0.1	0.05±0.1	0.15±0.1	6.2±0.5	6.7±0.4	6.4±0.5	497.9±157.1	559.3±90.4	624.6±118.7	533.1±137.3
3D-TCN	0.23±0.1	0.04±0.1	0.14±0.1	4.3±0.4	4.4±0.5	4.3±0.5	85.3±64.0	142.7±51.4	164.2±43.8	114.2±65.3
3D+time mJ-Net	**0.70±0.1**	0.42±0.3	**0.53±0.3**	2.6±0.6	1.9±0.8	2.2±0.9	35.1±36.1	18.3±26.3	2.7±2.6	25.7±32.4
4D mJ-Net	0.66±0.1	**0.44±0.3**	0.51±0.3	**2.3±0.6**	**1.3±0.7**	**1.7±1.0**	41.4±37.2	**6.1±6.3**	**0.0±0.0**	24.3±32.9
	Core									
Multi-input PMs [207]	**0.37±0.3**	**0.21±0.3**	**0.28±0.3**	1.2±0.8	**0.4±0.4**	0.8±0.8	9.4±20.3	**0.8±1.3**	0.5±0.5	5.3±15.3
mJ-Net [208]	0.27±0.2	**0.21±0.2**	0.22±0.2	1.5±0.7	0.8±0.6	1.2±0.6	**5.5±4.9**	1.0±1.2	1.0±1.1	**3.4±4.3**
2D-TCN [30]	0.02±0.0	0.01±0.0	0.01±0.0	1.9±0.7	1.5±0.6	1.7±0.7	11.8±13.3	8.1±8.2	11.0±11.2	10.3±11.4
3D-TCN-SE [31]	0.00±0.0	0.00±0.0	0.00±0.0	1.2±0.9	0.4±0.4	0.8±0.8	12.7±15.6	1.9±2.8	**0.0±0.0**	7.5±12.8
3D-TCN	0.02±0.0	0.01±0.0	0.01±0.0	1.4±0.8	0.8±0.4	1.1±0.7	12.0±14.3	1.9±2.1	2.4±1.9	7.3±11.6
3D+time mJ-Net	0.21±0.2	0.12±0.2	0.16±0.4	**1.1±0.7**	**0.4±0.4**	**0.7±0.7**	8.1±10.6	1.3±1.6	**0.0±0.0**	4.8±8.5
4D mJ-Net	0.29±0.2	**0.21±0.2**	0.23±0.2	1.6±0.9	0.5±0.4	1.0±0.9	25.9±37.0	1.4±2.2	**0.0±0.0**	14.3±29.6

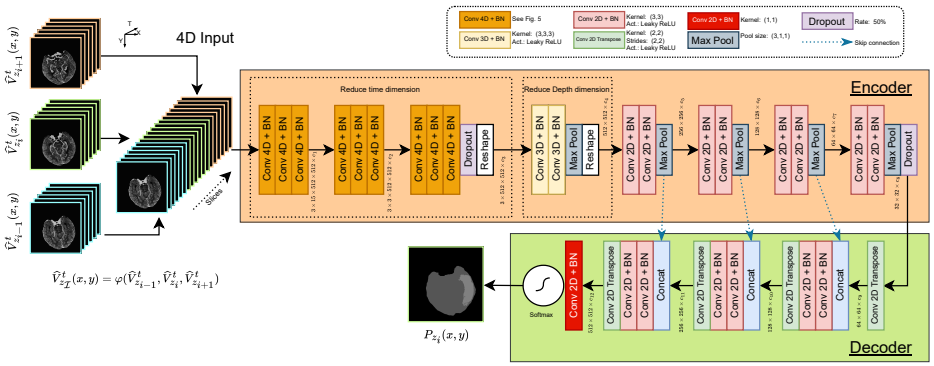


Figure 6.6: Illustration of the *4D mJ-Net* architecture. The 4D input $\widehat{V}^t_{z_\mathcal{I}} = \varphi(\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}})$ is the concatenation of a 2D+time volume $\widetilde{V}^t_{z_i}$ of a brain slice z_i at index i over all the time points t plus its neighbouring brain slice volumes $(\widetilde{V}^t_{z_{i-1}}, \widetilde{V}^t_{z_{i+1}})$. The network output is a 2D image $P_{z_i}(x, y)$. The orange section defines the encoder part of the network, while the green rectangle highlights the decoder section.

is inspected and reduced, and later, spatial information are captured. A single decoder, similar to the one for the *3D+time mJ-Net*, is implemented to generate the predicted ischemic regions (if any) of the brain slice z_i.

The two proposed architectures were trained and compared with the *Multi-Input PMs* (Paper II), the *mJ-Net* (Paper III), two systems presented in [30, 31], and a customized extension of [31]. Table 6.3 presents the outcomes of the proposed models plus the compared architectures. The *3D+time mJ-Net* network achieved the best results for the large ischemic areas delineation while underestimating the core class. The *4D mJ-Net* segmented penumbra regions in a superior way, regardless of patient groups, and it was the only model that didn't predict any ischemic areas in patients WIS, displaying very encouraging results.

Contributions of the author

- I developed the methodology in collaboration with my co-authors.
- I made all implementations and conducted all experiments.
- I wrote the original draft of the manuscript.

Chapter 7

Unsupervised Ischemic Stroke Segmentation

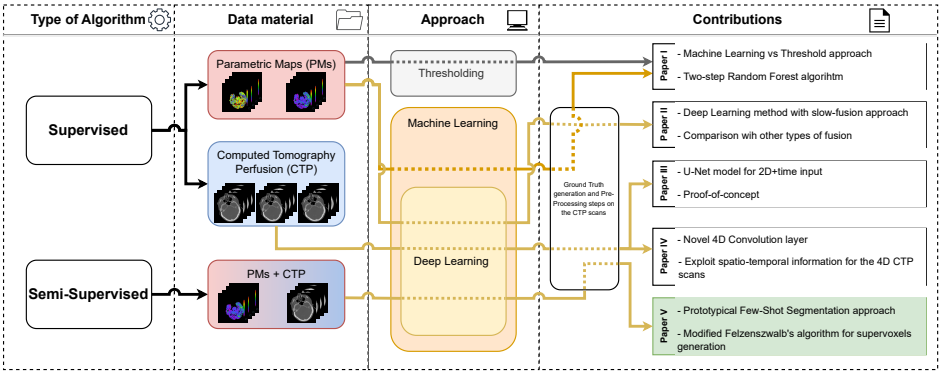


Figure 7.1: A general overview of the proposed pipeline and the topics addressed by the various papers included in the thesis. The pipeline accentuates the type of algorithms, the data material utilized, the diverse approaches considered, and the interrelation between every scientific paper and the various sections of the pipeline. Dashed lines emphasize the route through which data moves across the associated strategy. The topic of this chapter (the paper using a self-supervised approach) is highlighted in green.

The following chapter is dedicated to unsupervised IS segmentation. Fig. 7.1 presents an overview of the contents described in this chapter (highlighted in green). Paper V is dedicated to the topic, and the main methods, results, and contributions will be presented. In the thesis, Paper

V answers the sub-objective SO_3: Investigate if self-supervised and/or unsupervised DL methods are suitable for the AIS segmentation.

7.1 Segmentation task using Few-Shot Learning

While DL has shown great promise in many areas [82, 83, 112, 126], it also poses several challenges. One of the main challenges is the need for large amounts of labeled data to train DL models effectively. The need for a large amount of labeled data is highly relevant in the medical domain, where labeled data are scarce due to time constraints and the need for expert analysis. There has been an increasing trend in the utilization of self-supervised FSS approaches for image segmentation to circumnavigate the need for large amounts of labeled data [41, 56, 57].

FSS algorithms are trained and tested in an episodic approach, wherein a few labeled support images are employed to guide the segmentation of unlabeled query images [41]. To eliminate the requirement for manual annotation of training data, episodes are structured using a self-supervised method that automatically generates pseudolabels based on superpixels/supervoxels algorithms. Only a few labeled support images are necessary for the segmentation task during testing.

7.1.1 Paper V - Leveraging Domain Knowledge using Few-Shot Learning

The work of this paper tries to answer the following question: "Are unsupervised DL methods suitable for segmenting IS lesions?" which is connected with the thesis sub-objective SO_3.

Papers I-IV explored how supervised DL algorithms can be employed for AIS segmentation. Paper V was developed to answer the sub-research objective SO_3. A novel FSS architecture, inspired by the work of Hansen *et al.* [41], was tailored for the AIS segmentation. While recent state-of-the-arts FSS models [41, 57] rely on the extraction of superpixels/supervoxels from the raw input images, this paper leveraged domain knowledge generating supervoxels from the PMs.

Fig 7.2 gives a general overview of the various stages included in the pipeline. The proposed model, called ADNet, used CTP scans as input images and PMs for generating supervoxels implemented as pseudolabels during training. A set of PMs (CBV, CBF, TTP, T_{Max}) and MIP images

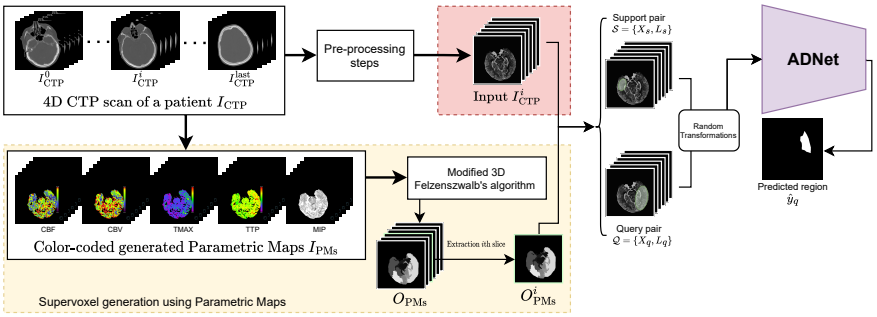


Figure 7.2: General overview of the various steps involved in the method presented in Paper V. Supervoxel regions are generated from the PMs and MIP with a modified 3D version of Felzenszwalb's image segmentation algorithm [122] (the area in yellow). Pre-processing steps are implemented to extract brain tissue and create a volume input (the region in red). Two random slices containing the same supervoxel are then sampled to act as the support and query slice, with the supervoxel region acting as the pseudolabel for the self-supervised task. Random transformations are then applied to either the support or query volume to provide the support-query pair that is used as input to train the ADNet model.

were stacked and fed to a modified 3D version of Felzenszwalb's image segmentation algorithm [122] to produce supervoxel regions (yellow area in Fig. 7.2). During the ADNet's training, a supervoxel was randomly selected as the pseudolabel. Then, two random brain slices containing the selected supervoxel were chosen; after applying arbitrary transformations, these slices acted as the support and query images for the model.

Raw CTP scans undergo a series of pre-processing steps to extract brain tissues and for contrast enhancement (Sec. 5.2). The proposed system was trained with two datasets: one was previously used in Papers I-II-IV, and the other contains only patients from the LVO group. The latter dataset is implemented to illustrate variation effects during training, and because ischemic lesions in this group are extended, hence easier to detect.

A series of experiments were run to validate the ADNet architecture. Firstly, a domain knowledge verification was performed, where different inputs for the network and the supervoxels generation were tested. Secondly, a sensitivity study of the supervoxel's size was accomplished to select the proper size for adequate IS lesions segmentation. Table 7.1 displays the results achieved by the proposed model compared to the baseline networks and the different datasets implemented during training. The work showed encouraging results, even if the performances differed from the supervised

Table 7.1: Mean (+ standard deviation) test results on the two datasets. All the methods (except Paper II's method) utilize a single labeled 2D+time slice during the inference phase. In Paper II, a supervised approach has been implemented and relies on the full labeled dataset for training and inference. Note that higher values are preferable for the DS and the Matthews Correlation Coefficient (MCC) [209] (⇑), while lower values are better for ΔV (⇓). Best performances are highlighted in **bold** (the last row is only used for comparing the supervised method and our self-supervised approaches).

Method	**Best** ρ	**Dataset**	**DS** ⇑			**MCC** ⇑			ΔV **(ml)** ⇓			
			LVO	Non-LVO	All	LVO	Non-LVO	All	LVO	Non-LVO	WIS	All
Self-Supervised Methods												
CTP-Baseline	1000	CTP-LVO	0.50±0.14	0.07±0.03	0.43±0.12	0.48±0.14	0.10±0.05	0.43±0.12	72.6±15.8	85.4±26.5	212.1±63.0	89.5±15.4
PMs-Baseline	1250		0.50±0.15	0.09±0.03	0.45±0.14	0.48±0.15	0.11±0.03	0.44±0.14	74.8±15.7	50.3±42.8	75.5±41.3	66.7±19.5
Proposed	1250		**0.60±0.16**	**0.13±0.05**	**0.58±0.16**	**0.58±0.17**	**0.14±0.05**	**0.57±0.16**	**56.4±14.7**	**23.0±40.0**	**16.0±50.0**	**41.6±25.4**
CTP-Baseline	1000	CTP-ALL	0.47±0.11	0.09±0.02	0.39±0.09	0.45±0.11	**0.14±0.04**	0.39±0.10	84.2±13.4	128.9±23.3	176.3±31.5	107.5±17.4
PMs-Baseline	1750		0.50±0.13	**0.10±0.05**	0.45±0.12	0.48±0.14	0.14±0.07	0.43±0.13	86.3±15.1	70.5±34.1	219.6±57.9	93.2±15.6
Proposed	1500		**0.55±0.10**	0.06±0.03	**0.51±0.10**	**0.54±0.10**	0.07±0.03	**0.50±0.10**	**60.4±10.9**	**32.3±6.8**	**13.8±12.0**	**46.8±9.8**
Supervised Method (used only for comparison)												
Paper II	N.A.	CTP-ALL	0.85±0.06	0.49±0.30	0.66±0.32	0.83±0.06	0.49±0.29	0.64±0.31	19.6±15.2	4.9±5.4	0.3±0.4	12.9±14.2

approaches proposed in Paper II (Sec. 6.2.2). This study is a first step in exploring few-shot learning approaches for IS lesion segmentations.

Contributions of the author

- I developed the methodology in collaboration with my co-authors.
- The implementation of the proposed architecture (ADNet) is based on the work of Hansen *et al.* [41]; I have expanded the architecture and conducted all the experiments.
- I wrote the original draft of the manuscript.

Chapter 8

Discussion and Conclusion

This chapter discusses the main achievements and challenges of segmenting AIS regions. The objective of the thesis, with its relative sub-objectives, was introduced in Chapter 1; in this chapter, a discussion of the various sub-objectives is given with the knowledge discussed in the previous chapters. Each paper included in this thesis will be analyzed by the type of input provided for the segmentation task or the learning approach. Finally, the conclusion of the thesis and possible future direction are discussed and considered.

8.1 Discussion

Early detection and intervention in AIS patients are of vital importance [137, 210, 211]. In the case of AIS, early detection and prompt treatment can prevent morbidity and mortality [9, 137]. Two ischemic regions can be defined in the early stage of an IS: the ischemic core and penumbra. The ischemic core is a term used to describe the irreversibly damaged brain tissue that is highly likely to become infarcted if fast reperfusion does not occur [19]. The ischemic penumbra is a region of hypoperfused but viable tissue surrounding the ischemic core [20].

Correctly identifying the ischemic penumbra is critical in tailoring the therapeutic strategy and improving the clinical outcome of AIS patients. This task is challenging due to the considerable variability in ischemic lesions shape and location within brain tissue and their dynamic nature, which can evolve. A system capable of identifying these regions automatically and

precisely can significantly benefit neuroradiologists and lessen their burden for manual delineations [9, 11–13]. Thus, the thesis explored various ML and DL networks that can accurately identify both penumbra and core areas in the brain tissue during an AIS.

8.1.1 Segmentation task using parametric maps

This section is directly related to the thesis sub-objective SO_1: "Use PMs (derived from CTP) as input for ML and DL architectures, mimicking what medical doctors are commonly using for assessing neurological disorders".

As established in the related work section in Chapter 3 (Sec. 3.3), PMs have been widely adopted as input for thresholding approaches [22–25, 33, 168, 177, 178], ML models [26, 183], and DL architectures [29, 185–188] to segment the ischemic areas (and FIAs). The creation of these PMs involves the exploitation of temporal information derived from CTP scans, as detailed in Sec. 3.2.5. Consequently, a 4D CTP scan is condensed into a 3D representation (Y × 3D), where Y denotes the number of PMs. This transformation facilitates human interpretation, enabling experts to study the data directly within the context of 3D PMs.

It is important to note that presently, neuroradiologists rely on visual inspection of PMs for diagnosis and treatment planning. Two papers, Paper I presented in Sec. 6.2.1 and Paper II summarized in Sec. 6.2.2, explored AI architectures to precisely segment the ischemic areas using the same PMs that a neuroradiologist would typically employ.

In Paper I, a multi-stage algorithm based on ML was presented. The proposed pipeline, using the RF algorithm with the *Single-Step* approach, generates noteworthy statistical performance over most metrics regardless of the stroke severity taken under consideration. The system achieved satisfactory results for both penumbra and core areas in patients affected by LVO, while the segmentation for Non-LVO patients was more challenging: small core regions can be problematic to segment.

A new approach was realized in Paper II for tackling the segmentation of small regions. A multi-input CNN with a slow fusion approach is introduced here. The proposed method is trained on patients with and without IS and with different vessel occlusion severities (the dataset is described in Sec. 4.1 and it is the same used in Paper I). This generalization helps the model correctly segment most ischemic regions, regardless of the patient group. Our proposed approach, implemented with a gradual fine-tuning approach

with PMs (CBV, CBF, TTP, T_{Max}) and NIHSS score as input, achieved encouraging results for the penumbra and core classes within the LVO group but showed unpromising results for the Non-LVO group. Medium/large size areas were more straightforward to predict by the proposed architecture than small regions, mainly located in the Non-LVO group. Moreover, false-positive predictions were detected with this architecture (and the other tested), possibly due to artifacts in the PMs. The network demonstrated the validity of using DL approaches for tackling the segmentation task of ischemic regions; nevertheless, relying on PMs as input model might lead to unsatisfactory results due to artifacts and the condensation of the 4D CTP scans into 3D representations.

8.1.2 Segmentation task using Computed Tomography Perfusion scans

The following section is connected to the thesis sub-objective SO_2: "Leverage of CTP scans as input for the segmentation task, exploiting the totality of the raw scans in DL architectures".

CTP is widely used in case of suspected AIS to identify ischemia positively and to make treatment decisions [33, 53, 142]. A CTP study contains many scans, making it well-suited for utilization by AI frameworks in detecting ischemic lesions. As described in Sec. 3.3, few ML and DL approaches with 4D CTP scans as input have been investigated for the FIAs segmentation tasks due to the high computational complexity of 4D data [27, 30, 31, 158, 189, 190]. There is a clear research gap concerning the segmentation of ischemic regions developing during the initial phases of an IS (penumbra and core), exploiting 4D CTP scans as input for AI architectures. It is noteworthy that employing the complete 4D CTP scans as input for a DL architecture holds the potential to offer significant advantages in the segmentation of these two ischemic regions within the context of an IS.

The DL architecture presented in Paper III (Sec. 6.3.1) proved the validity of using NN solely based on 4D CTP scans as input for segmenting penumbra and core areas. The results were promising for both the ischemic regions but with some limitations: a small dataset of 10 patients belonging to the LVO group was used for training and validation. Plus, the segmented ischemic regions exhibit regions with pixelation, lowering the segmentation quality. Thus, further work was required to validate the achieved results.

In Paper IV (Sec. 6.3.2), various architectures have been proposed to utilize the 4D CTP input for using the spatio-temporal information better than in existing approaches. Two systems were developed using the 4D CTP scans: the former, called *3D+time mJ-Net*, used an input dimension defined as 3D+time, while the latter, named *4D mJ-Net*, adopted a 4D tensor as its input. Due to promising achievements in Paper III, a train of its architecture (*mJ-Net*) with a larger and broader dataset was presented. Additionally, for comparison, we employed the architecture proposed in Paper II (here named *Multi-input PMs*) due to its satisfactory segmentation results attained using PMs as input. Lastly, to assess performance, we also utilized two models outlined in previous works by Amador *et al.* in [30, 31], along with a customized extension of the network from Amador *et al.* [31].

Results from the *mJ-Net* led to penumbra over-segmentation, which may increase treatment decision uncertainty. Prediction results from the *Multi-input PMs* present satisfactory results for large ischemic areas, although predictions for small regions are not optimal. Predictions from the networks presented in [30, 31] over-segment the penumbra class and poorly segment the core class, probably because these models were designed for a binary segmentation task. Results achieved by the *3D+time mJ-Net* and *4D mJ-Net* demonstrated that increasing the dimension fed to the architectures reduces over-segmentation, especially in the Non-LVO and WIS groups, where the ischemic areas are extremely small and burdensome to detect for a NN. Producing valid results in patients in the Non-LVO group can be valuable in a real-life scenario: Non-LVO cases comprise the majority of the AIS cases [198, 212, 213].

The findings from Table 6.3 demonstrate that the enhancement in performance for all *mJ-Net* pipelines is evident with the increase in input data dimension. Our approaches (*3D+time mJ-Net* and *4D mJ-Net*) yielded encouraging outcomes across all the included classes; the *3D+time mJ-Net* exhibits precise delineation of large ischemic areas but tends to underestimate the core class. On the other hand, the *4D mJ-Net* consistently and accurately segments penumbra regions, regardless of patient groups.

8.1.3 Segmentation task using Few-Shot Learning

This section is associated with the thesis sub-objective SO_3: "Investigate if self-supervised and/or unsupervised DL methods are suitable for the AIS segmentation".

Supervised learning networks require a substantial amount of annotated regions during training, a requirement that can be challenging to fulfill within the medical field, given the limited availability of annotated data. Considering the promising outcomes demonstrated by various FSS approaches [41, 56, 57], we explore an alternative strategy for AIS segmentation.

Paper V (Sec. 7.1.1) presents a tailored prototypical FSS model that relies on minimal annotated samples during the training phase. In the context of prototypical FSS, using domain knowledge has proven advantageous in improving the segmentation performance of the proposed network. Validation results have shown that incorporating PMs scans to generate supervoxel regions, subsequently used as pseudolabels in combination with CTP scans, enhances the overall segmentation accuracy in patients with LVO. It is worth noting that employing the same input type for generating the supervoxels and training the FSS model results in sub-optimal performance. While the segmentation outcomes are far from perfect, the study presented in Paper V represents an initial exploration of few-shot learning approaches for IS lesion segmentation.

8.1.4 Limitations

Each of the assessed methods shared common limitations. The scan data for each pipeline, originating exclusively from CT scanners, are produced by the same vendor, the "syngo.via" software from Siemens Healthineers (Sec. 4.1), resulting in the absence of data diversity. Additionally, the annotations employed as the ground truth delineate the penumbra and core, but they do not perfectly represent these regions. Consequently, there may be instances where small core segments extend into the penumbra tissue, and specific details of the penumbra may be inaccurately annotated as healthy brain tissue.

Most of the architectures presented in a supervised approach (Papers I-IV) produce either over-segmented or under-segmented predictions of a particular group or class due to the high variation in the size of ischemic regions within each group. Regarding the *4D mJ-Net*, it can detect small ischemic lesions with high precision and can correctly predict no ischemic regions in WIS patients. Nevertheless, sometimes it tends to overestimate the core class in LVO subjects. Including the entire spatio-temporal information of the CTP scan leads to more precise segmentation, especially when the lesion is meager, hence more challenging to detect. The architecture

presented in Paper V exhibits promising results for the segmentation of IS lesions and demonstrates the benefit of leveraging domain knowledge. However, it's important to note that the results are far from optimal, and additional research is still required.

8.2 Conclusions and future work

The results of the various papers mentioned above indicate that the semantic segmentation of ischemic regions, core and penumbra, during the first stages of an IS are more prone to be identified by an AI architecture with the usage of CTP scans rather than PMs. Adopting CTP as input for an AI method increases the segmentation performances for all groups of patients, especially for subjects with small or no lesions (Non-LVO and WIS groups). The three thesis sub-objectives helped explore different approaches and types of inputs to tackle the automatic segmentation of ischemic regions derived from cerebral stroke. Hence, from the three sub-objectives, it is possible to conclude the thesis's primary objective: we developed a fast and automatic diagnostic system for segmenting core and penumbra areas in patients suspected of AIS using CTP scans acquired at hospital admission. In future work, more validation experiments need to be conducted for the semi-supervised approach (Paper V). Moreover, reducing the over-segmentation for every class must be investigated to produce more precise outcomes. Furthermore, an analysis of future perspectives of predicting tissue outcome can be performed. Understanding tissue prone to bleeding complications and implementing clinical data in the risk analyses are also possible future works. Although we developed a fast and automatic diagnostic system, we need to implement it into a simple and usable pipeline that might aid neuroradiologists in diagnosis and treatment planning.

Part IV

Included Papers

Paper 1:
Machine learning algorithms versus thresholding to segment ischemic regions in patients with acute ischemic stroke

Machine learning algorithms versus thresholding to segment ischemic regions in patients with acute ischemic stroke

L. Tomasetti[1], L. J. Høllesli[1,2], K. Engan[1], K. D. Kurz[1,2], M. W. Kurz[3,4,5], M. Khanmohammadi[1]

[1] Department of Electrical Engineering and Computer Science, University of Stavanger

[2] Stavanger University Hospital, Stavanger medical imaging laboratory (SMIL)

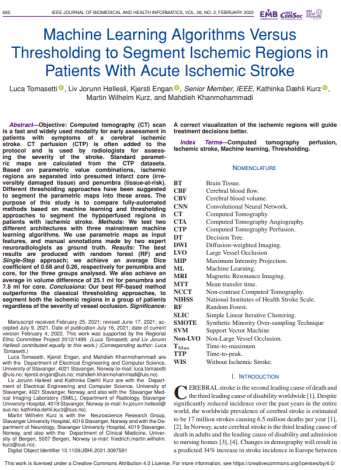
[3] Neuroscience Research Group, Stavanger University Hospital

[4] Department of Neurology, Stavanger University Hospital

[5] Department of Clinical Medicine, University of Bergen

Published by the IEEE Journal of Biomedical and Health Informatics, 2021

https://doi.org/10.1109/JBHI.2021.3097591



660 IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS, VOL. 26, NO. 2, FEBRUARY 2022

Machine Learning Algorithms Versus Thresholding to Segment Ischemic Regions in Patients With Acute Ischemic Stroke

Luca Tomasetti, Liv Jorunn Hallesli, Kjersti Engan, *Senior Member, IEEE*, Kathinka Dahli Kurz, Martin Wilhelm Kurz, and Mahdieh Khanmohammadi

***Abstract*—Objective: Computed tomography (CT) scan is a fast and widely used modality for early assessment in patients with symptoms of a cerebral ischemic stroke. CT perfusion (CTP) is often added to the protocol and is used by radiologists for assessing the severity of the stroke. Standard parametric maps are calculated from the CTP datasets, and by applying thresholds, tissue suspected of infarction regions are separated into presumed infarct core (irreversibly damaged tissue) and penumbra (tissue-at-risk). Different thresholding approaches have been suggested to segment the parametric maps into these areas. The purpose of this study is to compare fully-automated methods based on machine learning and thresholding approaches to segment the hypoperfused regions in patients with ischemic stroke. *Methods:* We test two different architectures with three mainstream machine learning algorithms. We use parametric maps as input features, and manual annotations made by two expert neuroradiologists as ground truth. *Results:* The best performing model, a U-Net architecture with a *Single-Step* approach, we achieve an average Dice coefficient of 0.68 and 0.26, respectively for penumbra and core, for the three groups analysed. We also achieve an average in volume difference of 25.1 ml for penumbra and 7.8 ml for core. *Conclusions:* Our best RF-based method outperforms the classical thresholding approaches, to segment both the ischemic regions in a group of patients regardless of the severity of vessel occlusion. *Significance:* A correct visualization of the ischemic regions will guide treatment decisions better.**

***Index Terms*—Computed tomography perfusion, ischemic stroke, Machine learning, Thresholding.**

Manuscript received February 25, 2021; revised June 17, 2021; accepted July 9, 2021. Date of publication July 16, 2021; date of current version February 4, 2022. This work was supported by the Regional Ethic Committee Project 2013/1499. (Luca Tomasetti, *and Liv Jorunn Hallesli contributed equally to this work.*) (*Corresponding author: Luca Tomasetti.*)

Luca Tomasetti, Kjersti Engan, and Mahdieh Khanmohammadi are with the Department of Electrical Engineering and Computer Science, University of Stavanger, 4021 Stavanger, Norway (e-mail: luca.tomasetti@uis.no; kjersti.engan@uis.no; mahdieh.khanmohammadi@uis.no).

Liv Jorunn Hallesli is with the Department of Electrical Engineering and Computer Science, University of Stavanger, 4021 Stavanger, Norway, and also with the Stavanger Medical Imaging Laboratory (SMIL), Department of Radiology, Stavanger University Hospital, 4019 Stavanger, Norway (e-mail: liv.jorunn.hallesli@sus.no; kathinka.dahli.kurz@sus.no).

Martin Wilhelm Kurz is with the Neuroscience Research Group, Stavanger University Hospital, 4011 Stavanger, Norway, and also with the Department of Neurology, Stavanger University Hospital, 4019 Stavanger, Norway, and also with the Department of Clinical Medicine, University of Bergen, 5007 Bergen, Norway (e-mail: friedrich.martin.wilhelm.kurz@uis.no).

Digital Object Identifier 10.1109/JBHI.2021.3097591

Nomenclature

BT	Brain Tissue.
CBF	Cerebral blood flow.
CBV	Cerebral blood volume.
CNN	Convolutional Neural Network.
CT	Computed Tomography.
CTA	Computed Tomography Angiography.
CTP	Computed Tomography Perfusion.
DT	Decision Tree.
DWI	Diffusion-weighted Imaging.
LVO	Large Vessel Occlusion
MIP	Maximum Intensity Projection.
ML	Machine Learning.
MRI	Magnetic Resonance Imaging.
MTT	Mean transfer time.
NCCT	Non-contrast Computed Tomography.
NIHSS	National Institutes of Health Stroke Scale.
RF	Random Forest.
SLIC	Simple Linear Iterative Clustering.
SMOTE	Synthetic Minority Over-sampling Technique
SVM	Support Vector Machine
Non-LVO	Non-Large Vessel Occlusion.
T_{Max}	Time-to-maximum
TTP	Time-to-peak.
WIS	Without Ischemic Stroke.

I. Introduction

CEREBRAL stroke is the second leading cause of death and the third leading cause of disability worldwide [1]. Despite significantly reduced incidence over the past years in the entire world, the worldwide prevalence of cerebral stroke is estimated to be 17 million strokes causing 6.5 million deaths per year [1], [2]. In Norway, acute cerebral stroke is the third leading cause of death in adults and the leading cause of disability and admission to nursing homes [3], [4]. Changes in demography will result in a predicted 34% increase in stroke incidence in Europe between

Abstract:
Objective: Computed tomography (CT) scan is a fast and widely used modality for early assessment in patients with symptoms of a cerebral ischemic stroke. CT perfusion (CTP) is often added to the protocol and is used by radiologists for assessing the severity of the stroke. Standard parametric maps are calculated from the CTP datasets. Based on parametric value combinations, ischemic regions are separated into presumed infarct core (irreversibly damaged tissue) and penumbra (tissue-at-risk). Different thresholding approaches have been suggested to segment the parametric maps into these areas. The purpose of this study is to compare fully-automated methods based on machine learning and thresholding approaches to segment the hypoperfused regions in patients with ischemic stroke. *Methods:* We test two different architectures with three mainstream machine learning algorithms. We use parametric maps as input features, and manual annotations made by two expert neuroradiologists as ground truth. *Results:* The best results are produced with random forest (RF) and *Single-Step* approach; we achieve an average Dice coefficient of 0.68 and 0.26, respectively for penumbra and core, for the three groups analysed. We also achieve an average in volume difference of 25.1ml for penumbra and 7.8ml for core. *Conclusions:* Our best RF-based method outperforms the classical thresholding approaches, to segment both the ischemic regions in a group of patients regardless of the severity of vessel occlusion. *Significance:* A correct visualization of the ischemic regions will guide treatment decisions better.

9.1 Introduction

Cerebral stroke is the second leading cause of death and the third leading cause of disability worldwide [214]. Despite significantly reduced incidence over the past years in the entire world, the worldwide prevalence of cerebral stroke is estimated to be 17 million strokes causing 6.5 million deaths per year [214, 215]. In Norway, acute cerebral stroke is the third leading cause of death in adults and the leading cause of disability and admission to nursing homes [216, 217]. Changes in demography will result in a predicted 34% increase in stroke incidence in Europe between 2015 and 2035, which is likely to be mirrored in other parts of the world [215]. Thus, cerebral stroke has a huge socio-economic impact on society and a tremendous impact on the quality of life for every single patient [8].

There are two broad categories of cerebral stroke; hemorrhagic and ischemic stroke. Approximately 20% of all strokes are due to hemorrhage, while approximately 80% are due to ischemia [4]. Both groups can further be divided into different subtypes. Ischemic stroke may be caused by arteriosclerosis, thrombi, emboli, dissections, or systemic hypoperfusion, all of them leading to ischemia due to reduced blood flow in regions of the brain.

The severity of ischemia usually varies within the area of reduced blood flow, and for clinical use, the area is divided into two distinct regions: ischemic core and penumbra. The ischemic core is defined as irreversibly damaged brain tissue [18]. The tissue within the penumbra is critically hypoperfused and is located around and adjacent to the infarct core. If blood flow is restored timely, this tissue may regain neurological function [18]. If the blood flow remains low, however, the area of penumbra will transfer into an irreversibly damaged infarct core. The ischemic penumbra was introduced by Astrup *et al.* as "a region of hypoperfused, electrically silent, and functionally impaired but viable tissue" [20]. Restoring blood flow and thereby preventing the penumbra from proceeding to irreversibly damaged infarct core, is the main treatment goal in patients with acute ischemic stroke (AIS). Penumbra may change into infarct core rapidly in AIS patients. Therefore, rapid recognition of stroke symptoms and acute treatment in a stroke center are of vital importance.

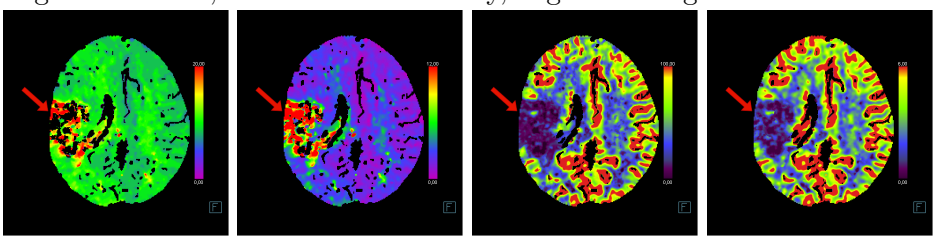
According to the European Stroke Organization guidelines, Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) are the two modalities recommended for diagnostic imaging in acute stroke patients [15]. MRI with diffusion-weighted imaging (DWI) is superior to CT scans

for detection of small acute infarctions and identification of some stroke mimics. Nevertheless, CT is the preferred imaging modality in many centers for acute stroke patients due to its widespread availability, rapid scan times, and its high sensitivity for detecting hemorrhage. DWI has been considered the gold standard for ischemic core estimation [22–24, 167, 168]; however, there are very few hospitals where MRI are used as the first imaging tool in acute stroke patients, since it is not always timely available on a 24/7 basis, plus, some patients have contraindications for this type of modality. MRI is usually performed within the first days after an AIS. Treatment, timing of treatment, and other variables will affect further development of the penumbra. Hence, any core of follow-up MRI might have developed after the acute imaging and might not be comparable with the imaging results in the acute setting. In the last years, DWI has been contested as the de-facto gold standard since it cannot accurately differentiate irreversibly ischemic tissue from salvageable tissue [19, 171], and it has been shown that the detected ischemic regions can be partially reverse, especially if DWI is performed in the early window time [19, 172, 218].

At Stavanger University Hospital (SUS), patients with suspected acute stroke are routinely investigated with non-contrast computed tomography (NCCT) of the head, CT angiography (CTA) of the precerebral and cerebral arteries, i.e. arch to vertex angiogram, and CT Perfusion (CTP) immediately after hospital admission. In most cases MRI including DWI is performed during the next days. In patients with suspected stroke with unknown time of symptom onset, MRI with DWI is used as a first-line diagnostic tool upon hospital admission.

Whether treatment is applied depends on time from symptom onset to hospital admission, but also largely depends on imaging results with CT Perfusion being the key-modality for patient selection. In CTP a time series of three-dimensional (3D) datasets are acquired during contrast agent injection. Based on the changes in the tissue density over time, color-coded parametric maps are calculated. The different parametric maps highlight spatio-temporal information from the passage of the contrast agent within the brain tissue. Generally, parametric maps based on CTP are generated in two steps: the first step acquires a time-density curve for each pixel based on the track of the contrast agent. The second step consists of extracting specific information from the generated time-density curves. Cerebral blood flow (CBF), cerebral blood volume (CBV), time-to-peak (TTP), mean transit time (MTT) and time-to-maximum (T_{Max}) are all examples of parametric maps [18]. Radiologists use parametric maps for diagnosis and

treatment planning and are indirectly assessing penumbra and core by evaluating such parametric maps. An example of the parametric maps of a single brain slice, involved in this study, is given in Fig. 9.1.



(a) TTP. (b) T_{Max}. (c) CBF. (d) CBV.

Figure 9.1: Parametric maps of a single slide of a patient's brain. In this patient there is an ischemic area on the right side in the vascular territory of the middle cerebral artery (pointed by a red arrow). TTP = time-to-peak; T_{Max} = time-to-maximum; CBF = relative cerebral blood flow; CBV = relative cerebral blood volume. Color in the online version.

Time is a fundamental factor for patients affected by an ischemic stroke. Automation of the recognition process for the ischemic regions, penumbra and core, can be immensely helpful for medical doctors for treatment decisions. Over the last decades, different methods and parameters were tested to find the most suitable approach to segment the ischemic regions using parametric maps as input.

Region growing is a technique to extract connected areas in an image based on pixel information; this method is defined as semi-automatic because the user manually selects a seed for the growing region algorithm. This technique was used by Matesin *et al.* [174] in relation with CT head images of stroke lesions, and by Dastidar *et al.* [175], for measuring the volumetric infarction using 3D T2 Fast Spin Echo MRI in patients affected by stroke. Their goal was to delineate ischemic areas, and not to distinct the core from the penumbra. The first delineation of both areas, using a region growing technique in combination with the parametric maps acquired by CTP analyses, was implemented by Contin *et al.* [176].

A series of studies have proposed experiments with threshold values on the derived parametric maps to improve the results achieved by the region's growing approaches. Different thresholds have been proposed for different parametric maps, generated from different vendors, and applied to various datasets [22–25] to estimate both the ischemic regions, or the infarct core, or penumbra. These studies have used follow-up images (such as DWI or NCCT), acquired hours later after the stroke onset, to delineate the

ground truth of the infarct regions and used them as a comparison for their predictions. For this reason, studies using DWI as follow-up imaging present some limitations: they only included patients who were later identified with infarct lesions in follow-up images, excluding the ones who underwent the same routine at the time of hospital admission but did not show any lesion in the follow-up DWI; they also excluded patients with contraindication for MRI. Moreover, since the threshold values were compared with final infarctions, assessed after the patient's treatment, they do not present a perfect estimation of the infarctions before treatment decision; thus, they are not the best candidates to help medical doctors during the treatment making decision. Furthermore, the studies have proposed quite distinct thresholding values due to the different vendors used for post-processing evaluation and the distinct window of time ($\leq$ 1 hour to 7 days) used for follow-up images to evaluate the ground truth for the ischemic regions. Thus, there is no real consensus to properly define the ischemic regions based on threshold values on the parametric maps derived from CTP.

In recent years, Machine Learning (ML) and neural network algorithms have achieved promising results in a large number of medical image analysis applications, and have also made their way into the stroke application [26–29, 208]. Kemmling *et al.* proposed a generalized linear model using the parametric maps as input and clinical data to quantify changes of tissue infarction [26]. Qiu *et al.* implemented a ML-based algorithm to detect early infarction in patients with AIS using NCCT as input and follow-up DWI as ground truth [27]. Kasasbeh *et al.* used a semi-automatic approach based on a convolutional neural network (CNN) with the entire set of parametric maps as input to classify the infarct core using follow-up DWI as ground truth [29]. However, these ML and CNN based methods were only trained to classify the infarct core regions and did not find the penumbra areas. Differently, Qiu *et al.* developed two distinct ML models, using a multiphase CTA as input and DWI/NCCT follow-up images as ground truth, to predict core and penumbra [28]. Their primary goal was to demonstrate the validity of using multiphase CTA in comparison to CTP imaging for evaluating ischemic regions, but they stated limitations in their data material. Nevertheless, using follow-up images for delineating the ischemic regions limits the usability for medical doctors since they might not be helpful for treatment decisions but just for comparison with the clinical outcome. Our research group was, to the best of our knowledge, the first using the entire 4D CTP data as input to a neural network to segment both penumbra and core simultaneously. A modified U-Net model

was used in a small pilot study to segment both penumbra and core regions using the entire 4D CTP volume as input and with ground truth generated with manual expert assessment directly from the parametric maps [208]. The results were promising, but they were based on a very small pilot study and need to be validated on a larger sample size.

Before continuing to use the entire 4D dataset as input, we wish to study the utility of automatically segmenting the penumbra and core based on the parametric maps that are already calculated in the standard software used in clinical practice. Based on the ideas and the shortcomings of the published methods, we propose in this paper a ML-based method using the parametric maps as input and both core and penumbra regions as output, in addition to healthy tissue. One can argue that CNN naturally fits this type of problem; nevertheless, several examples of classical ML methods with this application can be found in the literature [26–28] using follow-up images as ground truth, bearing with them the same issues mentioned earlier. Moreover, learning good CNN models usually require large datasets, and/or transfer-learning, and we have a limited dataset to work with. Thus, we aim to properly understand if well-established ML models, less data-hungry and complex than CNN models, can help to predict both the ischemic regions and have the potential to assist medical doctors during treatment decisions. We give a comparison of the proposed method with different parameters and with thresholding methods from the literature. This paper contributes with the following:

- Proposing a fully-automatic ML-based algorithm to segment both penumbra *and* infarct core regions in patients affected by AIS, since a correct visualization of the salvageable tissue will guide treatment decision better,
- Using the parametric maps as input, due to their wide usage by medical doctors for early assessment of ischemic strokes,
- Training the models using a dataset with different groups of patients based on their level of vessel occlusion, generalizing the models and the training data and not restricting the type of patients that can be tested,
- Adopting as ground truth, images annotated by expert neuroradiologists directly from the parametric maps based on CTP,

- And finally, testing different ML algorithms and parameters to find the most suitable approach. Both a single-step approach, segmenting normal brain, penumbra, and core in one go; and a two-step approach, segmenting penumbra and core individually before combining them, were tested. This was further compared to thresholding approaches.

9.2 Data Material

9.2.1 Dataset and ground truth

Context

Stavanger University Hospital (SUS) serves a population of 365.000. Close to 450 patients with AIS are annually admitted to the hospital. All consecutive patients with suspected AIS having received intravenous thrombolytic therapy are prospectively listed in a population-based database. Information about clinical severity measured by the National Institutes of Health Stroke Scale (NIHSS, scoring scale assessing neurological deficit) on admission, and at discharge are available. Long term functional outcome measured by the modified Rankin scale (mRS, scoring scale assessing long term functional outcome) at 90 days are also registered, in addition to mRS on hospital admission.

Dataset

The dataset in this study comprises CTP scans from 152 patients between January 2014 and August 2020. 137 of these patients had an AIS with visible perfusion deficit. Patients with AIS were divided into the following groups: 77 patients with large vessel occlusion (LVO), and 60 patients with non-large vessel occlusion (Non-LVO). Additionally, 15 patients without ischemic stroke (WIS) who were admitted with suspicion of stroke, but turned out not to have a stroke in the diagnostic workup, were included in the dataset. Age, gender, and NIHSS score for the groups are shown in Table 9.1.

LVO was defined using CT angiography; occlusion of the internal carotid artery, M1 and proximal M2 segment of the middle cerebral artery, A1 segment of the anterior cerebral artery, P1 segment of the posterior cerebral artery, basilar artery, and vertebral artery occlusion were regarded LVO. Non-LVO was defined as patients with perfusion deficits and affection of

Table 9.1: Patient characteristics.

		LVO	**Non-LVO**	**WIS**
Age (average/range)		72 (39-94) years	75 (41-94) years	60 (27-85) years
Gender	Male	49 (64%)	37 (62%)	8 (53%)
	Female	28 (36%)	23 (38%)	7 (47%)
NIHSS score (maximum /minimum /average)	On hospital admission	38/0/13	19/0/6	14/1/3
	On hospital discharge	25/0/5	10/0/2	1/0/0

more distal arteries or with perfusion deficits without visible proximal artery occlusion.

Ground truth

Ground truth images are manually annotated by two expert neuroradiologists. The manual annotations are done using the entire set of the CT examination including the parametric maps from the CTP (CBV, CBF, TTP, T_{Max}), the maximum intensity projection (MIP) images, calculated as the maximum Hounsfield unit value over the time sequence of the CTP, providing a 3D volume from the 4D acquisition of CTP. Furthermore, the MRI examination performed within 1 to 3 days after the CT examination was used in assistance to generate the ground truth images. In-house developed software was used for the annotations.

9.2.2 Imaging protocol and Analysis

The CT scanners used for image acquisition were Siemens Somatom Definition Flash (installed in 2012) and a Siemens Somatom Definition Edge (installed in 2014), Erlangen, Germany.

Patients with suspected acute cerebral stroke with symptom onset within 4,5 hours prior to hospital admission were routinely investigated by NCCT of the head. If contraindications were excluded, intravenous thrombolysis bolus-dose was administered in the CT lab. Then CTA and CTP were performed. Technical details about the protocols are shown in Table 9.2.

Further, the CTP images were analyzed using the software "syngo.via" from Siemens Healthineers with manufacturer default settings to generate color-coded parametric maps (CBF, CBV, TTP, MTT, and T_{Max}).

Table 9.2: Computed Tomography Technical Protocol for Acute Ischemic Stroke.

	NCCT of the head	**CTA of the cerebral arteries**	**CT perfusion**
Patient position	Head first, supine	Head first, supine	Head first, supine
Spiral/sequence	Spiral	Spiral	Spiral
kV	120	100	80
mAs	280	160	200
Rotation time (s)	1	0.28	0.28
Slice collimation	3 mm c 20 x 0.6 mm	0.6 mm c 128 x 0.6 mm	5 mm c 32 x 1.2 mm
Pitch	0.55	1.0	-
X-care	Yes	No	No
IV contrast	No	60 ml Omnipaque 350 mg I/ml + 40 ml NaCl	40 ml Omnipaque 350 mg I/ml + 40 ml NaCl
Flow rate	-	5 ml/second	6 ml/second
Start delay	-	4 seconds	4 seconds, ≥60 seconds after CTA
Scan direction	Caudocranial		

9.3 Ischemic segmentation by Thresholding

Several studies define threshold values on some of the parametric calculations or on a combination of them to segment the ischemic stroke regions. The variability in the chosen thresholding value(s) is mainly due to the various vendors used for post-processing the parametric maps, the different definitions of the ground truth for the ischemic regions. It also lies in the decision of using the entire brain or just the ipsilesional hemisphere in statistical evaluations. Table 9.3 lists some of them in addition to information about their dataset, the number of patients, NIHSS score, time of stroke onset, vendor used, and their defined threshold values on different parametric maps. It also shows the different optimal thresholds that are proposed in each of these studies to segment either core, penumbra, or both.

Most of the listed studies evaluated their method by testing the mismatch between values from parametric maps derived from CTP images and the

Table 9.3: Information about the dataset and the threshold value(s) of the various research method analyzed.

Article	Patients	NIHSS (mean)	Vendor	Stroke onset	Follow-up Images	Threshold	
						Penumbra	Core
Bathla *et al.* [177]	39	7	Siemens	N.A.	≤ 24h	T_{Max}>6s	CBF<20%
Wintermark *et al.* [22]	130	15.3	Philips	≤ 12h	≤7d	MTT > 145%	CBV ≤ 2.0ml/100g
Campbell *et al.* [23]	49	16.5	Philips	≤ 6h	≤1h	T_{Max}>6s	CBF<31% (with TTP >4s)
Cereda *et al.* [24]	103	16	In-house	≤ 8h	≤3h	N.A.	CBF<38% (with T_{Max}>4s)
Bivard *et al.* [167]	180	12	Toshiba	≤ 6h	≤24h	TTP>+5s	CBF<50%
Murphy *et al.* [25]	25	15.1	General Electric	≤ 7h	N.A.	CBF≤ 25ml/100g CBV≤ 2.15ml/100g	CBF ≤ 13.3ml/100g CBV≤ 1.12ml/100g
Schaefer *et al.* [168]	55	14	General Electric	≤ 9h	≤3h	N.A.	CBF≤15% + CBV≤30%

corresponding follow-up DWI, as the gold standard. The only study which did not use DWI as ground truth for the ischemic regions is Murphy *et al.* [25]. They defined the core region 5 to 7 days after the onset of stroke in the NCCT images, while the penumbra was the difference between the infarct and ischemic region. Nevertheless, they state that this difference "could lead to an underestimation of the final infarct size". All the approaches displayed in Table 9.3, with differences in their chosen parametric maps and the optimal values, demonstrate the lack of a consensus to define the ischemic regions based on thresholds.

Only the default setting used by "syngo.via" to define the ischemic regions after the parametric maps generation (CBF<27ml/100ml/min to define tissue at risk and CBV<1.2ml/100ml for non-viable tissue) and the thresholds proposed by Bathla *et al.* [177] were implemented for comparison with our best method due to the usage of the same vendor and software system as our input. We compare with a gold standard based on expert assessment of the parametric maps and manual delineation of the regions since these expert assessments are used normally for treatment decisions and are clinically relevant.

9.4 Machine Learning Approaches

Applying ML algorithms in the field of medical image analysis is rapidly growing [219]. To train state-of-the-art ML models, patient data sets that have the necessary size and quality of samples are needed. Given that the patient data is protected by strict privacy and security rules this can be a challenge, however, if the necessary training set is available to train appropriate ML algorithms, good prediction models can be obtained. The

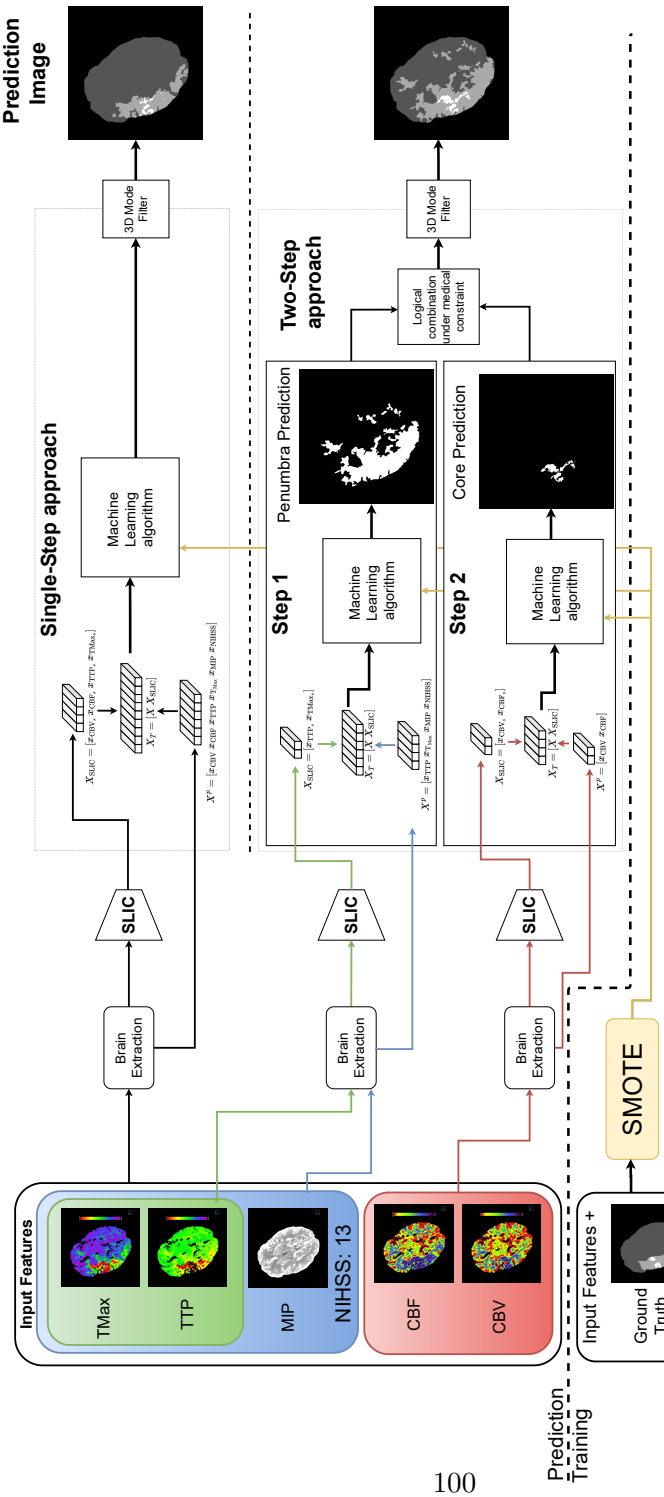


Figure 9.2: Visual description of the proposed multi-classification methods: for the *Single-Step* approach, all the parametric maps are adopted as input features for the ML algorithm to generate a final prediction image. The *Two-Step* approach work in a different way: Step1 takes in input six features for each pixel inside the brain and generates a binary map to classify the penumbra region(s) in a brain slice; Step2 takes in input 4 features, for each pixel, from different parametric maps and returns as output a binary map containing the predicted core region(s) if any. The final prediction combines the two binary maps only including the core regions that are inside the penumbra regions. A final post-processing step using a 3D mode filter is implemented. SLIC refers to the algorithm to extract superpixel regions. Color in the online version.

ML models tested in this study include Support Vector Machine, Decision Tree learning, and Random Forest. Each ML algorithm uses in input a training set $T = \{(x_1, y_1), \ldots, (x_T, y_T)\}$, composed of x_i features vectors and the relative y_i class label.

Support Vector Machine (SVM) is an algorithm used for binary classification that creates a line or a hyperplane, which separates the features from the input data into classes. In 1992, Boser *et al.* [203] proposed a supervised classification algorithm that has evolved into SVM as we know it today.

Decision Tree learning (DT), firstly introduced by Breiman *et al.* [204], is an efficient classification technique that creates a tree-like structure by computing the relationship between independent features and a target. DT covers both binary and multi-class classification. The tree splits into branches by using conditions at each internal node and the end of the branch that does not split anymore is the decision (leaf).

Random Forest (RF) is a supervised learning algorithm and the "forest" consists of an ensemble of decision trees. To classify a new object from an input vector, the input vector is fed to each tree in the forest and each tree casts a unit vote for the most popular class at the input vector. Finally, the forest chooses the classification having the most votes. Breiman *et al.* proposed this algorithm to minimize a possible overfitting problem generated by the usage of a single DT [205].

9.5 Proposed Method

In this paper, we test a single and a two-step method for segmenting core and penumbra in patients suspected of AIS using machine learning based on the parametric maps (CBF, CBV, TTP, and T_{Max}), derived from CTP datasets acquired at admission, the MIP map, and the NIHSS score. Various stages are performed during the proposed methods: (1) *Brain extraction and data imbalance:* extracting the brain tissue from the parametric maps to use only the pixel values inside the brain as input features, (2) *SLIC:* obtaining the 3D superpixel version of the parametric maps (CBF, CBV, T_{Max}, and TTP), (3) *Machine Learning algorithm:* Feeding the features from the parametric maps and their generated superpixel to our implemented machine learning algorithms to predict the ischemic regions.

Fig. 9.2 shows the flowchart of our proposed methods. In the reminder of the paper, we call them ***Single-Step*** and ***Two-Step*** approaches. The

features used for the proposed methods are the four parametric maps, the MIP map, and the NIHSS score. The input to the *Single-Step* method is all the aforementioned features (top part of Fig. 9.2) and it classifies both core and penumbra simultaneously. The *Single-Step* approach was tested with the DT and RF algorithms, but not with the SVM model since our implemented SVM model performs only binary classifications. In addition to the *Single-Step* method, we test another multi-stage classification method, which is simply adapted from the way neuroradiologists at SUS perform during the treatment decision process. The *Two-Step* approach is based on:

Step1: Takes as input the MIP, TTP, and T_{Max} maps, plus the NIHSS score; it performs a prediction of the penumbra region and outputs a binary image showing the predicted penumbra.

Step2: CBV and CBF parametric maps are used as input; it predicts the ischemic core resulting in a binary image.

9.5.1 Brain extraction and Data imbalance

We introduce a preprocessing step to extract the brain tissue from the whole image and work with pixels within the brain tissue (BT). In the reminder of the paper, the set of pixels belonging to the brain tissue for all patients p is called $\mathrm{BT} = \bigcup \mathrm{BT}^p$, while the various parametric maps are called $\mathrm{CBF}^p, \mathrm{CBV}^p, \mathrm{TTP}^p$, and $\mathrm{T_{Max}}^p$. This step helps to balance the classes inside the dataset. Moreover, we convert the pixel values into a $[0, 1]$ interval for each input feature based on the color bar on the right of each corresponding parametric map. Each input feature is mapped with the corresponding color in the bar and transformed into a value in the $[0, 1]$ interval, where the value 0 corresponds to the bottom value in the bar, while the value 1 indicates the top value. This was performed to reduce each input feature into a single value instead of keeping all three color channels.

9.5.2 Superpixel (SLIC)

A modified version of the Simple Linear Iterative Clustering (SLIC) algorithm [121] is employed to generate superpixel regions in the parametric maps. The regions are based on the initial segmentation of the intensity values of the maps. Using SLIC, we stacked each slice to obtain a 3D superpixel version for each parametric map and used it as extra features

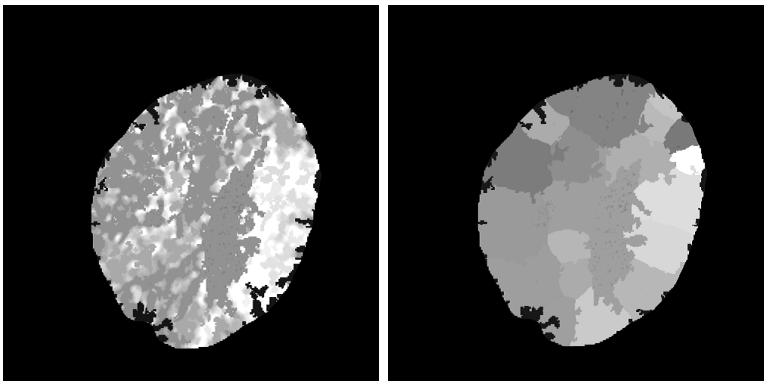


Figure 9.3: Visual comparison of a TTP map in grayscale (left) and the generated superpixel image (right) after the brain extraction.

as input to the model. These new features should help the models to consider the adjacent pixels along the third dimension (z-axis). In the reminder of the paper, the superpixel version of the parametric maps for a patient p are called: CBF $^{p}_{\text{SLIC}}$, CBV $^{p}_{\text{SLIC}}$, TTP $^{p}_{\text{SLIC}}$, and $\text{T}_{\text{Max}}{}^{p}_{\text{SLIC}}$. SLIC generates superpixel regions by clustering pixels utilizing their proximity and similarity in the image plane. An example of a normalized TTP map from one of the patients analyzed and the generated superpixel image is given in Fig. 9.3.

9.5.3 Machine Learning for core and penumbra

We implement three mainstream classical ML methods including, support vector machines, decision tree, and random forest. To the best of our knowledge, there exists no defined convention on which of the parametric maps should be used to detect core and which shows penumbra better. Let L^p be the number of pixels in BT^p. In the training phase, the totality of input features to these ML approaches are defined as a matrix. For a patient p, let the input features vector for CBV be:

$$x^p_{\text{CBV}} = \text{stack}(\text{CBV}^p(i,j))_{\forall (i,j) \in \text{BT}^p}$$

where x^p_{CBV} is a vector of size L^p. The stack function concatenates all the pixels in an image, row-by-row, into a vector. The input features totality of the parametric maps for a patient p is given by the matrix X^p. For simplicity we omit the p in the following notation where all definition are

on a single patient:

$$X = [x_{\text{CBV}}\ x_{\text{CBF}}\ x_{\text{TTP}}\ x_{\text{T}_{\text{Max}}}\ x_{\text{MIP}}\ x_{\text{NIHSS}}]$$

Defining $[\mathbf{1}]$ as a all-ones vector of length L^p, x_{NIHSS} is defined as $x_{\text{NIHSS}} = \text{NIHSS} \cdot [\mathbf{1}]$. In the same way, the input features totality for the superpixel version of the parametric maps is given by the matrix X^p_{SLIC}, defined as:

$$X_{\text{SLIC}} = [x_{\text{CBV}_s}\ x_{\text{CBF}_s}\ x_{\text{TTP}_s}\ x_{\text{T}_{\text{Max}s}}]$$

where x_{CBV_s} is represented as a vector:

$$x_{\text{CBV}_s} = \text{stack}(\text{CBV}^p_{\text{SLIC}}(i,j))_{\forall(i,j)\in \text{BT}^p}$$

The total matrix X_T is given by the combination of the two input features matrices depending on the model trained: $X_T = [X\ X_{\text{SLIC}}]$.

In the prediction phase, as shown in Fig. 9.2, the input features matrix X^p used for Step1 has 6 columns since the CBF and CBV parametric maps are excluded. Then, the model generates a binary map for the penumbra region over the entire image. Subsequently, the input feature matrix for the second step is derived only from CBV and CBF parametric maps plus their corresponding superpixel versions. This matrix has 4 columns as illustrated in Fig. 9.2. The selection of parametric maps is also in line with proposed methods in the literature [23, 167, 168] since TTP and T_{Max} are often used for detecting penumbra, while the other parametric maps are used for segmenting core regions.

To create the final prediction image, in the *Two-Step* approach, the binary predictions of core and penumbra are logically combined so the common white areas in both predictions indicate ischemic core in the final result. The logical AND combination simulate the medical constraint, where the ischemic core is limited to be inside the penumbra since the hypoperfused tissue always contains the dead tissue. For both the approaches (*Single-Step* and *Two-Step*), the patient's predictions pass through a 3D mode filter. This post-processing step helps to reduce unwanted noise and it also allows the predictions from a ML method to rely on the adjacent voxels in the z-axis, i.e. between adjacent slices.

9.6 Experiments and Results

9.6.1 Dataset division

In this paper data from 152 patients were used, 137 from AIS patients divided into two groups (LVO and Non-LVO) and 15 patients WIS but who

were admitted with suspicion of stroke. The dataset was randomly split into a training, validation, and holdout set, as described in Table 9.4, carefully dividing the LVO, Non-LVO, and WIS patients over the sets. The idea behind this division is to create a model that generalizes the classification of the ischemic regions working for all.

Table 9.4: Division in training, validation, and holdout dataset.

	Training (#; %)	**Validation (#; %)**	**Holdout (#; %)**	**Tot. (#; %)**
LVO	29; 37.7	29; 37.7	19; 24.6	77; 50.6
Non-LVO	24; 40	25; 41.7	11; 18.3	60; 30.5
WIS	6; 40	6; 40	3; 20	15; 9.8
Total	59; 38.8	60; 39.5	33; 21.7	152; 100

As many have reported, DWI is a questionable measure to describe the ischemic core [19, 24, 171, 172, 218], thus we propose to use manual annotations made by two expert neuroradiologists as the golden ground truth to assess both the ischemic regions during early stages and with different level of severity.

Even with removing the background and only considering the pixels inside the BT, the core and penumbra classes are still undersampled, leading to a class imbalance problem in the dataset. To overcome this problem, during the training phase we implement the Synthetic Minority Over-sampling Technique (SMOTE) algorithm [206] to over-sample the classes with a minor number of occurrences. SMOTE relies on the generation of synthetic examples on the difference between the feature vector under construction and its nearest neighbor. We over-sample the penumbra by a maximum of 5 times its standard amount and the core by a maximum of 20 times. These maximum values were chosen for their class importance and amounts. Before applying the SMOTE algorithm, the core and penumbra classes represent only 0.5% and 9.4% of the entire set respectively. After the application of the algorithm, they represent 7.6% and 36.5% of the dataset respectively.

9.6.2 Evaluation metrics

In all the experiments the predictions are compared with the ground truth and multi-class confusion matrices are generated. Our dataset is composed of three classes $\mathbf{C} \in \{\text{core}, \text{penumbra}, \text{healthy brain}\}$.

Table 9.5 presents a multi-class confusion matrix example for the core class: TP_c (True Positive) indicates the number of pixels predicted correctly as the core; FP_c (False Positive) represents the number of pixels classified

Table 9.5: Example of multi-class confusion matrix for the core class. TP =True Positive, FP = False Positive, FN = False Negative, and TN = True Negative.

		Predicted class		
		Core	**Penumbra**	**Healthy Brain**
Actual class	**Core**	TP_c	FN_c	FN_c
	Penumbra	FP_c	TN_c	TN_c
	Healthy Brain	FP_c	TN_c	TN_c

as core class but belonging to a different class; FN_c (False Negative) is the number of pixels predicted as a different class but labeled as the core in a ground truth image; TN_c (True Negative) displays the number of pixels that are classified as not core and belonging to one of the other classes. All the values in each multi-class confusion matrix are calculated based only on the number of voxels inside the BT, excluding all non-brain tissue voxels as the binary mask of brain vs background is found during pre-processing. From each confusion matrix of class $c \in \mathbf{C}$, we calculate the recall $\text{rec}_c = \frac{\text{TP}_c}{\text{TP}_c+\text{FN}_c}$, the precision $\text{prec}_c = \frac{\text{TP}_c}{\text{TP}_c+\text{FP}_c}$, and the Dice coefficient (equivalent to the F1-score) $\text{Dice}_c = \frac{2\cdot\text{prec}_c\cdot\text{red}_c}{\text{prec}_c+\text{rec}_c} = \frac{2\cdot\text{TP}_c}{2\cdot\text{TP}_c+\text{FP}_c+\text{FN}_c}$. The range for these values is $[0, 1]$. We also consider the Hausdorff distance between predictions and ground truth regions [220], and the absolute difference in the volume among the predictions (V_p [ml]) and the ground truth (V_g [ml]): $\Delta V = |V_g - V_p|$. The range value for the Hausdorff distance and ΔV is $[0, \infty]$ Bland-Altman plots were used to illustrate mean differences and limit of agreement between predicted volume and volume calculated from ground truth images.

9.6.3 Hyper-parameter optimization of ML algorithms

Before evaluating our methods, a series of hyper-parameter optimizations on the ML algorithms were performed using a Bayesian optimization. The input features for these optimizations, for a patient p, were solely based on X^p, without the usage of SLIC nor SMOTE algorithms. For DT and RF models, the hyper-parameters taken into consideration during the optimization were:

- the minimum number of leaf, with a range $[1, L^p/2]$,
- the maximum number of decision splits, in the range $[1, L^p - 1]$,

- Gini's diversity index, Twoing rule, and Cross-entropy for the split criterion to use,
- the number of decision trees in the model (1 for the DT algorithm, a range of $[1, 500]$ for the RF).

Table 9.6: Optimal hyper-parameters for the DT and RF algorithms divided by *Single-Step* and *Two-Step* approaches.

Method			**# DT**	**Split criterion**	**Min # Leaf**	**Max # Split**
DT	*Single-Step*		1	Cross-entropy	138	22489
	Two-Step	Step1		Cross-entropy	153	358000
		Step2		Cross-entropy	10	34427
RF	*Single-Step*		4	Gini	345	5535
	Two-Step	Step1	10	Cross-entropy	384	1400500
		Step2	10	Gini	2	20979

Differently, for the SVM model, we considered the following:

- Gaussian, Linear, and Polynomial kernel functions,
- the maximum penalty on the observations with a range of $[0.001, 1000]$,
- standardized vs not standardized features.

The values display in Table 9.6 show the best hyper-parameters for the DT and RF algorithms divided by *Single-Step* and *Two-Step* approaches, after an exhaustive set of experiments. Table 9.7 presents the optimal hyper-parameters for the SVM model. All the experiments described in the next sections use the same set of hyper-parameters defined in Table 9.6 and Table 9.7.

Table 9.7: Optimal hyper-parameters for the SVM model with the *Two-Step* approach.

Method			**Kernel Function**	**Max penalty**	**Standardize**
SVM	*Two-Step*	Step1	Gaussian	993.73	No
		Step2	Gaussian	0.487	No

9.6.4 Experiment 1 - ML algorithms and feature combination

For both the *Two-Step* and the *Single-Step* approaches, a series of six experiments were conducted to determine whether the inclusion of super-pixels as extra features is beneficial and to see if using SMOTE to balance

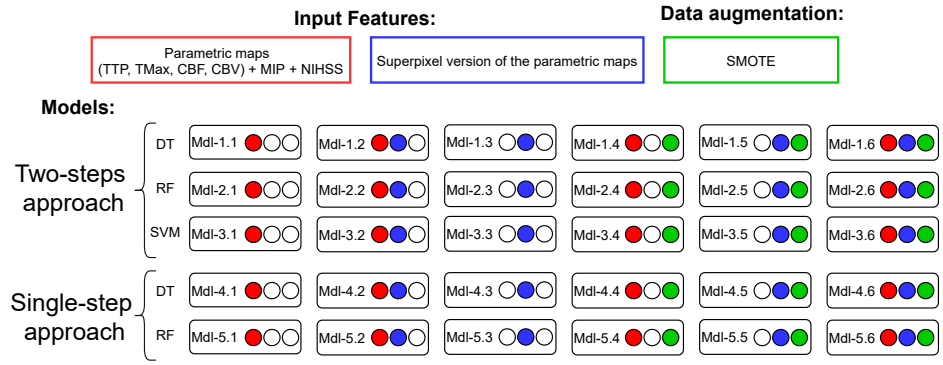


Figure 9.4: Description of the models implemented to test the two approaches, the input features used (the parametric maps with or without the superpixel regions), the usage of data augmentation (SMOTE). Experiments' names are included in the reminder of the paper. DT = Decision Tree; RF = Random Forest; SVM = Support Vector Machine. Color in the online version.

the classes during training gives better models. These six experiments were repeated for the different ML algorithms except SVM for *Single-Step* approach, due to our implementation of the approach which performs only binary classification.

Fig. 9.4 illustrates the 30 conducted experiments: (*Two-Step*×3 ML algorithms)×6 + (*Single-Step* ×2 ML algorithms)×6. The number of superpixel regions used for this set of experiments is 10. Fig. 9.5 shows the results for all models during the first experiment set taking into account all the various groups (LVO, Non-LVO, and WIS) together. The best model was selected mainly based on the averaging metrics in Fig. 9.5 for both the classes. Looking at Fig. 9.5, *Mdl-5.1* shows the best performances both for core and penumbra regardless of the group. Nevertheless, *Mdl-5.2* offers comparable results to *Mdl-5.1* in the majority of the metrics. Moreover, *Mdl-5.2* uses the superpixel regions as input features, on the contrary of Mdl-5.1, and the best number of superpixel regions should be investigated further.

9.6.5 Experiment 2 - Number of superpixels

After the first experiment set, we performed a series of empirical analyses on *Mdl-5.2* to choose the most adequate number of superpixel regions for the SLIC algorithm that produces the best results. We repeat a series of experiments using the *Mdl-5.2* starting with 25 total number of 3D

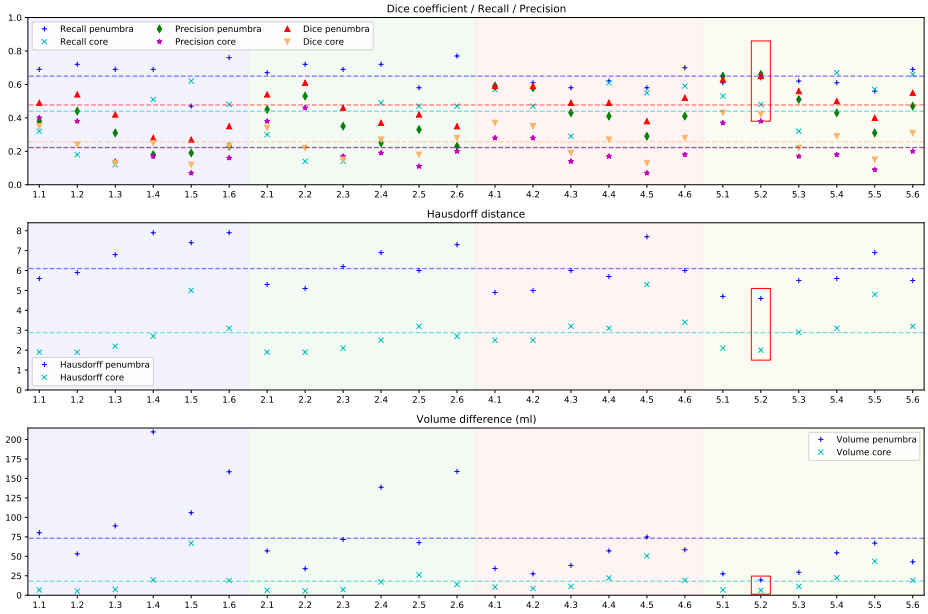


Figure 9.5: Results generated with the validation set on the 30 experiments described in Fig. 9.4. The x-axis contains the experiment IDs, while the y-axis refers to the statistic values. Each value represents the average of the patients in the validation set, including all the different severities. Note that for the top subplot we want high values, but for the mid and bottom subplots we want low values. All the experiments were tested with a number of superpixel regions equal to 10. The colored regions in the plot represent the division of the various experiments: blue, green, and red contain the experiments with the two steps approach using DT, RF, and SVM models respectively; yellow and purple have the experiments for DT and RF with the *Single-Step* approach. The colored horizontal lines display the average for the corresponding statistical measures. With the only exception of *Mdl-5.1*, *Mdl-5.2* (inside a red rectangle) is the one that presents the best tradeoff for all the evaluation metrics among the set of experiments. Color in the online version.

superpixel regions and continue by increasing the number until 600. The increment is 25 for each iteration. Fig. 9.6 presents the results obtained with different numbers of superpixel regions including 10 and also the total number of pixels in the image for *Mdl-5.2*. Fig. 9.6 shows the average metrics for the LVO and Non-LVO groups, and the average for the entire validation set (LVO, Non-LVO, and WIS). The combination of statistical metrics for both penumbra and core classes shows a clear difference when superpixel is used as shown in Fig. 9.6. As highlighted in the figure, 100 superpixel regions give slightly better results compared to the others. It is

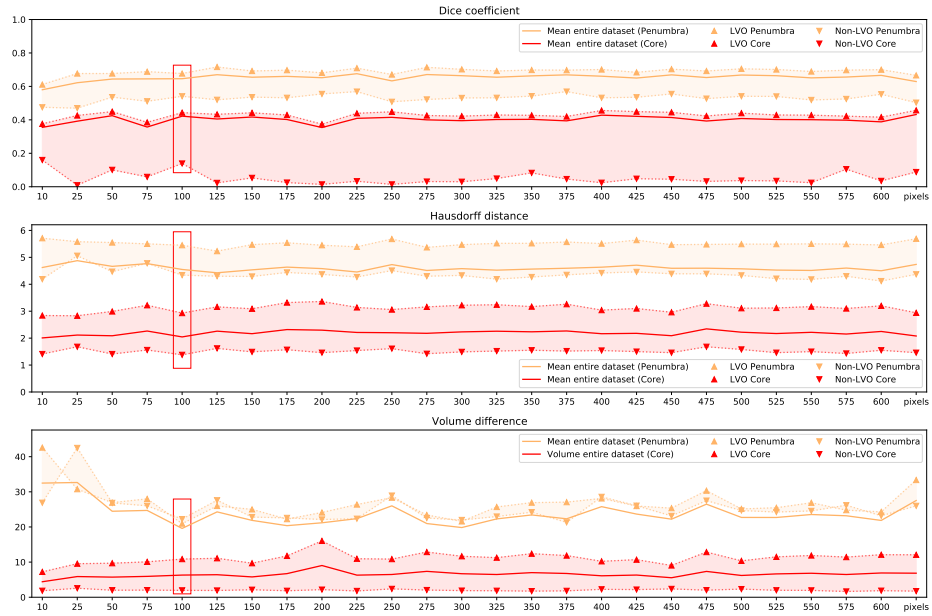


Figure 9.6: Various plots (Dice coeff., Hausdorff dist., ΔV) achieved with the validation set for selecting the best number of superpixel regions for *Mdl-5.2*. The x-axis indicates the number of superpixel regions. Results achieved by the best-performed model are highlighted with a red rectangle. Solid lines represent the average of the patients in the validation set, including all the different severities. Color in the online version.

noticeable that 100 superpixel regions yield the lowest volume difference for the penumbra class, which highly influenced the selection decision, the highest Dice coefficient for the core class on average, and significant results for the other metrics, and as such we propose to use 100 in further experiments.

9.6.6 Experiment 3 - Validate the superpixel result

The chosen number of superpixels, 100, was validated by repeating all the thirty experiments described in Fig. 9.4 and using 100 superpixel regions instead of 10 regions, which was used during the first evaluation round. Due to ineffective performance, SVM has been exempt from this validation step. Results are depicted in Fig. 9.7 showing the overall metrics for both the two ischemic regions. The model *Mdl-5.2* still shows the most promising results even compared with *Mdl-5.1*. It achieves the highest Dice coefficient and precision values for both the classes, and excellend ΔV results.

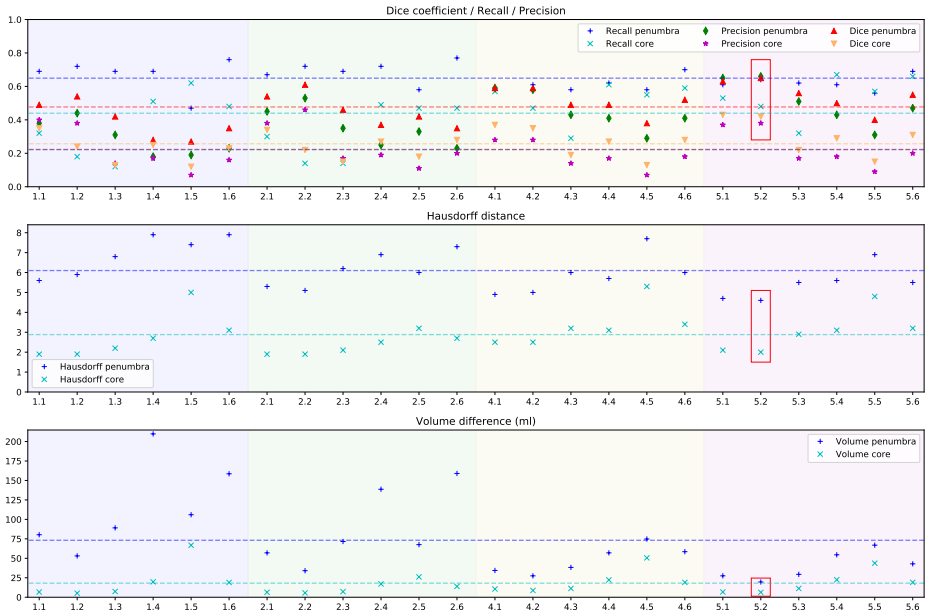


Figure 9.7: Statistical measures to select the best input data combination to use. All the methods were tested with the best number of superpixel regions (100). The best model (*Mdl-5.2*) is highlighted inside a red rectangle. Color in the online version.

9.6.7 Hyper-parameter optimization on the best model

The selected *Mdl-5.2* model went under a final step of performing optimization of its hyper-parameters with the current setting (100 superpixel regions) validated in the previous experiment set. We have taken into consideration the same hyper-parameters defined in Sec. 9.6.3 for RF. The new optimal hyper-parameters for *Mdl-5.2* are 48 number of DT, cross-entropy as the selected split criterion, 68982 as the maximum number of decision splits, and 315 as the minimum number of leaves.

9.6.8 Final test of the best model

We test the holdout set proposing *Mdl-5.2* as the best model, 100 as the most efficient number of superpixel regions, with the hyper-parameters defined in Sec. 9.6.7. A visual result of two sample predicted images along with ground truth and their corresponding parametric maps are shown in Fig. 9.8.

Furthermore, we remove the post-processing step (3D mode filter) and predict the regions to understand how the results are influenced by this step. Table 9.8 presents the results of the proposed best model, i.e. the *Single-Step* method with RF, *Mdl-5.2*, and 100 as the number of superpixels, in comparison with the same model without any post-processing step, the "syngo.via" default setting to define the ischemic regions, and the thresholding values proposed by Bathla *et al.* [177], since it is, to the best of our knowledge, the only research using "syngo.via" as vendor. Table 9.8 also depicts reported results from other thresholding methods ([22, 23, 25, 167]) which used other vendors for parametric maps acquisition and post-processing steps, thus a direct comparison with our model is not possible. Bland-Altman plots are used to visualize the predicted volume in comparison with the ground truth volume between the four methods compared in Table 9.8, shown in Fig. 9.9. For all rows, the statistical results are based solely on our holdout set to establish fair comparability with the other approaches. The results for two subsets of the data (LVO, Non-LVO) are presented separately, while for the WIS subset only ΔV is displayed.

Inter-observer variability

33 randomly selected patients (19 from the LVO subset, 11 from the Non-LVO subset, and 3 from the WIS group) were manually annotated by two different neuroradiologists, using the same criteria adopted for the creation of the ground truth images. The aim is to understand the inter-observer variability between two neuroradiologists. We investigate the inter-observer variability and compare it with the metrics of the automated method. Table 9.8 shows the inter-observer variability in the measurements of the ischemic regions for the two subsets of the data, LVO and Non-LVO, in comparison with the results achieved with our best method *Mdl-5.2*.

9.7 Discussion

We have proposed a multi-stage algorithm based on ML that automatically classifies ischemic core and penumbra regions in parametric maps generated from CTP images. The CTP scans were acquired from patients with AIS and WIS. In a real-life situation, medical doctors need to decide the treatment for a patient in a small time window; thus, an automatic

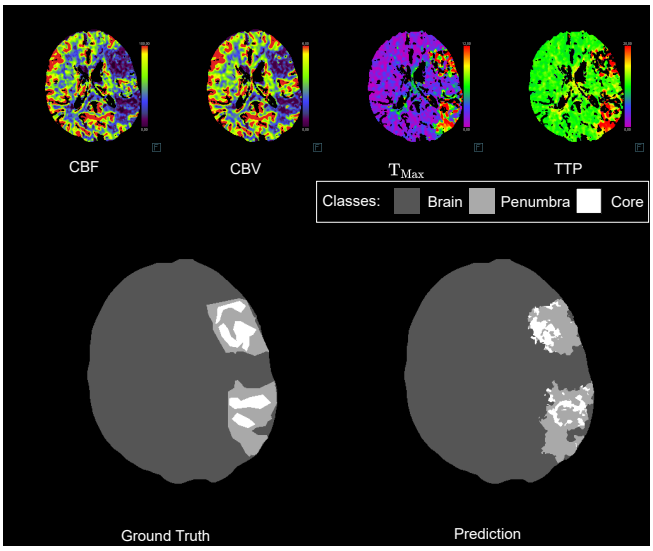
Table 9.8: Patients included in this table are all part of the holdout set. The results are presented for penumbra (core) regions. Comparison between various researches using thresholding values with the same vendor "syngo.via" (default setting and [177]) and our best model (*Mdl-5.2*). Predictions from [22, 23, 25, 167] are presented but they are not for comparison due to the usage of different vendor and/or post-processing steps for generating parametric maps. ‡ marks the results for the *Mdl-5.2* method without using any post-processing step. Inter-observer variability for two expert neuroradiologists (NR_1, NR_2) and the selected model *Mdl-5.2* is also presented. Note that for the Dice coefficient higher values are better (⇑), while for Hausdorff Distance and ΔV lower values are preferable (⇓).

Method	Vendor	Dice Coefficient ⇑			Hausdorff Distance ⇓			ΔV (ml) ⇓			
		LVO	Non-LVO	All	LVO	Non-LVO	All	LVO	Non-LVO	WIS	All
		Penumbra (Core)									
Best Method (*Mdl-5.2*) ‡	Siemens "syngo.via"	0.66 (**0.26**)	0.51 (0.03)	0.66 (0.26)	6.9 (4.8)	3.5 (0.9)	5.2 (3.1)	44.2 (16.2)	6.9 (**0.8**)	1.4 (**0.0**)	27.9 (9.6)
Best Method (*Mdl-5.2*)		**0.69** (**0.27**)	**0.56** (0.03)	**0.68** (**0.26**)	**6.5** (**4.3**)	**3.0** (**0.7**)	**4.8** (**2.7**)	**40.7** (**12.9**)	**4.9** (1.0)	**0.9** (**0.0**)	**25.1** (**7.8**)
Default Setting		0.31 (0.25)	0.11 (**0.04**)	0.27 (0.20)	7.8 (6.2)	5.6 (4.4)	6.6 (5.2)	67.5 (48.2)	51.8 (37.4)	3.7 (12.1)	58.2 (40.8)
Bathla *et al.* [177]		0.47 (0.17)	0.22 (0.03)	0.45 (0.14)	6.9 (6.9)	4.5 (4.7)	5.6 (5.7)	65.2 (65.3)	16.5 (44.5)	22.5 (6.6)	43.3 (53.5)
Other thresholding methods presented but not used for comparison											
Bivard *et al.* [167]	Toshiba	0.42 (0.19)	0.16 (0.03)	0.39 (0.15)	7.3 (6.5)	4.6 (4.4)	5.8 (5.4)	70.6 (52.9)	30.2 (36.0)	1.5 (9.1)	50.8 (43.3)
Cambell *et al.* [23]	Philips	N.A. (0.22)	N.A. (0.04)	N.A. (0.18)	N.A. (5.9)	N.A. (3.9)	N.A. (4.9)	N.A. (35.2)	N.A. (24.9)	N.A. (5.6)	N.A. (29.1)
Murphy *et al.* [25]	General Electric	0.17 (0.27)	0.08 (0.05)	0.16 (0.23)	7.5 (5.0)	4.8 (3.1)	6.1 (4.0)	96.7 (13.4)	21.1 (13.3)	8.6 (2.1)	63.5 (12.3)
Wintermark *et al.* [22]	Philips	N.A. (0.19)	N.A. (0.02)	N.A. (0.14)	N.A. (7.5)	N.A. (5.5)	N.A. (6.4)	N.A. (90.8)	N.A. (71.4)	N.A. (20.1)	N.A. (77.9)
Inter-observer variability											
NR_1 vs NR_2	Siemens "syngo.via"	0.80 (0.55)	0.67 (0.33)	0.79 (0.54)	5.1 (3.2)	1.9 (0.5)	3.6 (2.0)	33.3 (5.6)	5.5 (0.7)	0.0 (0.0)	21.0 (3.5)
Mdl-5.2 vs NR_1		0.69 (0.25)	0.51 (0.01)	0.68 (0.25)	6.6 (4.2)	3.3 (0.5)	5.0 (2.6)	53.6 (12.4)	8.7 (0.3)	0.9 (0.0)	33.8 (7.2)
Mdl-5.2 vs NR_2		0.71 (0.30)	0.56 (0.03)	0.70 (0.30)	6.4 (4.2)	3.0 (0.7)	4.8 (2.9)	38.6 (12.2)	4.9 (1.0)	0.9 (0.0)	23.9 (7.3)

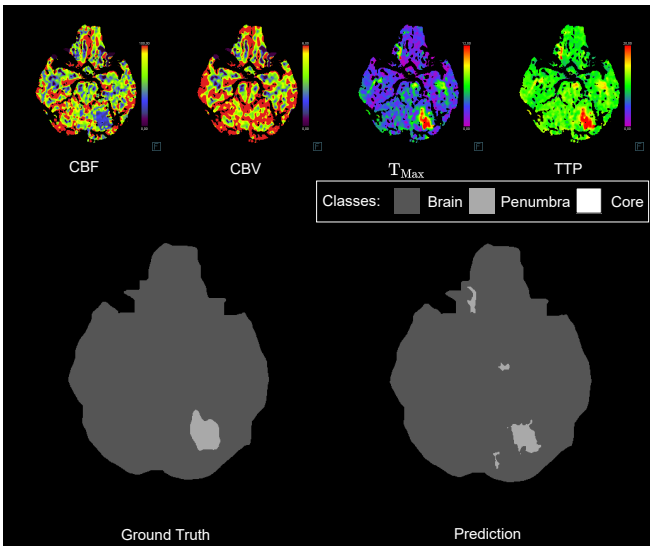
approach can be valuable. Expert assessments used as ground truth are commonly implemented in clinical use in many applications. We consider it to be a good method to interpret the ischemic regions, due to the lack of consensus on thresholding methods and the recent oppositions over the de-facto DWI as the gold standard [19, 171, 172, 218]. Nevertheless, these assessments present some variability among the experts (Table 9.8), thus an automatic approach might present some advantages during analysis and can aid medical doctors in rapid recognition of ischemic regions. We have trained our method with ground truth images directly acquired from the CTP parametric maps, MIP, and follow-up images. This results in better and more precise visualization of the two ischemic regions in the brain: the salvageable (penumbra) and the irreversibly damaged tissue (infarct core). Fast and correct visualization of the penumbra will guide the treatment better since it is fundamental to treat patients where relevant tissue can be saved, and not invest a lot of resources and time in trying to save tissue that is already irreversibly damaged and where the treatment might even harm the patient due to the risk of hemorrhage.

The criteria to select the best method was based on a study of various implemented experiments and their relative statistical results. First, we performed a set of thirty experiments described in Sec. 9.6.4 and in Fig. 9.4, to select the right features and model. From the relative outcomes in Fig. 9.5, the results provided by *Mdl-5.2* (RF with *Single-Step* approach using all parametric maps at once) produces considerable statistical measures in the majority of the metrics, regardless of the severity group or class. It is interesting to notice that the *Single-Step* approach generates better results or all metrics but the *Two-Step* approach with RF produces slightly better results in the Hausdorff distance for the core class. Results for irreversibly damaged tissue for SVM models were not taken into consideration since these models fail to predict the mentioned class.

Subsequently, we applied a different number of superpixel regions to *Mdl-5.2* to find one that gives the best prediction results (Sec. 9.6.5, and in Fig. 9.6). It is clear that the results are not the best without applying superpixel, however, there is not a clear difference between different numbers of superpixel regions; Dice coefficient and Hausdorff distance outcomes do not present large discrepancies during the increment of superpixel regions; the metric influencing the final decision was the volume difference due to its drastic drop for the selected number of superpixels for the penumbra class and a significantly low value for the core class. Another important factor that helped to select the best number of superpixel regions was



(a) A 71-year-old man with LVO and a baseline NIHSS score of 8.



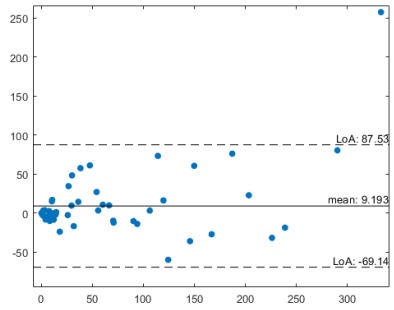
(b) A 60-year-old woman with Non-LVO and a baseline NIHSS score of 1.

Figure 9.8: Visual comparison with four parametric maps (top), ground truth images (left), and the corresponding predicted image with the best method (right) of one slice for two patients included in the testset, one labeled as LVO (a), the other as Non-LVO (b). The dark grey area is healthy brain tissue, the light gray area represents the penumbra, and the white region indicates the ischemic core. Color in the online version.

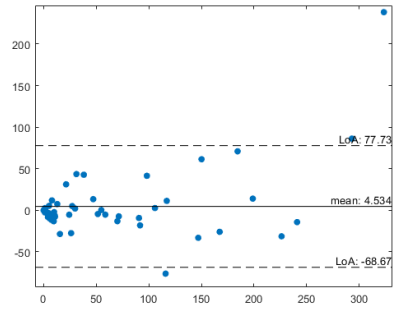
how the performances of the models differ with the various stroke severity groups. From Fig. 9.6 it is clear to notice that, among all the experiments in this set, *Mdl-5.2* presented the best tradeoff between the difference in volume and Dice coefficient for both the classes. One can argue that 125 superpixel regions give more or less similar results as 100 regions, however, ΔV is higher especially for penumbra regions, meaning that 125 superpixel regions provide an overestimation of the tissue at risk, especially for the LVO group.

Finally, we validated the selected superpixel number by applying it to the other experiments (Sec. 9.6.6). SVM was excluded from this step as it performed poorly from the beginning (reference to Fig. 9.5). As shown in Fig. 9.7, increasing the number of superpixel regions slightly improved the statistical measures for both classes. Moreover, results achieved by *Mdl-5.2* present higher precision and lower ΔV in comparison with the other models. The proposed method can classify correctly both penumbra and core in patients affected by a large vessel occlusion. The differences between the healthy and the ischemic tissue are more noticeable, in contrast with ischemic regions in patients with Non-LVO; an example is given in Fig. 9.8 for two brain slices of two patients affected by LVO (Fig. 9.8 (a)) and Non-LVO (Fig. 9.8 (b)). From the examples in Fig. 9.8 and the results in Table 9.8, our best method is shown to predict penumbra regions more precisely than core areas. In patients with LVO, the prediction of core regions achieved promising results. However, the detection of core regions in patients with Non-LVO is more challenging; the small core area can be difficult to classify correctly. This issue might be related to the limited number of samples for that particular class, since patients in the Non-LVO group does not always have a core region. We compared the performance of the proposed RF-based method with approaches based on thresholding suggested in the literature and the results are presented in Table 9.8; comparison is only performed with the default setting and values from Bathla *et al.* [177] due to the usage of the same vendor. Predictions from the other methods [22, 23, 25, 167] are just presented for visualization purposes; a comparison does not apply to the utilization of different vendors to generate the parametric maps, but it illustrates an important limitation of thresholding.

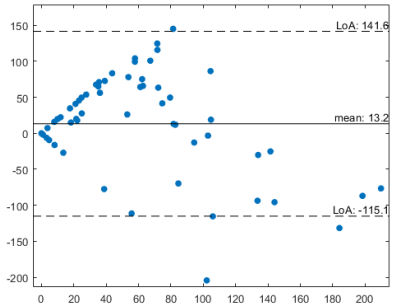
The proposed method (*Mdl-5.2*) performs better than the thresholding approaches concerning the evaluation metrics. The use of a post-processing step slightly increment the performances of the best method, as it is possible to evince from Table 9.8 and Fig. 9.9. The *Mdl-5.2* method (using a 3D



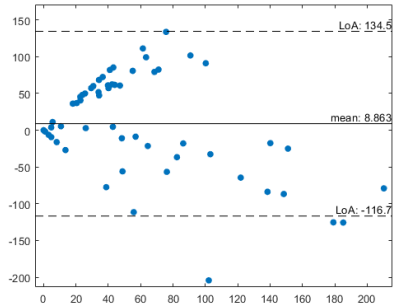
(a) *Mdl-5.2* without post-processing.



(b) *Mdl-5.2*



(c) "syngo.via" default setting



(d) Bathla *et al.* [177]

Figure 9.9: Bland-Altman plots of the volume calculated between the predictions and the ground truth images for model *Mdl-5.2* with (b), without (a) post-processing step, the "syngo.via" default setting (c), and the values presented by Bathla *et al.* [177] (d). Color in the online version.

mode filter as a post-processing step) achieved the highest metrics for all the classes regardless of the stroke severity level. The sole exception where the model does not perform well is with the core class for Non-LVO group since, as it is possible to evince from Table 9.8, it is the hardest class to predict correctly due to its limited number of samples and its narrow size in the BT.

Core predictions are slightly better than the one presented by the thresholding methods regardless of the group, while penumbra predictions are superior. This indicates a reliable understanding and agreement among ML predictions, threshold values, and neuroradiologists' annotations for the core regions. While, at the same time, it presents some uncertainties regarding the penumbra's definition. This might be related to the fact

that the infarct core and penumbra are two dynamic regions inside the brain and highly dependent on the acquisition time of CTP and DWI. The perfusion examination shows the perfusion at that specific time, the penumbra and core size may change rapidly. In many studies, MRI is not performed immediately after CTP. DWI, often used as the gold standard for defining the ischemic core, cannot define penumbra. Our method, relying the ground truth on both CTP generated right after hospital admission (parametric maps derived from CTP, and MIP) and follow-up images, seems to provide a reliable method to predict both penumbra and core. Note that we propose to make predictions only based on data available right after hospital admission. The areas defined as ground truth from the DWI sequence can over- or underestimate the ischemic core in individual patients, making it unrealistic to expect perfect concordance between ischemic core measurements on CTP and DWI [19, 24, 172, 218]. Other reasons are: first, they are not taking into consideration any spatial characteristics of an image; second, the values are very sensitive to image artifacts. Third, patients with contraindication to MRI, i.e. heart pacemaker, metal foreign body, might be excluded from studies where MRI and DWI images are used. Moreover, it is complicated to find an optimal threshold value for any group of patients. All the methods rely on selected thresholds, which might produce good results for a particular and predefined group, but it might not be the best for a single case study or the entire dataset studied. Their validation method relies on the comparison of the thresholding values with the clinical outcome of the patient; however, this is not perfect as the patient might have received treatment or the symptoms might have changed. Nevertheless, the delineation of the core should not be smaller due to treatment if the model delineates the core region correctly.

Table 9.8 shows the inter-observer variability in thirty patients divided by stroke severity into LVO and the Non-LVO subsets. There is a discrepancy between the results for the LVO and the Non-LVO subsets. Results for the LVO group have some similarities between the manual annotations and the *Mdl-5.2*. Nevertheless, manual annotations present better results in all the statistic measurements in comparison with the *Mdl-5.2* method in the Non-LVO subset. However, results in Table 9.8 illustrate the difficulties of achieving a consensus even among neuroradiologists.

9.8 Conclusion

We proposed an automatic multi-classification approach for segmenting both ischemic core and penumbra based on random forest using the parametric maps as input features, *Mdl-5.2*. We implemented other approaches based on thresholding, proposed in the literature, and compared them with our proposed method considering manual annotations as the ground truth generated from parametric maps. The method was trained with patients, both with AIS and WIS, grouped by different stroke severities. It shows good results for patients with large vessel occlusions, but not very good for patients with non-large vessel occlusions. Our method generates more precise results than the thresholding approaches for the two regions, but there is still room for improvement. We achieve an average Dice coefficient of 0.68 and 0.26, respectively for penumbra and core, for the three groups analyzed. We also achieve an average in volume difference of 25.1ml for penumbra and 7.8ml for core. Detecting ischemic core and penumbra regions in patients with non-large vessel occlusion can be very complicated, as shown in Fig. 9.8. Therefore, in the future, we plan to use approaches based on deep neural networks with 4D CTP volume as input instead of the parametric maps to work with the original acquired data.

Paper 2:
Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection

Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection

L. Tomasetti[1], M. Khanmohammadi[1], K. Engan[1], L. J. Høllesli[1,2], K. D. Kurz[1,2]

[1] Department of Electrical Engineering and Computer Science, University of Stavanger

[2] Stavanger University Hospital, Stavanger medical imaging laboratory (SMIL)

Published in the Proceedings of the Northern Lights Deep Learning Workshop (NLDL), 2022

https://doi.org/10.7557/18.6223

Multi-input segmentation of damaged brain in acute ischemic stroke patients using slow fusion with skip connection

Luca Tomasetti*[1], Mahdieh Khanmohammadi[1], Kjersti Engan[1], Liv Jorunn Høllesli[1,2], and Kathinka Dæhli Kurz[1,2]

[1]Department of Electrical Engineering and Computer Science, University of Stavanger, 4021 Stavanger, Norway
[2]Stavanger Medical Imaging Laboratory (SMIL), Department of Radiology, Stavanger University Hospital, 4019 Stavanger, Norway

Abstract

Time is a fundamental factor during stroke treatments. A fast, automatic approach that segments the ischemic regions helps treatment decisions. In clinical use today, a set of color-coded parametric maps generated from computed tomography perfusion (CTP) images are investigated manually to decide a treatment plan. We propose an automatic method based on a neural network using a set of parametric maps to segment the two ischemic regions (core and penumbra) in patients affected by acute ischemic stroke. Our model is based on a convolution-deconvolution bottleneck structure with multi-input and slow fusion. A loss function based on the focal Tversky index addresses the data imbalance issue. The proposed architecture demonstrates effective performance and results comparable to the ground truth annotated by neuroradiologists. A Dice coefficient of 0.81 for penumbra and 0.52 for core over the large vessel occlusion test set is achieved. The full implementation is available at: https://git.io/JtFGb.

1 Introduction

A cerebral stroke is the second most common cause of death among adults worldwide [21]. Cerebral stroke can be divided into two general categories: ischemic and hemorrhagic stroke. Ischemic stroke approximately represents 80% of the totality of the strokes [16]. The ischemic brain tissue is divided into two distinct regions during an ischemic stroke: the ischemic core (infarcted tissue) and the ischemic penumbra, a hypoperfused but viable tissue region. Fast and correct visualization of the salvageable penumbra and the irreversibly damaged core tissues can benefit medical doctors for treatment planning in acute stroke patients (AIS).

Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) are the two of the modalities used to diagnose acute stroke patients [6]. CT is preferred in many centers due to its high sensitivity for detecting hemorrhage, rapid scan times, and widespread availability. Information about clinical severity are calculated, at hospital admission, using the National Institutes of Health Stroke Scale (NIHSS), and color-coded parametric maps (2D PMs) are generated using the 4D CT Perfusion (CTP) imaging usually performed immediately after hospital admission. PMs are estimated to evaluate the changes in the tissue density over the injection of a contrast agent over time. Time-to-peak (TTP), time-to-maximum (T_{Max}), cerebral blood flow (CBF), and cerebral blood volume (CBV), are all examples of PMs, derived from pixel information of a time density curve, generated from a CTP study [11]. Also, the maximum intensity projection (MIP) is found as the maximum Hounsfield unit value over the time sequence of the CTP providing a 3D volume. In addition to diagnosing acute stroke, CT is also necessary for treatment decisions, with CTP being an essential modality with the ability to assess the penumbra and core.

Deep neural network (DNN) models have been proven to be an effective and beneficial tool for classification and segmentation tasks in many medical image analysis applications. Various research

*Corresponding Author: luca.tomasetti@uis.no

Abstract:
Time is a fundamental factor during stroke treatments. A fast, automatic approach that segments the ischemic regions helps treatment decisions. In clinical use today, a set of color-coded parametric maps generated from computed tomography perfusion (CTP) images are investigated manually to decide a treatment plan. We propose an automatic method based on a neural network using a set of parametric maps to segment the two ischemic regions (core and penumbra) in patients affected by acute ischemic stroke. Our model is based on a convolution-deconvolution bottleneck structure with multi-input and slow fusion. A loss function based on the focal Tversky index addresses the data imbalance issue. The proposed architecture demonstrates effective performance and results comparable to the ground truth annotated by neuroradiologists. A Dice coefficient of 0.81 for penumbra and 0.52 for core over the large vessel occlusion test set is achieved. The full implementation is available at: https://git.io/JtFGb.

10.1 Introduction

A cerebral stroke is the second most common cause of death among adults worldwide [138]. Cerebral stroke can be divided into two general categories: ischemic and hemorrhagic stroke. Ischemic stroke approximately represents 80% of the totality of the strokes [4]. The ischemic brain tissue is divided into two distinct regions during an ischemic stroke: the ischemic core (infarcted tissue) and the ischemic penumbra, a hypoperfused but viable tissue region. Fast and correct visualization of the salvageable penumbra and the irreversibly damaged core tissues can benefit medical doctors for treatment planning in acute stroke patients (AIS).

Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) are the two of the modalities used to diagnose acute stroke patients [15]. CT is preferred in many centers due to its high sensitivity for detecting hemorrhage, rapid scan times, and widespread availability. Information about clinical severity are calculated, at hospital admission, using the National Institutes of Health Stroke Scale (NIHSS), and color-coded parametric maps (2D PMs) are generated using the 4D CT Perfusion (CTP) imaging usually performed immediately after hospital admission. PMs are estimated to evaluate the changes in the tissue density over the injection of a contrast agent over time. Time-to-peak (TTP), time-to-maximum (T_{Max}), cerebral blood flow (CBF), and cerebral blood volume (CBV), are all examples of PMs, derived from pixel information of a time density curve, generated from a CTP study [18]. Also, the maximum intensity projection (MIP) is found as the maximum Hounsfield unit value over the time sequence of the CTP providing a 3D volume. In addition to diagnosing acute stroke, CT is also necessary for treatment decisions, with CTP being an essential modality with the ability to assess the penumbra and core.

Deep neural network (DNN) models have been proven to be an effective and beneficial tool for classification and segmentation tasks in many medical image analysis applications. Various research groups have focused their effort on the study of ischemic strokes, and some methods have been developed for classifying and segmenting the infarct core [29, 182, 184, 221]. These methods rely on a set of PMs as input and ground truth images generated through follow-up images acquired hours after the stroke onset. Nevertheless, the methods mentioned above only segment *core* regions. However, it could be more beneficial to acquire knowledge also about the penumbra regions since it is crucial for the treatment decisions during the first stages of the ischemic stroke [25, 222]. To the best of our knowledge,

Tomasetti *et al.* was the first research group to segment both *core and penumbra* using machine learning and deep learning approaches [208, 222].

It is highly time-consuming to collect and label medical data, and transfer learning is a popular approach to solve problems related to medical images [76, 223]. Over the past years, there have been numerous examples of transfer learning architectures used for various tasks in disparate domains pre-trained with the ImageNet dataset [76, 83]. Additionally, early and slow fusion approaches, with or without inflation, have been proven to improve accuracy in video classification [83, 114], and medical diagnosis [46, 224]. An early fusion approach combines input information at the beginning of the process, allowing a network to increase the performances of the system using cross-correlation between data [225]. The slow fusion approach slowly merges input information throughout the model permitting higher layers to access more global information [114]. An inflating technique has been proposed to use pre-trained weights from image classification networks in video classification models, expanding the filters from 2D (image-based) to 3D (video-based) [83].

A proper understanding of *both* ischemic regions is a major requisite for initial treatment decisions; however, it has not been fully explored in the previous researches; thus, in this work, we propose a DNN architecture to simultaneously segment both core and penumbra regions in AIS patients. We implemented a structure that was inspired by a multi-scale model proposed by Wetteland *et al.* [76]; however, while Wetteland's model used the same image but with different magnifications as input, we propose a different approach. Our model uses a multi-input CNN with slow fusion, based on transfer learning from VGG-16 models pre-trained on ImageNet. We want to investigate if the usage of the PMs in combination with MIP volume and NIHSS as input can produce meaningful results in the segmentation of ischemic regions, as this is already calculated and in use in clinical settings. The paper contributes with:

- A fully-automatic DNN method (Fig. 10.1) to segment both *core and penumbra*, using color-coded PMs acquired shortly after hospital admission when an AIS is expected.
- A slow fusion multi-input approach is tested combining the PMs with other images and/or patient information.
- The model is trained and tested using a dataset of patients affected by different levels of vessel occlusion for generalizing the input.
- Manual annotations based on experts' assessment of the CTP with PMs and MIP are used as ground truth.

10.2 Dataset

The study was approved by the Regional ethic committee project 2012/1499. The dataset included CTP scans from 152 patients collected at Stavanger University Hospital (SUS) between January 2014 and August 2020. NIHSS score was available for all patients. Based on the level of vessel occlusion using CT angiography, large vessel occlusion (LVO) was defined as occlusion of a large, proximal artery. Non-LVO was defined as patients with perfusion deficits with occlusion of a smaller, more distal artery or with perfusion deficits without visible artery occlusion. Patients were divided into three groups based on the vessel occlusion severity: 77 patients with LVO, 60 Non-LVO patients, and 15 without ischemic stroke (WIS).

Two neuroradiologists with 16 and 3.5 years of clinical experience delineated ground truth core and penumbra regions in an in-house developed software tool. These delineations were based on visual information in the CBF, CBV, TTP, T_{Max} and MIP images acquired directly after the CTP at hospital admission. PMs and MIP have a 512×512 pixel resolution, displaying only the brain region. Each patient study contains a number of brain slices between 13 and 27 for each scan. Furthermore, the MRI examination performed within one to three days after the CT examination was studied and used in assistance to generate the ground truth images since follow-up MRI exhibits the final infarct core. The dataset was randomly split into training (58%), validation (20%), and testing sets (22%). The training set resulted in 89 patients (42 LVO, 38 Non-LVO, 9 WIS); the validation set resulted in 30 patients (16 LVO, 11 Non-LVO, 3 WIS), and the remaining 33 patients for testing (19 LVO, 11 Non-LVO, 3 WIS).

10.3 Method

The baseline model takes as input four PMs (CBF, CBV, TTP, T_{Max}) of each 2D brain slice, MIP images and, for some experiments, we also use the patient's NIHSS score as input. The slow fusion is done both by concatenating the feature vectors from the different inputs and introducing skip connections from the different levels of the convolutional part.

The architecture presents a structure that resembles the U-Net model [112]. Fig. 10.1 displays a representation of the architecture. The convolutional part of the network is based on four to five distinct VGG-16 networks. This part is used to extract low-resolution features. A detailed overview

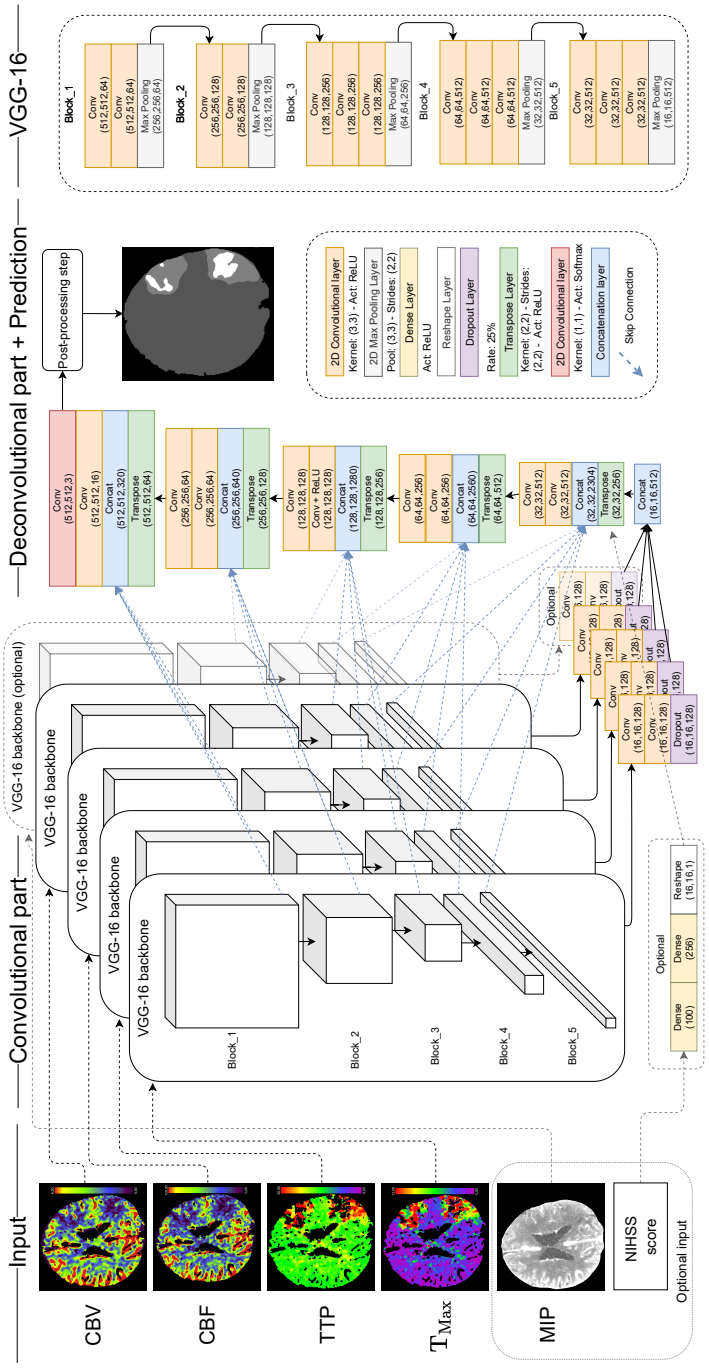


Figure 10.1: Proposed architecture with multi-input and slow fusion. The network has a convolutional part composed of four to five VGG-16 models. Skip connections combined with feature concatenation from VGG-16 models ensure slow data fusion. On the right, an overview of the feature extraction part of the VGG-16 architecture.

of a single VGG-16 architecture is displayed on the right of Fig. 10.1. At the end of each VGG-16 network, two convolutional layers followed by a dropout layer are inserted to regularize the model. The VGG-16 models were pre-trained with the ImageNet dataset. The deconvolutional section is a series of transpose layers followed by convolutional layers. The transpose layers receive in input a concatenation of the previous layer and the skip connection of the last convolutional layer for each block in the VGG-16 architectures. This step is performed through a concatenation layer, where various inputs are concatenated together along the channel axis, providing a slow fusion of the low-level features and the features of the VGG-16 models. All the convolutional layers use a ReLU activation function, except the last layer, which uses a softmax activation function that generates a probability vector for the three classes involved (i.e., core, penumbra, and healthy brain).

The model's output is a single 2D brain slice image with the same height and width dimensions as the input PMs. The 3D volume is generated by concatenating all the 2D brain slice images sequentially. To evaluate the accuracy of the predicted outputs, we use three distinct metrics: the Dice Coefficient, the Hausdorff distance, and the difference in volume among the predictions (V_p) and the ground truth images (V_g): $\Delta V = |V_g - V_p|$.

10.3.1 Class imbalance & Loss function

The dataset has imbalanced classes: 93.1% of the pixels belong to the healthy brain class, 6.2% penumbra, and the remaining 0.7% core. This issue is even more pronounced in the Non-LVO group, where 0.2% is core, 2.2% penumbra, and the remaining 97.6% belongs to healthy brain tissue. To overcome this issue, we build our model to focus on two aspects: the *loss function* and the *Non-LVO group*.

A generalized focal loss, based on the Tversky index (TI) [100, 226] was adopted. The TI is a generalization of the Dice similarity coefficient. The selected loss function was developed to address the data imbalance problem in medical image segmentation, improving the trade-off between precision and recall when training on small structures. γ, α, and β are hyper-parameters of the Focal Tversky loss (FTL) [100]. Furthermore, during training, we emphasized the misclassification of penumbra and core class, in patients in the Non-LVO group, with a higher penalty in the loss because of the evident imbalance among classes in this sub-group. A post-processing step is performed before generating the predicted outcome:

a binary mask of the entire brain slice is created based on the MIP image to force the segmentation inside a valid area (the brain tissue). Subsequently, from the softmax activation function, the highest probability value for each pixel was selected.

10.4 Experiments & Results

In the reminder of the paper we define the models as: $\mathrm{SF}_w(input)$, where $w \in$ {**F**rozen, **U**nfrozen, **G**radual fine-tuning} and the input are combination of PMs, MIP images (M), and NIHSS score (N). We use the FTL as the loss function for our network. We perform three different experiments: *Exp-1)* Hyper-parameter search for the poposed method (Fig. 10.1). *Exp-2)* combination of inputs and freeze/unfreeze variations of VGG-16; *Exp-3)* comparison of different input-fusion methods (Fig. 10.2). For all experiments, the same setting was used: Adam [104] was used as the optimizer function. The batch size was set to 2, and each model was trained for 1000 epochs. The validation FTL was monitored, and an early stopping was invoked if there was no improvement after 25 consecutive epochs.

In *Exp-1* a hyper-parameter search is done running a large number of hyper-parameter combinations (over the same model) for finding the optimal values for the given task. We ran experiments with distinct values for FTL hyper-parameters $\gamma \in [1, 3], \alpha \in [0, 1]$, and $\beta \in [0, 1]$ as shown in Table 10.1, where each value represents the average of the patients in the validation set for the different severity levels.

For *Exp-2* we have combined our slow fusion multi-input baseline model with different combinations of inputs to understand if diversity in the input can improve training the model. The various combinations are shown in Table 10.2 with the corresponding results. From the baseline input (PMs), we added MIP images as input or the NIHSS score or combined both MIP images and NIHSS score. Models for all these experiments were based on a multi-input with slow fusion. VGG-16 models were trained with frozen weights, unfrozen weights, and a gradual fine-tuning approach for each experiment. The latter setting was developed in three steps: first, the model was trained with all the VGG-16 weights frozen; secondly, after monitoring the validation loss having no improvements for 25 consecutive epochs, the bottom half of the weights were unfrozen, and the training continued; at last, when no improvement in the validation loss was detected

Table 10.1: *Exp-1*: hyper-parameters search for the FTL loss, model SF_F(PMs). Each value represents the average of the patients in the validation set for the different severity levels and their standard deviation (SD).

Model	Parameters			Dice Coefficient (Avg.) ± SD			
				LVO		Non-LVO	
	γ	α	β	Penumbra	Core	Penumbra	Core
SF_F(PMs)	1	1	1	0.68±0.2	0.29±0.3	**0.30±0.3**	0.13±0.2
		0.3	0.7	0.36±0.2	0.37±0.3	0.06±0.1	0.21±0.3
		0.5	0.5	0.66±0.2	0.35±0.3	**0.30±0.4**	0.19±0.2
		0.7	0.3	0.68±0.2	0.31±0.3	0.28±0.3	0.17±0.2
	4/3	0.3	0.7	0.68±0.2	0.35±0.3	0.29±0.4	0.17±0.2
		0.5	0.5	0.69±0.2	0.36±0.3	**0.30±0.3**	0.17±0.3
		0.7	0.3	**0.71±0.1**	0.37±0.3	0.27±0.3	**0.22±0.3**
	1.5	0.3	0.7	0.67±0.2	0.30±0.3	0.27±0.3	0.14±0.2
		0.5	0.5	0.67±0.2	0.30±0.3	0.29±0.3	0.14±0.2
		0.7	0.3	0.70±0.3	0.37±0.2	0.29±0.3	0.20±0.3
	2	0.3	0.7	0.67±0.2	0.00±0.0	**0.30±0.4**	0.00±0.0
		0.5	0.5	0.70±0.2	0.00±0.0	0.28±0.3	0.00±0.0
		0.7	0.3	0.70±0.2	**0.39±0.3**	0.29±0.3	0.20±0.3
	3	0.3	0.7	0.00±0.0	0.00±0.0	0.00±0.0	0.00±0.0
		0.5	0.5	0.70±0.2	0.00±0.0	**0.30±0.4**	0.00±0.0
		0.7	0.3	0.68±0.2	0.37±0.3	0.29±0.3	0.19±0.3

again, all weights were unfrozen, the training continued, and the validation loss was monitored. We have selected SF_G(PMs,N) as the proposed model.

To understand if using multi-input and slow fusion is suitable for this task, in *Exp-3* we compared it with two models: an early fusion (EF_G(PMs,N)), and early fusion with inflation (EFI_G(PMs,N)), adopting the same multi-input idea but with different fusion approaches. The inflation approach converts 2D into 3D layers, adding a temporal dimension. The inflation followed the idea of the I3D network by Carreira *et al.* [83], where they introduced video classification models with an inflated ImageNet pre-trained image classification architecture. The setting and hyper-parameters of these two new architectures are the same as the selected model to maintain a fair comparison. Fig. 10.2 presents an overview of the model architectures. For EF_G(PMs,N), the four input parametric maps are concatenated together over the channel axis to generate a single input volume passing through a

Table 10.2: Statistical results over the validation set for the models divided for *Exp-2* and *Exp-3*. The last two rows contain results for *Exp-3*. Dice coefficient, Hausdorff distance, and the volume difference are the metrics considered to select the best model. Each value exhibits the average of the patients in the validation set and the standard deviation (SD) for the distinct groups. For the metrics: ⇑ indicates that higher values are better, while with ⇓ lower values are preferable. Highlighted values represent the best result for a specific class and metric. The selected model is highlighted inside a red rectangle.

Model	Input			Dice Coeff. (Avg.) ± SD ⇑		Hausdorff Dist. (Avg.) ± SD ⇓		ΔV (Avg.) ± SD (ml) ⇓		
				LVO	Non-LVO	LVO	Non-LVO	LVO	Non-LVO	WIS
	PMs	MIP	NIHSS	Penumbra				Core		
Exp-2 - Layer weights: **Frozen**										
SF_F(PMs)	X			0.71±0.1 / **0.37±0.3**	0.27±0.3 / 0.22±0.3	5.9±1.0 / 2.7±1.8	3.2±1.5 / 0.9±0.8	27.0±28 / 9.4±20	10.0±15 / 0.8±1.3	9.8±8.0 / 0.5±0.5
SF_F(PMs,M)	X	X		0.69±0.2 / 0.36±0.3	0.29±0.3 / 0.20±0.3	5.9±0.9 / 3.0±1.7	3.1±1.4 / 1.0±0.8	25.9±27 / 10.0±21	5.5±6.0 / 0.8±1.0	8.1±7.0 / 0.5±0.3
SF_F(PMs,N)	X		X	0.70±0.2 / 0.36±0.3	0.29±0.3 / 0.16±0.2	5.6±1.2 / **2.3±1.9**	2.3±1.3 / 0.7±0.6	27.0±37 / 5.0±9.0	4.1±4.8 / 0.6±1.2	4.0±3.4 / 0.1±0.1
SF_F(PMs,M,N)	X	X	X	0.68±0.2 / 0.34±0.3	0.3±0.3 / 0.18±0.3	5.7±1.4 / 2.5±1.8	2.4±1.2 / 0.8±0.7	29.9±37 / 6.3±12	**2.8±3.0** / 0.5±1.0	3.9±4.0 / **0.0±0.0**
Exp-2 - Layer weights: **Unfrozen**										
SF_U(PMs)	X			0.70±0.2 / 0.34±0.3	0.29±0.4 / 0.24±0.3	5.4±1.3 / 2.7±1.7	2.1±1.5 / 0.7±0.7	29.8±36 / 6.5±14	4.2±6.9 / 0.5±0.7	**0.1±0.1** / **0.0±0.0**
SF_U(PMs,M)	X	X		0.70±0.2 / 0.36±0.3	0.34±0.3 / **0.24±0.3**	5.6±1.4 / 2.5±1.8	2.3±1.6 / 0.6±0.6	28.1±40 / 6.5±8.1	4.9±9.8 / 0.4±0.8	0.9±1.5 / **0.0±0.0**
SF_U(PMs,N)	X		X	**0.72±0.2** / 0.36±0.3	0.29±0.3 / 0.23±0.3	5.6±1.1 / 2.6±1.8	2.7±1.4 / 0.9±0.7	28.8±28 / 5.2±10	7.8±12 / 0.5±0.7	3.4±2.7 / 0.1±0.1
SF_U(PMs,M,N)	X	X	X	0.71±0.2 / 0.36±0.3	0.32±0.3 / 0.22±0.3	5.7±1.0 / 2.8±1.7	2.4±1.6 / 1.0±0.8	29.0±25 / 7.0±14	9.0±18 / 0.8±0.9	5.5±4.5 / 0.1±0.1
Exp-2 - Layer weights: **Gradual Fine-tuning**										
SF_G(PMs)	X			0.71±0.2 / 0.35±0.3	0.30±0.3 / 0.19±0.3	5.8±1.0 / 2.4±1.8	3.0±1.4 / 0.8±0.7	29.3±25 / 4.7±7.6	11.8±22 / 0.6±0.8	6.2±5.5 / **0.0±0.0**
SF_G(PMs,M)	X	X		0.69±0.2 / 0.35±0.3	**0.34±0.6** / 0.22±0.4	5.4±1.4 / 2.4±1.8	**1.9±1.2** / **0.6±0.5**	27.8±38 / 5.2±9.4	3.3±3.7 / **0.3±0.3**	2.4±3.8 / **0.0±0.0**
SF_G(PMs,N)	X		X	**0.72±0.2** / **0.37±0.3**	0.31±0.3 / 0.21±0.3	**5.3±1.4** / 2.4±1.7	2.3±1.4 / 0.7±0.6	26.0±35 / **4.4±7.0**	6.5±14 / 0.5±0.7	1.0±1.7 / **0.0±0.0**
SF_G(PMs,M,N)	X	X	X	0.68±0.3 / 0.34±0.3	0.31±0.3 / 0.18±0.3	5.5±1.3 / 2.5±1.8	2.3±1.3 / 0.7±0.6	26.9±35 / 5.1±9.6	5.5±9.2 / 0.5±0.8	0.9±1.1 / **0.0±0.0**
Exp-3 - Layer weights: **Gradual Fine-tuning**										
EF_G(PMs,N)	X		X	0.68±0.2 / 0.26±0.3	0.32±0.3 / 0.19±0.3	5.6±1.4 / 2.4±1.9	2.3±1.5 / 0.7±0.6	31.8±47 / 8.0±21	5.3±10 / 0.7±0.9	1.8±0.3 / **0.0±0.0**
EFI_G(PMs,N)	X		X	0.67±0.2 / 0.29±0.3	0.29±0.3 / 0.16±0.2	6.0±0.8 / 2.5±1.9	3.3±1.5 / 0.7±0.6	**25.5±28** / 5.3±10	6.3±7.1 / 0.6±1.2	6.7±4.7 / **0.0±0.0**

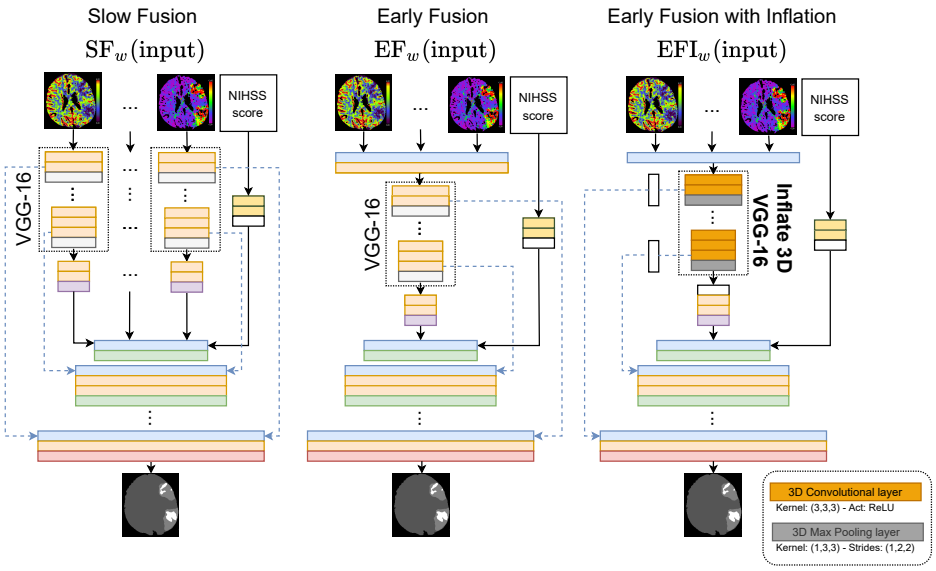


Figure 10.2: Overview of the three models used for comparison in *Exp-3*: the early fusion model ($EF_w(input)$); the early fusion with inflation ($EFI_w(input)$); $SF_w(input)$ is described in Fig. 10.1.

2D convolutional layer to reduce the channel dimension to feed it to a single VGG-16 backbone. Differently, EFI_G(PMs,N)'s inputs are concatenated over the time dimension; the filters and pooling kernels of the VGG-16 architecture are inflated, remodeling squared filters into cubic filters. The bottom of Table 10.2 shows the results of these two models in comparison with the other experiments performed.

SF_G(PMs,N) is seen as having good overall performance and is chosen as the proposed model. The model is tested with a previously unseen test set, and the performance is compared with manual annotations from two different experts. Table 10.3 presents the test results and the inter-observer variability in comparison with two expert neuroradiologists. Fig. 10.3 shows examples of predictions from six patients from the test set using the proposed model.

10.5 Discussion

We developed a multi-input CNN with early and slow fusion using transfer learning in this work. The proposed network aims to simultaneously

segment dead (core) and salvageable tissues (penumbra) in AIS patients. The proposed method is learned on patients with or without ischemic stroke and for different vessel occlusion severities. This generalization helps the model correctly segment most ischemic regions, regardless of the patient group. After a series of experiments, multi-input including NIHSS, by a slow fusion approach, SF_G(PMs,N), is chosen as the proposed method based on the total of the results.

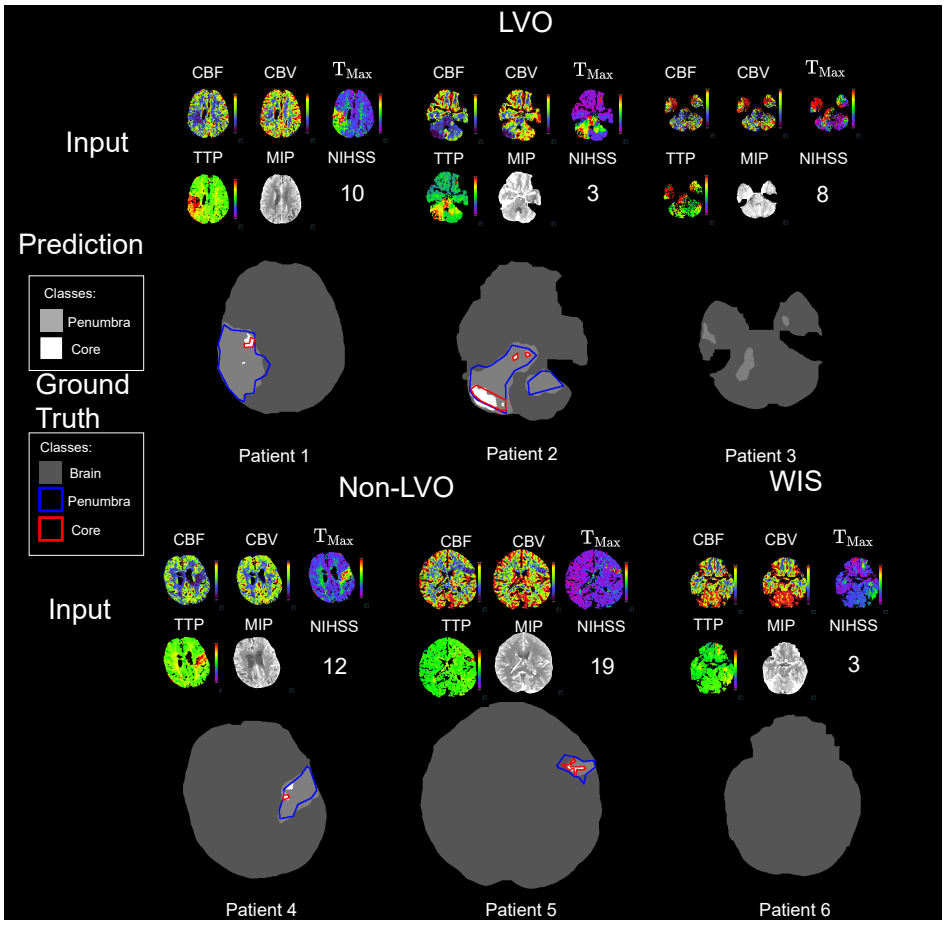


Figure 10.3: Prediction results for six test patients using the selected model, the set of all possible inputs, and the relative ground truth images.

A hyper-parameter search was performed for the first experiment. Table 10.1 shows the average Dice coefficient for selecting the optimal hyper-parameters for the FTL function. Our observations showed that $\gamma = 4/3$,

Table 10.3: Evaluation metrics for the selected method (SF_G(PMs,N)) over the 33 test patients (19 LVO, 11 Non-LVO, 3 WIS) manually annotated by two experts neuroradiologists (NR_1 and NR_2).

Model	Dice Coefficient (Avg.) ± SD ⇑			Hausdorff Distance (Avg.) ± SD ⇓			ΔV (Avg.) ± SD (ml) ⇓					
	LVO	Non-LVO	All	LVO	Non-LVO	All	LVO	Non-LVO	WIS	All		
	Penumbra						Core					
SF_G(PMs,N) vs (NR_1 & NR_2)	0.52±0.2 / 0.81±0.1	0.12±0.2 / 0.48±0.3	0.34±0.3 / 0.63±0.3	3.2±1.5 / 5.0±1.2	0.7±1.0 / 2.2±0.9	2.1±2.8 / 3.7±2.0	6.1±5.1 / 15.5±12	0.7±1.4 / 4.4±4.4	0.0±0.0 / 0.2±0.4	3.7±4.8 / 10.5±11		
NR_1 vs. NR_2	0.44±0.2 / 0.78±0.1	0.15±0.2 / 0.65±0.2	0.30±0.3 / 0.67±0.2	3.2±1.4 / 5.1±±1.0	0.5±0.9 / 1.9±1.4	2.0±1.8 / 3.6±2.2	5.6±4.3 / 33.3±28	0.7±1.9 / 5.5±9.2	0.0±0.0 / 0.0±0.0	3.5±4.2 / 21.0±26		
SF_G(PMs,N) vs. NR_1	0.57±0.2 / 0.84±0.1	0.12±0.2 / 0.48±0.3	0.37±0.3 / 0.64±0.3	3.0±1.4 / 4.9±1.1	0.7±1.0 / 2.2±1.0	1.9±1.7 / 3.6±1.9	6.2±5.7 / 11.2±11	0.7±1.4 / 4.4±4.4	0.0±0.0 / 0.3±0.4	3.8±5.2 / 7.9±9.4		
SF_G(PMs,N) vs. NR_2	0.44±0.2 / 0.78±0.1	0.1±0.2 / 0.43±0.3	0.29±0.3 / 0.59±0.3	3.3±1.4 / 5.4±1.1	0.5±0.8 / 2.7±1.4	2.1±1.8 / 4.0±2.1	6.8±5.9 / 29.2±30	0.3±0.6 / 8.2±9.6	0.0±0.0 / 0.3±0.4	4.0±5.5 / 19.6±26		

$\alpha = 0.7$, and $\beta = 1 - \alpha$ give satisfactory results for all the classes in the two groups. Thus we apply these parameters in the following experiments. From the validation results of Table 10.2 it can be seen that unfreezing the weights of the VGG-16 feature extractors improves the models but gives an overestimation of the volume for both the classes and also that the gradual fine-tuning approach gives a slight improvement when compared to unfreezing all weights from the start. The choice of freezing the parameters reduces training time since a smaller set of parameters needs to be learned; however, the statistical results are less than satisfactory; this could be since PMs are not included in the ImageNet dataset, then the weights are not optimized for these images. Therefore, at the cost of a longer training time, fully unfreezing the weights or using a gradual fine-tuning technique will allow the model to familiarize and learn this particular dataset more accurately.

From the validation results of the different gradual fine-tuned models, it is clear that multi-input fusion, including the NIHSS or MIP images, is better than only PMs. However, it is not entirely clear if including both NIHSS and MIP images improves the models compared to only including NIHSS in addition to PMs. Based on the results presented in Table 10.2, we select SF_G(PMs,N) as the proposed model. One can argue that SF_U(PMs,N) yields similar results, but SF_G(PMs,N) is chosen because of its lower ΔV for the LVO group, high Dice coefficient over the entire dataset, and satisfactory results for all the other metrics. Furthermore, *Exp-3* favored the proposed slow fusion approach over the two early fusion approaches, EF_G(PMs,N) and EFI_G(PMs,N).

Results from the 33 randomly selected patients constituting the test set using our selected proposed model (Table 10.3) show an average Dice coefficient of 0.34 for core and 0.63 for penumbra over the entire test set and, on the LVO set, 0.81 and 0.52, respectively. These results present higher or analogous values compared to the inter-observer variability in most of the metrics, regardless of the stroke's severity. A separate comparison of the predicted outputs with both the neuroradiologists' annotations demonstrates that the proposed architecture can achieve high statistic values, regardless of the neuroradiologist. This achievement can be considered valuable throughout the first stages of an AIS.

Predictions from six brain slices by six patients in the test set are shown in Fig. 10.3. Results display comparable regions as the ground truth images, both with LVO and Non-LVO groups, showing high Dice coefficient and promising results with the model. However, the third example in

Fig. 10.3 shows false-positive regions in the brain: this is possibly due to artifacts present in the generated color-coded PMs. We have noticed similar false-positive with all the other architectures and hyper-parameters as well. These false-positive might be avoided using 4D raw CTP datasets instead of the pre-generated PMs.

Several researchers have proposed methods with promising results to segment the ischemic core [29, 182, 184]. They all use PMs derived from CTP studies for their architectures. Nevertheless, their only focus was to segment the ischemic core without considering the penumbra; thus, excluding a critical aspect for medical treatment decisions. Furthermore, there was no differentiation in the severity level of AIS for the patients involved in their research.

10.6 Conclusion

The ability to achieve comparable results with two expert neuroradiologists for all the segmented classes (core and penumbra) is valuable to neuroradiologists during the first stages of an ischemic stroke. The proposed model might be a supportive tool that can help doctors to make treatment decisions.

Paper 3: CNN Based Segmentation of Infarcted Regions in Acute Cerebral Stroke Patients From Computed Tomography Perfusion Imaging

CNN Based Segmentation of Infarcted Regions in Acute Cerebral Stroke Patients From Computed Tomography Perfusion Imaging

L. Tomasetti[1], K. Engan[1], M. Khanmohammadi[1], K. D. Kurz[1,2]

[1] Department of Electrical Engineering and Computer Science, University of Stavanger

[2] Stavanger University Hospital, Stavanger medical imaging laboratory (SMIL)

Published in the proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB), 2020

https://doi.org/10.1145/3388440.3412470

Paper 4:
CT Perfusion is All We Need: 4D CNN Segmentation of Penumbra and Core in Patients With Suspected Acute Ischemic Stroke

CT Perfusion is All We Need: 4D CNN Segmentation of Penumbra and Core in Patients With Suspected Acute Ischemic Stroke

L. Tomasetti[1], K. Engan[1], L. J. Høllesli[1,2], K. D. Kurz[1,2], M. Khanmohammadi[1]

[1] Department of Electrical Engineering and Computer Science, University of Stavanger

[2] Stavanger University Hospital, Stavanger medical imaging laboratory (SMIL)

Published by the IEEE Access, 2023

https://doi.org/10.1109/ACCESS.2023.3336590

IEEE Access

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier xxxx

CT Perfusion is All We Need: 4D CNN Segmentation of Penumbra and Core in Patients With Suspected Acute Ischemic Stroke

Luca Tomasetti[1], (Member, IEEE), Kjersti Engan[1], (Senior Member, IEEE), Liv Jorunn Hollesli[1,2], Kathinka Dæhli Kurz[1,2], and Mahdieh Khanmohammadi[1],(Member, IEEE)

[1]Department of Electrical Engineering and Computer Science, University of Stavanger, Stavanger, 4021, Norway

[2]Stavanger Medical Imaging Laboratory (SMIL), Department of Radiology, Stavanger University Hospital, Stavanger, 4068, Norway

Corresponding author: Luca Tomasetti (e-mail: luca.tomasetti@uis.no).

All authors are with the BioMedical Data analysis group (https://www.uis.no/en/bmdlab). This study was supported and approved by the Regional ethic committee project 2012/1499.

ABSTRACT Stroke is the second leading cause of death worldwide, and around 87 % of strokes are ischemic strokes. Accurate and rapid prediction techniques for identifying ischemic regions, including dead tissue (core) and potentially salvageable tissue (penumbra), in patients with acute ischemic stroke (AIS) hold great clinical importance, as this can provide valuable information for diagnosis and treatment planning. Computed Tomography Perfusion (CTP) is often used as a primary tool for assessing stroke location, severity, and the volume of ischemic regions. Current automatic segmentation methods for CTP typically utilize pre-processed 3D parametric maps, traditionally used for clinical interpretation by radiologists. An alternative approach is to use the raw CTP data slice by slice as 2D+time input, where the spatial information over the volume is overlooked. Additionally, these methods primarily focus on segmenting core regions, yet predicting penumbra regions can be crucial for treatment planning.

This paper investigates different methods to utilize the entire raw 4D CTP as input to fully exploit the spatio-temporal information, leading us to propose a 4D convolution layer in a 4D CNN network. Our comprehensive experiments on a local dataset of 152 patients divided into three groups show that our proposed models generate more precise results than other methods explored. Adopting the proposed *4D mJ-Net*, a Dice Coefficient of 0.53 and 0.23 is achieved for segmenting penumbra and core areas, respectively. Using the entire 4D CTP data for AIS segmentation offers improved precision and potentially better treatment planning in patients suspected of this condition.

INDEX TERMS 4D Convolution, Acute Ischemic Stroke, Computed Tomography Perfusion, Deep Neural Network, Image Segmentation.

I. INTRODUCTION

NEUROLOGICAL disorders are the primary contributor to disability-adjusted life years and the second most prevalent cause of death globally [1], with cerebral stroke as the leading cause of these disorders. This study focuses on ischemic stroke, which constitutes the majority of cerebral strokes [2]. A precise and fast comprehension of the brain tissue affected by an ischemic stroke holds substantial value in guiding decision-making and treatment planning. An acute ischemic stroke (AIS) generally occurs if a segment of the supplying arteries of the brain is occluded by a blood clot and prevents the regular flow of oxygen-rich blood to the capillaries in the brain tissue. The ischemic area can roughly be divided into two different types: 1) *penumbra*, areas where the tissue is still vital but critically hypoperfused [3]; and 2) *core*, referring to non-salvageable tissue. If blood flow is not restored timely, penumbra regions may develop rapidly into irreversibly damaged core regions. Therefore, a fast and accurate understanding of ischemic areas to plan the treatment and tailor further procedures to every single patient is fundamental.

The recommended modalities for diagnostic imaging in AIS patients are Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) [4]. In the initial stages of an AIS, CT Perfusion (CTP) has proven to be a fast and beneficial tool for evaluating both diagnosis and prognosis [5]. MRI

Abstract:
Stroke is the second leading cause of death worldwide, and around 87 % of strokes are ischemic strokes. Accurate and rapid prediction techniques for identifying ischemic regions, including dead tissue (core) and potentially salvageable tissue (penumbra), in patients with acute ischemic stroke (AIS) hold great clinical importance, as this can provide valuable information for diagnosis and treatment planning. Computed Tomography Perfusion (CTP) is often used as a primary tool for assessing stroke location, severity, and the volume of ischemic regions. Current automatic segmentation methods for CTP typically utilize pre-processed 3D parametric maps, traditionally used for clinical interpretation by radiologists. An alternative approach is to use the raw CTP data slice by slice as 2D+time input, where the spatial information over the volume is overlooked. Additionally, these methods primarily focus on segmenting core regions, yet predicting penumbra regions can be crucial for treatment planning.
This paper investigates different methods to utilize the entire raw 4D CTP as input to fully exploit the spatio-temporal information, leading us to propose a 4D convolution layer in a 4D CNN network. Our comprehensive experiments on a local dataset of 152 patients divided into three groups show that our proposed models generate more precise results than other methods explored. Adopting the proposed *4D mJ-Net*, a Dice Coefficient of 0.53 and 0.23 is achieved for segmenting penumbra and core areas, respectively. Using the entire 4D CTP data for AIS segmentation offers improved precision and potentially better treatment planning in patients suspected of this condition.

12.1 Introduction

neurological disorders are the primary contributor to disability-adjusted life years and the second most prevalent cause of death globally [1], with cerebral stroke as the leading cause of these disorders. This study focuses on ischemic stroke, which constitutes the majority of cerebral strokes [4]. A precise and fast comprehension of the brain tissue affected by an ischemic stroke holds substantial value in guiding decision-making and treatment planning. An acute ischemic stroke (AIS) generally occurs if a segment of the supplying arteries of the brain is occluded by a blood clot and prevents the regular flow of oxygen-rich blood to the capillaries in the brain tissue.

The ischemic area can roughly be divided into two different types: 1) *penumbra*, areas where the tissue is still vital but critically hypoperfused [20]; and 2) *core*, referring to non-salvageable tissue. Fig. 12.1 represents a visual description of penumbra, in orange, and core, in red, after a scan at hospital admission. If blood flow is not restored timely, penumbra regions may develop rapidly into irreversibly damaged core regions. Therefore, a fast and accurate understanding of ischemic areas to plan the treatment and tailor further procedures to every single patient is fundamental.

The recommended modalities for diagnostic imaging in AIS patients are Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) [15]. In the initial stages of an AIS, CT Perfusion (CTP) has proven to be a fast and beneficial tool for evaluating both diagnosis and prognosis [53]. MRI with Diffusion-weighted imaging (DWI) or non-contrast CT (NCCT) are commonly utilized after treatment to assess the *final infarct areas* (FIAs)[167] (in dark gray color in Fig. 12.1). These imaging modalities are obtained hours or days after the patient's treatment.

CTP is a four-dimensional (4D) spatio-temporal examination to assess the passage of blood in the brain. It is performed by acquiring a series of three-dimensional (3D) CT scans of a specific portion of the brain at time intervals during contrast agent injection. By using an iodinated contrast agent, density changes in the brain tissue over time can be analyzed. The shape and height of the time density curve depend on the brain tissue's perfusion [18]. The abundance of images in the raw 4D CTP poses a challenge for neuroradiologists in detecting ischemic strokes. The 4D volume of CTP requires simultaneous evaluation of the propagation of contrast fluid across all spatial regions of the specific portion of the brain over time to identify ischemic regions. However, this becomes impractical due to the complexity and the sheer volume of images involved. This

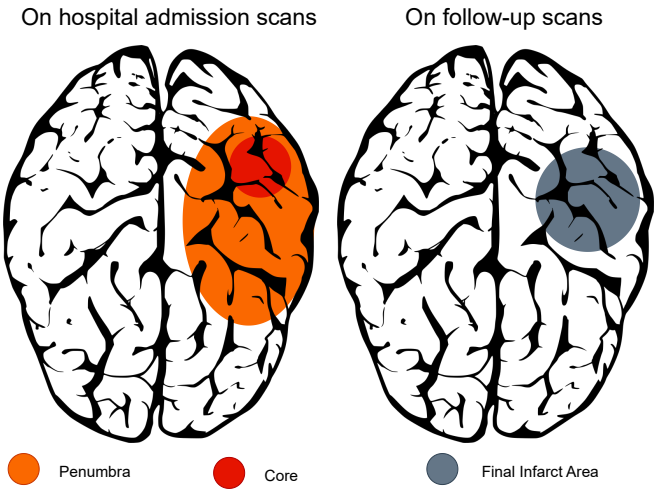


Figure 12.1: Visual representation of the various regions that can be defined after an IS: penumbra (in orange) and core (in red) can be localized during the initial stage of the neurological conditions, while the final infarct area (in dark gray) is determined during a follow-up study, after treatment.

complexity significantly prolongs the time required for analyzing the CTP study, which is highly detrimental in situations requiring accurate diagnosis and prompt treatment decisions.

To overcome this challenge, medical doctors rely on software estimating a set of clinically interpretable parameters related to the propagation of the contrast fluid combining all the temporal information for each pixel generated from the 4D CTP scan. This gives a set of 3D parametric maps (PMs) [18, 161]. Commonly used PMs are cerebral blood flow (CBF), cerebral blood volume (CBV), time-to-maximum (TMAX), and time-to-peak (TTP). CBF represents the blood supply in the brain at a given time; CBV refers to the blood volume present at a given time in a brain region; TMAX is the flow-scaled residue function in the tissue, while TTP shows the time until the contrast agent reaches the tissue [18]. Maximum intensity projection (MIP) is also usually generated. MIP images are calculated as the maximum Hounsfield unit (HU) value over the time sequence of the CTP, providing a 3D volume from the 4D acquisition of CTP. Although PMs provide helpful information about ischemic brain tissue, extracting them from the 4D CTP scans limits the spatio-temporal information only to specific subsets of information [189].

In this study, the objective is to develop a fully automated method to

segment the penumbra and core regions in AIS patients based on the raw 4D CTP input. The main contributions of this work can be summarized in three points:

(i) We propose a 4D convolution layer and use that to propose a DNN model, *4D mJ-Net*, to segment ischemic core and penumbra areas from 4D CTP scans.

(ii) We extend multiple existing methods for 3D CTP to 4D CTP to perform a comparison with the proposed 4D convolution solution.

(iii) To assess the results, we use manual annotations obtained by two expert neuroradiologists from the 4D CTP data upon patients' admission. We also demonstrate the feasibility of our proposed methods by comparing their performances with existing models that rely on different inputs.

12.2 Previous work

Several methods [24, 167, 238–240] have used thresholding techniques to predict the ischemic areas from the PMs. However, simple thresholding approaches over-simplify the complexity in AIS [181, 241].

In the past years, Deep Neural Networks (DNNs), and especially Convolutional Neural Networks (CNNs), have been successfully applied in numerous medical applications: image classification tasks [34, 64, 65, 242, 243], automatic video analysis [46, 244], and activity recognition [44, 245, 246]. Automatic image segmentation adopting U-Net structure [112] and its numerous variants have produced innovative outcomes for several applications [76, 229, 247–249].

Several DNNs have been proposed for AIS applications to predict and segment *only* the FIAs using CT studies in combination with PMs derived from CTP scans as input [182, 184–187]. Other researchers have proposed architectures to segment the ischemic lesion (i.e., the core) from the images obtained at hospital admission. Kasasbeh *et al.* [29] were the first to implement a CNN with a set of PMs as input for ischemic core segmentation. Tomasetti *et al.* [250] proposed a few-shot self-supervised architecture for hypoperfused (core + penumbra) tissue segmentation using a combination of PMs and raw scans as input of the model. The work demonstrated the feasibility of using self-supervised techniques to segment this tissue

type. Wedinger *et al.* [183] introduced a machine learning segmentation method to delineate hypoperfused tissue, demonstrating the capabilities of this methodology over classic thresholding approaches. They used four PMs as input features for their model. However, a general problem with all the methods mentioned above is relying on commercial CTP software and using heavily pre-processed information (i.e., PMs) rather than taking advantage of the totality of the raw 4D CTP scans.

DNNs are more suitable for discovering information from raw data. Nevertheless, relying on raw data (directly exploiting the temporal and spatial dimensions) is scarcely explored in the literature for AIS applications. The task is challenging because of the low contrast and low signal-to-noise ratio in the CTP scans. Relatively few studies proposed DNN models with encouraging results, exploiting the temporal dimension to assess acute stroke lesions using 4D CTP scans. Soltanpour *et al.* [189] utilized CTP images to create 2D matrices in which each row is a voxel, and each column is a time point. The 2D matrices are used as input for a model that shows encouraging results in differentiating healthy tissue from FIAs. Vries *et al.* [251] promoted a 2D+time symmetry-aware CNN-based architecture to segment FIAs using solely CTP scans. Their work estimated the irreversibly damaged areas, demonstrating the possibilities of using 4D CTP images for this task. Bertels *et al.* [190] used a U-Net-like structure for segmenting FIAs using CTP scans as an input plus contra-lateral information. Results were promising, but further research is needed due to their far-from-ideal registration of the contra-lateral information. Rosa *et al.* [252] introduced a two-step model for estimating FIAs using the 4D CTP series as input. They first generate an arterial input function and later deconvolve it with a singular value decomposition approach to find the infarction. Amador *et al.* [30] designed a framework based on the Temporal Convolution Network to predict AIS FIAs from 4D CTP studies. Due to memory constraints, they independently processed each 2D slice of the 4D CTP dataset. In their recent work, Amador *et al.* [31] also proposed an extension of their model where 3D+time tensors of the ipsilateral stroke hemisphere are used as input to predict FIAs. Robben *et al.* [158] proposed a DNN that predicts the FIAs directly from 4D raw CTP plus patients metadata. Their proposed architecture relied on a series of 3D Convolution layers; the input is a list of 4D CTP scans sampled at different resolutions. Their method presented promising segmentation results; however, their main target was to estimate the final infarct volume, allowing clinicians to simulate different treatments and gain insight into the procedures. They were not taking

into consideration the penumbra in their study. Plus, the quality of the ground truth images is debatable since they rely on NCCT follow-up images acquired between 24 hours and five days after patient's admission. It has been reported that FIAs can grow after 24 hours in NCCT measurements [159].

All the segmentation methods in the previous paragraph rely on ground truth labels obtained from DWI and/or NCCT hours or days after the patient's admission since they predict FIAs. Even though follow-up images (DWI and NCCT) represent the gold standard for estimating core [167], there are some limitations with these techniques [19, 171]. Follow-up images can only be used to assess FIAs but not penumbra regions. Plus, some studies have demonstrated that the detected FIAs can be partially reverse in DWI performed in an early time window [19, 172, 218].

Previous studies have indicated the potential of 4D data in AIS prediction [30, 31, 158, 189, 190, 251, 252], with a critical gap as they only consider predicting FIAs. An appropriate method is still required to simultaneously handle the spatio-temporal information for segmenting the ischemic *core* and *penumbra* regions. Understanding the penumbra's extension during the ischemic stroke's first stages is crucial for treatment decision [25, 253]. To the best of our knowledge, our work in Tomasetti *et al.* [253] using machine learning, and later in Tomasetti *et al.* [207, 208] using DNN, were the first and only to segment *both* core and penumbra areas. In [207, 253], the PMs were used as input, and in [208] 2D + time CTP images were segmented slice-by-slice.

Building upon our previous works [207, 208, 253] and filling the critical gap left by previous studies [30, 31, 158, 189, 190, 251, 252], in this paper, we present and investigate three novel models to segment the two ischemic regions (core and penumbra), where the input is the entire 4D CTP scans arranged in different ways to exploit the spatio-temporal nature of the data. We compare all models with previous work based on PMs [207] and slice-by-slice CTP [208], and two methods from the literature proposed by Amador *et al.* [30, 31].

12.3 Data Material

A section of the brain is repeatedly scanned during the passage 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 to highlights

changes in the tissue; the scan delay was four seconds. Each brain slice contains a fixed number of time points $t_{\max}$ representing the temporal dimension. The width and height of each image are 512×512 pixels with a resolution of 0.4258 mm/pixel and a slice thickness of 5mm. The first twenty time points are acquired every 1s, and the remaining ten images every 2s.

CTP scans from 152 patients collected between January 2014 and August 2020 formed the dataset. 137 of these patients had an AIS with a visible perfusion deficit. During the diagnostic workup, the remaining 15 patients were admitted with suspected strokes but were determined not to have suffered from a stroke episode after the diagnostic workup. The raw CTP studies are saved as DICOM files. Raw perfusion data from the CTP examination was used to generate PMs with the software "syngo.via" from Siemens Healthineers, with manufacturer default settings. The arterial input function was automatically selected, with few exceptions where it was chosen manually (e.g., severe cardiac failure).

The patients were divided into the following groups: 77 patients with large vessel occlusion (LVO), 60 patients with non-large vessel occlusion (Non-LVO), and the remaining 15 patients without ischemic stroke (WIS). Based on CT angiography, LVO was defined as occlusion of any of the following arteries: the internal carotid artery, M1 and proximal M2 segment of the middle cerebral artery, A1 segment of the anterior cerebral artery, P1 segment of the posterior cerebral artery, basilar artery, and vertebral artery occlusion. Non-LVO was defined as patients with perfusion deficit with more distal artery occlusion or with perfusion deficit without visible artery occlusion. The dataset is randomly split into a training, validation,

Table 12.1: Division in training, validation, and test dataset.

	Training (#; %)	**Validation (#; %)**	**Test (#; %)**	**Tot. (#; %)**
LVO	42; 54.5	16; 20.8	19; 24.7	77; 50.6
Non-LVO	36; 60	13; 21.7	11; 18.3	60; 30.5
WIS	9; 60	3; 20	3; 20	15; 9.8
Total	87; 58.6	32; 19.7	33; 21.7	152; 100

and test set. The percentage of the three subsets (LVO, Non-LVO, WIS) is equally distributed among the sets, as shown in Table 12.1.

12.3.1 Ground truth

The manual annotations are based on the entire CT dataset, including the PMs derived from CTP. MRI performed during the first days after hospital admission was also utilized. Two expert neuroradiologists manually annotated ground truth images by utilizing the complete set of the CT examination (NCCT, CT angiography, and CTP), which includes PMs from the CTP (CBV, CBF, TTP, TMAX) and the MIP images. The PMs were visually assessed. In general, ischemic regions with increased TTP and TMAX and reduced CBF but preserved CBV were considered as penumbra, while areas with additionally reduced CBV were deemed as core. Additionally, the MRI examination, including DWI, obtained within 1 to 3 days after the CT examination, and clinical information, was used to assist in generating the ground truth images. The annotations were performed using an in-house developed software in Matlab[1].

12.4 Background theory

12.4.1 Notation

Table 12.2 presents the various formal notations adopted in the remainder of the paper. Let the data obtained from a CTP scan be defined as a 4D tensor $V \in \mathbb{R}^{(X \times Y \times Z \times T)}$. After a series of pre-processing steps (details in Sec. 12.4.2), we define the 4D tensor as $\widetilde{V} \in \mathbb{R}^{(X \times Y \times Z \times T)}$. The four dimensions of a CTP scan are defined as width (X), height (Y), depth (Z), and time (T). The list of time points in the time dimension is given by $t = [t_j | \forall j \in \{1, 2, \cdots, t_{\max}\}]$, where $t_{\max}$ is the last time point of the list. We indicate how the notation superscript adopts the time dimension in the various inputs. Furthermore, we define $z = [z_i | \forall i \in \{1, 2, \cdots, z_{\max}\}]$ as the list of brain slices in the depth dimension, where $z_{\max}$ corresponds to the last slice. We illustrate how the depth dimension is being used in the inputs through the notation subscript. Fig. 12.3 displays the input combination of all the techniques.

All methods return a 3D output, segmenting the images P_{z_i} slice-by-slice. The segmented 2D image P_{z_i} corresponds to a brain slice z_i at index i. The predicted image P_{z_i} contains brain tissue segmented with the classes $\mathcal{C}$ (if any): healthy brain, penumbra, and core.

[1]The code is publicly available at the following link https://github.com/Biomedical-Data-Analysis-Laboratory/CTP-Matlab

Table 12.2: List of formal notations used in the paper.

Notation	Description
$t = [t_j \vert \forall j \in \{1, 2, \cdots, t_{\max}\}]$	List of all the time points.
$t_{\max}$	Last time point in the time dimension.
$z = [z_i \vert \forall i \in \{1, 2, \cdots, z_{\max}\}]$	List of all the slices.
$z_{\max}$	Last slice in the depth dimension.
$\mathcal{I} = \{i-1, i, i+1\}$	Set of indexes i, plus its neighbours $i-1$ and $i+1$.
$z_{\mathcal{I}} = \{z_{i-1}, z_i, z_{i+1}\}$	Set of slices: z_{i-1}, z_i, z_{i+1}.
$V \in \mathbb{R}^{(X \times Y \times Z \times T)}$	4D **raw** CTP input.
$\mathcal{C} = \{\text{healthy brain, penumbra, core}\}$	Set of classes.
$I_{z_i}^{t_j} \in \mathbb{R}^{(X \times Y)}$	2D brain slice z_i at time point t_j.
$P_{z_i} \in \mathbb{R}^{(X \times Y)}$	2D probability output of brain slice z_i.
$\widetilde{\cdot}$	Input **after pre-processing steps**.
$\bar{\cdot}$	**List** of inputs
$\varphi(\cdot)$	Concatenation function.
$\widehat{\cdot}$	**Concatenated** inputs after passing through $\varphi(\cdot)$.
$\bar{V}_{z_i}^{t} = [\widetilde{I}_{z_i}^{t_j} \vert \forall t_j \in t] \in \mathbb{R}^{(X \times Y)}$	List of 2D images $\widetilde{I}_{z_i}$ for all the time points t. Input for *2D-TCN* (Sec. 12.5.1).
$\widehat{V}_{z_i}^{t} = \varphi(\widetilde{I}_{z_i}^{t_j} \vert \forall t_j \in t) \in \mathbb{R}^{(X \times Y \times T)}$	2D+time volume of slice z_i. Input for *mJ-Net* (Sec. 12.5.1).
$\widehat{V}_{z_{\mathcal{I}}}^{t_j} = \varphi(\widetilde{V}_{z_{i-1}}^{t_j}, \widetilde{V}_{z_i}^{t_j}, \widetilde{V}_{z_{i+1}}^{t_j}) \in \mathbb{R}^{(X \times Y \times Z)}$	3D volume of slices $z_{\mathcal{I}}$ for a time point t_j.
$\bar{V}_{z_{\mathcal{I}}}^{t} = [\widehat{V}_{z_{\mathcal{I}}}^{t_j} \vert \forall t_j \in t] \in \mathbb{R}^{(X \times Y \times Z)}$	List of 3D volumes of slices $z_{\mathcal{I}}$ for all the time points t. Input for *3D-TCN* (Sec. 12.5.2).
$\bar{V}_{z_{\mathcal{I}}}^{t} = [\widehat{V}_{z_{i-1}}^{t}, \widehat{V}_{z_i}^{t}, \widehat{V}_{z_{i+1}}^{t}] \in \mathbb{R}^{(X \times Y \times Z \times T)}$	List of 2D+time volumes for slices $z_{\mathcal{I}}$. Input for *3D+time mJ-Net* (Sec. 12.5.2).
$\widehat{V}_{z_{\mathcal{I}}}^{t} = \varphi(\widehat{V}_{z_{i-1}}^{t}, \widehat{V}_{z_i}^{t}, \widehat{V}_{z_{i+1}}^{t}) \in \mathbb{R}^{(X \times Y \times Z \times T)}$	4D Tensor of slices $z_{\mathcal{I}}$ over all the time points t. Input for *4D mJ-Net* (Sec. 12.5.2).

12.4.2 Pre-processing steps

The 4D CTP dataset underwent a series of pre-processing steps to extract brain tissue from the raw CTP scans.

Algorithm 2 describes in detail the various steps for pre-processing all

Algorithm 2 Pre-processing steps for one single patient study

Input: 4D CTP scan: $V(x, y, z, t)$
ref $\leftarrow V(x, y, z, t_1)_{\text{HU}}$ {Get the 1st time point image as frame of reference}

for $j = 2$ to $t_{\max}$ **do**
Co-register $V(x, y, z, t_j)$ using ref as frame of reference.
end for
$V(x, y, z, t)_{\text{HU}} \leftarrow \text{ConvertToHU}(V(x, y, z, t))$
$\dot{V}(x, y, z, t) \leftarrow \text{BrainExtraction}(V(x, y, z, t)_{\text{HU}})$ {The brain extraction function is designed by Najm *et al.*[201]}
$\ddot{V}(x, y, z, t) \leftarrow \text{GammaCorrection}(\dot{V}(x, y, z, t))$
$z_{\text{high}} \leftarrow \text{GetSliceWithHighestIntensityValue}(\ddot{V}(x, y, z, t))$
bins $\leftarrow 2^{16}$
$\widetilde{V}(x, y, z_{\text{high}}, t), \mathcal{T}_{z_{\text{high}}} \leftarrow \text{HistEq}(\ddot{V}(x, y, z_{\text{high}}, t), \text{bins})$ {$\mathcal{T}_{z_{\text{high}}}$ is the grayscale transformation for z_{high}}
$\widetilde{V}(x, y, z_{\text{high}}, t) \leftarrow \frac{\widetilde{V}(x,y,z_{\text{high}},t) - \text{mean}(\widetilde{V}(x,y,z_{\text{high}},t))}{\sigma(\widetilde{V}(x,y,z_{\text{high}},t))}$ {Standardization of the data}
for $i = 1$ to $z_{\max}$ **do**
if $i \neq z_{\text{high}}$ **then**
$\widetilde{V}(x, y, z_i, t) \leftarrow \text{HistEq}(\ddot{V}(x, y, z_i, t), \mathcal{T}_{z_{\text{high}}})$
$\widetilde{V}(x, y, z_i, t) \leftarrow \frac{\widetilde{V}(x,y,z_i,t) - \text{mean}(\widetilde{V}(x,y,z_i,t))}{\sigma(\widetilde{V}(x,y,z_i,t))}$
end if
end for
return Processed 4D CTP scan: $\widetilde{V}(x, y, z, t)$

the data in a patient study. Furthermore, each step can be summarized as follows:

(i) Co-registration of all the images in the 4D CTP scan using the first time point image as the frame of reference in order to correct possible motion artifacts. An intensity-based image registration with similarity transformation was used in this step.

(ii) All the registered CTP scans were encoded into HU values to have a known quantitative scale to describe radiodensity efficiently. We used the following equation to calculate the HU value for a voxel V with a rescale slope (RS) and a rescale intercept (RI) extracted from the DICOM header: $V(x, y, z, t)_{\text{HU}} = V(x, y, z, t) \cdot \text{RS} + \text{RI}$

(iii) Brain extraction of CT studies plays an essential role in stroke imaging research [201, 202]. An automatic brain extraction method designed by Najm *et al.* [201] was selected for this purpose due to its proven efficiency with CT datasets and public availability.

(iv) Gamma correction ($\gamma = 0.5$) and histogram equalization (HE) were also performed after step 3 to increase contrast and visual interpretability on the CTP scans.

(v) Finally, z-score (z) on the enhanced 4D tensor is applied to normally distribute the data. Thus, every study has a mean of 0 and a standard deviation of 1, ensuring that all CTP studies have a similar scale.

The input for all the methods (except for the *Multi-input PMs*'s approach) follows the same pre-processing steps. These steps were performed to improve the quality of the images by enhancing the contrast. For a detailed explanation of the pre-processing steps, we refer the reader to [208, 250]. An additional re-sampling step for all the images was performed to ensure uniform distribution in the temporal dimension. The pre-processing steps and re-sampling effects are examined in an ablation study in Sec. 12.6.3.

12.4.3 Convolution in many dimensions

A handful of DNN methods have been proposed to exploit 4D data with a full 4D Convolution (4D-Conv) layer. Using 4D-Conv for 4D data produced better performances than using multiple 3D Convolutions (3D-Conv) on the same data. Gessert *et al.* [254] and Bengs *et al.* [255] proposed a 4D spatio-temporal convolutional network to optical coherence tomography force estimation. They demonstrated that using the full 4D data information yields better performances than 3D data. Myronenko *et al.* [256] introduced a 4D CNN to segment cardiac volumetric sequences using CT scans, showing the advantages of using their proposed architecture compared to a classic 3D CNN.

In the remainder of this manuscript, let's define $I(x, y, z, t) \in \mathbb{R}^4$, $\mathcal{H}(w, h, d, p) \in \mathbb{R}^4$ as a 4D tensor and a 4D kernel, respectively. The x and w indicate the width of the 4D structures; y and h express the height dimension; z and d define the depth dimension, while t and p represent the time dimension of the 4D structures. Like a 3D-Conv can be represented as the sum of multiple 2D Convolution (2D-Conv) along the depth dimension, a 4D-Conv operation can be described as the sum of numerous 3D-Conv

along the temporal dimension. The loop rearrangement to avoid repeated 3D-Conv operations allows a true (non-separable) 4D convolution operation [256].

A 4D-Conv $g''''(x, y, z, t)$ of a 4D input $I(x, y, z, t) \in \mathbb{R}^4$ and a 4D kernel $\mathcal{H}(w, h, d, p) \in \mathbb{R}^4$ can be defined as:

$$\begin{aligned} g''''(x, y, z, t) &= \mathcal{H}(w, h, d, p) \circledast I(x, y, z, t) \\ &= \sum_{l=0}^{p-1} \sum_{k=0}^{d-1} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \mathcal{H}(i, j, k, l) I(\tilde{x}, \tilde{y}, \tilde{z}, \tilde{t}), \end{aligned}$$

where $\tilde{x} \equiv x + \widetilde{w} - i$, $\tilde{y} = y + \widetilde{h} - j$, $\tilde{z} \equiv z + \widetilde{d} - k$, and $\tilde{t} \equiv t + \widetilde{p} - l$. For a detailed description of the 2D-Conv and 3D-Conv, we refer to the additional supplemental material linked with the manuscript.

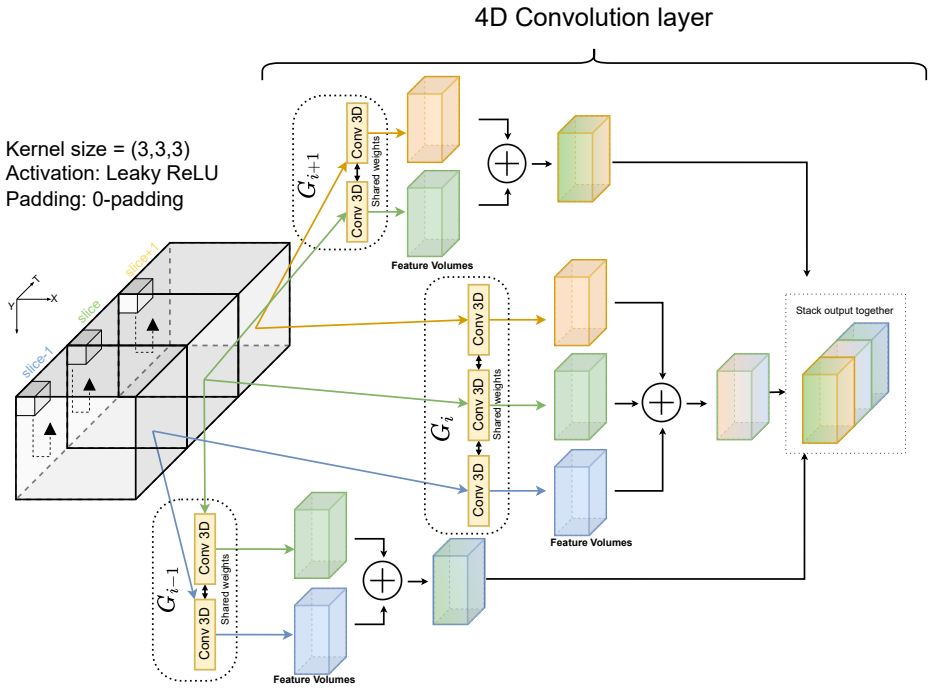


Figure 12.2: Visual representation of a 4D-Conv layer. The input is $\widehat{V}^t_{z_\mathcal{I}} = \varphi(\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}}) \in \mathbb{R}^{(X \times Y \times Z \times T)}$. A series of 3D-Conv operations are calculated over a 4D input. Several groups (G_{i-1}, G_i, G_{i+1}) are generated, one for each volume involved in the operation.

We define a 4D-Conv $g''''(x, y, z, t)$ and a 4D kernel $\mathcal{H}(w, h, d, p)$ as the sum of multiple 3D-Conv over a specific dimension, i.e., the third dimension. The 4D kernel $\mathcal{H}(w, h, d, p)$ can be seen as a list of $\mathcal{H}(w, h, k, p)|\forall k \in d-1$, where $\mathcal{H}(w, h, k, p)$ is a 2D+time volume of the kth slice over the entire p elements in the temporal dimension:

$$\begin{aligned} g''''(x, y, z, t) &= H(w, h, d, p) \circledast I(x, y, z, t) \\ &= \sum_{k=0}^{d-1} \mathcal{H}(w, h, k, p) I(x, y, z + \widetilde{d} - k, t) \\ &= \sum_{k=0}^{d-1} g'''(x, y, z + \widetilde{d} - k, t), \end{aligned}$$

where $I(x, y, z + \widetilde{d} - k, t) \in \mathbb{R}^3$ is a 2D+time volume at slice $z + \widetilde{d} - k$ over the totality t element in the temporal dimension, and $[z + \widetilde{d} - k \in z_m | \forall m \in \{1, \ldots, z_{\max}\}]$

Fig. 12.2 visually explains the proposed 4D-Conv layer. The input for our 4D-Conv layer is a 4D tensor $\widehat{V}^t_{z_\mathcal{I}}$, where $z_\mathcal{I}$ is a set of neighboring slices $\{z_{i-1}, z_i, z_{i+1}\}$ (Table 12.2). The 2D+time volume of the ith brain slice $\widehat{V}^t_{z_i}$ over all the time points t, and the two 2D+time volumes of the neighboring brain slices ($\widehat{V}^t_{z_{i-1}}$ and $\widehat{V}^t_{z_{i+1}}$). The 4D-Conv layer uses a loop rearrangement with three 3D-Conv groups (G_{i-1}, G_i, G_{i+1}), one for each volume slice involved in the operation. In each group G_i, several 3D-Conv layers are created. All convolution layers in each group shared the weights. Each 3D-Conv layer is used for a single input volume. The number of layers depends on the legal subscript indexes, i.e., for the group G_{i-1}, there are two 3D-Conv layers since the legal subscript indexes are $\{i-1, i\}$: the indexes are given by the current volume slice z_{i-1} and the only neighboring volume slice z_i. The output of each group is a set of feature volumes summed together. The resulting feature volumes are stacked together to keep the same dimension as the input.

12.5 Existing methods & Proposed 4D methods

This paper presents three novel deep learning (DL) approaches that accommodate 4D input data. In the remainder of the paper, they are called

3D-TCN (Sec. 12.5.2), *3D+time mJ-Net* (Sec. 12.5.2), and *4D mJ-Net* (Sec. 12.5.2).

Together with the proposed approaches, we implemented and compared the *2D-TCN* [30] (Sec. 12.5.1), the *3D-TCN-SE* [31] (Sec. 12.5.1), and the *mJ-Net* [208] (Sec. 12.5.1) to validate the performances of our models. We also compare the models with a method that uses a set of PMs as input [207]. In the remainder of the paper, we call this architecture *Multi-input PMs* (Sec. 12.5.1). Fig. 12.3 visually compares the input utilized for the various approaches.

12.5.1 Existing methods

Approach 1: Multi-input PMs

The *Multi-input PMs* model was proposed in [207]. This architecture was used as a baseline study because all the PMs were input for the model. The input for the architecture is a list of PMs for each brain slice z_i: PMs_{z_i}, as shown in Table 12.3. The loss function implemented is the Focal Tversky loss (FTL) [100]; for a specific class c, the FTL is defined as:

$$\text{FTL}(x, y) = \sum_{c}^{\mathcal{C}} (1 - \text{TI}_c)^{1/\gamma},$$

where $\gamma \geq 1$ is a hyper-parameter that forces the loss function to focus more on less accurate predictions that have been misclassified [100]. Denoting $x_{i,c} \in [0, 1]$ as the probability of the ith predicted pixel to belong to class c; $y_{i,c} \in \{0, 1\}$ as the pixel i with class c in a ground truth image, TI_c is the Tversky index (TI) for a class c defined as:

$$\text{TI}_c = \frac{\sum_{i=1}^{M \times N} x_{i,c} y_{i,c}}{\sum_{i=1}^{M \times N} x_{i,c} y_{i,c} + \alpha \sum_{i=1}^{M \times N} \widehat{x_{i,c}} y_{i,c} + \beta \sum_{i=1}^{M \times N} x_{i,c} \widehat{y_{i,c}}},$$

where $\widehat{x_{i,c}} = 1 - x_{i,c}$ is the probability that the ith pixel is not of class c, and $\widehat{y_{i,c}} = 1 - y_{i,c}$ represents the complement of pixel i in a ground truth image. The hyper-parameters α and β control the trade-off between precision and recall. We refer the reader to [207] for a more extensive explanation and discussion about this approach.

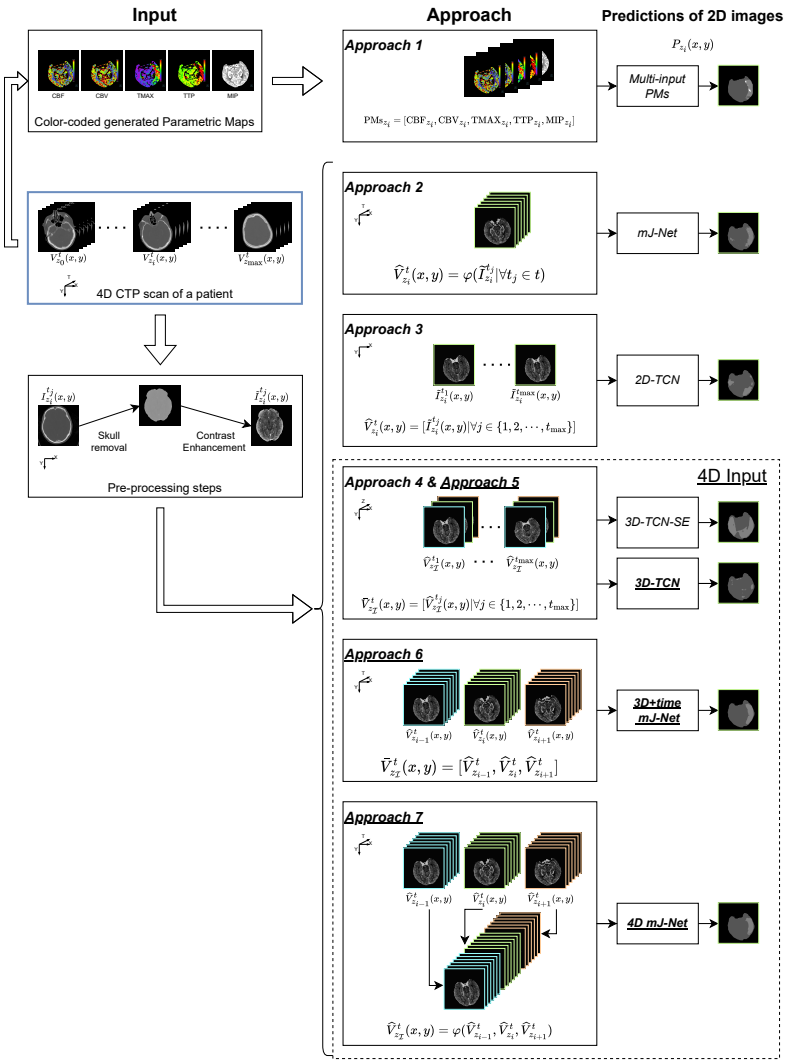


Figure 12.3: Visual comparison of the input for each implemented approach. Every 4D CTP patient's study $V \in \mathbb{R}^{(X \times Y \times Z \times T)}$ undergoes a series of pre-processing steps to enhance each CTP scan. Approach 1 (*Multi-input PMs*) [207] accepts a list of PMs generated from a CTP study in input. Approach 2 (*mJ-Net*) [208] use a 2D+time volume $\widehat{V}^t_{z_i}(x, y)$ as input. Approach 3 (*2D-TCN*) and Approach 4 (*3D-TCN-SE*) follows the model proposed by Amador *et al.* [30] and Amador *et al.* [31], respectively. The resulting 4D tensor is fed to one of the approaches. The proposed approach 5 (*3D-TCN*), approach 6 (*3D+time mJ-Net*), and approach 7 (*4D mJ-Net*) take in input the entire 4D CTP processed data $\widetilde{V}$.

Approach 2: mJ-Net

The *mJ-Net* approach was proposed in [208]. As presented in Table 12.3, the input $\widehat{V}^t_{z_i}(x, y)$ for *mJ-Net* is a 2D+time volume of the same brain slice z_i at index i over all the time points t. We define the dimension of this input as 2D+time; the first dimension of the input is time.

The loss function used for this method is the soft Dice Coefficient loss (SDCL) [99]. The SDCL is a modified version of the Dice Coefficient score mainly used in medical domains where the classes to predict are highly unbalanced due to a small region of interest compared to the background of the scans. The SDCL can be written as:

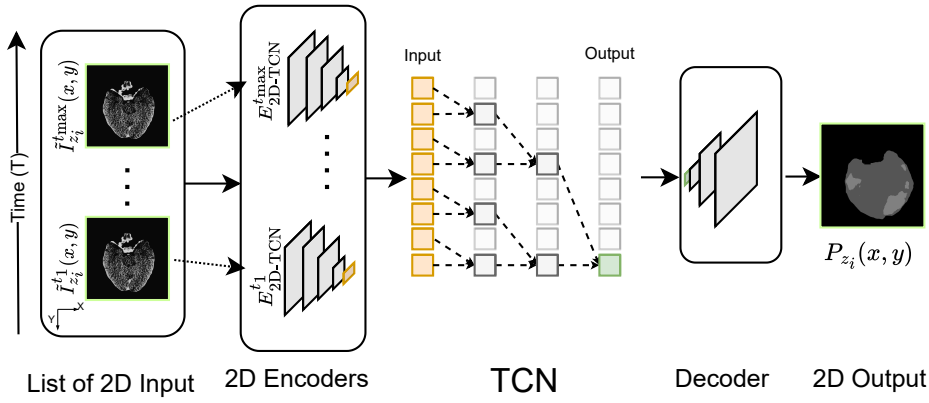
$$\text{SDCL}(x, y) = \sum_{c}^{\mathcal{C}} \left(1 - \frac{2 \sum_{i}^{M \times N} x_{i,c} y_{i,c}}{\sum_{i}^{M \times N} x_{i,c}^2 + \sum_{i}^{M \times N} y_{i,c}^2} \right)$$

The first section of the model contains 3D-Conv layers to extract information from the temporal dimension, while the second part follows the classic U-Net structure [112]. For more details about the *mJ-Net* approach, we refer the reader to [208].

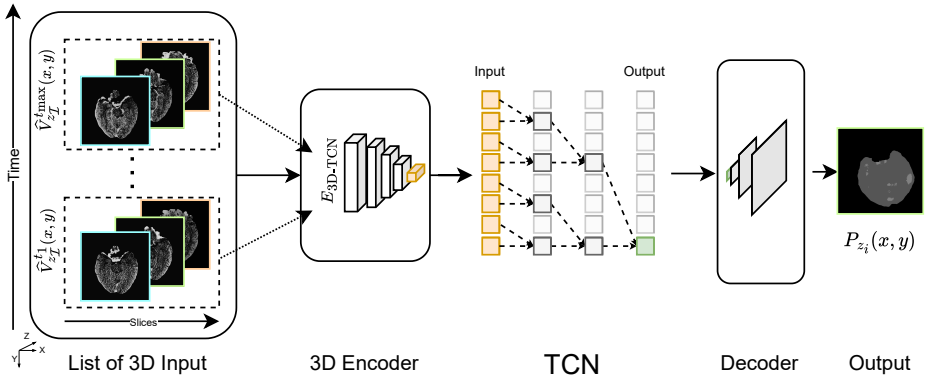
Approach 3: 2D Temporal Convolutional Network

For comparison reasons, we implemented the method proposed by Amador *et al.* [30]. We call this architecture *2D-TCN* in the remainder of the paper. Fig. 12.4a provides a general architecture overview. They proposed a Temporal Convolutional Network (TCN), which has been shown to outperform conventional neural networks in different tasks [257]. Moreover, a TCN has a lower memory requirement for training than other Recurrent Neural Networks [257].

The *2D-TCN* was trained with the exact implementation as the original work (see Table 12.3 for more details). The *2D-TCN* model receives the 4D CTP scans in input re-sampled to 1 second per time point. The 4D input is processed as a list of 2D brain slices z_i for each time point t. Thus, the actual input for the *2D-TCN* is a list $\bar{V}^t_{z_i}$, as mentioned in Table 12.3. The list $\bar{V}^t_{z_i}$ contains all the time points of the brain slice z_i. Every 2D input image of the list $\tilde{I}^{t_j}_{z_i}$ at time point t_j is fed to a 2D encoder $E^{t_j}_{\text{2D-TCN}}$ to extract features in the latent space. Each $E^{t_j}_{\text{2D-TCN}}$ encoder returns a $(4 \times 4 \times \text{Ch})$ feature vector, where Ch corresponds to the number of channels. The architecture merges the low-level feature vectors across



(a) *2D-TCN* architecture proposed by Amador *et al.* [30].



(b) *3D-TCN-SE* proposed by Amador *et al.* [31]

Figure 12.4: Visual comparison between (a) the *2D-TCN* architecture [30] and (b) the *3D-TCN-SE* [31].

the different t_j time points to capture the spatio-temporal information. The merged feature vector $\text{ETOT}_{\text{2D-TCN}} = [E^{t_j}_{\text{2D-TCN}} | \forall t_j \in t]$ is used as input to the TCN, which yields a one-dimensional vector $O_{\text{2D-TCN}}$ of 64 elements. Finally, a decoder takes the $O_{\text{2D-TCN}}$ and generates a final 2D image $P_{z_i}(x, y)$. The Dice Coefficient loss (DCL), the same as the original

paper, was implemented as the loss function as follows:

$$\mathrm{DCL}(x, y) = \sum_{c}^{\mathcal{C}} \left(1 - \frac{2 \sum_{i}^{M \times N} x_{i,c} y_{i,c}}{\sum_{i}^{M \times N} x_{i,c} + \sum_{i}^{M \times N} y_{i,c}} \right)$$

For more details about the *2D-TCN* approach, we refer the reader to [30].

Approach 4: 3D Temporal Convolutional Network Single Encoder

We implemented a similar method from Amador *et al.* [31]. We call this approach *3D-TCN-SE* due to using a single encoder (SE). Fig. 12.4b shows a simplified version of the proposed architecture, emphasizing the input difference between the *2D-TCN* and this model.

The *3D-TCN-SE* model receives the 4D CTP scans re-sampled to 1 second per time point. The input is a list of t 3D volumes $\bar{V}_{z_\mathcal{I}}^{t}$ (Table 12.3). Each 3D input volume in the list $\widehat{V}_{z_\mathcal{I}}^{t_j}(x, y)$ corresponds to the concatenation of the ith brain slice z_i plus its neighbouring slices z_{i-1} and z_{i+1} over a specific time point t_j. The *3D-TCN-SE* approach uses a single encoder $E_{\text{3D-TCN}}$ for all the elements in the input list. It is worth mentioning that the *3D-TCN-SE* model is trained with the entire brain images for comparison reasons and not with just the ipsilateral hemisphere, as in the original paper [31].

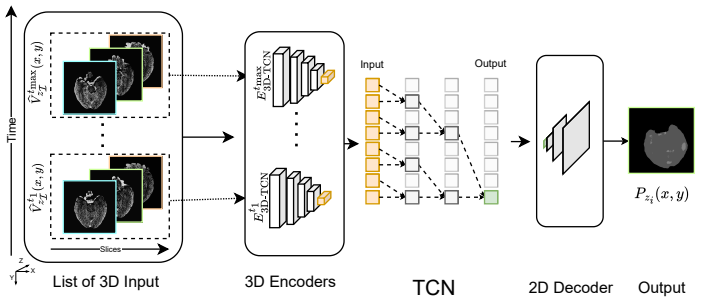
12.5.2 Proposed 4D methods

The proposed methods[2] adopt the entire 4D CTP scan as input to segment the ischemic regions (core and penumbra) in patients suspected of AIS. The main difference lies in how the 4D input is processed. The *3D-TCN* is based on a *2D-TCN* [30], modified to receive a list of 3D input volumes. The *3D+time mJ-Net* inputs a list of 2D+time brain volumes from a CTP dataset, while the *4D mJ-Net* uses the entire 4D structure of a CTP dataset as input. Fig. 12.5 compares these architectures with their respective inputs.

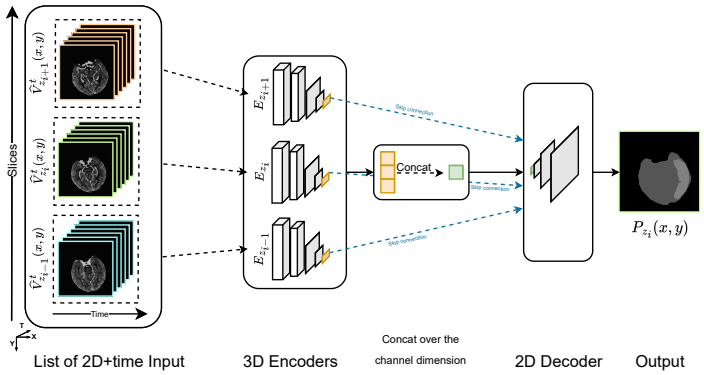
Approach 5: 3D Temporal Convolutional Network

We extend the architecture proposed by Amador *et al.* [30] for our application to exploit further the information in the depth dimension. In the

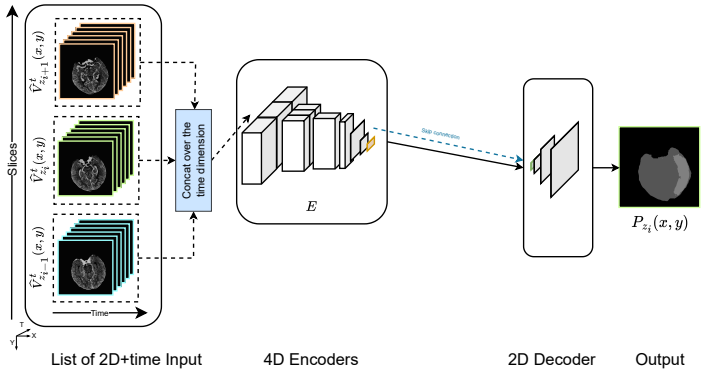
[2]The code is publicly available at the following link https://github.com/Biomedical-Data-Analysis-Laboratory/4D-mJ-Net



(a) *3D-TCN* architecture (Sec. 12.5.2)



(b) *3D+time mJ-Net* architecture (Sec. 12.5.2)



(c) *4D mJ-Net* architecture (Sec. 12.5.2)

Figure 12.5: Visual comparison between the proposed architectures: (a) the *3D-TCN*, (b) the *3D+time mJ-Net*, and (c) *4D mJ-Net*.

remainder of the paper, we call our architecture *3D-TCN*. The main differences between the proposed *3D-TCN* and the *3D-TCN-SE* (Sec. 12.5.1) rely on the usage of a 3D encoder for each input element, instead of a 3D single encoder, plus the possibility to segment both core and penumbra regions, in comparison with segmenting only the core areas.

The 4D CTP scans for the *3D-TCN* are all re-sampled to 1 second per time point. As described in Sec. 12.5.1, the *3D-TCN* architecture feeds each element of the input list $\widehat{V}_{z_\mathcal{I}}^{t_j}$ at time point t_j to a specific 3D encoder $E_{\text{3D-TCN}}^{t_j}$ to extract low-level features. Each $E_{\text{3D-TCN}}^{t_j}$ encoder returns a $(4 \times 4 \times C)$ feature vector, where C corresponds to the number of channels. Each feature vector is merged to create a single input $\text{ETOT}_{\text{3D-TCN}} = [E_{\text{3D-TCN}}^{t_j} | \forall t_j \in t]$. The $\text{ETOT}_{\text{3D-TCN}}$ is used in the TCN, which generates a one-dimensional vector $O_{\text{3D-TCN}}$ of 64 elements. The TCN's output $O_{\text{3D-TCN}}$ is then given in input to the decoder to create the final predicted 2D image $P_{z_i}(x, y)$ of a brain slice z_i at index i

Approach 6: 3D+time mJ-Net

We propose a model called *3D+time mJ-Net*, an extension of the work of Tomasetti *et al.* [208]. The proposed model inputs a list of 2D+time matrices; thus, the dimension of this input can be defined as 3D+time. The input and output are presented in Table 12.3, whereas a visual example of the input for the model is given in Fig. 12.3. Each element of the input list coincides with a possible input for the *mJ-Net* (details in Sec. 12.5.1). $\bar{V}_{z_\mathcal{I}}^{t}$ consists of a list of 2D+time volumes, where $z_\mathcal{I} = \{z_{i-1}, z_i, z_{i+1}\}$ is a set of brain slices containing the ith slice z_i analyzed and its neighboring slices z_{i-1} and z_{i+1}. In case the index i corresponds to the first (or last) brain slice, $\widehat{V}_{z_{i-1}}^{t}$ (and equivalently $\widehat{V}_{z_{i+1}}^{t}$) is set equal to $\widehat{V}_{z_i}^{t}$. Every 2D+time volume from the input list $\widehat{V}_{z_i}^{t}(x, y)$ is trained separately in the model through a series of encoders $(E_{z_{i-1}}, E_{z_i}, E_{z_{i+1}})$ composed of 3D-Conv and 2D-Conv layers. Each section is independent of the other: the convolution layers have no shared weights.

Fig. 12.6 illustrates the model's architecture. Attention layers [258] and 2D Upsampling layers were implemented in the decoder section. Attention layers benefit the architecture by focusing on target structures and help increase the segmentation performances. With the sole exception of the last convolution layer, each convolution layer uses a kernel of dimension 3 and a Leaky ReLU activation function [259] with $\alpha = 1/3$.

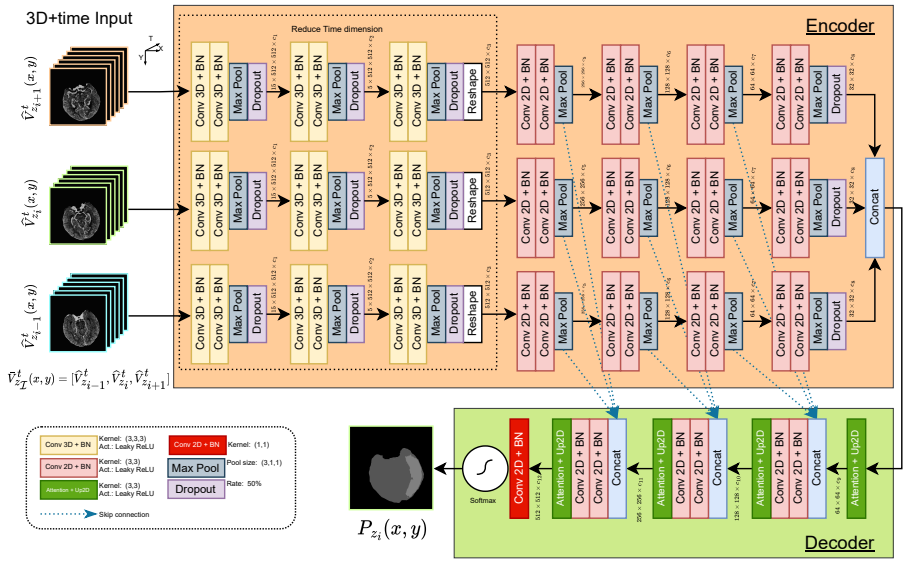


Figure 12.6: Illustration of the *3D+time mJ-Net* model. The list of 2D+time input $\bar{V}^t_{z_\mathcal{I}}(x,y) = [\widehat{V}_{z_{i-1}}, \widehat{V}_{z_i}, \widehat{V}_{z_{i+1}}]$ is trained in parallel, where $z_\mathcal{I} = \{z_{i-1}, z_i, z_{i+1}\}$. The output is a 2D image $P_{z_i}(x,y)$. The first max-pooling layer of each block in the convolution section has a pool size of (2,1,1) to reduce the first dimension by a factor of 2. The second max-pooling layer uses a pool size of (3,1,1), while the third has a pool size of (5,1,1). The selection of these pool sizes is due to reducing the time dimension. The remaining max-pooling layers have a pool size of (2,1,1). The Attention layers utilize a kernel of dimension 3 and a Leaky ReLU activation function. The 2D Upsampling layers have an upsampling factor of 2. The last convolution layer has a kernel of 1 and a Softmax activation function to produce a probability score for every class.

Approach 7: 4D mJ-Net

We propose another model called *4D mJ-Net*. We introduce this method to avoid the three paths to process the 4D data presented in the previous architecture (Sec. 12.5.2). We still use a sliding window technique over the depth dimension to limit the amount of input data fed to the model, using three consecutive brain slices simultaneously. Using a neighborhoods operation is a common image processing technique to include, in the computation, information from a additional dimension. Thus, adopting three consecutive brain slices enhances the model's ability to capture the information from the depth dimension effectively. Like the *3D+time mJ-Net* model, also this approach is an extension of the work of Tomasetti *et al.* [208]. Information on this approach is given in Table 12.3. The 4D input

tensor $\widehat{V}^t_{z_\mathcal{I}}$ contains both the time dimension and the neighboring slices of the ith brain slice. The $\widehat{V}^t_{z_\mathcal{I}}$ is a concatenation of a 2D+time volume $\widehat{V}^t_{z_i}$ of a brain slice z_i at index i over all the time points t together with its neighbouring 2D+time volumes $\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_{i+1}}$. This model can be considered an early-fusion approach since the 4D input tensors $\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}}$ are concatenated before being fed to the encoder's model.

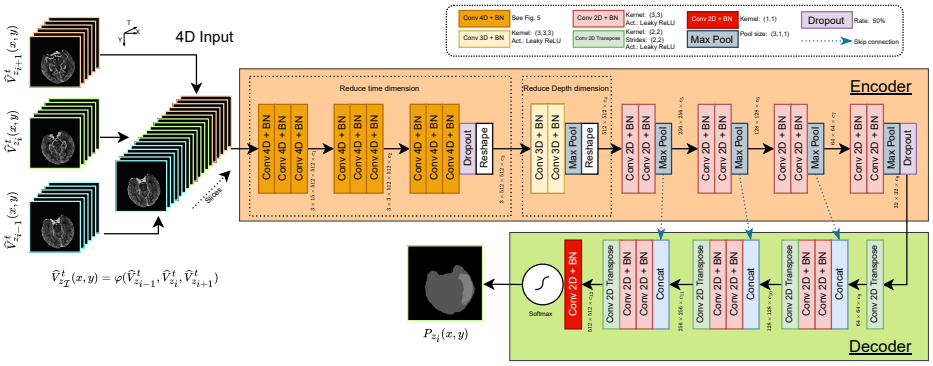


Figure 12.7: Illustration of the *4D mJ-Net* architecture. The 4D input $\widehat{V}^t_{z_\mathcal{I}} = \varphi(\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}})$ is the concatenation of a 2D+time volume $\widetilde{V}^t_{z_i}$ of a brain slice z_i at index i over all the time points t plus its neighbouring brain slice volumes $(\widetilde{V}^t_{z_{i-1}}, \widetilde{V}^t_{z_{i+1}})$. Two MonteCarlo dropout layers [260] are added at the end of the 4D and 2D Convolution blocks. The rate was set to 50%. These layers were added to reduce uncertainties in the final predictions. Additional details are given in the supplemental material. The last convolution layer has a kernel of (1×1) and a Softmax activation function to produce a probability score for every class.

The proposed *4D mJ-Net* model is a combination of both *3D+time mJ-Net* (Sec. 12.5.2) and *mJ-Net* (Sec. 12.5.1). The proposed approach uses the same input type that the *3D+time mJ-Net* exploits. However, rather than a list of 2D+time volumes, the model concatenates the input into a single 4D tensor of dimensions $(X \times Y \times Z \times T)$.

Unlike 1D, 2D, and 3D Convolution layers, 4D Convolution layers are not available in public DL frameworks (i.e., Keras[3] or PyTorch[4]). Thus, for this model, we implemented a novel 4D-Conv layer (details in Sec. 12.4.3), which uses the convolutional layers defined in the public DL frameworks to replicate a 4D convolution operation.

[3] https://keras.io/
[4] https://pytorch.org/

The architecture of the *4D mJ-Net* is displayed in Fig. 12.7. No Attention layers [258] were included in these models due to a considerable performance decline. The output of the 4D-Conv layers is a tensor where the temporal dimension has been squeezed and reduced; information are extrapolated from the temporal dimension. Thus, the output resulting from the 4D-Conv layers contains only three dimensions ($X \times Y \times Z$) plus the channel dimension. 3D-Conv layers are implemented to reduce the depth dimension Z and produce a 2D vector ($X \times Y$) plus the channel dimension.

A weighted categorical cross-entropy (WCC) loss [101] was the loss function implemented for this method. The loss can be written as:

$$\mathrm{WCC}(x, y) = \sum_{c}^{\mathcal{C}} \sum_{i}^{M \times N} (y_{i,c} \log x_{i,c}) \cdot (w_{i,c} y_{i,c}),$$

where $w_{i,c}$ corresponds to the weight of the ith pixel for a class $c \in \mathcal{C}$; $x_{i,c}$ is the ith predicted pixel, and $y_{i,c}$ is the corresponding ground truth pixel.

12.5.3 Implementation details

Table 12.3: Summary of the approaches. C-Time stands for computational time per brain slice in seconds. The C-Time values are expressed as mean ± standard deviation.

Approach	Input	Output	Loss	C-Time (s)
Approach 1 [207]: *Multi-input PMs*	$\mathrm{PMs}_{z_i} = [\mathrm{CBF}_{z_i}, \mathrm{CBV}_{z_i}, \mathrm{TMAX}_{z_i}, \mathrm{TTP}_{z_i}, \mathrm{MIP}_{z_i}]$	$P_{z_i}(x, y)$	FTL [100]	0.07±0.01
Approach 2 [208]: *mJ-Net*	$\widehat{V}^t_{z_i}(x, y) = \varphi(\tilde{I}^{t_j}_{z_i}(x, y) \vert \forall t_j \in t)$.	$P_{z_i}(x, y)$	SDCL [99]	0.19±0.02
Approach 3 [30]: *2D-TCN*	$\bar{V}^t_{z_i} = [\tilde{I}^{t_j}_{z_i} \vert \forall t_j \in t]$ of 2D images $\tilde{I}^{t_j}_{z_i}$	$P_{z_i}(x, y)$	DCL	0.40±0.07
Approach 4 [31]: *3D-TCN-SE*	$\bar{V}^t_{z_\mathcal{I}} = [\widehat{V}^{t_j}_{z_\mathcal{I}} \vert \forall t_j \in t]$	$P_{z_i}(x, y)$	SDCL [99]	0.54±0.06
Approach 5: *3D-TCN*	$\bar{V}^t_{z_\mathcal{I}} = [\widehat{V}^{t_j}_{z_\mathcal{I}} \vert \forall t_j \in t]$	$P_{z_i}(x, y)$	SDCL [99]	0.61±0.02
Approach 6: *3D+time mJ-Net*	$\bar{V}^t_{z_\mathcal{I}}(x, y) = [\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}}]$	$P_{z_i}(x, y)$	SDCL [99]	0.65±0.02
Approach 7: *4D mJ-Net*	$\widehat{V}^t_{z_\mathcal{I}} = \varphi(\widehat{V}^t_{z_{i-1}}, \widehat{V}^t_{z_i}, \widehat{V}^t_{z_{i+1}})$	$P_{z_i}(x, y)$	WCC [101]	0.38±0.03

Table 12.3 provides information about all the methods. All the methods mentioned in Sec. 12.5.1 and Sec. 12.5.2 utilize Adam as the optimizer [104] with a learning rate of 0.0003 and a step-based decay rate of 0.95 every ten epochs. The batch size is set to 2. An early stopping function is called if there is no decrement in the validation loss after 25 epochs. During

training, L1 and L2 regularizations are applied in the kernels, plus a max norm constraint is also used in the kernel and bias weights. The average computational time for all the brain slices during inference phase is also highlighted in Table 12.3. All experiments were implemented in Python using Keras (2.3.1) with Tensorflow as the backend and trained using an NVIDIA Tesla V100 GPU (32 GB memory).

12.6 Experiments & Results

We assess the proposed methods[5] on a local dataset of CTP scans from 152 patients (Sec. 12.3). All experiments are performed with the same training set and evaluated over the validation set (details in Table 12.1). The test set is used only to make predictions with the best models with CTP scans that the methods have not seen before. Since the Non-LVO group has smaller ischemic areas than the LVO patients, we set a higher penalty for every misclassification of penumbra and core classes for this sub-group during training.

12.6.1 Evaluation metrics

Three evaluation metrics are used to assess the various experiments' models. The Dice Coefficient (DC), the Hausdorff Distance (HD) [220], and the absolute difference in the volumes (ΔV). We employ the DC to compare the model predictions with the ground truth segmentations. The DC between two segmentations x and y is given by the following equation:

$$\mathrm{DC}(x, y) = 2\frac{|x \cap y|}{|x| + |y|}$$

where the range for the DC is $[0, 1]$; thus a $\mathrm{DC}(x, y) = 1$ corresponds to a perfect match between the prediction x and ground truth y segmentations.

The HD measures how two subsets $(\mathcal{A}, \mathcal{B})$ are distant from each other, and it is formulated as follows:

$$\mathrm{HD}(\mathcal{A}, \mathcal{B}) = \max\{\, h(\mathcal{A}, \mathcal{B}),\, h(\mathcal{B}, \mathcal{A}) \,\},$$

[5]The code is publicly available at the following link: https://github.com/Biomedical-Data-Analysis-Laboratory/4D-mJ-Net

where $h(\mathcal{A}, \mathcal{B}) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} ||a - b||$. The range value for the HD is $[0, \infty]$.

The absolute difference in the volumes ΔV between the prediction volume V_x and the ground truth volume V_y can be expressed as:

$$\Delta V(V_y, V_x) = |V_y - V_x|$$

The range for ΔV is $[0, \infty]$, and $\Delta V(V_y, V_x) = 0$ represents a perfect match between the two volumes. The $\Delta V(V_y, V_x)$ is an essential evaluation metric for the WIS group due to the lack of ground truth segmentations in this group. The other metrics are not suitable for understanding how the predictions will be since the ground truth will always be empty.

The best scenario for a model is to produce high DC with low HD and ΔV: this implies a strong correlation between the predicted areas and the ground truth regions. If the results show high ΔV (or HD) with low DC, an over-segmentation of the ischemic areas is perceived. On the other hand, promising outcomes of ΔV (or HD) with mediocre DC results imply an under-segmentation of the predicted regions.

12.6.2 Comparison with other methods

The proposed *3D+time mJ-Net*, *4D mJ-Net*, and *3D-TCN* methods are compared with alternative models: the *2D-TCN* [30], the *mJ-Net* [208], the *3D-TCN-SE* [31], and the *Multi-input PMs* [207].

Table 12.4 presents the evaluation metrics' results over the validation set. Results are presented for each group distinctly (LVO, Non-LVO, WIS, and all) to highlight the strengths and weaknesses of each model over the various groups composing the dataset. An extensive number of experiments are performed for all the analyzed models. However, to present a fair comparison among the various models, we only introduce the methods with a combination of parameters that yield the best results, omitting the other combinations tested during experiments. Qualitative comparison results of random brain slices extracted from the validation set are provided in Fig. 12.8.

12.6.3 Ablation Study

To demonstrate the effects of the pre-processing steps (Sec. 12.4.2), we conduct an ablation study on the *4D mJ-Net* architecture. Moreover, we

Table 12.4: Experiment results for the validation set. Values in bold exhibit the best results for each column and each class. Mean results plus standard deviation for Dice Coefficient (DC), Hausdorff Distance (HD), and ΔV are presented. Results are for the penumbra and core areas divided by the distinct patient groups (LVO, Non-LVO, WIS, and All). Note that for the DC, higher values are better ($\Uparrow$), while for HD and ΔV, lower values are preferable ($\Downarrow$).

Method	DC $\Uparrow$			HD (mm) $\Downarrow$			ΔV (ml) $\Downarrow$			
	LVO	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**WIS**	**All**
	Penumbra									
Multi-input PMs [207]	**0.70±0.1**	0.27±0.3	0.47±0.3	2.9±0.4	1.4±0.7	2.0±0.8	27.0±28.6	10.0±15.5	9.8±8.1	**19.0±24.2**
mJ-Net [208]	0.66±0.2	0.39±0.3	0.50±0.3	2.9±0.5	2.6±0.6	2.7±0.7	**25.5±20.0**	24.7±29.2	45.5±39.1	27.2±26.0
2D-TCN [30]	0.12±0.1	0.02±0.0	0.07±0.1	4.1±0.5	3.8±0.6	4.0±0.6	81.3±65.6	80.6±57.8	131.6±93.1	86.0±66.5
3D-TCN-SE [31]	0.25±0.1	0.05±0.1	0.15±0.1	6.2±0.5	6.7±0.4	6.4±0.5	497.9±157.1	559.3±90.4	624.6±118.7	533.1±137.3
3D-TCN	0.23±0.1	0.04±0.1	0.14±0.1	4.3±0.4	4.4±0.5	4.3±0.5	85.3±64.0	142.7±51.4	164.2±43.8	114.2±65.3
3D+time mJ-Net	**0.70±0.1**	0.42±0.3	**0.53±0.3**	2.6±0.6	1.9±0.8	2.2±0.9	35.1±36.1	18.3±26.3	2.7±2.6	25.7±32.4
4D mJ-Net	0.66±0.1	**0.44±0.3**	0.51±0.3	**2.3±0.6**	**1.3±0.7**	**1.7±1.0**	41.4±37.2	**6.1±6.3**	**0.0±0.0**	24.3±32.9
	Core									
Multi-input PMs [207]	**0.37±0.3**	**0.21±0.3**	**0.28±0.3**	1.2±0.8	**0.4±0.4**	0.8±0.8	9.4±20.3	**0.8±1.3**	0.5±0.5	5.3±15.3
mJ-Net [208]	0.27±0.2	**0.21±0.2**	0.22±0.2	1.5±0.7	0.8±0.6	1.2±0.6	**5.5±4.9**	1.0±1.2	1.0±1.1	**3.4±4.3**
2D-TCN [30]	0.02±0.0	0.01±0.0	0.01±0.0	1.9±0.7	1.5±0.6	1.7±0.7	11.8±13.3	8.1±8.2	11.0±11.2	10.3±11.4
3D-TCN-SE [31]	0.00±0.0	0.00±0.0	0.00±0.0	1.2±0.9	0.4±0.4	0.8±0.8	12.7±15.6	1.9±2.8	**0.0±0.0**	7.5±12.8
3D-TCN	0.02±0.0	0.01±0.0	0.01±0.0	1.4±0.8	0.8±0.4	1.1±0.7	12.0±14.3	1.9±2.1	2.4±1.9	7.3±11.6
3D+time mJ-Net	0.21±0.2	0.12±0.2	0.16±0.4	**1.1±0.7**	**0.4±0.4**	**0.7±0.7**	8.1±10.6	1.3±1.6	**0.0±0.0**	4.8±8.5
4D mJ-Net	0.29±0.2	**0.21±0.2**	0.23±0.2	1.6±0.9	0.5±0.4	1.0±0.9	25.9±37.0	1.4±2.2	**0.0±0.0**	14.3±29.6

Table 12.5: Ablation study for the *4D mJ-Net* model showing how various pre-processing steps (HE, γ, z) and re-sampling ($\uplus$) affect the Dice Coefficient (DC) for the validation results. Penumbra and core DC scores are shown for all the classes together. Note that for the DC, higher values are better ($\Uparrow$).

Ablation Setting				**DC** $\Uparrow$			
				LVO		**Non-LVO**	
HE	γ	z	$\uplus$	**Penumbra**	**Core**	**Penumbra**	**Core**
-	-	-	-	0.42±0.2	0.25±0.2	0.20±0.2	0.16±0.2
✓	-	-	-	0.32±0.3	0.24±0.2	0.13±0.2	0.16±0.3
-	✓	-	-	0.00±0.0	0.07±0.1	0.00±0.0	0.06±0.1
✓	✓	-	-	0.48±0.2	0.28±0.2	0.24±0.2	0.20±0.3
-	-	✓	-	0.01±0.0	0.14±0.2	0.01±0.0	0.08±0.1
✓	-	✓	-	0.53±0.2	0.22±0.2	0.35±0.3	0.12±0.2
-	✓	✓	-	0.28±0.2	0.17±0.2	0.08±0.1	0.05±0.1
✓	✓	✓	-	**0.66±0.1**	**0.29±0.2**	**0.44±0.3**	**0.21±0.2**
-	-	-	✓	0.00±0.0	0.00±0.0	0.00±0.0	0.00±0.0
✓	-	-	✓	0.07±0.1	0.26±0.2	0.01±0.0	0.05±0.1
-	✓	-	✓	0.00±0.0	0.00±0.0	0.00±0.0	0.00±0.0
✓	✓	-	✓	0.41±0.3	0.29±0.2	0.11±0.2	0.12±0.2
-	-	✓	✓	0.00±0.0	0.00±0.0	0.00±0.0	0.00±0.0
✓	-	✓	✓	0.56±0.2	0.24±0.2	0.37±0.3	0.18±0.2
-	✓	✓	✓	0.00±0.0	0.00±0.0	0.00±0.0	0.00±0.0
✓	✓	✓	✓	0.59±0.2	**0.29±0.2**	0.40±0.3	0.20±0.2

re-sampled the CTP scans to handle the irregular temporal dimension and studied the effect of using re-sampled scans during the model's training. Different CT scan vendors have different imaging acquisition protocols; thus, re-sampling the scans to a fixed time-sampling rate is a reasonable step to increase the versatility and usability across hospitals. DC is illustrated in Table 12.5, showing performances of the network for all the groups trained with the datasets using different types of pre-processing steps and re-sampled scans. The study aims to systematically analyze the contribution of each pre-processing step toward improving the overall results. We begin by defining a baseline configuration consisting of the raw input images without pre-processing (first row in Table 12.5). Subsequently, we incrementally introduce and evaluate individual pre-processing steps, such as histogram equalization (HE), gamma correction (γ), and z-score (z).

Table 12.6: Inter-observer variability results for test set. Values are presented for the two best-proposed architectures (*3D+time mJ-Net*, *4D mJ-Net*) in relation to manual annotations generated separately by two expert neuroradiologists (NR_1, NR_2) over the test set. An investigation of the inter-observer variability between NR_1 and NR_2 is performed (last row or each class). Note that for the DC, higher values are better (⇑), while for HD and ΔV, lower values are preferable (⇓).

Method	DC ⇑			HD (mm) ⇓			ΔV (ml) ⇓			
	LVO	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**All**	**LVO**	**Non-LVO**	**WIS**	**All**
	Penumbra									
3D+time mJ-Net vs (NR_1 vs NR_2)	0.70±0.1	0.32±0.3	0.51±0.3	2.6±0.5	1.5±0.6	2.1±0.9	36.7±36.4	6.5±4.9	10.1±3.1	24.2±31.1
4D mJ-Net vs (NR_1 vs NR_2)	0.67±0.1	0.25±0.3	0.47±0.3	2.4±0.5	0.9±0.5	1.7±1.0	34.1±30.6	**5.3±6.6**	**0.0±0.0**	21.4±27.7
NR_1 vs NR_2	**0.78±0.1**	**0.65±0.2**	**0.67±0.2**	**2.2±0.4**	**0.8±0.6**	**1.6±1.0**	**33.3±27.7**	5.5±9.2	**0.0±0.0**	**21.0±25.9**
	Core									
3D+time mJ-Net vs (NR_1 vs NR_2)	0.19±0.2	0.01±0.0	0.12±0.2	1.6±0.8	0.3±0.4	1.0±1.0	14.6±18.2	0.9±2.3	**0.0±0.0**	8.7±15.4
4D mJ-Net vs (NR_1 vs NR_2)	0.28±0.2	0.03±0.1	0.18±0.2	1.8±0.8	0.3±0.5	1.1±1.0	21.2±31.7	1.8±4.5	**0.0±0.0**	12.8±25.9
NR_1 vs NR_2	**0.44±0.2**	**0.15±0.2**	**0.30±0.3**	**1.4±0.6**	**0.2±0.4**	**0.9±0.8**	**5.6±4.3**	**0.7±1.9**	**0.0±0.0**	**3.5±4.2**

12.6.4 Inter-observer variability

Two expert neuroradiologists (NR_1, NR_2) manually annotated the scans of 33 randomly selected patients: 19 from the LVO group, 11 from the Non-LVO, and 3 from the WIS subset. The manual annotation images were generated using the same criteria endorsed for creating the ground truth images, as explained in Sec. 12.3.1. An investigation of the inter-observer variability between NR_1, NR_2, and the two best-proposed models is presented in Table 12.6.

12.7 Discussion

Early detection and intervention in AIS patients are of vital importance [137, 210, 211]. In this study, we have proposed different architectures to utilize the 4D CTP input to use the spatio-temporal information better than in existing approaches. We suggest expanding the *mJ-Net* and showing two ways of segmenting ischemic areas in patients suspected of AIS. In addition, we expand another method (*3D-TCN*) for comparison reasons. We use the entire raw 4D CTP data and feed different combinations as input to our proposed approaches to prevent possible loss of spatio-temporal information. Studying the data as an independent volume and neglecting its spatio-temporal nature can lead to the loss of relevant information. All proposed approaches return a series of 2D segmented images as output, later stacked together to produce a 3D volume. Returning a list of 2D images as output is less computationally expensive and less memory intensive than directly returning 3D volumetric data as output [261].

Few studies have adopted 4D datasets in DNN models to detect ischemic lesions in patients affected by a stroke [30, 31, 158, 189]. This is rooted in the high computational complexity of 4D data and the lack of ground truth for the whole set. The limitations that these approaches encounter are as follows: 1) datasets used for the training and evaluation take into account only a subset of the entire population; 2) segmentations are only performed on the core areas, excluding penumbra regions; 3) ground truth images derived from follow-up DWI or NCCT present some limitations [19, 171].

To our knowledge, this is the first study using 4D CTP data to segment both the ischemic regions, penumbra and core. Additionally, we include data from all patients, regardless of stroke severity, to train our models. Rather than entrusting ground truth images from follow-up DWI or NCCT

studies that are usually taken 24 hours or some days after stroke onset, our proposed methods were trained with ground truth images obtained from the CTP captured at admission, including PMs, and follow-up scans (Sec. 12.3.1).

We use three evaluation metrics to assess the models' performances: DC, HD, and ΔV compared with our previously developed algorithms and other state-of-the-art algorithms. Results in Table 12.4 demonstrate that increasing the input dimension benefits achieving more precise segmentation, especially for the Non-LVO and WIS groups, regardless of the class. Thus, when a smaller portion of the brain is affected, the whole dataset's usage helps achieve better segmentation results. The ablation study (Table 12.5) shows how including the pre-processing steps and not re-sampling the CTP scans helped improve the overall segmentation performances. It is possible to evince that by combining multiple pre-processing steps, the overall segmentation performances increased, regardless of whether using or not the re-sampling technique. Adopting the HE step with other pre-processing steps is beneficial for the AIS segmentations; the absence of the HE step decreases the overall performance. It is worth mentioning that using the pre-processing steps and re-sampled CTP scans yields the second-best overall results (last row in Table 12.5), establishing the validity of the pre-processing sequence.

Additionally, we evaluated the computational time for predicting each brain slice (Table 12.3), highlighting how fast each method performs. From the results in Table 12.3, it is possible to notice that the *Multi-input PMs* architecture has the lowest computational cost due to the usage of PMs instead of CTP scans as input. Among the methods utilizing CTP scans as input, the *mJ-Net* has the best computational time since it consists of a more straightforward structure compared to the other models.

Visual results of random validation brain slices are shown in Fig. 12.8, where we can see that our proposed approaches (*3D+time mJ-Net*, *4D mJ-Net*) are less prone to over-segment, especially in the Non-LVO and WIS groups. It is reported that LVO cases are less common compared to Non-LVO. On average, LVOs are estimated to represent around 30% of all AIS cases [198]. Thus, a neural network that can accurately segment patients in the Non-LVO group can be valuable in a real-life scenario. Nonetheless, patients with LVO represent a clinically significant proportion of patients presenting with AIS, especially considering the grim natural course of the disease.

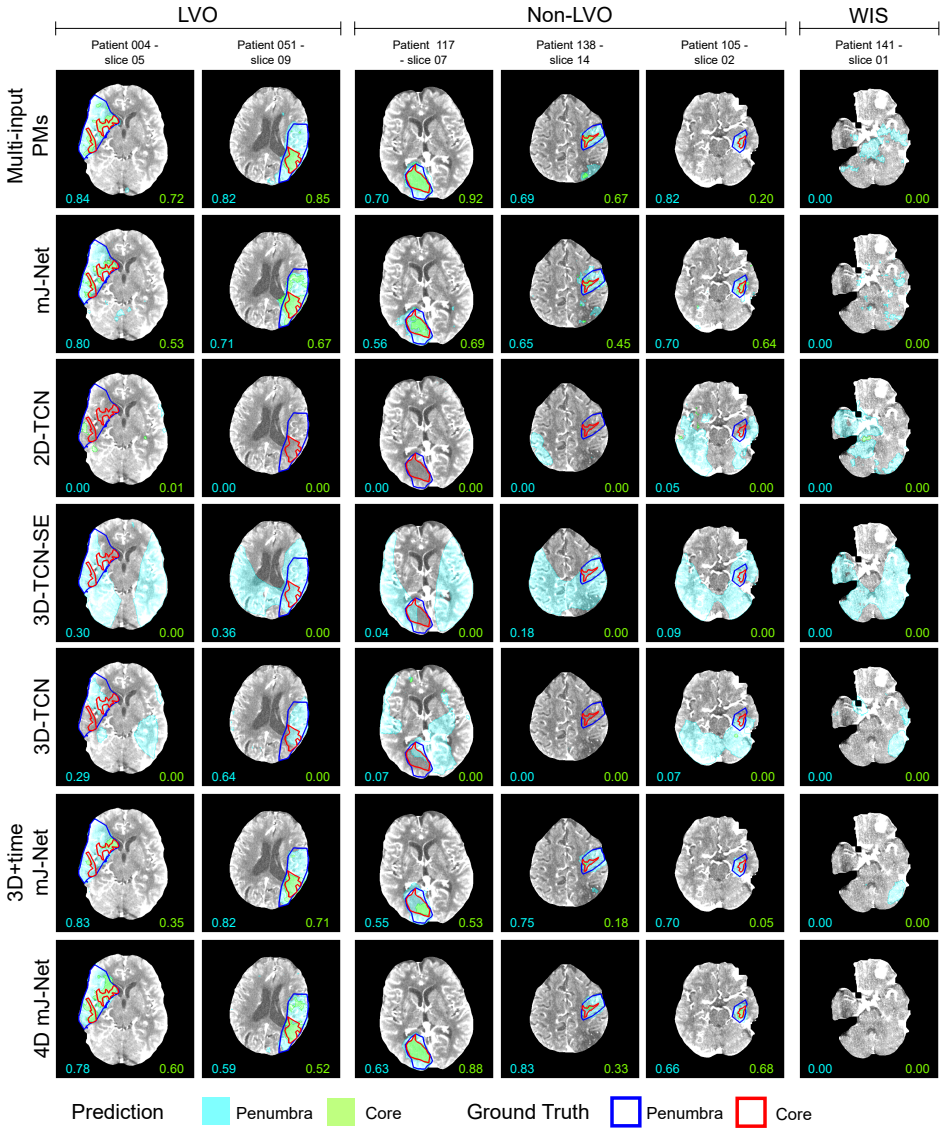


Figure 12.8: Qualitative comparisons for the tested models. The brain slices are taken randomly from distinct patients from the validation set, divided by group (left to right). Each row represents the results for each model involved in the study. The numbers on the bottom sides of every image represent the DC for penumbra (in light blue) and core (in green).

The results presented in Table 12.4 indicate that all *mJ-Net* models have improved where the input data dimension has increased, regardless of the patients' group. Fig. 12.8 shows that using 2D+time input for the *mJ-Net* [208] led to over-segmentation of penumbra class in separate brain tissue sections, brain slice 4-6. The visual results for the Non-LVO and WIS groups highlight the limitations of this model: the over-segmentation of the penumbra regions might affect the usage in a real-life scenario, and an overestimation of the penumbra area can generate uncertainties for treatment decisions.

Adding depth as an extra dimension to the input of models (*3D+time mJ-Net* and *4D mJ-Net*) determines an increment in the performances for both classes in the three patient groups. A significant increase is noticeable for the DC metric in the Non-LVO group, regardless of the class. An essential difference between these two architectures is how they exploit their structures' input. The *3D+time mJ-Net* is considered a late-fusion approach as the data sources are used independently and fused close to decision-making. Statistical results presented for the *3D+time mJ-Net* show promising general performances for the LVO group. However, an underestimation of the core class, regardless of the patient group, can be noticed from the visual results in Fig. 12.8 and the low HD metric. Nevertheless, *3D+time mJ-Net* achieved the best HD for the core class in all the groups. The *3D+time mJ-Net* can precisely segment ischemic regions with large areas, as shown by the first three slices in Fig. 12.8. This can also be manifested in the high DC score achieved for the LVO group for both classes (Table 12.4).

The *4D mJ-Net* network has learned to precisely segment the ischemic regions even without re-sampling the CTP scans, as shown in Table 12.5. The model fuses the data before they are fed to the network. Visual results in Fig. 12.8 and values in Table 12.4 indicate that the *4D mJ-Net* model segments the penumbra class more precisely compared to the other approaches that use raw CTP as their input. This promising performance follows in all patient groups. The *4D mJ-Net* achieved the highest DC metric for core and penumbra regions in patients with Non-LVO. This approach gives the best HD for penumbra in all the groups. The *4D mJ-Net* showed high precision in detecting small ischemic areas, as shown in sample brain slices 3 to 5 in Fig. 12.8. Furthermore, the *4D mJ-Net* model can correctly predict no ischemic regions in WIS patients, as demonstrated by the results for the ΔV in the WIS group. However, it over-segments the core class in patients with LVO. This means that including the complete

spatio-temporal information of the data and following an early fusion approach leads to better prediction in Non-LVO and WIS groups, where small areas are of interest.

Models based on TCN generally showed poor results statistically in Table 12.4 and visually in Fig. 12.8. They extremely over-segment the penumbra class and poorly segment the core class. The original *2D-TCN* and *3D-TCN-SE* were designed to segment only one class, the ischemic core. This can explain the poor performance of segmenting the two classes. Besides, in Amador *et al.* [31] (*3D-TCN-SE*), the model was trained to use only the ipsilateral hemisphere. For a fair comparison, the model's training was done over both hemispheres, which can cause over-segmentation in penumbra regions.

As the name indicates, the *Multi-input PMs* model [207] takes parametric maps and pre-processed data obtained from CTP scans. The experiment results of this model show a high DC value for the penumbra class in the LVO group, as also seen in the first three brain slices of Fig. 12.8. This highlights that this method presents satisfactory results for large ischemic areas. However, when the region's volume is small or vacant, the predictions are not optimal: see brain slices 4 and 6 in Fig. 12.8. Although HD and ΔV are encouraging, DC values show under-segmentation in the core and penumbra classes for the Non-LVO set. Using PMs derived from CTP scans limits the machine to only learn from specific pre-processed information.

The inter-observer variability results, highlighted in Table 12.6, show promising outcomes for the proposed methods with the results achieved by the two expert neuroradiologists (NR_1, NR_2). Similar statistic values can be observed between NR_1 vs. NR_2 and the *4D mJ-Net* for the penumbra class in connection with the LVO group. The same results as the neuroradiologists were achieved by the *4D mJ-Net* for the ΔV in the WIS group. The proposed *3D+time mJ-Net* model produces higher results for the DC compared to the *4D mJ-Net* in association with the penumbra class. However, the results for the core regions could be more satisfactory. The inter-observer variability outcomes for the HD and ΔV highlight substantial similarity among the proposed approaches and the neuroradiologists, except for the core class connected with the LVO group. The difference can be due to an over-segmentation of this particular class, which can be highly complex to detect for the models, considering its small size.

To summarize the discussion, the proposed *3D+time mJ-Net* model can precisely segment large ischemic penumbra regions while presenting an

underestimation of the core class. The proposed *4D mJ-Net* network can segment small ischemic areas with high precision and predict the absence of ischemic regions; nevertheless, it presents an overestimation of the core class in LVO patients.

12.7.1 Common limitations

All the assessed approaches have faced general limitations. The images used during the training of each model are from CT scanners of the same vendor. This causes a lack of diversity in the data. The annotations used as ground truth surround the essential ischemic regions (penumbra and core) but do not represent the areas perfectly. They might leave out small parts of the core spread into the penumbra tissue and details of the penumbra misclassified as healthy brain tissue [208]. Additional qualitative results for all the methods, showing their worst segmentation results, are presented in the supplemental material.

12.8 Conclusions

Fast and precise diagnosis and treatment are of vital importance in AIS patients. In this paper, we proposed to use 4D CTP as input to extract spatio-temporal information for segmenting core and penumbra areas in patients with AIS. This is presented primarily by expanding the *mJ-Net* in two ways (*3D+time mJ-Net* and *4D mJ-Net*). Furthermore, we introduced a novel 4D-Conv layer to exploit spatio-temporal information. Two of our approaches (*3D+time mJ-Net* and *4D mJ-Net*) achieved promising results for all the classes involved. The *3D+time mJ-Net* can precisely delineate large ischemic penumbra areas while underestimating the core class. Our best network (*4D mJ-Net*) can correctly segment penumbra regions, regardless of patient groups, with a 0.53 DC score on average. However, with an average of 0.23 DC score, it overestimates the core class for the LVO group.

We used the entire 4D CTP dataset of all patients and compared models using different input types. We demonstrated that relying only on images derived from the CTP scans (i.e., PMs) or on a restricted number of dimensions (i.e., 2D, 2D+time, 3D) limits the prediction accuracy in DNN-based approaches. Moreover, we segmented both penumbra and core regions

in ischemic brain tissue since an accurate and fast understanding of both is essential for quick treatment decisions in AIS.

Further studies with larger datasets, including images from different vendors and various acquisition parameters, are still needed to validate our methods. The ISLES18 dataset [55] can be used in future work; the dataset uses FIAs as ground truth labels, which is not in the scope of the architectures mentioned above; thus, some changes must be implemented for validating the methods. Due to complex and time-consuming work for manual annotations, further work on optimizing the segmentation using unsupervised neural networks is encouraged.

Acknowledgment

We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

12.9 Supplementary Material

12.9.1 2D and 3D Convolution operations

The following section describes a 2D Convolution (2D-Conv) and a 3D Convolution (3D-Conv).

Let define a 2D-Conv $g''(x, y)$, where a 2D input image $I(x, y) \in \mathbb{R}^2$ is convolved with a 2D kernel $\mathcal{H}(w, h) \in \mathbb{R}^2$ as:

$$\begin{aligned} g''(x, y) &= \mathcal{H}(w, h) \circledast I(x, y) \\ &= \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \mathcal{H}(i, j) I(x + \widetilde{w} - i, y + \widetilde{h} - j), \end{aligned}$$

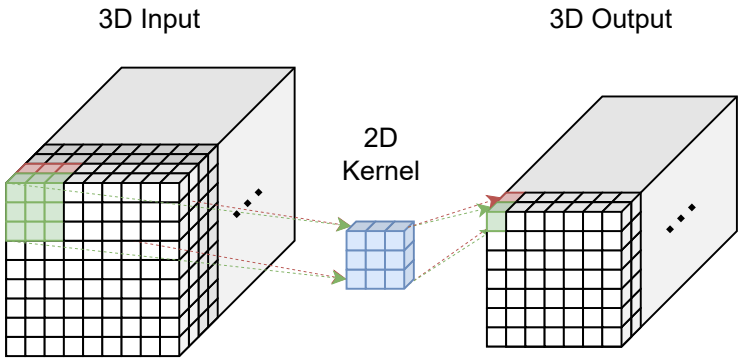
where $\circledast$ is the convolution operation. $\widetilde{h} \equiv \lfloor \frac{1}{2}(h-1) \rfloor, \widetilde{w} \equiv \lfloor \frac{1}{2}(w-1) \rfloor$ correspond to the half-width, and half-height of the kernel $\mathcal{H}$.

Thereafter, let us define a 2D-Conv $g''(x, y, z)$ with a 2D kernel $\mathcal{H}(w, h) \in \mathbb{R}^2$ and a 3D input $I(x, y, z) \in \mathbb{R}^3$. Since the convolution operation is performed slice by slice over the third dimension, the 3D input can be seen

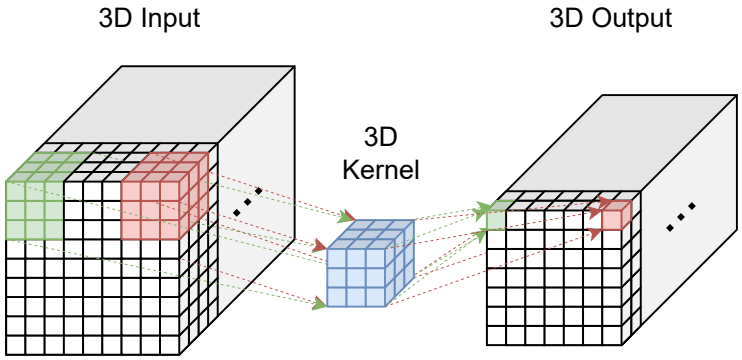
as a list of 2D input $I(x, y, z) = \{I(x, y, z_m) | \forall m \in \{1, \ldots, z_{\max}\}\}$, where $I(x, y, z_m) \in \mathbb{R}^2$ are the coordinates (x, y) at slice z_m:

$$g''(x, y, z) = \mathcal{H}(w, h) \circledast I(x, y, z)$$
$$g''(x, y, z_m) = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \mathcal{H}(i, j) I(\tilde{x}, \tilde{y}, z_m) \; \forall m \in \{1, \ldots, z_{\max}\},$$

where $\tilde{x} \equiv x + \widetilde{w} - i$ and $\tilde{y} = y + \widetilde{h} - j$. Fig. 12.9a presents a visual representation of the 2D-Conv $g''(x, y, z)$ with a 3D input and a 2D kernel.



(a) 2D-Conv with a 3D input and a 2D kernel



(b) 3D-Conv with a 3D input and a 3D kernel

Figure 12.9: Visual examples of (a) 2D-Conv and (b) 3D-Conv. Both examples utilize a 3D input, and no padding is applied. Each box corresponds to a pixel value.

Furthermore, let define a 3D-Conv operation $g'''(x, y, z)$ of a 3D kernel

$\mathcal{H}(w,h,d) \in \mathbb{R}^3$ and a 3D input volume $I(x,y,z) \in \mathbb{R}^3$ as:

$$g'''(x,y,z) = \mathcal{H}(w,h,d) \circledast I(x,y,z)$$
$$= \sum_{k=0}^{d-1} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \mathcal{H}(i,j,k) I(x+\widetilde{w}-i, y+\widetilde{h}-j, z+\widetilde{d}-k),$$

where $\widetilde{d} \equiv \lfloor \frac{1}{2}(d-1) \rfloor$ is the half-depth of $\mathcal{H}$. Fig. 12.9b displays an example of a 3D-Conv with a 3D input and 3D kernel. If we define the 3D input $I(x,y,z)$ as before, then the output of a 3D-Conv $g'''(x,y,z)$ can be rewritten as the sum of multiple 2D-Conv operations over the third dimension:

$$\begin{aligned} g'''(x,y,z) &= \mathcal{H}(w,h,d) \circledast I(x,y,z) \\ &= \sum_{k=0}^{d-1} \mathcal{H}(w,h,k) I(x,y,z+\widetilde{d}-k) \\ &= \sum_{k=0}^{d-1} g''(x,y,z+\widetilde{d}-k), \end{aligned}$$

where $\mathcal{H}(w,h,k) \in \mathbb{R}^2$ is a 2D kernel at index k, while $I(x,y,z+\widetilde{d}-k) \in \mathbb{R}^2$ is a 2D image at slice $z+\widetilde{d}-k$ where $[z+\widetilde{d}-k \in z_m | \forall m \in \{1,\ldots,z_{\max}\}]$. The output dimension is defined as: $\dim(g''') = \dim(I) - \dim(\mathcal{H}) + 1$; thus, if the $\dim(I) \equiv \dim(\mathcal{H})$, then $\dim(g''') = 1$.

Moreover, lets define a 3D-Conv operation $g'''(x,y,z,t)$ of a 3D kernel $\mathcal{H}(w,h,d) \in \mathbb{R}^3$ with a 4D input tensor $I(x,y,z,t) \in \mathbb{R}^4$. The 4D tensor $I(x,y,z,t)$ can be seen as a list of $I(x,y,z,t) = [I(x,y,z,t_n) | \forall n \in \{1,\ldots,t_{\max}\}]$, where each element in the list corresponds to the coordinates (x,y,z) of a t_n element in the temporal dimension. Then, the 3D-Conv operation $g'''(x,y,z,t)$ can be seen as:

$$g'''(x,y,z,t) = \mathcal{H}(w,h,d) \circledast I(x,y,z,t)$$
$$= \sum_{k=0}^{d-1} \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} \mathcal{H}(i,j,k) I(\tilde{x},\tilde{y},\tilde{z},t_n) \; \forall n \in \{1,\ldots,t_{\max}\},$$

where $\tilde{x} \equiv x+\widetilde{w}-i$, $\tilde{y} = y+\widetilde{h}-j$, and $\tilde{z} \equiv z+\widetilde{d}-k$.

If we use a 2D+time kernel $\mathcal{H}(w,h,p) \in \mathbb{R}^3$ and we define $I(x,y,z,t)$ as a list of $I(x,y,z,t) = [I(x,y,z_m,t) | \forall m \in \{1,\ldots,z_{\max}\}]$, where each element

in the list corresponds to the coordinates (x, y, t) of a slice z_m, then the 3D-Conv operation $g'''(x, y, z, t)$ can be rewritten as:

$$\begin{aligned} g'''(x, y, z, t) &= \mathcal{H}(w, h, p) \circledast I(x, y, z, t) \\ g'''(x, y, z_m, t) &= \mathcal{H}(w, h, p) \circledast I(x, y, z_m, t) \; \forall m \in \{1, \ldots, z_{\max}\} \\ &= \sum_{l=0}^{p-1} \mathcal{H}(w, h, l) I(x, y, z_m, t + \widetilde{p} - l) \; \forall m \in \{1, \ldots, z_{\max}\}, \end{aligned}$$

where $\widetilde{p} \equiv \lfloor \frac{1}{2}(p-1) \rfloor$ is the temporal dimension of the kernel $\mathcal{H}$ halved.

12.9.2 4D Convolution Layers

All the layers are written in in Python using Keras (2.3.1)[6] with Tensorflow as the backend and trained using an NVIDIA Tesla V100 GPU (32 GB memory)[7].

Reshape Layer

All the Reshape layers used are directly implemented from Keras[8] due to its public availability and flexibility of reshaping tensors of any dimension.

Dropout Layer

MonteCarlo dropout layers are implemented at the end of the 4D and 2D Convolution blocks. The rate was set to 50%. The MonteCarlo dropout layers were added to reduce uncertainties in the final predictions. The implementation follows the one created by Keras[9] with a single change: the "training" flag is always set to true for the layer to add dropout both during training and inference phases.

[6] https://keras.io/

[7] The code is publicly available at https://github.com/Biomedical-Data-Analysis-Laboratory/4D-mJ-Net

[8] https://keras.io/api/layers/reshaping_layers/reshape/

[9] https://keras.io/api/layers/regularization_layers/dropout/

12.9.3 Additional validation results

Fig. 12.10 compares the worst segmentations from the validation set for each proposed model. It is worth noticing that, regardless of how well the models performed on average, they still show poor segmentation results in particular situations. In Fig. 12.10, some edge cases are shown, such as brain slices with no ischemic region but with false positive segmentations or small ischemic areas hard to detect in the Non-LVO group.

The *Multi-input PMs* architecture shows high false positive results when no ischemic region is present in the brain tissue (first row of Fig. 12.10). The *mJ-Net* model displays difficulties in the ischemic stroke segmentation when the lesion is small or non-existent (second row of Fig. 12.10). We also show two situations where the segmentation yields poor results with a large ischemic region (first two brain slices on the left in the second row of Fig. 12.10).

The brain slices predicted by the *2D-TCN*, *3D-TCN-SE*, and *3D-TCN* models (from third to fifth rows in Fig. 12.10) over-segment the ischemic regions, regardless of the dimension of the area. Even when no ischemia exists in the brain tissue, the models miss-predict penumbra and core areas.

The *3D+time mJ-Net* network achieves promising results overall. Still, it tends to over-segment ischemic regions when the ischemia is narrow or non-existent (sixth row of Fig. 12.10). Finally, the *4D mJ-Net* architecture presents promising results as the *3D+time mJ-Net* model. Nevertheless, it is possible to notice an underestimation of the lesion in the LVO group (first two brain slices on the left in the last row of Fig. 12.10). It is worth mentioning that there is no false positive segmentation in any slice from the WIS group.

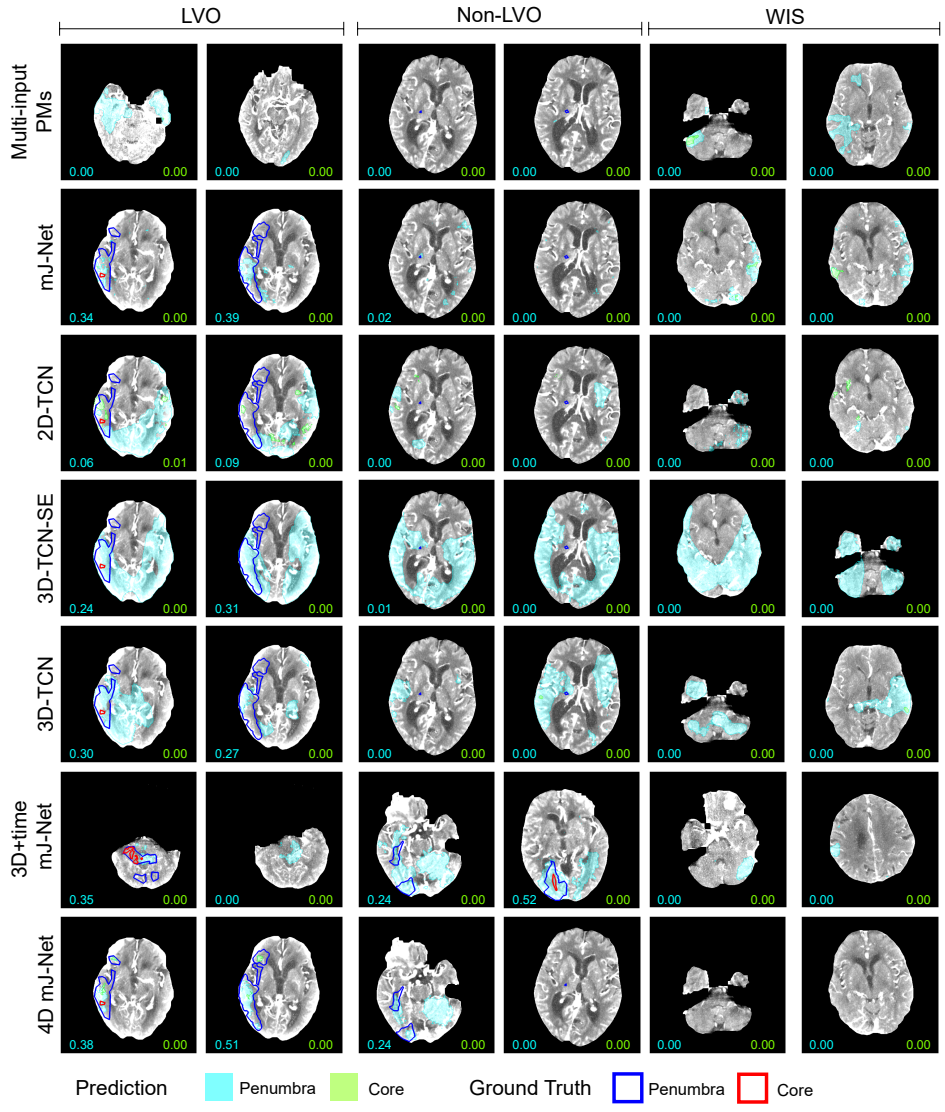


Figure 12.10: Qualitative comparisons for the tested models. The brain slices are the worst segmentations from distinct patients from the validation set, divided by group (left to right). Each row represents the results for each model involved in the study. The numbers on the bottom of every image represent the DC for penumbra (in light blue) and core (in green).

Paper 5:
Self-Supervised Few-Shot Learning for Ischemic Stroke Lesion Segmentation

Self-Supervised Few-Shot Learning for Ischemic Stroke Lesion Segmentation

L. Tomasetti[1], S. Hansen[2], M. Khanmohammadi[1], K. Engan[1], L. J. Høllesli[1,3], K. D. Kurz[1,3], M. Kampffmeyer[2]

[1] Department of Electrical Engineering and Computer Science, University of Stavanger

[2] Department of Physics and Technology, UiT The Arctic University of Norway

[3] Stavanger University Hospital, Stavanger medical imaging laboratory (SMIL)

Published by IEEE, 20th IEEE International Symposium on Biomedical Imaging (ISBI), 2023

https://doi.org/10.1109/ISBI53787.2023.10230655