

# Towards a data-driven treatment of epilepsy: computational methods to overcome low-data regimes in clinical settings

A dissertation in partial fulfilment  
of the requirements for the degree of  
**Doctor of Philosophy**

written at the  
Department of Medical Physics and Biomedical Engineering  
**University College London**

supervised by  
Prof. Sébastien Ourselin  
Prof. John S. Duncan  
and  
Dr Rachel Sparks

by  
*Fernando Pérez García*

London, January 19, 2022

DRAFT – January 19, 2022

©2022

Fernando Pérez García

All Rights Reserved

DRAFT – January 19, 2022

I, Fernando Pérez García, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

DRAFT – January 19, 2022

# Abstract

---

Epilepsy is the most common neurological disorder, affecting around 1% of the population. One third of epilepsies are drug-resistant. If the epileptogenic zone can be localized precisely, curative resective surgery may be performed. However, only 40 to 70% of patients remain seizure-free after the surgery. Presurgical evaluation is a complex, multimodal process that requires subjective clinical decisions, often relying mostly on the multidisciplinary team's experience. Thus, the clinical pathway could benefit from data-driven methods for clinical decision support.

In the last decade, deep learning has seen great advancements thanks to the improvement of graphics processing units (GPUs), the development of new algorithms and the large amounts of data available for training. Using deep learning in clinical settings is challenging as large datasets are rare due to privacy concerns and expensive annotation processes.

This thesis introduces computational methods that pave the way towards a more data driven clinical pathway for the treatment of epilepsy, overcoming the relatively small datasets available. We used transfer learning from human action recognition datasets to characterize epileptic seizures from video-telemetry data. We developed a software framework to predict the location of the epileptogenic zone given seizure semiologies, based on retrospective information from the literature. We trained models using self-supervised and semi-supervised learning to perform quantitative analysis of resective surgery by segmenting resection cavities on brain magnetic resonance images (MRIs). Finally, we shared datasets and software tools that will accelerate research in medical image computing, particularly in the field of epilepsy.

DRAFT – January 19, 2022

# Impact statement

---

The algorithms and software developed in this thesis have potential benefits both outside and inside academia. The publications and tools that I have shared and present here are already accelerating research and **facilitating clinical practice**.

**Inside the academic environment, the outputs of this thesis are being used and built upon by multiple institutions.** We curated the EPISURG dataset, comprising 699 MRIs from 430 patients who underwent epilepsy surgery at the National Hospital for Neurology and Neurosurgery (NHNN) between 1990 and 2018, including 200 manual annotations from three different human raters. To the best of our knowledge, EPISURG is the first open annotated database of post-resection MRI for epilepsy patients. EPISURG has been made open-access and can be freely downloaded from the UCL Research Data Repository [**Pérez-García** 20d]. As we proposed in [**Pérez-García** 21a] (a collaboration between the NHNN and hospitals in Milan, Marseille and Paris), our method to simulate brain lesions is being used in research for brain tumor segmentation [Zhang 21], which demonstrates its generalizability and potential to solve related problems.

TorchIO, our software library for medical image processing, has been cited over 70 by researchers at numerous hospitals and leading academic institutions in various countries including the United Kingdom, the United States, Canada, Australia, Japan, China, Korea, Singapore, India, Iran, Pakistan, Saudi Arabia, Egypt, Kazakhstan, Russia, Switzerland, Germany, Austria, France, Sweden, Spain, Italy, Belgium and the Netherlands<sup>1</sup>. The package is downloaded from the Python Package Index (PyPI) over 9000 times per month<sup>2</sup> and the GitHub repository has

---

<sup>1</sup><https://scholar.google.co.uk/scholar?oi=bibs&cites=11818021599290863762>

<sup>2</sup><https://pypistats.org/packages/torchio>

over 1200 stars<sup>3</sup>. I was invited to present TorchIO at the UCL Centre for Medical Image Computing (CMIC), the UCL Wellcome / EPSRC Centre for Interventional and Surgical Sciences (WEISS), the UCL Institute of Nuclear Medicine (INM), and the School of Biomedical Engineering & Imaging Sciences (BMEIS) at King's College London. Additionally, I was invited to present the library at multiple events organized by Meta: the PyTorch Ecosystem Day 2021<sup>4</sup>, the PyTorch Community Voices Webinar Series<sup>5</sup>, and the PyTorch Developer Day 2021<sup>6</sup>. TorchIO was also disseminated at the *LATAM Minds* podcast from the Latin American School of Artificial Intelligence<sup>7</sup> and within an entry to the MICCAI Educational Challenge 2020<sup>8</sup>.

Outside academia, the presented work paves the way for a data-driven paradigm to diagnose and treat epilepsy. Our methods may allow clinicians to use data-driven approaches for clinical decision support, as we overcome the scarcity of annotated data in clinical settings. Our work on classification of seizure videos could be further developed into a commercial product, such as a mobile phone application, to analyze seizures inside and outside the hospital. This application could be used a clinical decision support tool to give patients higher priority for surgery if their seizures imply a higher risk of death, or to help in presurgical evaluation. The developed software for visualization of seizure semiology may be used for a faster and more objective planning of electrodes implantation and resective surgery.

---

<sup>3</sup><https://github.com/fepellar/torchio>

<sup>4</sup><https://pytorch.org/ecosystem/pted/2021>

<sup>5</sup><https://www.youtube.com/watch?v=UEUVSw5-M9M>

<sup>6</sup><https://pytorch.org/blog/pytorch-developer-day-2021/>

<sup>7</sup><https://open.spotify.com/episode/03gYyFmwWmiY38U30xJ0Zp?si=4bd8e97ca1a74685>

<sup>8</sup><https://github.com/fepellar/miccai-educational-challenge-2020>

# Acknowledgments

---

DRAFT – January 19, 2022

# List of publications

---

## First-author articles in peer-reviewed journals

1. **Fernando Pérez-García**, Reuben Dorent, Michele Rizzi, Francesco Cardinale, Valerio Fazzini, Vincent Navarro, Caroline Essert, Irène Ollivier, Tom Vercauteren, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *A self-supervised learning strategy for postoperative brain cavity segmentation simulating resections*. International Journal of Computer Assisted Radiology and Surgery, June 2021
2. **Fernando Pérez-García**, Rachel Sparks and Sébastien Ourselin. *TorchIO: A Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning*. Computer Methods and Programs in Biomedicine, volume 208, page 106236, September 2021

## First-author articles in peer-reviewed conferences with published proceedings

1. **Fernando Pérez-García**, Roman Rodionov, Ali Alim-Marvasti, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Simulation of Brain Resection for Cavity Segmentation Using Self-supervised and Semi-supervised Learning*. In MICCAI 2020, Lecture Notes in Computer Science, pages 115–125, Cham, 2020. Springer International Publishing
2. **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures*. In Medical Image Computing and

Computer Assisted Intervention – MICCAI 2021, Lecture Notes in Computer Science, pages 334–344, Cham, 2021. Springer International Publishing

## Co-author articles in peer-reviewed journals

1. Vejay N. Vakharia, Rachel E. Sparks, Kuo Li, Aidan G. O’Keeffe, **Fernando Pérez-García**, Lucas G. S. França, Andrew L. Ko, Chengyuan Wu, Joshua P. Aronson, Brett E. Youngerman, Ashwini Sharan, Guy McKhann, Sébastien Ourselin and John S. Duncan. *Multicenter validation of automated trajectories for selective laser amygdalohippocampectomy*. Epilepsia, volume 60, number 9, pages 1949–1959, 2019
2. Ali Alim-Marvasti, **Fernando Pérez-García**, Karan Dahele, Gloria Romagnoli, Beate Diehl, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson and John S. Duncan. *Machine Learning for Localizing Epileptogenic-Zone in the Temporal Lobe: Quantifying the Value of Multimodal Clinical-Semiology and Imaging Concordance*. Frontiers in Digital Health, volume 3, page 8, 2021
3. Alejandro Granados, **Fernando Pérez-García**, Martin Schweiger, Vejay Vakharia, Sjoerd B. Vos, Anna Misericordi, Andrew W. McEvoy, John S. Duncan, Rachel Sparks and Sébastien Ourselin. *A generative model of hyperelastic strain energy density functions for multiple tissue brain deformation*. International Journal of Computer Assisted Radiology and Surgery, volume 16, number 1, pages 141–150, January 2021
4. Ali Alim-Marvasti, Gloria Romagnoli, Karan Dahele, Hadi Modarres, **Fernando Pérez-García**, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson, Fahmida Chowdhury, Beate Diehl and John S. Duncan. *Probabilistic Landscape of Seizure Semiology Localising Values*. Under submission (Brain Communications), 2021

5. Ali Alim-Marvasti, **Fernando Pérez-García**, Gloria Romagnoli, Peter N. Taylor, Vejay Vakharia, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson and John S. Duncan. *Mapping Epileptic Symptomatology to Cortical Epileptogenicity in MNI Space: Seizure Semiology-to-Brain Visualisation Tool.* Under submission (NeuroImage: Clinical), 2021

## Abstracts in conferences

1. **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation.* In International League Against Epilepsy (ILAE) British Branch Virtual Annual Scientific Conference, 2020
2. **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation.* In American Epilepsy Society (AES) Annual Meeting, 2020
3. **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures.* In American Epilepsy Society (AES) Annual Meeting, 2021

DRAFT – January 19, 2022

# Table of contents

---

<b>Abstract</b>	<b>5</b>
<b>Impact statement</b>	<b>7</b>
<b>Acknowledgments</b>	<b>9</b>
<b>List of publications</b>	<b>11</b>
<b>List of abbreviations</b>	<b>18</b>
<b>List of figures</b>	<b>21</b>
<b>List of tables</b>	<b>23</b>
<b>1 Introduction</b>	<b>25</b>
1.1 Motivation . . . . .	25
1.1.1 Video-telemetry and SUDEP . . . . .	26
1.1.2 Presurgical evaluation . . . . .	27
1.1.3 Resective surgery . . . . .	28
1.2 Thesis outline . . . . .	29
<b>2 Classification of epileptic seizure videos</b>	<b>33</b>
2.1 Introduction . . . . .	33
2.2 Materials and methods . . . . .	37
2.2.1 Video acquisition . . . . .	37
2.2.2 Dataset description and ground-truth definitions . . . . .	37
2.2.3 Snippet-level classification . . . . .	38
2.2.4 Seizure-level classification . . . . .	38
2.3 Experiments and results . . . . .	39
2.3.1 Evaluation of feature extractors for snippets . . . . .	40
2.3.2 Aggregation for seizure-level classification . . . . .	41
2.4 Discussion . . . . .	44

<b>3 Visualization of the epileptogenic zone on brain images</b>	<b>47</b>
3.1 Introduction . . . . .	48
3.2 System architecture . . . . .	49
3.2.1 Implementation . . . . .	52
3.2.2 Data loading . . . . .	52
3.2.3 Input semiologies . . . . .	53
3.2.4 Querying the database . . . . .	54
3.2.5 Visualization . . . . .	55
3.3 Example of clinical usage . . . . .	56
3.4 Discussion . . . . .	58
<b>4 Segmentation of postoperative brain resection cavities</b>	<b>61</b>
4.1 Introduction . . . . .	62
4.1.1 Motivation . . . . .	62
4.1.2 Related works . . . . .	62
4.1.3 Contributions . . . . .	64
4.2 Methods . . . . .	64
4.2.1 Learning strategy . . . . .	64
4.2.2 Resection simulation for self-supervised learning . . . . .	69
4.3 Experiments and results . . . . .	74
4.3.1 Data . . . . .	74
4.3.2 Network architecture and implementation details . . . . .	77
4.3.3 Processing during training . . . . .	78
4.3.4 Experiments . . . . .	80
4.4 Discussion . . . . .	86
<b>5 TorchIO: a software library for medical image processing</b>	<b>89</b>
5.1 Introduction . . . . .	89
5.1.1 Motivation . . . . .	91
5.1.2 Related work . . . . .	96
5.2 Methods . . . . .	98
5.2.1 Data . . . . .	99
5.2.2 Transforms . . . . .	103
5.3 Results . . . . .	111
5.3.1 Code availability . . . . .	111
5.3.2 Usage examples . . . . .	112
5.4 Discussion . . . . .	114

---

<b>6 Discussion</b>	<b>117</b>
6.1 Contributions . . . . .	118
6.1.1 Classification of epileptic seizure videos . . . . .	118
6.1.2 Visualization of the epileptogenic zone on brain images . . . .	118
6.1.3 Segmentation of postoperative brain resection cavities . . . .	119
6.1.4 TorchIO: a software library for medical image processing . . . .	120
<b>Bibliography</b>	<b>121</b>

# List of abbreviations

---

<b>3DMMI</b>	3D multimodality imaging
<b>ADNI</b>	Alzheimer’s Disease Neuroimaging Initiative
<b>AED</b>	anti-epileptic drug
<b>AMP</b>	automatic mixed precision
<b>API</b>	application programming interface
<b>ATP</b>	automatic trajectory planning
<b>BITE</b>	Brain Images of Tumors for Evaluation
<b>BLSTM</b>	bidirectional LSTM
<b>BMEIS</b>	School of Biomedical Engineering & Imaging Sciences
<b>CI</b>	continuous integration
<b>CLI</b>	command-line interface
<b>CMIC</b>	Centre for Medical Image Computing
<b>CNN</b>	convolutional neural network
<b>CPU</b>	central processing unit
<b>CQV</b>	coefficient of quartile variation
<b>CSF</b>	cerebrospinal fluid
<b>CT</b>	computerized tomography
<b>DAG</b>	directed acyclic graph
<b>DICOM</b>	Data Imaging and Communications in Medicine
<b>DLTK</b>	Deep Learning Toolkit
<b>dMRI</b>	diffusion MRI
<b>DSC</b>	Dice score coefficient
<b>EEG</b>	electroencephalography
<b>EMU</b>	epilepsy monitoring unit
<b>EZ</b>	epileptogenic zone
<b>FFT</b>	Fast Fourier Transform

<b>fMRI</b>	functional MRI
<b>FOS</b>	focal onset seizure
<b>FOV</b>	field of view
<b>FPS</b>	frames per second
<b>GESTURES</b>	Generalized Epileptic Seizure classification from video-Telemetry Using REcurrent convolutional neural networkS
<b>GIF</b>	geodesical information flows
<b>GMM</b>	Gaussian mixture model
<b>GPU</b>	graphics processing unit
<b>GUI</b>	graphical user interface
<b>HAR</b>	human action recognition
<b>HEVC</b>	High Efficiency Video Coding
<b>iEEG</b>	intracranial EEG
<b>INM</b>	Institute of Nuclear Medicine
<b>ITK</b>	the Insight Toolkit
<b>IXI</b>	Information eXtraction from Images
<b>LSTM</b>	long short-term memory
<b>MNI</b>	Montreal Neurological Institute
<b>MONAI</b>	the Medical Open Network for AI
<b>MRI</b>	magnetic resonance image
<b>NHNN</b>	National Hospital for Neurology and Neurosurgery
<b>NIfTI</b>	Neuroimaging Informatics Technology Initiative
<b>MIP</b>	maximum intensity projection
<b>OASIS</b>	Open Access Series of Imaging Studies
<b>PGES</b>	postictal generalized EEG suppression
<b>PIP</b>	Pip Installs Packages
<b>PNG</b>	Portable Network Graphics
<b>PReLU</b>	parametric rectified linear unit
<b>PyPI</b>	Python Package Index
<b>RC</b>	resection cavity

<b>RGB</b>	red-green-blue
<b>RNN</b>	recurrent neural network
<b>SEEG</b>	stereoelectroencephalography
<b>SFCNN</b>	single-frame CNN
<b>STCNN</b>	spatiotemporal CNN
<b>SUDEP</b>	sudden and unexpected death in epilepsy
<b>SVT</b>	Semiology Visualization Tool
<b><math>T_1\text{w}</math></b>	$T_1$ -weighted
<b><math>T_1\text{wCE}</math></b>	$T_1\text{w}$ MRI with contrast enhancement
<b>TCS</b>	focal to bilateral tonic-clonic seizure
<b>TSN</b>	temporal segment network
<b>TTA</b>	test-time augmentation
<b>US</b>	ultrasound
<b>VTK</b>	the Visualization Toolkit
<b>WHO</b>	World Health Organization
<b>WEISS</b>	Wellcome / EPSRC Centre for Interventional and Surgical Sciences

# List of Figures

---

1.1	Examples of decerebrate posturing . . . . .	26
1.2	Electrodes used for intracranial EEG . . . . .	28
1.3	Examples of brain resection cavities after curative epilepsy surgery . . . . .	30
2.1	Overview of the GESTURES architecture . . . . .	42
2.2	Probability distributions to sample snippets from video segments . . . . .	42
2.3	Quantitative results for seizure-level classification . . . . .	43
3.1	General architecture of the Semiology Visualization Tool . . . . .	51
3.2	Semiology visualization module after loading a patient’s data . . . . .	53
3.3	List of pre-defined semiologies in the GUI . . . . .	54
3.4	Adding a custom semiology to the semiology visualization module . . . . .	55
3.5	Results of querying the database with the <i>Epigastric</i> semiology . . . . .	56
3.6	Demonstration of advanced visualization settings . . . . .	57
3.7	Querying the database using a retrospective real case . . . . .	58
3.8	Comparison of the probability map and the postoperative MRI . . . . .	59
4.1	Learning strategies for cavity segmentation . . . . .	65
4.2	Geodesic polyhedron from a subdivided icosahedron . . . . .	70
4.3	Example of a sphere with perturbed vertices . . . . .	71
4.4	Simulation of the ground-truth cavity label . . . . .	71
4.5	Example of a resectable hemisphere mask . . . . .	73
4.6	Simulation of resected image using alpha blending . . . . .	74
4.7	Simulation of resection cavities with increasing complexity . . . . .	81
4.8	Dice score without before and after fine-tuning . . . . .	82
4.9	Qualitative evaluation of fine-tuning . . . . .	82
4.10	Generating reliable pseudolabels for semi-supervised learning . . . . .	84
4.11	Qualitative evaluation on postoperative brain tumor data . . . . .	85
4.12	Qualitative result on an intraoperative MRI . . . . .	86
5.1	Importance of spatial metadata in medical imaging processing . . . . .	92
5.2	General diagram of TorchIO, its dependencies and its interfaces . . . . .	99
5.3	Usage example of different data structures in TorchIO . . . . .	100

5.4	Data pipelines for training with whole volumes and patches . . . . .	102
5.5	A selection of data augmentation techniques in TorchIO v0.18.0 . .	107
5.6	Graph representation of a composed transform . . . . .	109
5.7	TorchIO documentation . . . . .	112
5.8	Graphical user interface for TorchIO . . . . .	113

# List of Tables

---

2.1	Performance of the feature extractors . . . . .	41
3.1	Result of querying an imaginary database with one semiology . . . . .	50
4.1	Datasets summary . . . . .	75
4.2	Lobar distribution of resection types in EPISURG. . . . .	77
5.1	Transforms included in TorchIO v0.18.0 . . . . .	104

DRAFT – January 19, 2022

# CHAPTER 1

# Introduction

---

## Table of contents

<b>1.1 Motivation . . . . .</b>	<b>25</b>
1.1.1 Video-telemetry and SUDEP . . . . .	26
1.1.2 Presurgical evaluation . . . . .	27
1.1.3 Resective surgery . . . . .	28
<b>1.2 Thesis outline . . . . .</b>	<b>29</b>

---

## 1.1 Motivation

Epilepsy is a neurological condition characterized by abnormal brain activity that gives rise to recurring seizures, affecting about 600 000 people in the UK and 50 million people worldwide [NICE 12, Fiest 17]. This means that ~~around 1%~~ of the world population live with epilepsy.

Epilepsy is **associated with “stigma, psychiatric comorbidity, and high economic costs”**, and it has been ranked by the World Health Organization (WHO) as the second most burdensome neurological disorder in terms of disability-adjusted life years [Fiest 17]. Epileptic seizures, especially repeated, prolonged and uncontrolled tonic-clonic seizures, induce neuronal loss and have long-term behavioral and cognitive consequences [Sutula 03]. The overall risk of dying is 1.6 to 3 times higher ~~in people with epilepsy than in~~ the general population [Forsgren 05]. There are 21 epilepsy-related deaths in the UK every week<sup>1</sup>, half of which are sudden and unexpected deaths in epilepsy (SUDEPs).

---

<sup>1</sup><https://sudep.org/epilepsy-deaths>



Figure 1.1: Examples of decerebrate posturing during focal to bilateral tonic-clonic seizures (TCSs).

### 1.1.1 Video-telemetry and SUDEP

SUDEP is **more** formally defined as “the sudden, unexpected, witnessed or unwitnessed, non-traumatic, and non-drowning death in patients with epilepsy with or without evidence for a seizure, and excluding documented status epilepticus, in which postmortem examination does not reveal a structural or toxicological cause for death” [Nashef 97]. **It is the most common category of epilepsy-related deaths [Devinsky 16].**

Although the underlying mechanisms of SUDEP are not fully understood, several **prognostic risk factors** have been identified [So 08, Jha 21]. Seizure semiology, the analysis of clinical signs during an epileptic seizure, is an important tool to predict the risk of SUDEP. Some motor semiologies such as decerebration (Fig. 1.1) **have been associated** with postictal generalized EEG suppression (PGES), which has in turn **been** associated with a higher risk of SUDEP [Alexandre 15, Vilella 21]. Therefore, classifying seizures ~~by~~ motor semiologies could ~~be used to~~ assess the risk of SUDEP and ~~modify~~ the treatment of epilepsy or give higher priority for surgery to patients with a higher risk. However, manual assessment of seizure videos by neurophysiologists is time-consuming, as videos can be very long, and **presents a high intra- and inter-rater variability**, especially between observers from different epilepsy centers [Tufenkjian 12].



**Our research** on automatic classification of seizure videos is presented in Chapter 2.

### 1.1.2 Presurgical evaluation

Anti-epileptic drugs (AEDs) are normally used to treat epilepsy. The aim of these drugs is not to cure the epilepsy, but to decrease the seizure frequency. In roughly one third of the patients, ~~antiepileptic drugs~~ do not adequately control seizures. These patients are described as being medically refractory. Half of the medically refractory epileptic patients have focal epilepsy, which may be treated by curative resective surgery.

The objective of resective epilepsy surgery is the complete resection or complete disconnection of the epileptogenic zone (EZ), which is defined as “the area of cortex indispensable for the generation of clinical seizures” [Rosenow 01]. The surgery is performed if the EZ can be definitely identified and is located in a part of the brain that may be removed without causing neurological, cognitive or neuropsychiatric deficit [Jobst 15].



To locate the EZ, several preoperative imaging scans such as  $T_1$ -weighted ( $T_1w$ ) magnetic resonance images (MRIs) are acquired ~~in order~~ to identify structural cerebral abnormalities, such as focal cortical dysplasia [Kabat 12], hippocampal sclerosis [Thom 14] or brain tumors. If a structural lesion is found that is concordant with the results of electroencephalography (EEG) and video-telemetry, the patient can be recommended for surgery after a functional MRI (fMRI) study to assess language lateralization [Duncan 16]. However, 15 to 30% of patients with focal epilepsy are MRI-negative, meaning they have no distinct abnormalities visible from imaging or ~~have~~ discordant video EEG telemetry [Bien 09]. Results are discordant when they suggest different EZ localizations. For example, an MRI can show a lesion near the motor cortex, but EEG shows abnormal activity in the occipital lobe. In such cases, intracranial electrodes may be implanted to acquire intracranial EEG (iEEG) signals that for precise localization of the EZ (Fig. 1.2).

The brain structures in which the electrodes are implanted are chosen by clinicians after interpretation of the aforementioned non-invasive data acquisition modalities,

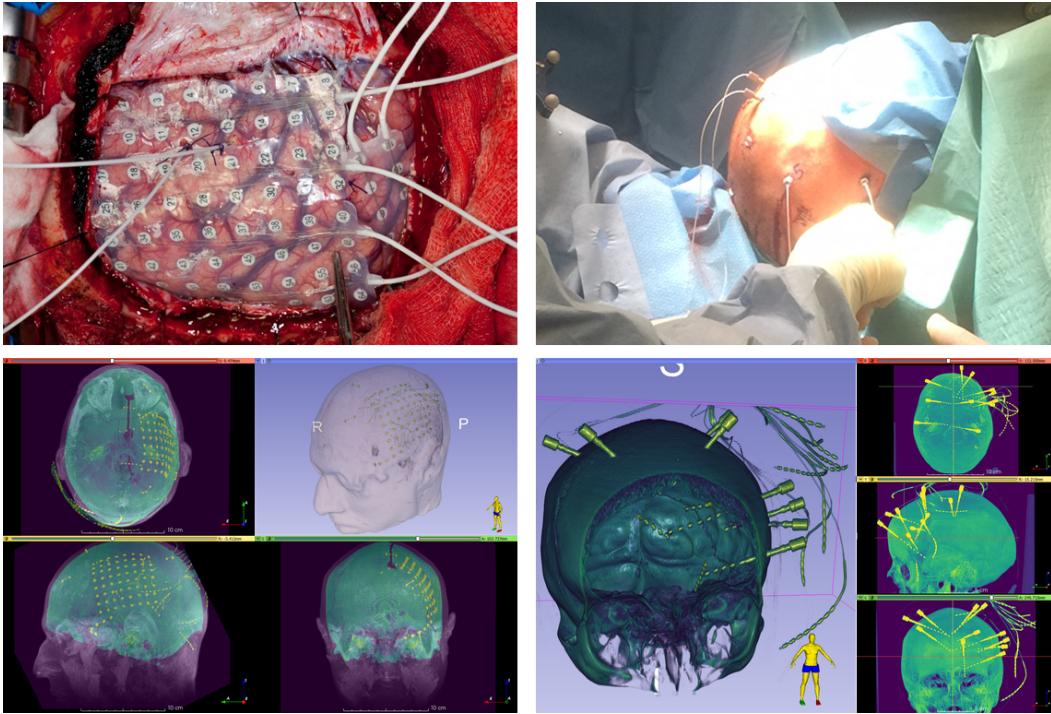


Figure 1.2: Electodes used for iEEG. Left: subdural grid electrode; right: stereoelectroencephalography (SEEG) depth electrodes. Top: intraoperative photos after electrode placement for each implantation procedure; bottom: volume renderings and maximum intensity projections (MIPs) of post-implantation images. The skull is shown in green. The electrodes and cables are shown in yellow.

particularly seizure semiology, video-EEG and MRI. There are variations in implantation strategies between centers, based on views regarding the relationship between semiological features and brain regions involved.

To determine an objective implantation plan for the iEEG electrodes, automatic data-driven methods may be used. Our tool to objectively map seizure semiologies to brain structures based on the literature is presented in Chapter 3.

### 1.1.3 Resective surgery

If the iEEG findings enable a definitive localization of the EZ, surgery to resect the determined EZ may be performed. Currently, only 40% to 70% of patients with refractory focal epilepsy are seizure-free<sup>2</sup> after surgery [Jobst 15]. This is, in part, due

<sup>2</sup>In this context, “seizure freedom” refers to one year without seizures from the surgery.

to limitations identifying the EZ. Retrospective studies relating presurgical clinical features and resected brain structures to surgical outcome provide useful insight A to guide EZ resection [Jobst 15]. To quantify resected structures, the resection cavity, which is mostly composed of cerebrospinal fluid (CSF) (Fig. 1.3), must be segmented on the postoperative MRI. A preoperative image with a corresponding brain parcellation can then be registered to the postoperative MRI to identify resected structures.

Manual segmentation of brain resection cavities on 3D images is a time-consuming process requiring highly trained individuals, and a high inter-rater variability is usual [Havaei 17]. A tool for automatic segmentation would facilitate and accelerate the research to better understand the relation between the clinical features and surgical outcomes. Our work on automatic segmentation of brain resection cavities is presented in Chapter 4.

## 1.2 Thesis outline

This thesis is structured in six chapters. The order of Chapters 2 to 4 follows the temporal order of the associated elements of the clinical pathway. The literature review related to each topic is presented in the corresponding chapters.

In the **current chapter**, an overview of epilepsy was first presented. Then, the relevant elements in the clinical pathway for the treatment of epilepsy and their corresponding challenges were described.

The **second chapter** (Classification of epileptic seizure videos) proposes Generalized Epileptic Seizure classification from video-Telemetry Using REcurrent convolutional neural networkS (GESTURES), our open-source framework for automatic classification of seizures from videos.

The **third chapter** (Visualization of the epileptogenic zone on brain images) introduces the Semiology Visualization Tool (SVT), a piece of software to visualize a probability map of the EZ on neuroimages given a set of observed seizure semiologies,

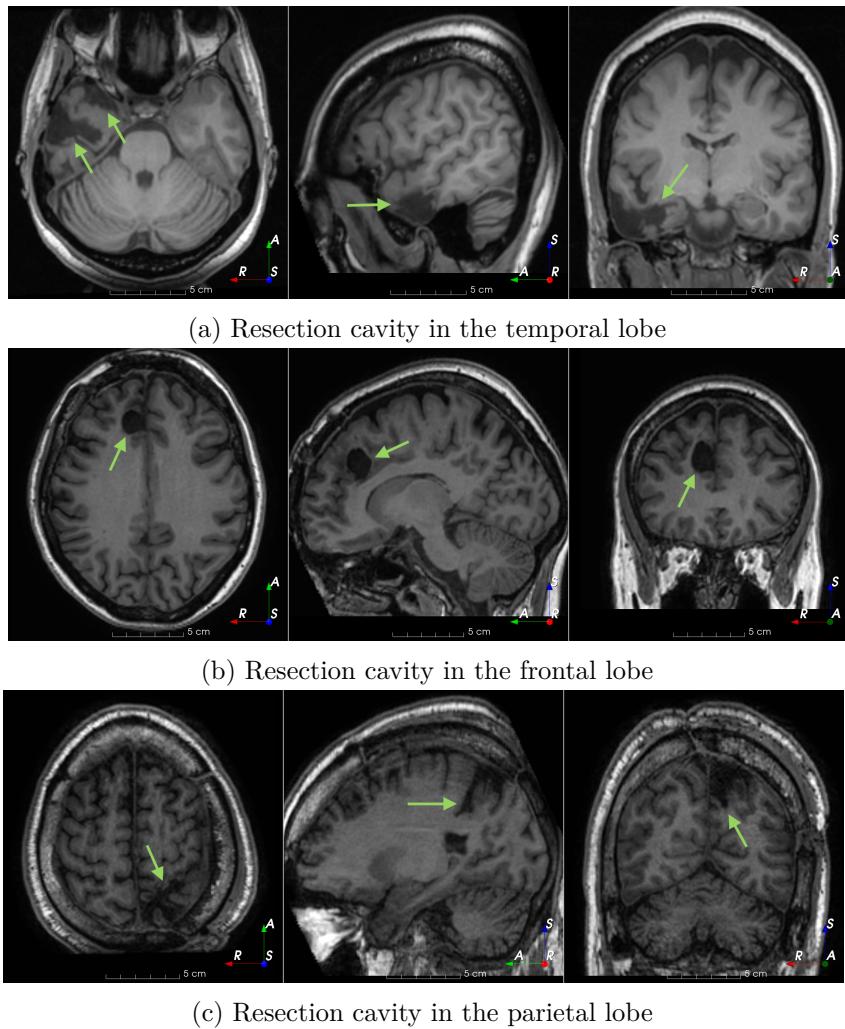


Figure 1.3: Examples of brain resection cavities (marked with green arrows) after curative epilepsy surgery.

using an evidence-based approach.

The **fourth chapter** (Segmentation of postoperative brain resection cavities) presents our framework for segmentation of brain resection cavities from postoperative MRIs.

The **fifth chapter** (TorchIO: a software library for medical image processing) describes our open-source Python library TorchIO [Pérez-García 21e], which was initially developed in the context of the work presented in Chapter 4.

Finally, the **sixth chapter** (Discussion) concludes the thesis with a summary of the contributions presented in each chapter and outlines potential future research directions.

DRAFT – January 19, 2022

## CHAPTER 2

# Classification of epileptic seizure videos

---

## Table of contents

<b>2.1</b>	<b>Introduction</b>	<b>33</b>
<b>2.2</b>	<b>Materials and methods</b>	<b>37</b>
2.2.1	Video acquisition	37
2.2.2	Dataset description and ground-truth definitions	37
2.2.3	Snippet-level classification	38
2.2.4	Seizure-level classification	38
<b>2.3</b>	<b>Experiments and results</b>	<b>39</b>
2.3.1	Evaluation of feature extractors for snippets	40
2.3.2	Aggregation for seizure-level classification	41
<b>2.4</b>	<b>Discussion</b>	<b>44</b>

**Foreword** This chapter is adapted and contains content from the work presented in:

- **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures*. In American Epilepsy Society (AES) Annual Meeting, 2021
- **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures*. In Medical Image Computing and Computer Assisted Intervention – MICCAI 2021, Lecture Notes in Computer Science, pages 334–344, Cham, 2021. Springer International Publishing

## 2.1 Introduction

Seizure semiology, “the historical elicitation or observation of certain symptoms and signs” during seizures, provides context to infer epilepsy type [Fisher 17]. Focal

onset seizures (FOSs) start in a region of one hemisphere. If they spread to both hemispheres, they are said to *generalize*, becoming focal to bilateral tonic-clonic seizures (TCSs) [Fisher 17]. In TCSs, the patient first presents semiologies associated with a FOS, such as head turning or mouth and hand automatisms. This is followed by a series of phases, in which muscles stiffen (tonic phase) and limbs jerk rapidly and rhythmically (clonic phase). TCSs put patients at risk of injury and, if the seizure does not self-terminate rapidly, can result in a medical emergency. The risk of sudden and unexpected death in epilepsy (SUDEP) depends on epilepsy and seizure characteristics as well as living conditions. TCSs in particular increase SUDEP risk substantially [Nashef 12]. In a small number of SUDEP cases occurring in epilepsy monitoring units (EMUs), death was preceded by a TCS followed by cardiorespiratory dysfunction minutes after seizure offset [Ryvlin 13]. Identifying semiologies related to increased risk of SUDEP to appropriately target treatment is an open research question. One limitation determining SUDEP risk factors is that inter-rater reliability based on qualitative visual analysis is poor for most semiological features (e.g., limb movement, head pose or eye gaze), especially between observers from different epilepsy centers [Tufenkjian 12]. Therefore, automatic and quantitative analysis of video-recorded seizures is needed to standardize assessment of seizure semiology across multicenter studies [Ahmedt-Aristizabal 17].

Early quantitative analysis studies of epileptic seizures evaluated patient motion by attaching infrared reflective markers to key points on the body or using cameras with color and depth streams [Li, Z. 02, Cunha 03, O'Dwyer 07, Cunha 16]. These methods are not robust to occlusion by bed linens or clinical staff, differences in illumination and pose, or poor video quality caused by compression artifacts or details out of focus.

Neural networks can overcome these challenges by automatically learning features from the training data that are more robust to variations in the data distribution. Most related works using neural networks focus on classifying the *epilepsy type* by

predicting the location of the epileptogenic zone (EZ), e.g., “temporal lobe epilepsy” vs. “extratemporal lobe epilepsy”, from short ( $\leq 2$  s) snippets extracted from videos of one or more seizures [Ahmedt-Aristizabal 18c, Ahmedt-Aristizabal 18a, Ahmedt-Aristizabal 18b, Maia 19, Karácsony 20]. Typically, this is done as follows. First, the bed is detected in the first frame and the entire video is cropped so the field of view (FOV) is centered on the bed. During training, a convolutional neural network (CNN) is used to extract features for each frame in a sampled snippet. Then, a recurrent neural network (RNN) aggregates the features into a *snippet-level* representation and a fully-connected layer predicts the epilepsy type. Finally, a *subject-level* prediction is obtained by averaging all snippet-level predictions. This approach has several disadvantages. First, it is not robust to incorrect bed detection or changes in the FOV due to zooming or panning. Second, the order of semiologies is ignored, as the epilepsy type is predicted from short snippets independently of their occurrence during a seizure. Moreover, patients with the same epilepsy type may present different ~~seizure~~ types. Finally, training neural networks from small datasets, as is often the case in clinical settings, leads to **limited results**.

The goal of this work is to compute *seizure-level* representations of arbitrarily long videos when a small dataset is available~~x~~, which is typically the case in EMUs.

To overcome the challenge of training with small datasets, transfer learning from spatiotemporal CNNs (STCNNs) trained for human action recognition (HAR) can be used [Karácsony 20]. Although seizures are, strictly speaking, not actions, HAR models are expected to encode strong representations of human motion that may be relevant for seizure characterization. These methods are typically designed to classify human actions by aggregating predictions for snippets sampled from short clips ( $\approx 10$  s). Epileptic seizures, however, can last from seconds to tens of minutes [Jenssen 06]. A common aggregation method is to average predictions from randomly sampled snippets [Carreira 17, Ghadiyaram 19, Simonyan 14]. Averaging predictions typically works because most video datasets considered are trimmed, i.e.,

the same action occurs along most of the video duration. In our dataset, due to the nature of TCSs, more than half the frames are labeled as non-generalizing in 49/79 (62%) of the TCS videos. Therefore, simply averaging snippet-level predictions would result in a large number of seizures being misclassified as FOSs. Temporal segment networks (TSNs) [Wang 19b] split videos of any duration into  $n$  non-overlapping segments and a consensus function aggregates features extracted from each segment. Therefore, we propose the use of TSNs to capture semiological features across the entirety of the seizure. We use an RNN as a consensus function to model the sequence of feature vectors extracted from the segments.

We present a novel neural network architecture combining TSNs and RNNs, which we denote Generalized Epileptic Seizure classification from video-Telemetry Using REcurrent convolutional neural networkS (GESTURES), that provides full representations of arbitrarily long seizure videos. These representations could be used for tasks such as classification of seizure types, seizure description using natural language, or triage. To model the relevant patient motion during seizure without the need for object detection, we use a STCNN trained on large-scale HAR datasets (over 65 million videos from Instagram and 250,000 from YouTube) [Ghadiyaram 19] to extract features from short snippets. Then, an RNN is used to learn a representation for the full duration of the seizure.

We chose as a proof of concept to distinguish between FOSs and TCSs, because the key distinction, if the discharge spreads across hemispheres, is only observed later in the seizure. This task demonstrates that we can train a model to take into account features across the entirety of the seizure. The main challenge, apart from the typical challenges in video-telemetry data described above, is distinguishing between TCSs and hyperkinetic FOSs, which are characterized by intense motor activity involving the extremities and trunk.

## 2.2 Materials and methods

### 2.2.1 Video acquisition

Patients were recorded using two full high-definition ( $1920 \times 1080$  pixels, 30 frames per second (FPS)) cameras installed in the EMU as part of standard clinical practice. Infrared is used for acquisition in scenes with low light intensity, such as during nighttime. The acquisition software (Micromed, Treviso, Italy) automatically resizes one of the video streams ( $800 \times 450$ ), superimposes it onto the top-left corner of the other stream and stores the montage using MPEG-2. See the supplementary materials for six examples of videos in our dataset.

### 2.2.2 Dataset description and ground-truth definitions

A neurophysiologist annotated for each seizure the following times: clinical seizure onset  $t_0$ , onset of the clonic phase  $t_G$  (TCSs only) and clinical seizure offset  $t_1$ . The annotations were confirmed using electroencephalography (EEG).

We curated a dataset comprising 141 FOSs and 77 TCSs videos from 68 epileptic patients undergoing presurgical evaluation at the National Hospital for Neurology and Neurosurgery, London, United Kingdom. As patients with only generalized onset seizures are typically not considered for surgery [Duncan 16], our dataset does not contain any seizure of this type. To reduce the seizure class imbalance, we discarded seizures where  $t_1 - t_0 < 15$  s, as this threshold is well under the shortest reported time for TCSs [Jenssen 06]. After discarding short videos, which accounted for 1% of the duration of the initial dataset, there were 106 FOSs. The ‘median (min, max)’ number of seizures per patient is 2 (1, 16). The duration of FOS and TCS is 53 (16, 701) s and 93 (51, 1098) s, respectively. The total duration of the dataset is 298 minutes, 20% of which correspond to TCS phase (i.e., the time interval  $[t_G, t_1]$ ). Two patients had only FOS, 32 patients had only TCS, and 34 had seizures of both types. The ‘mean (standard deviation)’ of the percentage of the seizure duration before the appearance of generalizing semiology, i.e.,  $r = (t_G - t_0)/(t_1 - t_0)$ , is 0.56

(0.18), indicating that patients typically present generalizing semiological features in the second half of the seizure.

Let a seizure video be a sequence of  $K$  frames starting at  $t_0$ . Let the time of frame  $k \in \{0, \dots, K-1\}$  be  $t_k = t_0 + \frac{k}{f}$ , where  $f$  is the video frame rate. We use 0 and 1 to represent FOS and TCS labels, respectively. The ground-truth label  $y_k \in \{0, 1\}$  for frame  $k$  is defined as  $y_k := 0$  if  $t_k < t_G$  and 1 otherwise, where  $t_G \rightarrow \infty$  for FOSs.

Let  $\mathbf{x} \in \mathbb{R}^{3 \times l \times h \times w}$  be a stack of frames or *snippet*, where 3 denotes the RGB channels,  $l$  is the number of frames, and  $h$  and  $w$  are the number of rows and columns in a frame, respectively. The label for a snippet starting at frame  $k$  is

$$Y_k := \begin{cases} 0 & \text{if } \frac{t_k + t_{k+l}}{2} < t_G \\ 1 & \text{otherwise} \end{cases} \quad (2.1)$$

### 2.2.3 Snippet-level classification

The probability  $\hat{Y}_k$  that a patient presents generalizing features within snippet  $\mathbf{x}_k$  starting at frame  $k$  is computed as

$$\hat{Y}_k = \Pr(Y_k = 1 \mid \mathbf{x}_k) = \mathcal{F}_{\theta_{z,x}}(\mathcal{C}_{\theta_x}(\mathbf{x}_k)) = \mathcal{F}_{\theta_{z,x}}(\mathbf{z}_k) \quad (2.2)$$

where  $\mathcal{C}_{\theta_x}$  is an STCNN parameterized by  $\theta_x$  that extracts features  $\mathbf{z}_k \in \mathbb{R}^m$  is a vector of  $m$  features representing  $\mathbf{x}_k$  in a latent space, and  $\mathcal{F}_{\theta_{z,x}}$  is a fully-connected layer parameterized by  $\theta_{z,x}$  followed by a sigmoid function that maps logits to probabilities. In this work, we do not update  $\theta_x$  during training.

### 2.2.4 Seizure-level classification

#### 2.2.4.1 Temporal segment network

Let  $V = \{\mathbf{x}_k\}_{k=1}^{K-l}$  be the set of all possible snippets sampled from a seizure video. We define a sampling function  $f : (V, n, \gamma) \mapsto S$  that extracts a sequence  $S$  of  $n$  snippets

by splitting  $V$  into  $n$  non-overlapping segments and randomly sampling one snippet per segment. There are two design choices: the number of segments  $n$  and the probability distribution used for sampling within a segment. If a uniform distribution is used, information from two adjacent segments might be redundant. Using the middle snippet of a segment minimizes redundancy, but reduces the proportion of data leveraged during training. We propose using a symmetric beta distribution ( $\text{Beta}(\gamma, \gamma)$ ) to model the sampling function, where  $\gamma$  controls the dispersion of the probability distribution (Fig. 2.2). The set of latent snippet representations is  $Z = \{\mathcal{C}_{\theta_x}(\mathbf{x}_i)\}_{i=1}^n$ .

#### 2.2.4.2 Recurrent neural network

To perform a seizure-level prediction  $\hat{\mathbf{Y}}$ ,  $Z$  is aggregated as follows:

$$\hat{\mathbf{Y}} = \Pr(\mathbf{Y} = 1 \mid S) = \mathcal{F}_{\theta_{z,s}}(\mathcal{R}_{\theta_s}(Z)) = \mathcal{F}_{\theta_{z,s}}(\mathbf{z}) \quad (2.3)$$

where  $\mathcal{R}_{\theta_s}$  is an RNN parameterized by  $\theta_s$ ,  $\mathcal{F}_{\theta_{z,s}}$  is a fully-connected layer parameterized by  $\theta_{z,s}$  which uses a softmax function to output probabilities, and  $\mathbf{z}$  is a feature-vector representation of the entire seizure video, corresponding to the last hidden state of  $\mathcal{R}_{\theta_s}$ .

## 2.3 Experiments and results

All videos were preprocessed by separating the two streams into different files (replacing the small embedded view with black pixels), resampling to 15 FPS and  $320 \times 180$  pixels, and reencoding using High Efficiency Video Coding (HEVC). To avoid geometric distortions while maximizing the FOV and resolution, videos were cropped horizontally by removing 5% of the columns from each side, and padded vertically so frames were square. Snippets were resized as imposed by the corresponding architecture to  $224 \times 224$  for single-frame CNNs (SFCNNs) and

$112 \times 112$  for STCNNs. For realism, six video streams in which the patient was completely outside of the FOV were discarded for training but used for evaluation. Recorded audio signals were discarded.

Experiments were implemented in PyTorch 1.7.0. We used a stratified 10-fold cross-validation, generated to ensure the total duration of the videos and ratio of FOSs to TCSs were similar across folds. Both views from the same video were assigned to the same fold, but videos from the same patient were not. This is because individual patients can present with both FOSs or TCSs, so data leakage at the patient level is not a concern. We minimized the weighted binary cross-entropy loss to overcome dataset imbalance, using the AdamW optimizer [Loshchilov 19]. The code is available at <https://github.com/fepellar/gestures-miccai-2021>.

For each fold, evaluation is performed using the model from the epoch with the lowest validation loss. At inference time, the network predicts probabilities for both video streams of a seizure, and these predictions are averaged. The final binary prediction is the consensus probability thresholded at 0.5. We analyzed differences in model performance using a one-tailed Mann-Whitney  $U$  test (as metrics were not normally distributed) with a significance threshold of  $\alpha = 0.05$ , and Bonferroni correction for each set of  $e$  experiments:  $\alpha_{\text{Bonf}} = \frac{\alpha}{e(e-1)}$ .

### 2.3.1 Evaluation of feature extractors for snippets

Despite recent advances in STCNNs for HAR, these architectures do not always outperform SFCNNs pre-trained on large generic datasets [Hutchinson 20]. We assessed the ability of different feature extractors to model semiologies by training a classifier for snippet-level classification (Section 2.2.3).

We used two pre-trained versions of the STCNN R(2+1)D-34 [Ghadiyaram 19] that take as inputs 8 frames ( $\approx 0.5$  s) or 32 frames ( $\approx 2.1$  s). Models were trained using weakly supervised learning on over 65 million Instagram videos and fully supervised learning on over 250,000 YouTube videos of human actions. We selected

Table 2.1: Performance of the feature extractors. The number of parameters is shown in millions. AUC is the area under the precision-recall curve. Accuracy is computed for TCSs and FOSs, while  $F_1$ -score and AUC only for TCSs, represented by an asterisk (\*). Metrics are expressed as ‘median (interquartile range)’.

Model (frames)	Parameters	Features	Accuracy	$F_1$ -score*	AUC*
Wide R2D-50-2 (1)	66.8 M	2048	80.3 (33.2)	67.4 (30.2)	75.7 (38.4)
R2D-34 (1)	21.2 M	512	89.7 (27.7)	73.9 (23.6)	84.3 (28.7)
R(2+1)D-34 (8)	63.5 M	512	93.9 (18.3)	81.6 (16.9)	93.7 (13.4)
R(2+1)D-34 (32)	63.5 M	512	96.9 (12.9)	84.7 (13.4)	94.7 (11.9)

two pre-trained SFCNNs with 34 (R2D-34) and 50 (Wide R2D-50-2) layers, trained on ImageNet [Zagoruyko 17]. The SFCNNs were chosen so the numbers of layers (34) and parameters ( $\approx$  65 million) were similar to the STCNNs.

To ensure that all features datasets have the same number of training instances, we divided each video into segments of 32 frames. Then, we use the models to extract features from snippets of the required length (8, 32, or 1) such that all snippets are centered in the segments. The datasets of extracted feature vectors are publicly available [Pérez-García 21b]. We trained a fully-connected layer for 400 epochs on each feature set, treating views from the same video independently. We used an initial learning rate  $10^{-3}$  and mini-batches of 1024 feature vectors. We minimized a weighted binary cross-entropy loss, where the weight for TCSs was computed as the ratio of FOS frames to TCS frames.

For evaluation, a sliding window was used to infer probabilities for all snippets. STCNNs performance was significantly better than SFCNNs ( $p < 10^{-7}$ ) (Table 2.1). The difference between STCNNs was not significant ( $p = 0.012$ ).

### 2.3.2 Aggregation for seizure-level classification

In this experiment, we compared the performance of three aggregation methods to perform seizure-level classification, using 1) the mean, 2) an RNN with 64 long short-term memory (LSTM) units and 3) an RNN with 64 bidirectional LSTM (BLSTM) units to aggregate the  $n$  feature vectors sampled from the video segments. We used

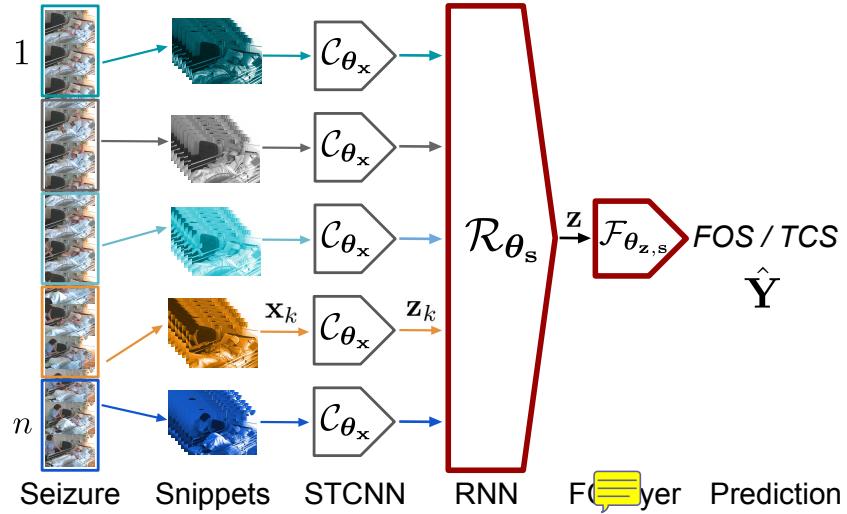


Figure 2.1: Our architecture: Generalized Epileptic Seizure classification from video-Telemetry Using REcurrent convolutional neural networkS (GESTURES). We train only the models with thick red borders.

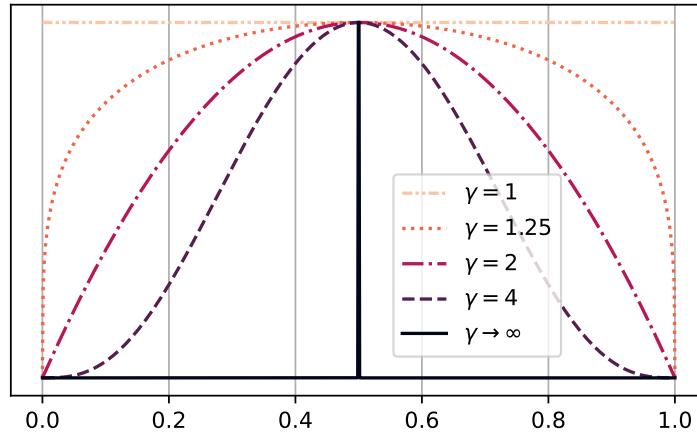


Figure 2.2: Probability distributions used to sample snippets from video segments.

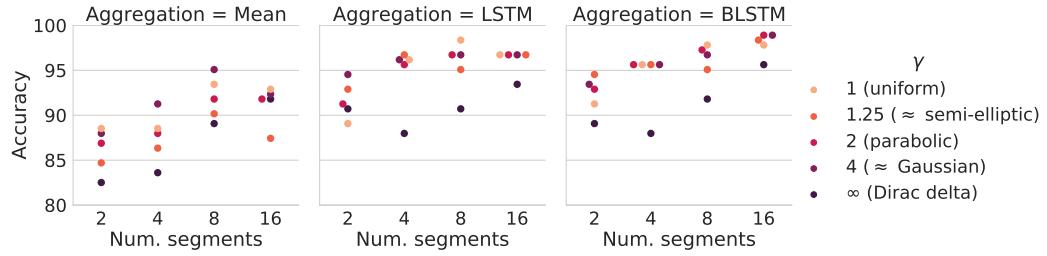


Figure 2.3: Quantitative results for seizure-level classification. Marker brightness is proportional to the dispersion associated with the probability distribution used to sample snippets from the video segments (see Fig. 2.2).

the dataset of feature vectors generated by R(2+1)D-34 (8) (Section 2.3.1). For the task of classifying FOS and TCS, the number of segments **should be** selected to ensure snippets after  $t_G$ , when generalizing semiologies begin, are sampled. The theoretical minimum number of segments needed to sample snippets after  $t_G$  is  $n_{\min} = \lceil 1/(1 - r_{\max}) \rceil$ , where  $r_{\max}$  is the largest possible ratio of non-generalizing to generalizing seizure durations (Section 2.2.2). We can estimate  $r_{\max}$  from our dataset:  $r_{\max} = \max(r_1, \dots, r_{n_{\text{TCS}}}) = 0.93$ , where  $n_{\text{TCS}}$  is the number of TCSs, which yields  $n_{\min} = 15$  segments. We evaluated model performance using  $n \in \{2, 4, 8, 16\}$  segments per video and a sampling distribution using  $\gamma \in \{1, 1.25, 2, 4, \infty\}$ , corresponding to uniform, near semi-elliptic, parabolic, near Gaussian and Dirac's delta distributions, respectively. For evaluation, we used  $\gamma \rightarrow \infty$ , i.e., only the central snippet of each segment. We trained using mini-batches with sequences sampled from 64 videos, and an initial learning rate of  $10^{-2}$ . We used a weighted binary cross-entropy loss for training, where the weight for TCSs was the ratio of FOSs to TCSs.

The highest accuracies were obtained using  $n = 16$  segments,  $\gamma \in \{2, 4\}$  and the BLSTM aggregator (Fig. 2.3). The model with the highest accuracy (98.9%) and  $F_1$ -score (98.7%) yielded 77 true positives, 104 true negatives, 2 false positives and 0 false negatives, where TCS is the positive class (Section 2.2.2). See the supplementary materials for examples of videos classified correctly and incorrectly, with different levels of confidence.

## 2.4 Discussion

Objective assessment of seizure semiology from videos is important to determine appropriate treatment for the diagnosed epilepsy type and help reduce SUDEP risk. Related works focus on EZ localization by averaging classifications of short snippets from multiple seizures, ignoring order of semiologies, and are not robust to variations seen in real world datasets such as changes in the FOV. Moreover, their performance is limited by the size of the training datasets, which are small due to the expense of curating datasets. Methods that take into account the sequential nature of semiologies and represent the entirety of seizures are needed.

We presented GESTURES, a method combining TSNs and RNNs to model long-range sequences of seizure semiologies. GESTURES can classify seizures into FOSs and TCSs with high accuracy. To overcome the challenge of training on limited data, we used a network pre-trained on large HAR datasets to extract relevant features from seizure videos, highlighting the importance of transfer learning in medical applications. GESTURES can take videos from multiple cameras, which makes it robust to patients being out of the FOV.

In Section 2.3.1 we compared STCNNs to SFCNNs for snippet-level classification. To make comparisons fair, we selected models with a similar number of layers (R2D-34) or parameters (Wide R2D-50-2). We found the larger SFCNN had worse performance, due to overfitting to the training dataset. Classification accuracy was proportional to snippet duration (Table 2.1), meaning that both STCNNs outperformed SFCNNs. We selected R(2+1)D-34 (8) for the aggregation experiment (Section 2.3.2), as performance between the two STCNNs was similar and this model is less computationally expensive.

Using LSTM or BLSTM units to aggregate features from snippets improved accuracy compared to averaging (Fig. 2.3), confirming that modeling the order of semiologies is important for accurate seizure representation. Model performance was proportional to the number of temporal segments, with more segments providing a

denser sampling of seizure semiologies. Ensuring some dispersion in the probability distributions used to sample snippets improved classification. One of the two false positives was caused by the patient being out of the FOV in one of the video streams. We did not observe overfitting to unrelated events in the videos, such as nurses in the room, to predict TCS, and models correctly discriminated between TCSs and hyperkinetic FOSSs.

We demonstrated that methods designed for HAR can be adapted to learn deep representations of epileptic seizures. This enables a fast, automated and quantitative assessment of seizures. GESTURES takes arbitrarily long videos and is robust to occlusions, changes in FOV and multiple people in the room. In the future, we will investigate the potential of GESTURES to classify different types of TCSs and to localize the EZ, using datasets from multiple EMUs.

DRAFT – January 19, 2022

## CHAPTER 3

# Visualization of the epileptogenic zone on brain images

---

## Table of contents

---

3.1	Introduction	48
3.2	System architecture	49
3.2.1	Implementation	52
3.2.2	Data loading	52
3.2.3	Input semiologies	53
3.2.4	Querying the database	54
3.2.5	Visualization	55
3.3	Example of clinical usage	56
3.4	Discussion	58

---

**Foreword** This chapter describes the software developed in the context of a collaboration with neurologists Ali Alim-Marvasti and Gloria Romagnoli.

My contributions to this project are:

- A 3D Slicer module [Fedorov 12] that reads the output of the querying tool and generates a 3D visualization on a parcellated brain magnetic resonance image (MRI), where the brightness associated to each brain structure is proportional to the probability of the EZ being in the structure.
- The software engineering aspects of the project: a Pip Installs Packages (PIP)-installable Python package for the querying tool, including continuous integration (CI), and an application programming interface (API) to access the Python package from 3D Slicer or from the command line.
- The implementation of the online demo, which does not require installing 3D Slicer<sup>a</sup>.

---

<sup>a</sup><https://github.com/fepegar/SVT-web>

## 48 Chapter 3. Visualization of the epileptogenic zone on brain images

Ali and Gloria performed the systematic literature review to generate the database (which is an Excel spreadsheet) that maps seizure semiologies to brain regions. Ali wrote most of the code to query the database, and contributed to the Slicer module. The code and the database are available on GitHub<sup>a</sup>.

The relevant publications are:

- **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation*. In International League Against Epilepsy (ILAE) British Branch Virtual Annual Scientific Conference, 2020
- **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation*. In American Epilepsy Society (AES) Annual Meeting, 2020
- Ali Alim-Marvasti, Gloria Romagnoli, Karan Dahele, Hadi Modarres, **Fernando Pérez-García**, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson, Fahmida Chowdhury, Beate Diehl and John S. Duncan. *Probabilistic Landscape of Seizure Semiology Localising Values*. Under submission (Brain Communications), 2021
- Ali Alim-Marvasti, **Fernando Pérez-García**, Gloria Romagnoli, Peter N. Taylor, Vejay Vakharia, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson and John S. Duncan. *Mapping Epileptic Symptomatology to Cortical Epileptogenicity in MNI Space: Seizure Semiology-to-Brain Visualisation Tool*. Under submission (NeuroImage: Clinical), 2021

---

<sup>a</sup><https://github.com/fepegar/Semiology-Visualisation-Tool>

### 3.1 Introduction

Anti-epileptic drugs (AEDs) are normally used to treat epilepsy. One third of epilepsies are drug-resistant [Engel 16]. Curative resective surgery can be performed to remove the epileptogenic zone (EZ) (Chapter 4). The location of the EZ, “the area of cortex indispensable for the generation of clinical seizures” [Rosenow 01], is normally inferred by a multidisciplinary team following non-invasive evaluation such as video-electroencephalography (EEG), magnetic resonance image (MRI) or neuropsychological tests. If the information regarding the location of the EZ is discordant between the different tests, intracranial EEG (iEEG) electrodes may be

implanted to localize it precisely. To determine the brain regions ~~that~~ need to be recorded, i.e., the targets for the iEEG electrodes, the team leverages information from the non-invasive examinations, including seizure semiology from the recorded videos (Chapter 2). The choice of targets is therefore influenced by the team's subjective experience and personal knowledge of the literature. This leads to substantial variations of implantation strategies across different epilepsy centers [Tufenkjian 12]. The diagnostic pathway for surgical planning could be supported by an objective tool to aid clinicians in deducing the EZ location from seizure semiology.

Res~~ew~~lers who do not wish to code would benefit from a graphical user interface (GUI) to query the database. Moreover, visualizing the probability of each structure containing the EZ on 3D multimodality imaging (3DMMI) could help plan the resection and iEEG implantation strategies [Nowell 17, Nowell 15], potentially using automatic trajectory planning (ATP) [Sparks 17]. Finally, a 3DMMI visualization could be used to perform qualitative and quantitative analyses of the retrospective information contained in the *Semio2Brain*.

In this work, we present a software tool that, given an observed list of seizure semiologies and other patient data such as the dominant hemisphere, generates a table with the number of datapoints associated to each brain region. Each datapoint represents a patient presenting the observed semiologies who became seizure-free after resection of the corresponding brain structure (Table 3.1). The usage examples in this thesis use the *Semio2Brain* database and the corresponding software used to query the database [Alim-Marvasti 21a, Alim-Marvasti 21c], which were defined based on the Neuromorphometrics atlas parcellation<sup>1</sup>.

## 3.2 System architecture

In this section, we describe the elements that compose our Semiology Visualization Tool (SVT) and ~~their interconnections~~ (Fig. 3.1). The visualization module depends

<sup>1</sup><http://www.neuromorphometrics.com>

## **50 Chapter 3. Visualization of the epileptogenic zone on brain images**

Table 3.1: Result of querying an imaginary database with the semiology *Head version* and exemplar brain structures A, B and C, assuming that the brain has been parcellated into only three structures. In this example, according to the literature analyzed to build the database, structure C was one of the resected structures in 20 patients presenting *Head version* who became seizure-free after surgery, suggesting a high probability of the EZ being associated with that structure. Structure B was resected for five patients who became seizure-free after surgery. Structure A was never resected in patients who presented *Head version* and became seizure-free after surgery. This result would support the decision of implanting electrodes in structure C (and possibly B), as they are likely to be associated with the EZ according to the retrospective information in the literature. The actual list of brain structures would depend on the method used to parcellate the brain.

	<b>Structure A</b>	<b>Structure B</b>	<b>Structure C</b>
<b>Head version</b>	0	5	20

on an external application programming interface (API) that must be able to provide the following:

- At start-up, a list of pre-defined semiologies that are used to generate the GUI and a list of additional settings to perform the query
- ~~When queried~~ with a list of semiologies and additional settings, a table with datapoints in the literature associated to each brain structure

At start-up, the user is asked to load a patient's MRI that will be used as reference image and a corresponding brain parcellation that must be compatible with the **querying module**. If no patient data are available, data from an Montreal Neurological Institute (MNI) template is used. Then, a list of pre-defined semiology terms and additional settings compatible with the API are displayed, and the user fills in the observed semiologies. Finally, the query is submitted using the API, which returns the table of datapoints that is used to generate the 3D EZ probability map. These steps and the implementation details are described in Sections 3.2.2 to 3.2.5 and Section 3.2.1, respectively.

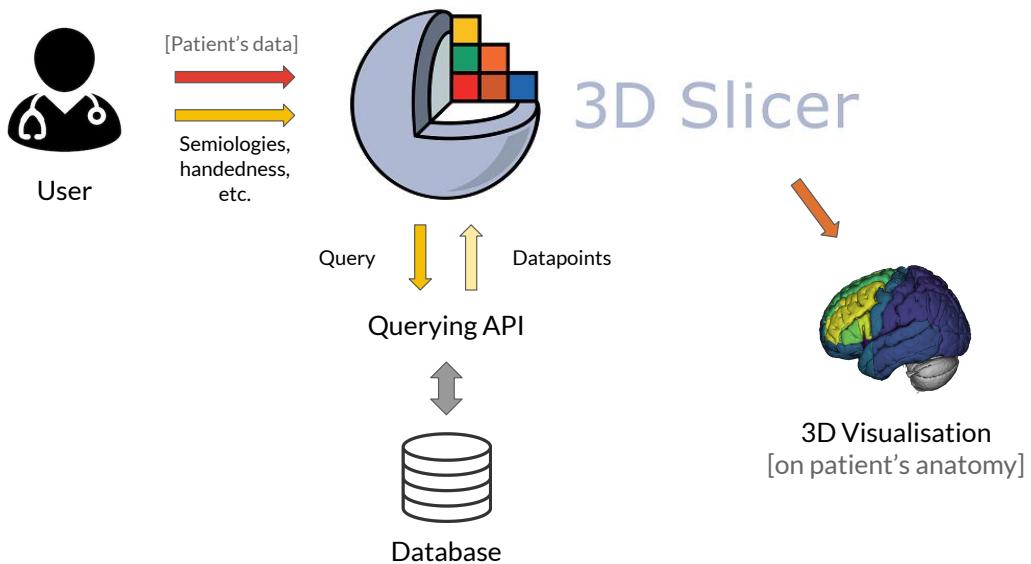


Figure 3.1: General architecture of the SVT. The user inputs a list of observed seizure semiologies and (optionally) an MRI of the patient for visualization in the patient's space. The GUI is used to input the observed semiologies to the software, without the need to code. The Slicer module generates a structured query that can be processed by the database querying module. The database is queried and the retrieved datapoints are converted into a 3DMMI visualization showing the EZ probability map.

### **3.2.1 Implementation**

Our SVT is built on top of 3D Slicer [Fedorov 12]. 3D Slicer, or simply ‘Slicer’, is “a free, open source and multi-platform software package widely used for medical, biomedical, and related imaging research”<sup>2</sup>.

We developed a Slicer Python module that leverages the Insight Toolkit (ITK) for image processing [McCormick 14], the Visualization Toolkit (VTK) for visualization [Schroeder 06] and Qt for the GUI<sup>3</sup>. The Python package used to query the *Semio2Brain* database, called `mega_analysis`, can be installed with Pip Installs Packages (PIP). The code, written in Python 3, is available on GitHub<sup>4</sup>. We have also developed an lite online version hosted on Binder [Bussonnier 18], which does not require installation of 3D Slicer and can be accessed through a web browser<sup>5</sup>. Our SVT can be used on all major platforms: Windows, Linux and macOS.

### **3.2.2 Data loading**

At start-up, the ~~querying Python module~~ is installed in the Slicer environment in the background, if it was not installed previously.

Then, the user is asked to load the patient’s MRI and a corresponding brain parcellation. If the patient’s data is not available or the SVT is being used to explore the database and not for surgical planning, a generic MNI template is loaded instead [Fonov 09]. As the EZ is expected to be in the gray matter, the parcellation is automatically stripped from the white matter and other irrelevant structures. The expected brain parcellation must be compatible with the Neuromorphometrics atlas as it was used to build the *Semio2Brain* database. However, the presented method is agnostic to the atlas choice. All figures in this chapter show parcellations based on the Neuromorphometrics atlas, generated using geodesical information flows (GIF) [Cardoso 15]. Finally, a Slicer segmentation node is instantiated from

---

<sup>2</sup><https://www.slicer.org/>

<sup>3</sup><https://www.qt.io>

<sup>4</sup><https://github.com/fepegar/Semiology-Visualisation-Tool>

<sup>5</sup><https://mybinder.org/v2/gh/fepegar/SVT-web/HEAD?filepath=SVT-web.ipynb>

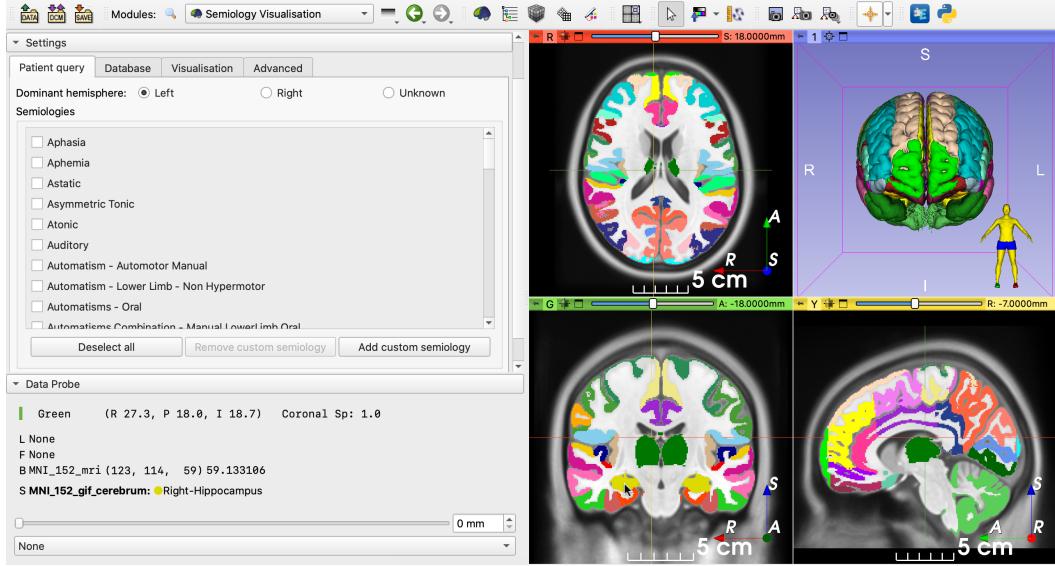


Figure 3.2: Semiology visualization module after loading an MNI template as image reference (as opposed to a patient's data) and its corresponding brain parcellation. Left: main panel containing a list of suggested semiology terms to be selected by the user; right: neuroimages in the patient's space. Images are shown using the radiological convention, i.e., the left hemisphere is shown on the right and vice versa.

the parcellation label map, 3D meshes are generated for visualization using the marching cubes algorithm [Lorensen 87, Pinter 19] and the user is presented with the preprocessed data (Fig. 3.2).

### 3.2.3 Input semiologies

Once the data have been loaded, a pre-defined list of common semiologies is shown on the left-hand side of the screen (Fig. 3.3). Certain semiologies such as *Head version* require a laterality (left or right). Semiologies such as *Hypermotor* may not have an associated laterality, meaning it was observed for both sides. Some lateralities such as *Ictal speech* never have an associated laterality. The dominant hemisphere may also be specified, if known. The used API allows for custom semiologies to also be entered, which may be matched to one of the pre-defined semiologies using regular expressions [Alim-Marvasti 21c]. Matching is performed in real time every time the characters in the widget are modified, as long as at least three characters

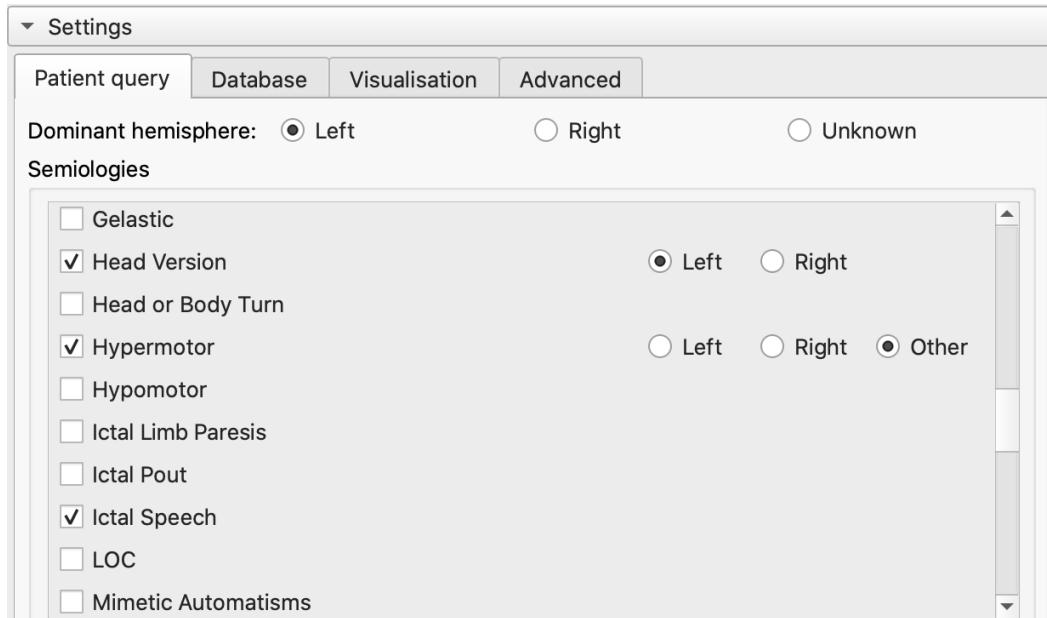


Figure 3.3: List of pre-defined semiologies in the GUI. Users can select a semiology in the list or add a custom one, which might be matched to a pre-defined semiology using regular expressions to match concepts associated to that semiology. For example, ‘butterflies’ and ‘déjà vu’ would be matched with epigastric and psychic auras, respectively.

have been typed (Fig. 3.4).

### 3.2.4 Querying the database

The Slicer module reads the patient’s semiologies and settings from the GUI and generates a machine-readable query for the querying module. As the waiting time to query the database is sometimes in the order of tens of seconds, previous queries result are cached in the disk for faster retrieval. The result from querying the database is a table containing the number of datapoints for each brain structure (Table 3.1). If multiple semiologies are selected, results are combined and a number between 0 and 1 is computed for each brain structure by the querying module. Details on how the datapoints for multiple semiologies are combined can be found in [Alim-Marvasti 21c].

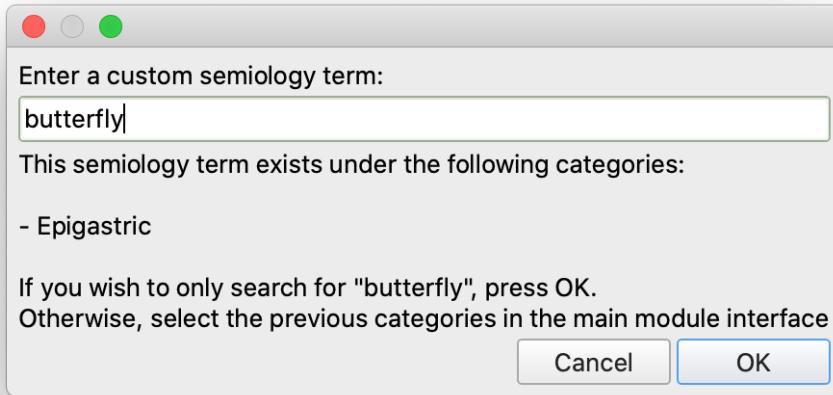


Figure 3.4: Adding a custom semiology term to the semiology visualization module. In this case, the term *butterfly* was entered as the patient described “a butterfly feeling in my stomach”. The querying module matches the input term with the  *gastric* semiology term, and therefore the Slicer module suggests matching *Epigastric* to the list of selected semiologies.

### 3.2.5 Visualization

The datapoints table is displayed on the GUI, ~~sorting the structures~~ by number of datapoints, showing first the structures with the highest number of datapoints in the database (Fig. 3.5).

The probability map is generated and displayed on the 2D slice views and the 3D view, ~~and~~ the 2D slice views are centered on the brain structure with the highest number of associated datapoints. Brain structures without datapoints are hidden from the 2D slice views and shown in gray on the 3D view. To emphasize the importance of brain structures with a high number of datapoints, the opacity of each structure on the 2D slice views is ~~linearly proportional~~ to the number of associated datapoints. We chose an open-source perceptually uniform sequential colormap, *viridis*, as default for this application. This implies that the brightness of each structure is also linearly proportional to the number of associated datapoints.

## 56 Chapter 3. Visualization of the epileptogenic zone on brain images

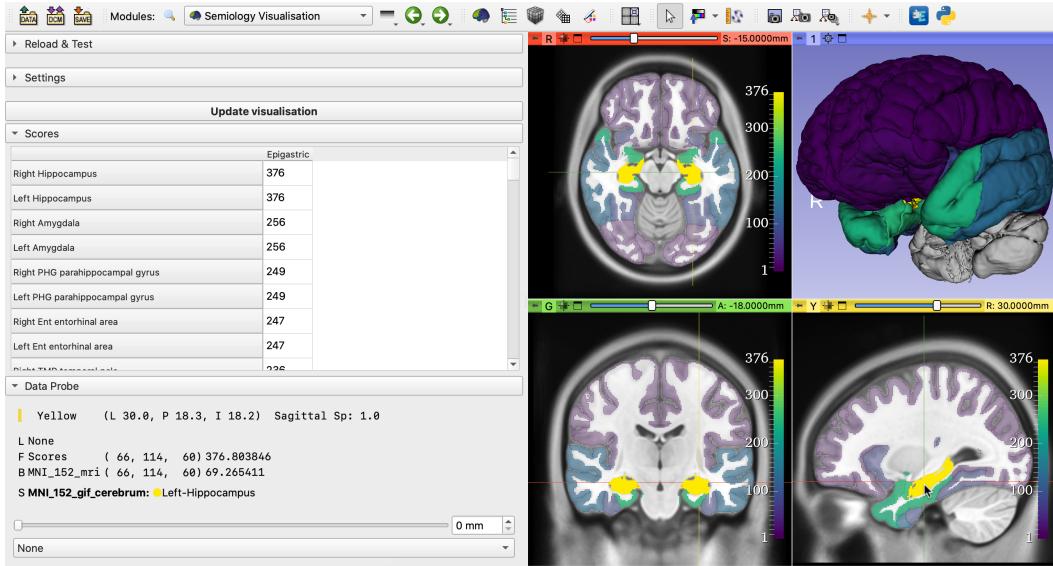


Figure 3.5: Results of querying the database with the *Epigastric* semiology. Left: table showing the number of datapoints associated to each brain structure; right: EZ probability map, where brightness (and opacity, on the 2D views) is linearly proportional to the number of datapoints.

However, other colormaps are available if desired. The colorbars on the 2D slice views help mapping colors to number of datapoints.

Advanced visualization settings may be selected, such as showing only one hemisphere on the 3D view, setting the minimum opacity on the 2D views or enabling the color blind mode, in which the color-blind-friendly *cividis* colormap is used [Nuñez 18] (Fig. 3.6).

### 3.3 Example of clinical usage

In this section, we demonstrate a usage example of our SVT with a retrospective case of a patient who underwent epilepsy resective surgery at the National Hospital for Neurology and Neurosurgery (NHNN) (Queen Square, London, UK). We use the `mega_analysis` querying module, which uses the *Semio2Brain* database [Alim-Marvasti 21c, Alim-Marvasti 21a].

The patient was right-handed and presented head version to the right at the

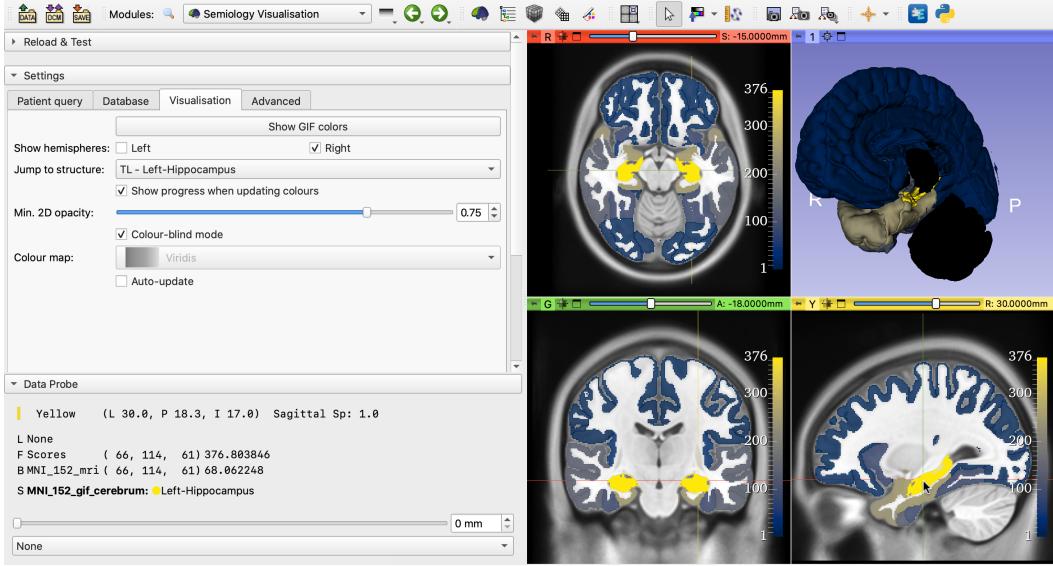


Figure 3.6: Demonstration of advanced visualization settings. The selected settings are: show only the right hemisphere, center the views on the right thalamus, set minimum opacity to 75% (default is 25%) and enable color-blind mode.

beginning of seizures. Ictal EEG was non lateralizing and interictal EEG showed bitemporal sharp waves. We used the patient's preoperative  $T_1$ -weighted ( $T_{1w}$ ) MRI as reference for the visualization. The Neuromorphometrics brain parcellation was generated using GIF [Cardoso 15].

We queried the database using the semiology *Head version (right)* and setting the dominant hemisphere to *Left*. The most highlighted structures concentrate in the left frontal lobe (Fig. 3.7).

As this is a retrospective case from a patient who underwent resective surgery some years ago, the postoperative MRI and the outcome are available. We registered the preoperative and postoperative MRIs for visualization purposes [Ourselin 00]. As non-rigid brain deformations may happen after the surgery and the registration was rigid, the alignment is only approximated, albeit accurate enough for a visual analysis. When visualizing the aligned images, including the probability map, we observed an overlap between the highlighted areas (i.e., the brain structures with the highest number of datapoints) and the resection cavity (Fig. 3.8). The surgery

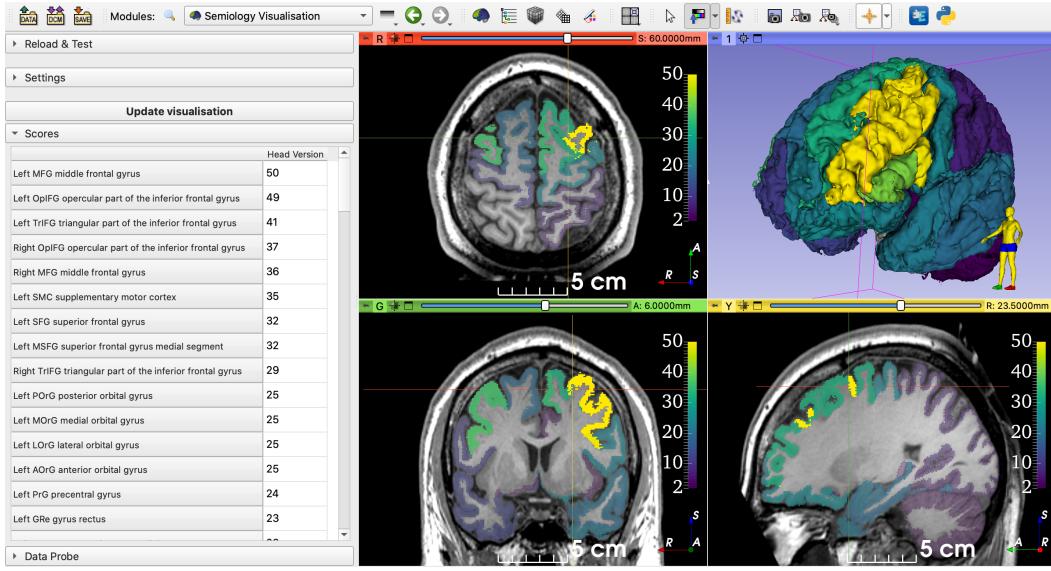


Figure 3.7: Querying the database using data from a retrospective real case. The selected semiology term was *Head version* (right), and the dominant hemisphere was *Left*. The regions with highest numbers of datapoints concentrate around the left frontal lobe (represented on the right side of the axial and coronal views).

resulted in complete seizure freedom.

### 3.4 Discussion

The choice of targets for iEEG electrodes implantation would benefit from an objective, data-driven method. In this work, we present an open-source software tool to visualize regions of the brain with a high probability of being associated with the EZ, given a set of observed seizure semiologies.

Our tool is not designed to replace clinicians in planning the iEEG implantation or the resective surgery, but to support their decisions using a data-driven approach that represents patterns in the literature intuitively.

In the future, we will improve the generalizability of our framework to improve the compatibility with custom databases and other brain parcellation strategies, such as the Desikan-Killiany atlas [Desikan 06]. We will also develop an online version of our SVT that uses a well-established cloud infrastructure service, such

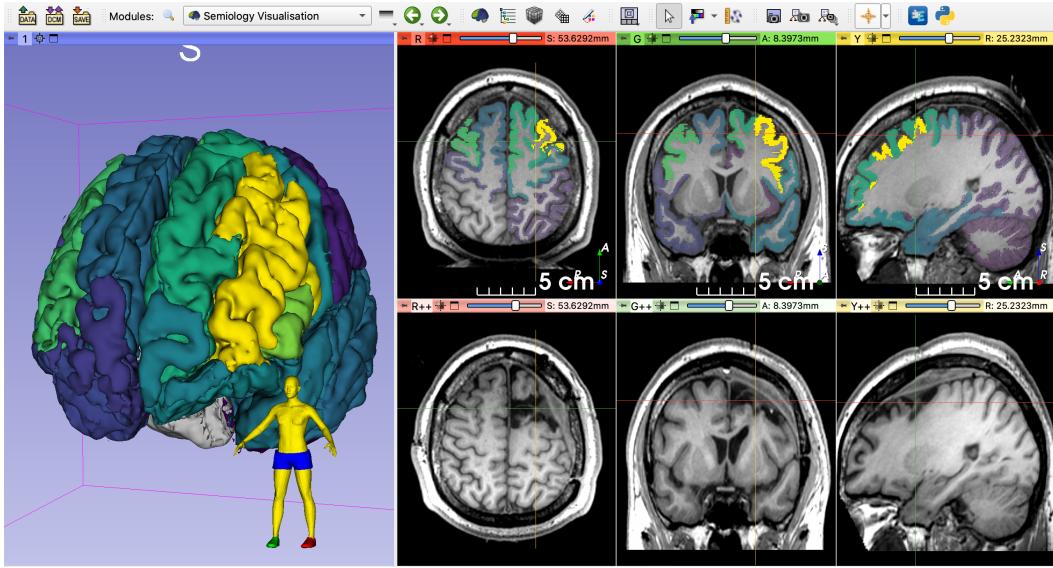


Figure 3.8: Qualitative comparison of the EZ probability map and the postoperative MRI. Note the overlap between the highlighted structures and the resection cavity. The images were rigidly registered, and the crosshairs are centered on approximately the same brain region. The patient became seizure-free after resective surgery.

as Azure or the Google Cloud Platform, which would allow for a fast and seamless user experience. Another potential improvement is adding support to parcellate the brain at loading time using deep learning [Li 17, Pérez-García 19], according to the requirements of the API. This would spare the user the need to wait for hours before the parcellation is generated, which is typically the case for GIF [Cardoso 15] or FreeSurfer<sup>6</sup>.

<sup>6</sup><https://surfer.nmr.mgh.harvard.edu/>

DRAFT – January 19, 2022

## CHAPTER 4

# Segmentation of postoperative brain resection cavities

---

## Table of contents

---

<b>4.1</b>	<b>Introduction</b>	<b>62</b>
4.1.1	Motivation	62
4.1.2	Related works	62
4.1.3	Contributions	64
<b>4.2</b>	<b>Methods</b>	<b>64</b>
4.2.1	Learning strategy	64
4.2.2	Resection simulation for self-supervised learning	69
<b>4.3</b>	<b>Experiments and results</b>	<b>74</b>
4.3.1	Data	74
4.3.2	Network architecture and implementation details	77
4.3.3	Processing during training	78
4.3.4	Experiments	80
<b>4.4</b>	<b>Discussion</b>	<b>86</b>

---

**Foreword** This chapter is adapted and contains content from the work presented in:

- **Fernando Pérez-García**, Roman Rodionov, Ali Alim-Marvasti, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Simulation of Brain Resection for Cavity Segmentation Using Self-supervised and Semi-supervised Learning*. In MICCAI 2020, Lecture Notes in Computer Science, pages 115–125, Cham, 2020. Springer International Publishing
- **Fernando Pérez-García**, Reuben Dorent, Michele Rizzi, Francesco Cardinale, Valerio Frazzini, Vincent Navarro, Caroline Essert, Irène Ollivier, Tom Vercauteren, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *A self-supervised learning strategy for postoperative brain cavity segmentation simulating resections*. International Journal of Computer Assisted Radiology and Surgery, June 2021

## **4.1 Introduction**

### **4.1.1 Motivation**

Approximately one third of epilepsies are drug-resistant. If the epileptogenic zone (EZ), i.e., “the area of cortex indispensable for the generation of clinical seizures” [Rosenow 01], can be localized, resective surgery to remove the EZ may be curative. Currently, only 40% to 70% of patients with refractory focal epilepsy are seizure-free after surgery [Jobst 15]. This is, in part, due to limitations identifying the EZ. Retrospective studies relating presurgical clinical features and resected brain structures to surgical outcome provide useful insight to guide EZ resection [Jobst 15]. To quantify resected structures, first, the resection cavity (RC) must be segmented on the postoperative magnetic resonance image (MRI). A preoperative image with a corresponding brain parcellation can then be registered to the postoperative MRI to identify resected structures.

RC segmentation is also necessary in other applications. For neuro-oncology, the gross tumor volume, which is the sum of the RC and residual tumor volumes, is estimated for postoperative radiotherapy [Ermiş 20].

Despite recent efforts to segment RCs in the context of brain cancer [Meier 17, Ermiş 20], little research has been published in the context of epilepsy surgery. Furthermore, previous work is limited by the lack of benchmark datasets, released code or trained models, and evaluation is restricted to single-institution datasets used for both training and testing.

### **4.1.2 Related works**

After surgery, RCs fill with cerebrospinal fluid (CSF). This causes an inherent uncertainty in delineating RCs adjacent to structures such as sulci, ventricles or edemas. Nonlinear registration has been presented to segment the RC for epilepsy [Chitphakdithai 10] and brain tumor [Chen 15] surgeries by detecting non-corresponding regions between pre- and postoperative images. However, evaluation

of these methods was restricted to a very small number of images. Furthermore, in cases with intensity changes due to the resection (e.g., brain shift, atrophy, fluid filling), non-corresponding voxels may not correspond to the RC.

Decision forests were presented for brain cavity segmentation after glioblastoma surgery, using four MRI modalities [Meier 17]. These methods, which aggregate hand-crafted features extracted from all modalities to train a classifier, can be sensitive to signal inhomogeneity and unable to distinguish regions with intensity patterns similar to CSF from RCs. Recently, a 2D convolutional neural network (CNN) was trained to segment the RC on MRI slices in 30 glioblastoma patients [Ermiş 20]. They obtained a ‘median (interquartile range)’ Dice score coefficient (DSC) of 84 (10) compared to ground-truth labels by averaging predictions across anatomical axes to compute the 3D segmentation. While these approaches require four modalities to segment the RC, some of the modalities are often unavailable in clinical settings [Dorent 21]. Furthermore, code and datasets are not publicly available, hindering a fair comparison across methods. Applying these techniques requires curating a dataset with manually obtained annotations to train the models, which is expensive.

Unsupervised learning methods can leverage large, unlabeled medical image datasets during training. In self-supervised learning, training instances are generated automatically from unlabeled data and used to train a model to perform a pretext task. The model can be fine-tuned on a smaller labeled dataset to perform a downstream task [Chen 19]. The pretext and downstream tasks may be the same. For example, a CNN was trained to reconstruct a skull bone flap by simulating craniectomies on CT scans [Matzkin 20]. Lesions simulated in chest CT of healthy subjects were used to train models for nodule detection, improving accuracy compared to training on a smaller dataset of real lesions [Pezeshk 17].

Semi-supervised learning may be used when a large amount of unlabeled data is available. A model trained on a labeled dataset (which may have been generated in a self-supervised setting) can generate pseudolabels for unlabeled data. Uncertainty

estimation may be used to select pseudolabeled instances with a low uncertainty for medical image segmentation tasks, improving model performance compared to using a random subset [Venturini 20].

#### 4.1.3 Contributions

We present a self-supervised learning approach to train a 3D CNN to segment brain RCs from  $T_1$ -weighted ( $T_1w$ ) MRI without annotated data, by simulating resections during training. We ensure our work is reproducible by releasing the source code for resection simulation and CNN training, the trained CNN, and the evaluation dataset. To the best of our knowledge, we introduce the first open annotated dataset of postoperative MRI for epilepsy surgery.

This work extends our conference paper [**Pérez-García** 20e] as follows: 1) we performed a more comprehensive evaluation, assessing the effect of the resection simulation shape on performance and evaluating datasets from different institutions and pathologies; 2) we formalized our transfer learning strategy; 3) we used uncertainty estimation as a selection criterion for pseudolabeled instances within our semi-supervised learning setting.

## 4.2 Methods

We describe our learning strategy in Section 4.2.1 (Fig. 4.1) and introduce our RC simulation in Section 4.2.2.

### 4.2.1 Learning strategy

#### 4.2.1.1 Problem statement

The overall objective is to automatically segment RCs from postoperative  $T_1w$  MRI using a CNN  $f_\theta$  parameterized by weights  $\theta$ . Let  $\mathbf{X}_{\text{post}} : \Omega \rightarrow \mathbb{R}$  and  $\mathbf{Y}_{\text{cavity}} : \Omega \rightarrow \{0, 1\}$  be a postoperative  $T_1w$  MRI and its cavity segmentation label, respectively, where  $\Omega \subset \mathbb{R}^3$ .  $\mathbf{X}_{\text{post}}$  and  $\mathbf{Y}_{\text{cavity}}$  are drawn from the data distribution  $\mathcal{D}_{\text{postop}}$ .

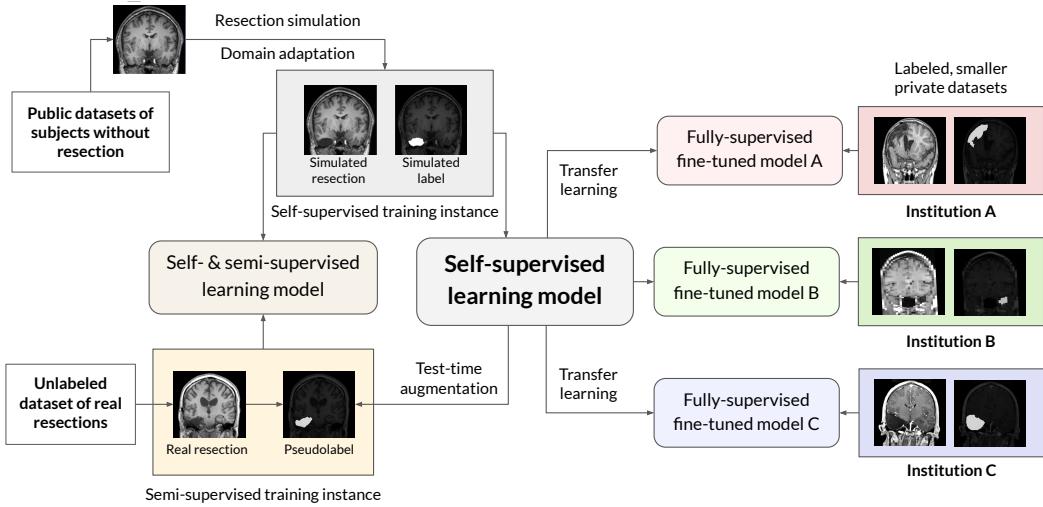


Figure 4.1: Learning strategies. 3D images without resections (top left) are modified by our resection simulation method to mimic postoperative images and their corresponding labels, generating training instances. These instances are used to train a baseline model in a self-supervised manner (middle). The baseline model generates pseudolabels from unlabeled images of patients who underwent resective surgery (bottom left). Instances from the RC simulation and pseudolabeled dataset are used to train a new model in a self- and semi-supervised learning approach (left). The baseline model may be fine-tuned to improve its performance on small labeled datasets containing real resections from a single institution, using a standard fully-supervised learning approach (right).

## 66 Chapter 4. Segmentation of postoperative brain resection cavities

In model training, the aim is to minimize the expected discrepancy between the label  $\mathbf{Y}_{\text{cavity}}$  and network prediction  $f_{\theta}(\mathbf{X}_{\text{postop}})$ . Let  $\mathcal{L}$  be a loss function that estimates this discrepancy (e.g., Dice loss). The optimization problem for the network parameters  $\theta$  is:

$$\theta^* = \arg \min_{\theta} \mathbb{E}_{\mathcal{D}_{\text{postop}}} [\mathcal{L}(f_{\theta}(\mathbf{X}_{\text{postop}}), \mathbf{Y}_{\text{cavity}})] \quad (4.1)$$

In a fully-supervised setting, a labeled dataset  $D_{\text{postop}} = \{(\mathbf{X}_{\text{post}_i}, \mathbf{Y}_{\text{cavity}_i})\}_{i=1}^{n_{\text{postop}}}$  is employed to estimate the expectation defined in Eq. (4.1) as:

$$\mathbb{E}_{\mathcal{D}_{\text{postop}}} [\mathcal{L}(f_{\theta}(\mathbf{X}_{\text{postop}}), \mathbf{Y}_{\text{cavity}})] \approx \frac{1}{n_{\text{postop}}} \sum_{i=1}^{n_{\text{postop}}} \mathcal{L}(f_{\theta}(\mathbf{X}_{\text{post}_i}), \mathbf{Y}_{\text{post}_i}) \quad (4.2)$$

In practice, CNNs typically require an annotated dataset with a large  $n_{\text{postop}}$  to generalize well for unseen instances. However, given the time and expertise required to annotate scans,  $n_{\text{postop}}$  is often small. We present a method to artificially increase  $n_{\text{postop}}$  by simulating postoperative MRIs and associated labels from preoperative scans.

### 4.2.1.2 Simulation for domain adaptation and self-supervised learning

Let  $D_{\text{preop}} = \{\mathbf{X}_{\text{pre}_i}\}_{i=1}^{n_{\text{preop}}}$  be a dataset of preoperative  $T_1w$  MRI, drawn from the data distribution  $\mathcal{D}_{\text{preop}}$ . We propose to generate a simulated postoperative dataset  $D_{\text{sim}} = \{(\mathbf{X}_{\text{sim}_i}, \mathbf{Y}_{\text{sim}_i})\}_{i=1}^{n_{\text{sim}}}$  using the preoperative dataset  $D_{\text{preop}}$ . Specifically, we aim to build a generative model  $\phi_{\text{sim}} : \mathbf{X}_{\text{preop}} \mapsto (\mathbf{X}_{\text{sim}}, \mathbf{Y}_{\text{sim}})$  that transforms preoperative images into simulated, annotated postoperative images that imitate instances drawn from the postoperative data distribution  $\mathcal{D}_{\text{postop}}$ .  $D_{\text{sim}}$  can then be used to estimate the expectation in Eq. (4.1):

$$\mathbb{E}_{\mathcal{D}_{\text{postop}}} [\mathcal{L}(f_{\theta}(\mathbf{X}_{\text{postop}}), \mathbf{Y}_{\text{cavity}})] \approx \frac{1}{n_{\text{sim}}} \sum_{i=1}^{n_{\text{sim}}} \mathcal{L}(f_{\theta}(\mathbf{X}_{\text{sim}_i}), \mathbf{Y}_{\text{sim}_i}) \quad (4.3)$$

Simulated images can be generated from any unlabeled preoperative dataset.

Therefore, the size of the simulated dataset can be much greater than the annotated dataset  $D_{\text{postop}}$ , i.e.,  $n_{\text{sim}} \gg n_{\text{postop}}$ . The network parameters  $\theta$  can be optimized by minimizing Eq. (4.3) using stochastic gradient descent, leading to a trained predictive function  $f_{\theta_{\text{sim}}}$ . Finally,  $f_{\theta_{\text{sim}}}$  can be fine-tuned on  $D_{\text{postop}}$  to improve performance on the postoperative domain  $\mathcal{D}_{\text{postop}}$ .

#### 4.2.1.3 Leveraging unlabeled data from the target domain

Let  $D_{\text{unlabeled}} = \{\mathbf{X}_{\text{postop}_i}\}_{i=1}^{n_{\text{unl}}}$  be a dataset comprising  $n_{\text{unl}}$  unlabeled postoperative images. We propose to leverage  $D_{\text{unlabeled}}$  to build a better predictive model than using simulated resections only, employing a semi-supervised learning approach. First, pseudolabels for each image  $\mathbf{X}_{\text{postop}} \in D_{\text{unlabeled}}$  can be generated with  $f_{\theta_{\text{sim}}}$  using data distillation, i.e., ensembling multiple predictions from transformed versions of  $\mathbf{X}_{\text{postop}}$ . Using the multiple predictions generated for pseudolabels, we estimate image-level segmentation uncertainty so that only instances with high reliability are used for training. Finally, simulated resection instances and the selected pseudolabeled instances are used to train a new model  $f_{\theta_{\text{sim+unl}}}$  in a hybrid self- and semi-supervised setting.

**Generating pseudolabels using data distillation** Data distillation is a method that ensembles predictions from multiple transformations applied to data, using a single model [Radosavovic 17]. We use Monte Carlo simulation to generate each pseudolabel with test-time augmentation (TTA), which can improve the performance of segmentation models [Wang 19a]. Let  $n_u$  represent the total number of simulation runs. In the  $i$ -th simulation run, the TTA intensity and spatial transforms  $T_\alpha$  and  $T_\beta$  (Section 4.3.3.2) are applied to  $\mathbf{X}_{\text{postop}}$ .  $f_{\theta_{\text{sim}}}$  is used to predict  $\widetilde{\mathbf{Y}}'_{\theta_{\alpha\beta}}$ , the probability of each voxel belonging to the cavity in the transformed space. Finally,  $T_\beta^{-1}$  is used to transform  $\widetilde{\mathbf{Y}}'_{\theta_{\alpha\beta}}$  back onto the space of  $\mathbf{X}_{\text{postop}}$ :

$$\widetilde{\mathbf{Y}}'_{\text{cavity}_i} = T_\beta^{-1} \circ f_{\theta_{\text{sim}}} \circ T_\beta \circ T_\alpha \circ \mathbf{X}_{\text{postop}} = T_\beta^{-1} (\widetilde{\mathbf{Y}}'_{\theta_{\alpha\beta}}) \quad (4.4)$$

We ensure that  $T_\beta$  is invertible by using diffeomorphic spatial transformations. To preserve image quality and ensure that probabilities stay within  $[0, 1]$ , we use tricubic and trilinear interpolation for  $T_\beta$  and  $T_\beta^{-1}$ , respectively.

Predictions  $P = \{\widetilde{\mathbf{Y}}'_{\text{cavity}_i}\}_{i=1}^{n_u}$  are averaged to obtain  $\widetilde{\mathbf{Y}}_{\text{cavity}} : \Omega \rightarrow [0, 1]$ , and the corresponding binary pseudolabel  $\widetilde{\mathbf{Y}}_{\text{cavity}} : \Omega \rightarrow \{0, 1\}$  is obtained applying a threshold of 0.5 to  $\widetilde{\mathbf{Y}}'_{\text{cavity}}$ .

### Uncertainty estimation as selection criterion for pseudolabeled instances

Images in  $D_{\text{unlabeled}}$  might have artifacts that limit the quality of the segmentation or include RCs not modeled by  $\phi_{\text{sim}}$ . The corresponding noisy pseudolabels would hinder training of machine learning models.

As  $n_{\text{unl}}$  might be large, rather than performing manual quality control to select pseudolabels with high reliability for training, we use uncertainty estimation as an automated selection criterion [Venturini 20].

We use  $n_{\text{unc}}$  TTA predictions to estimate aleatoric uncertainty, which captures noise inherent in the observation [Kendall 17]. Aleatoric uncertainty can indicate segmentation quality and is a successful selection criterion of pseudolabels in semi-supervised learning settings for medical image segmentation [Wang 19a, Venturini 20].

Let  $L = \{l_i\}_{i=1}^{n_{\text{unc}}}$  denote the set of (soft) volumes of the segmented cavity for each prediction, where  $l_i$  is the sum of all probabilities in the  $i$ -th prediction  $\widetilde{\mathbf{Y}}'_{\text{cavity}_i} \in P$ . We use the coefficient of quartile variation (CQV) of the volumes [Zwillinger 99, Wang 19a] to estimate the image-level uncertainty  $u : L \rightarrow [0, 1]$ :

$$u = \frac{q_3 - q_1}{q_3 + q_1} \quad (4.5)$$

where  $q_1$  and  $q_3$  are the first and third quartiles of  $L$ , respectively. The CQV is agnostic to the volume of the segmented RC and therefore avoids bias introduced by naturally-occurring uncertainty along the resection boundaries [Jungo 20]. Finally,

training instances with an associated prediction uncertainty  $u(f_{\theta\alpha\beta}, \mathbf{X}_{\text{postop}}, n_{\text{unc}})$  below a threshold  $t_{\text{unc}}$  are combined with self-labeled instances (Section 4.2.1.2) to train a new model.

### 4.2.2 Resection simulation for self-supervised learning

$\phi_{\text{sim}}$  takes images from  $\mathcal{D}_{\text{preop}}$  to generate training instances by simulating a realistic shape, location and intensity pattern for the RC. We present simulation of cavity shape and label in Sections 4.2.2.1 and 4.2.2.2, respectively. In Section 4.2.2.3, we present our method to generate the resected image.

#### 4.2.2.1 Initial cavity shape

To simulate a realistic RC, we consider its topological and geometric properties: it is a single volume with a non-smooth boundary. We generate a geodesic polyhedron with frequency  $f$  by subdividing the edges of an icosahedron  $f$  times and projecting each vertex onto a parametric sphere with a unit radius centered at the origin (Fig. 4.2). This polyhedron models a spherical surface  $S = \{V, F\}$  with vertices  $V = \{\mathbf{v}_i \in \mathbb{R}^3\}_{i=1}^{n_V}$  and faces  $F = \{\mathbf{f}_k \in \mathbb{N}^3\}_{k=1}^{n_F}$ , where  $n_V$  and  $n_F$  are the number of vertices and faces, respectively. Each face  $\mathbf{f}_k = \{i_1^k, i_2^k, i_3^k\}$  is a sequence of three non-repeated vertex indices.

To create a non-smooth surface,  $S$  is perturbed with simplex noise [Perlin 02], a procedural noise generated by interpolating pseudorandom gradients on a multidimensional simplicial grid. We chose simplex noise as it simulates natural-looking textures or terrains and is computationally efficient for multiple dimensions. The noise  $\eta : \mathbb{R}^3 \rightarrow [-1, 1]$  at point  $\mathbf{p} \in \mathbb{R}^3$  is a weighted sum of the noise contribution for  $\omega$  different octaves, with weights  $\{\gamma^{n-1}\}_{n=1}^\omega$  controlled by the persistence parameter  $\gamma$ . The displacement  $\delta$  of a vertex  $\mathbf{v}$  is:

$$\delta(\mathbf{v}) = \eta \left( \frac{\mathbf{v} + \boldsymbol{\mu}}{\zeta}, \omega, \gamma \right) \quad (4.6)$$

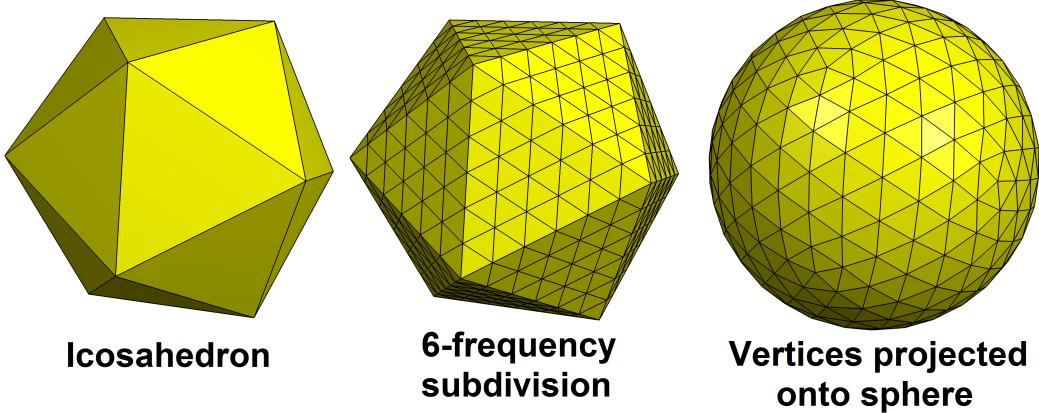


Figure 4.2: Geodesic polyhedron from a subdivided icosahedron. Courtesy of Wikipedia.

where  $\zeta$  is a scaling parameter to control smoothness and  $\mu$  is a shifting parameter that adds stochasticity (equivalent to a random number generator seed). Each vertex  $\mathbf{v}_i$  is displaced radially to create a perturbed sphere:  $V_\delta = \left\{ \mathbf{v}_i + \delta(\mathbf{v}_i) \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|} \right\}_{i=1}^{n_V} = \{\mathbf{v}_{\delta i}\}_{i=1}^{n_V}$ . This process is illustrated in Fig. 4.3

Next, a series of transforms is applied to  $V_\delta$  to modify the mesh's volume and shape. To add stochasticity, random rotations around each axis are applied to  $V_\delta$  with the rotation transform  $T_R(\boldsymbol{\theta}_r) = R_x(\theta_x) \circ R_y(\theta_y) \circ R_z(\theta_z)$ , where  $\circ$  indicates a transform composition and  $R_i(\theta_i)$  is a rotation of  $\theta_i$  radians around axis  $i$ .  $T_S(\mathbf{r})$  is a scaling transform, where  $(r_1, r_2, r_3) = \mathbf{r}$  are semiaxes of an ellipsoid with volume  $v$  used to model the cavity shape. The semiaxes are computed as  $r_1 = r$ ,  $r_2 = \lambda r$  and  $r_3 = r/\lambda$ , where  $r = (3v/4)^{1/3}$  and  $\lambda$  controls the semiaxes length ratios<sup>1</sup>. These transforms are applied to  $V_\delta$  to define the initial RC surface  $S_E = \{V_E, F\}$ , where  $V_E = \{T_S(\mathbf{r}) \circ T_R(\boldsymbol{\theta}_r)(\mathbf{v}_{\delta i})\}_{i=1}^{n_V}$ .

#### 4.2.2.2 Cavity label

The simulated RC should not span both hemispheres or include extracerebral tissues such as bone or scalp. This section describes our method to ensure that the RC appears in anatomically plausible regions.

<sup>1</sup>Note the volume of an ellipsoid with semiaxes  $(a, b, c)$  is  $v = \frac{4}{3}\pi abc$ .

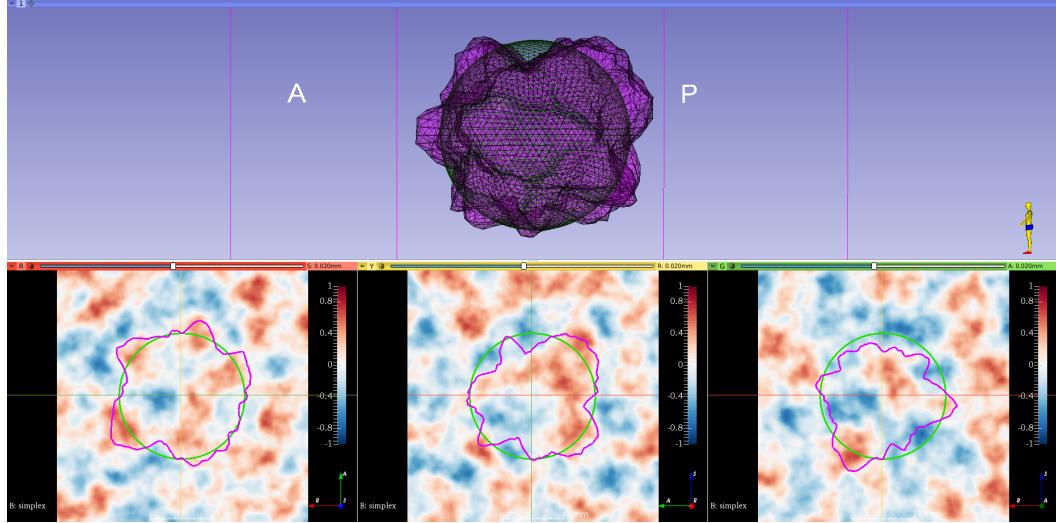


Figure 4.3: Example of a sphere with perturbed vertices. Green: original sphere  $E$ ; magenta: noisy sphere  $E_\epsilon$  with perturbed vertices. Top: 3D reconstruction of both surfaces. Bottom: 3D volume of simplex noise with positive values in red and negative values in blue, and intersections with both surfaces. Note that points of  $E$  that lie on blue voxels get closer to the centre, whereas points on red voxels get farther and points on white voxels (which correspond to a noise value of 0) remain in their original position. The noise volume has been generated for visualization purposes, as the noise needs to be computed only on the vertices.

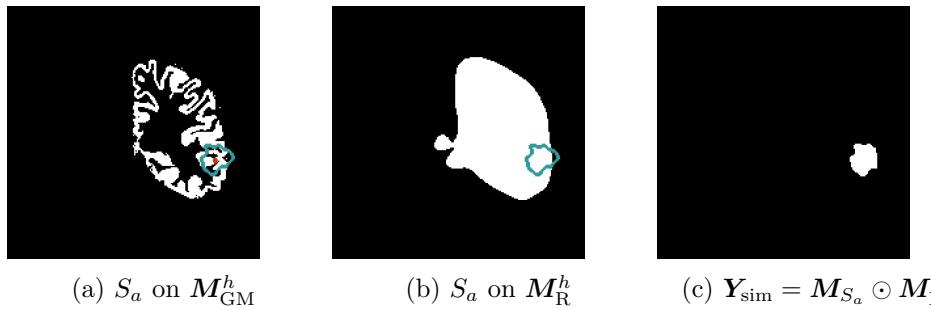


Figure 4.4: Simulation of the ground-truth cavity label.  $S_a$  (blue) is computed by centering  $S_E$  on  $\mathbf{a}$ , a random positive voxel (red) of  $M_{GM}^h$  (a).  $M_{S_a}$  is a binary mask derived from  $S_a$ .  $Y_{sim}$  (c) is the intersection of  $M_{S_a}$  and  $M_R^h$  (b).

A  $T_1$ w MRI is defined as  $\mathbf{X}_{\text{preop}} : \Omega \rightarrow \mathbb{R}$ . A full brain parcellation  $\mathbf{P} : \Omega \rightarrow Z$  is generated [Cardoso 15] for  $\mathbf{X}_{\text{preop}}$ , where  $Z$  is the set of segmented structures. A cortical gray matter mask  $\mathbf{M}_{\text{GM}}^h : \Omega \rightarrow \{0, 1\}$  of hemisphere  $h$  is extracted from  $\mathbf{P}$ , where  $h$  is randomly chosen from  $H = \{\text{left, right}\}$  with equal probability.

A “resectable hemisphere mask”  $\mathbf{M}_{\text{R}}^h$  is generated from  $\mathbf{P}$  and  $h$  such that  $\mathbf{M}_{\text{R}}^h(\mathbf{p}) = 1$  if  $\mathbf{P}(\mathbf{p}) \neq \{M_{\text{BG}}, M_{\text{BT}}, M_{\text{CB}}, M_{\hat{h}}\}$  and 0 otherwise, where  $M_{\text{BG}}$ ,  $M_{\text{BT}}$ ,  $M_{\text{CB}}$  and  $M_{\hat{h}}$  are the labels in  $Z$  corresponding to the background, brainstem, cerebellum and contralateral hemisphere, respectively.  $\mathbf{M}_{\text{R}}^h$  is smoothed using a series of binary morphological operations, for realism.

A random voxel  $\mathbf{a} \in \Omega$  is selected such that  $\mathbf{M}_{\text{GM}}^h(\mathbf{a}) = 1$ . A translation transform  $T_{\text{T}}(\mathbf{a} - \mathbf{c})$  is applied to  $S_{\text{E}}$  so  $S_a = T_{\text{T}}(\mathbf{a} - \mathbf{c})(S_{\text{E}})$  is centered on  $\mathbf{a}$ .

A binary image  $\mathbf{M}_{S_a} : \Omega \rightarrow \{0, 1\}$  is generated from  $S_a$  such that  $\mathbf{M}_{S_a}(\mathbf{p}) = 1$  for all  $\mathbf{p}$  within  $S_a$  and  $\mathbf{M}_{S_a}(\mathbf{p}) = 0$  outside. Finally,  $\mathbf{M}_{S_a}$  is restricted by  $\mathbf{M}_{\text{R}}^h$  to generate the cavity label  $\mathbf{Y}_{\text{sim}} = \mathbf{M}_{S_a} \odot \mathbf{M}_{\text{R}}^h$ , where  $\odot$  represents the Hadamard product. Fig. 4.4 illustrates the process, and Fig. 4.5 shows an example.

#### 4.2.2.3 Simulating cavities filled with CSF

Brain RCs are typically filled with CSF. To generate a realistic CSF texture, we create a ventricle mask  $\mathbf{M}_{\text{V}} : \Omega \rightarrow \{0, 1\}$  from  $\mathbf{P}$ , such that  $\mathbf{M}_{\text{V}}(\mathbf{p}) = 1$  for all  $\mathbf{p}$  within the ventricles and  $\mathbf{M}_{\text{V}}(\mathbf{p}) = 0$  outside. Intensity values within the ventricles are assumed to have a normal distribution [Gudbjartsson 95] with a mean  $\mu_{\text{CSF}}$  and standard deviation  $\sigma_{\text{CSF}}$  calculated from voxel intensity values in  $\{\mathbf{X}_{\text{preop}}(\mathbf{p}) \mid \mathbf{p} \in \Omega \wedge \mathbf{M}_{\text{V}}(\mathbf{p}) = 1\}$ . A CSF-like image is then generated as  $\mathbf{X}_{\text{CSF}}(\mathbf{p}) \sim \mathcal{N}(\mu_{\text{CSF}}, \sigma_{\text{CSF}}), \forall \mathbf{p} \in \Omega$ .

We use  $\mathbf{Y}_{\text{sim}}$  to guide blending of  $\mathbf{X}_{\text{CSF}}$  and  $\mathbf{X}_{\text{preop}}$  as follows. A Gaussian filter is applied to  $\mathbf{Y}_{\text{sim}}$  to obtain a smooth alpha channel  $\mathbf{A}_{\alpha} : \Omega \rightarrow [0, 1]$  defined as  $\mathbf{A}_{\alpha} = \mathbf{Y}_{\text{sim}} * \mathbf{G}_{\mathcal{N}}(\boldsymbol{\sigma})$ , where  $*$  is the convolution operator and  $\mathbf{G}_{\mathcal{N}}(\boldsymbol{\sigma})$  is a 3D Gaussian kernel with standard deviations  $\boldsymbol{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ . Then,  $\mathbf{X}_{\text{CSF}}$  and  $\mathbf{X}_{\text{preop}}$

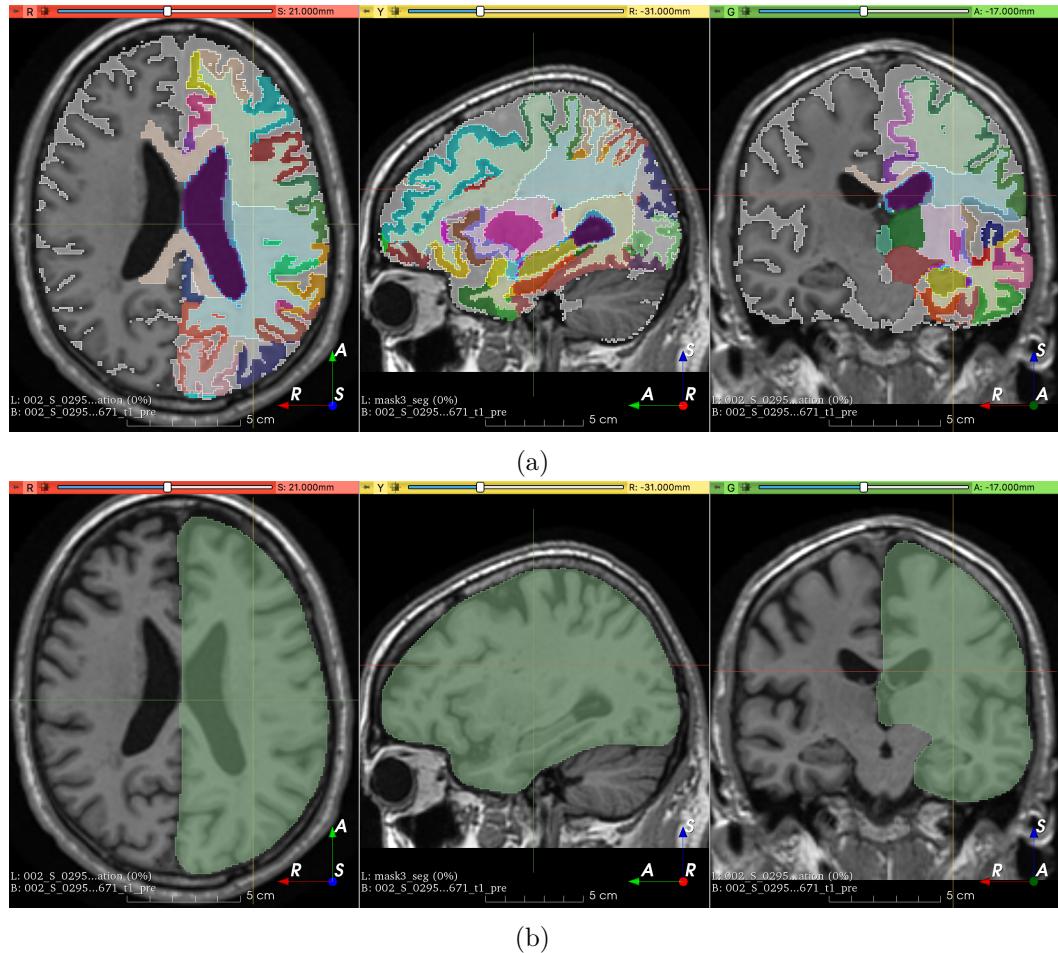


Figure 4.5: Example of a resectable hemisphere mask. Structures in  $\mathcal{P}$  used to create the initial resectable hemisphere mask  $M_R^h$  (a). Resectable hemisphere mask after binary morphological operations (b).

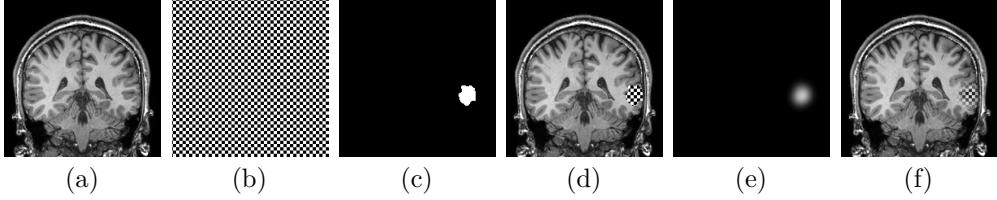


Figure 4.6: Simulation of resected image  $\mathbf{X}_{\text{sim}}$ . We use a checkerboard for visualization. Two scalar-valued images  $\mathbf{X}_{\text{preop}}$  (a) and  $\mathbf{X}_2$  (b) are blended using  $\mathbf{Y}_{\text{sim}}$  (c) and  $\sigma_i = 0 \text{ mm}$  to create an image with hard boundaries (d) and  $\sigma_i = 5 \text{ mm}$  (e) for an image with soft boundaries (f), mimicking partial-volume effects.

are blended by the convex combination

$$\mathbf{X}_{\text{sim}} = \mathbf{A}_\alpha \odot \mathbf{X}_{\text{CSF}} + (1 - \mathbf{A}_\alpha) \odot \mathbf{X}_{\text{preop}} \quad (4.7)$$

We use  $\sigma > 0$  to mimic partial-volume effects at the cavity boundary. The blending process is illustrated in Fig. 4.6.

## 4.3 Experiments and results

### 4.3.1 Data

Table 4.1 summarizes the datasets used for evaluation.

#### 4.3.1.1 Public data for simulation

$T_{1\text{w}}$  MRIs were collected from publicly available datasets Information eXtraction from Images (IXI)<sup>2</sup>, Alzheimer’s Disease Neuroimaging Initiative (ADNI) [Jack 08], and Open Access Series of Imaging Studies (OASIS) [LaMontagne 19], for a total of 1813 images.

These datasets are used as control subjects in our self-supervised experiments (Section 4.2.1.2). Although we use the term ‘control’ to refer to subjects that have not undergone resective surgery, they may have other neurological conditions. For example, subjects in ADNI may suffer from Alzheimer’s disease.

---

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

Table 4.1: Datasets used in this study. Where multiple resolutions are present, the minimum, mean and maximum for each dimension are shown. ‘ $T_1wCE$ ’ indicates that gadolinium was administered for contrast enhancement.

Dataset	Modality	Resolution (mm)	Subjects	Surgery	Annotated
<b>IXI</b>	$T_1w$	$0.94 \times 0.94 \times 1.20$			
		$0.94 \times 0.94 \times 1.20$	566	-	-
		$0.98 \times 0.98 \times 1.20$			
<b>ADNI</b>	$T_1w$	$1.00 \times 1.00 \times 1.00$	467	-	-
<b>OASIS</b>	$T_1w$	$1.00 \times 1.00 \times 1.00$			
		$1.05 \times 1.01 \times 1.02$	780	-	-
		$1.20 \times 1.05 \times 3.00$			
<b>EPISURG</b>	$T_1w$	$0.75 \times 0.75 \times 0.75$			
		$0.96 \times 0.96 \times 1.08$	430	Epilepsy	133
		$1.09 \times 1.09 \times 1.60$			
<b>Milan</b>	$T_1w$	$0.46 \times 0.46 \times 0.90$	20	Epilepsy	20
<b>Strasbourg</b>	$T_1w$ & $T_1wCE$	$0.23 \times 0.23 \times 0.50$			
		$0.61 \times 0.61 \times 2.79$	33	Epilepsy	33
		$1.00 \times 1.00 \times 5.00$			
<b>Paris</b>	$T_1w$	$0.47 \times 0.47 \times 0.49$			
		$0.82 \times 0.76 \times 1.06$	19	Epilepsy	19
		$1.20 \times 0.98 \times 1.20$			
<b>BITE</b>	$T_1wCE$	$1.00 \times 0.47 \times 0.47$			
		$2.31 \times 0.53 \times 0.53$	13	Tumor	0
		$5.50 \times 0.55 \times 0.55$			

#### 4.3.1.2 Multicenter epilepsy data

We evaluate the generalizability of our approach to data from several institutions: *Milan* ( $n = 20$ ), *Paris* ( $n = 19$ ), *Strasbourg* ( $n = 33$ ), and EPISURG ( $n = 133$ ).

We curated the EPISURG dataset using images from patients with refractory focal epilepsy who underwent resective surgery between 1990 and 2018 at the National Hospital for Neurology and Neurosurgery (NHNN), London, United Kingdom. These were anonymized data that had been previously acquired as a part of clinical care, so individual patient consent was not required. All images in EPISURG were defaced using a predefined face mask in the Montreal Neurological Institute (MNI) space to preserve patient identity. In total, there were 430 patients with postoperative  $T_1w$  MRI, 268 of which had a corresponding preoperative MRI. The distribution of resection types is shown in Table 4.2.

Three human raters annotated 200 of the postoperative images in EPISURG [Pérez-García 20e]. Annotations used for evaluation in this study were performed semi-automatically using a fast grow-cut algorithm implemented in 3D Slicer 4.10 [Zhu 14, Fedorov 12].

EPISURG is available online and can be freely downloaded [Pérez-García 20d]<sup>3</sup>.

The same human rater (F.P.G.) annotated all images from *Milan*, *Paris* and *Strasbourg* using the same protocol that was used for EPISURG.

#### 4.3.1.3 Brain tumor datasets

The Brain Images of Tumors for Evaluation (BITE) dataset [Mercier 12] consists of ultrasound and MRI of patients with brain tumors. We use the 13 postoperative  $T_1w$  MRI with contrast enhancement ( $T_1wCE$ ) in BITE to perform a qualitative assessment of the potential of our models to generalize to images from a substantially different domain (images are contrast-enhanced) and different pathology, which may require different surgical techniques that could affect RC appearance.

---

<sup>3</sup><https://doi.org/10.5522/04/9996158.v1>

Table 4.2: Lobar distribution of resection types in EPISURG.

Lobe	Type	Subjects
Temporal	lobectomy	317
Temporal	lesionectomy	30
Temporal-frontal	lobectomy	2
Temporal-parietal	lobectomy	1
Frontal	lobectomy	47
Frontal	lesionectomy	10
Parietal	lesionectomy	11
Parietal	lobectomy	4
Occipital-parietal	lobectomy	2
Occipital	lobectomy	2
-	multiple subpial	2
-	hemispherectomy	2
<b>Total</b>		430

#### 4.3.1.4 Preprocessing

For all images, the brain was segmented using ROBEX [Iglesias 11]. Voxels within the brain were used to register the images to the nonlinear symmetric ICBM152 MNI template [Fonov 09, Fonov 11] using a pyramidal approach to compute the affine transformation [Modat 14]. All images were resampled into the MNI space using sinc interpolation to preserve image quality. After resampling, images had a 1-mm isotropic resolution and size  $193 \times 229 \times 193$ .

#### 4.3.2 Network architecture and implementation details

We used the PyTorch deep learning framework [Paszke 19], training with automatic mixed precision (AMP) on two 32-GB TESLA V100 graphics processing units (GPUs). In the AMP setting, some operations such as convolution are computed using half precision (i.e., 16-bit floating point) to reduce the computational burden while maintaining a similar performance.

We implemented a variant of 3D U-Net [Cicek 16] using two downsampling and upsampling blocks, upsampling with trilinear interpolation for the synthesis path, and 1/4 of the filters for each convolutional layer. We used dilated convolutions

[Chen 17], starting with a dilation factor of one, then increased or decreased in steps of one after each downsampling or upsampling block, respectively. This results in a model with the same receptive field (a cube of length 88 mm) but  $\approx 77\times$  fewer parameters (246 156) than the original 3D U-Net, reducing overfitting and computational burden.

Convolutional layers were initialized using He’s method and followed by batch normalization and nonlinear parametric rectified linear unit (PReLU) activation functions [Ioffe 15, He 15]. We used adaptive moment estimation (AdamW) [Kingma 14, Loshchilov 19] to adjust the learning rate during training, with weight decay of  $10^{-2}$  and a learning scheduler that divides the learning rate by ten every 20 epochs. We optimized our network to minimize the mean soft Dice loss [Milletari 16] of each mini-batch, for all the experiments. A mini-batch size of ten images (five per GPU) was used for training. Self-supervised training took about 27 hours. Fine-tuning on a small annotated dataset took about seven hours.

We used Sacred [Greff 17] and TensorBoard [Abadi 16] to configure, log and visualize our experiments.

### 4.3.3 Processing during training

#### 4.3.3.1 Resection simulation

We perform the resection simulation on the fly, i.e., during training. Simulation requires around 0.5 s for an image of size  $193 \times 229 \times 193$ . In practice, we perform expensive operations such as convolutions on subvolumes to reduce computational burden. The simulation is implemented using SimpleITK [Lowekamp 13], VTK [Schroeder 06] and NumPy [van der Walt 11]. To generate the noisy sphere, we used `pyDome`<sup>4</sup> and `noise`<sup>5</sup>.

---

<sup>4</sup><https://github.com/badassdatascience/pyDome>

<sup>5</sup><https://github.com/caseman/noise>

### 4.3.3.2 Preprocessing and augmentation

We use TorchIO transforms to load, preprocess and augment our data during training [Pérez-García 21e] (Chapter 5). Instead of preprocessing the images with denoising or bias removal, we simulate different artifacts in the training instances so that our models are robust to them.

Our preprocessing and augmentation transforms are described below. For transforms that are not applied to all images, we show the probability  $p$  of the transform being applied:

1. Random resection simulation (for self-supervised training only)
2. Histogram standardization [Nyúl 00]
3. Simulation of low resolution artifacts ( $p = 0.75$ ). Sampled uniformly from
  - (a) Random simulation of anisotropic spacing [Billot 20b] and
  - (b) Gaussian blurring with random variance
4. Random simulation of MRI ghosting artifacts [Shaw 20] ( $p = 0.2$ )
5. Random simulation of MRI spike artifacts [Shaw 20] ( $p = 0.2$ )
6. Random simulation of MRI motion artifacts [Shaw 19] ( $p = 0.2$ )
7. Random simulation of bias field inhomogeneity [Sudre 17] ( $p = 0.5$ )
8. Foreground standardization to zero-mean and unit variance using only voxels with intensity above the mean to compute the statistics
9. Gaussian noise with random variance ( $p = 0.75$ )
10. Diffeomorphic spatial transform, sampled from either
  - (a) Random rotation and anisotropic scaling ( $p = 0.9$ ) or
  - (b) Random elastic deformation ( $p = 0.1$ )
11. Random flip around the sagittal plane ( $p = 0.5$ )
12. Crop to a tight bounding box around the brain of size of  $176 \times 216 \times 160$  voxels.

We refer the reader to the GitHub repository<sup>6</sup> for details on the transforms

---

<sup>6</sup><https://github.com/fepegar/resseg-ijcars>

parameters used for our experiments.

#### 4.3.4 Experiments

Overlap measurements are reported as ‘median (interquartile range)’ DSC. No postprocessing is performed for evaluation, except thresholding at 0.5. We analyzed differences in model performance using a one-tailed Mann-Whitney  $U$  test (as DSCs were not normally distributed) with a significance threshold of  $\alpha = 0.05$ , and Bonferroni correction for  $n$  experiments:  $\alpha_{\text{Bonf}} = \frac{\alpha}{n(n-1)}$ .

##### 4.3.4.1 Self-supervised learning: training with simulated resections only

In our first experiment, we assess the relation between the resection simulation complexity and the segmentation performance of the model. We train our model with simulated resections on the publicly available dataset  $D_{\text{preop}} = \{\mathbf{X}_{\text{preop}_i}\}_{i=1}^{n_{\text{preop}}}$ , where  $n_{\text{preop}} = 1813$  (Section 4.3.1). We use 90% of the images in  $D_{\text{preop}}$  for the training set  $D_{\text{pre,train}}$  and 10% for the validation set. At each training iteration,  $b$  images from  $D_{\text{pre,train}}$  are loaded, resected, preprocessed and augmented to obtain a mini-batch of  $b$  training instances  $\{(\mathbf{X}_{\text{sim}_i}, \mathbf{Y}_{\text{sim}_i})\}_{i=1}^b$ . Note that the resection simulation is performed on the fly, which ensures that the network never sees the same resection during training. Models were trained for 60 epochs, using an initial learning rate of  $10^{-3}$ . We use the model weights from the epoch with the lowest mean validation loss obtained during training for evaluation. Models were tested on the 133 annotated images in EPISURG.

To investigate the effect of the simulated cavity shape on model performance, we modify  $\phi_{\text{sim}}$  to generate cuboid- (Fig. 4.7a) or ellipsoid-shaped (Fig. 4.7b) resections, and compare with the baseline “noisy” ellipsoid (Fig. 4.7c). The cuboids and ellipsoid meshes are not perturbed using simplex noise, and cuboids are not rotated.

Best results were obtained by the baseline model (80.5 (18.7)), trained using ellipsoids perturbed with procedural noise. Models trained with cuboids and rotated

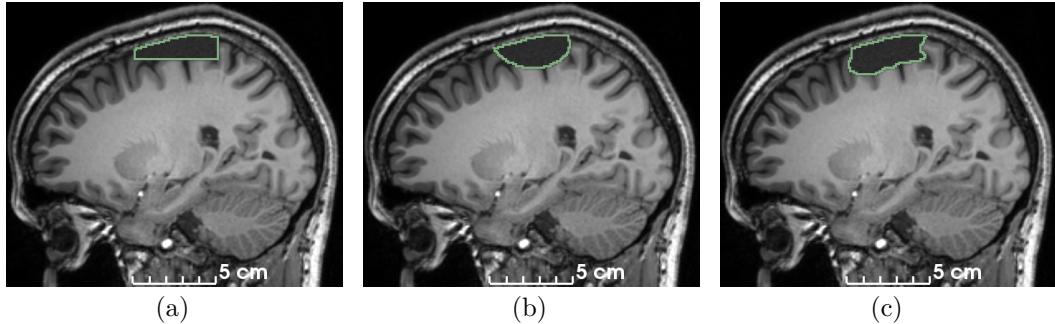


Figure 4.7: Simulation of RCs with increasing shape complexity (Section 4.2.2): cuboid (a), ellipsoid (b) and ellipsoid perturbed with simplex noise (c).

ellipsoids performed significantly (57.9 (73.1),  $p < 10^{-8}$ ) and marginally (79.0 (20.0),  $p = 0.123$ ) worse.

#### 4.3.4.2 Fine-tuning on small clinical datasets

We assessed the generalizability of our baseline model by fine-tuning it on small datasets from four institutions that may use different surgical approaches and acquisition protocols (including contrast enhancement and anisotropic spacing in *Strasbourg*) (Section 4.3.1.2). Additionally, we fine-tuned the model on 20 cases from EPISURG with the lowest DSC in Section 4.3.4.1.

For each dataset, we load the pretrained baseline model, initialize the optimizer with an initial learning rate of  $5 \times 10^{-4}$ , initialize the learning rate scheduler and fine-tune all layers simultaneously for 40 epochs using 5-fold cross-validation. We use model weights from the epoch with the lowest mean validation loss for evaluation. To minimize data leakage, we determined the above hyperparameters using the validation set of one fold in the *Milan* dataset.

We observed a consistent increase in DSC for all fine-tuned models, up to a maximum of 89.2 (13.3) for the *Milan* dataset. For comparison, inter-rater agreement between human annotators in our previous study was 84.0 (9.9) [Pérez-García 20e]. Quantitative and qualitative evaluations are illustrated in Figs. 4.8 and 4.9, respectively.

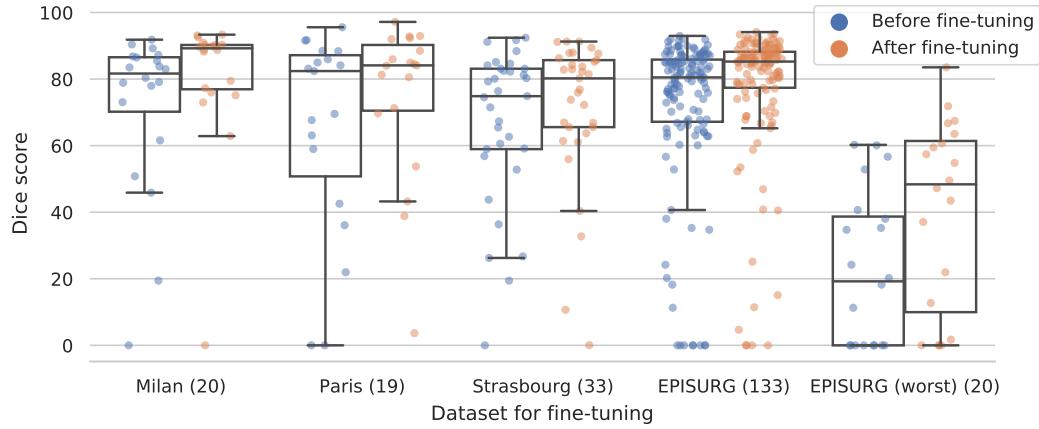


Figure 4.8: DSC without (blue) and with (orange) fine-tuning of the model training using self-supervision. Horizontal lines in the boxes represent the first, second (median) and third quartiles. The ‘EPISURG (worst)’ dataset comprises the 20 cases from EPISURG with the lowest DSC in the experiment described in Section 4.3.4.1. Numbers in parentheses indicate subjects per dataset.

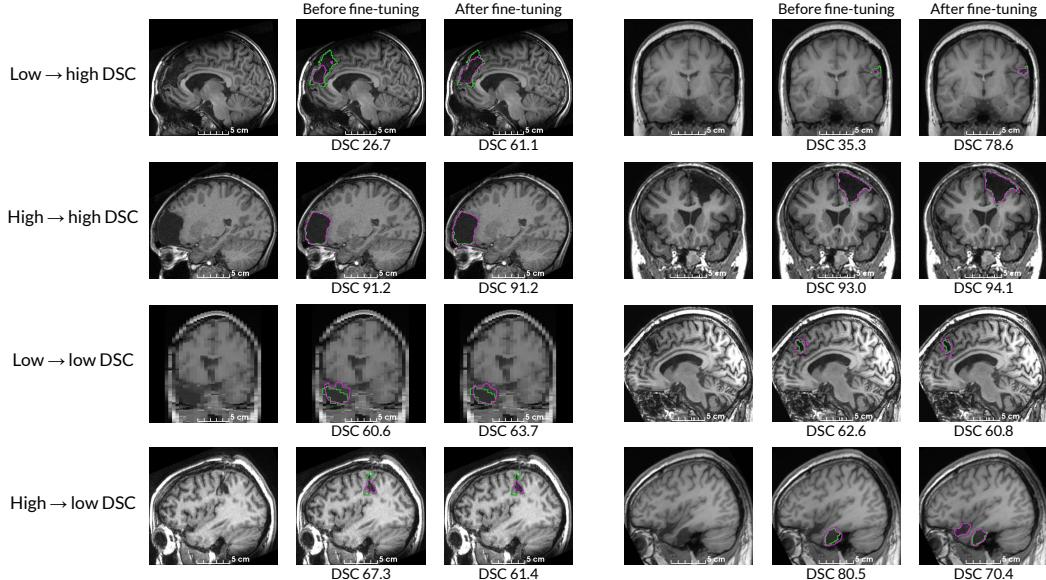


Figure 4.9: Qualitative evaluation of fine-tuning for the *Strasbourg* (left) and EPISURG (right) datasets. Rows correspond, from top to bottom, to cases for which the DSC 1) increased, 2) remained high, 3) remained low and 4) decreased after fine-tuning the self-supervised model. Manual annotations (green) and model predictions (magenta) are overlaid.

#### 4.3.4.3 Semi-supervised learning: leveraging real unlabeled resections

In this experiment, we assessed the ability of semi-supervised learning to improve the performance of our baseline model. We first computed the uncertainty  $u(f_{\alpha\beta}, \mathbf{X}_{\text{postop}}, n_{\text{unc}})$  for  $D_{\text{unlabeled}}$  (Section 4.2.1.3), all unlabeled images in EPISURG, using the baseline model and the preprocessing and augmentation transforms (Section 4.3.3.2). We generated pseudolabels and estimated uncertainty from  $n_{\text{unc}} = 50$  Monte Carlo TTA iterations (Section 4.2.1.3). To obtain the final dataset  $D_{\text{pseudo}} = \{(\mathbf{X}_{\text{postop}_i}, \widetilde{\mathbf{Y}}_{\text{cavity}_i})\}_{i=1}^{n_{\text{pseudo}}}$ , we selected pseudolabels with  $u(\mathbf{X}_{\text{postop}_i}, \cdot) < t_{\text{unc}} = 0.2$ , resulting in  $n_{\text{pseudo}} = 256$ .

We used  $D_{\text{pre,train}}$  (Section 4.3.4.1) in addition to  $D_{\text{pseudo}}$  to train a new model  $f_{\theta_{\text{sim,semi}}}(\cdot)$ , using the same hyperparameters as in the self-supervised setting (Section 4.3.4.1). To ensure that all batches contain real resections, we use  $b - b_{\text{pseudo}}$  images from  $D_{\text{pre,train}}$  and  $b_{\text{pseudo}}$  images from  $D_{\text{pseudo}}$  to compose each mini-batch of size  $b$ . We chose  $b_{\text{pseudo}} = 2$  for our experiments, i.e., 20% of the images in each mini-batch contain real RCs.

Semi-supervised learning improved the performance of the baseline model from 80.5 (18.7) to 81.5 (17.8) ( $p = 0.474$ ).

#### 4.3.4.4 Qualitative evaluation on brain tumor resection dataset

We used the BITE dataset [Mercier 12] to evaluate the ability of our self-supervised model to segment RCs on images from a different institution, modality and pathology than the datasets used for quantitative evaluation. For postprocessing, all but the largest binary connected component were removed. The model successfully segmented the RC on 11/13 images, even though some contained challenging features (Fig. 4.11).

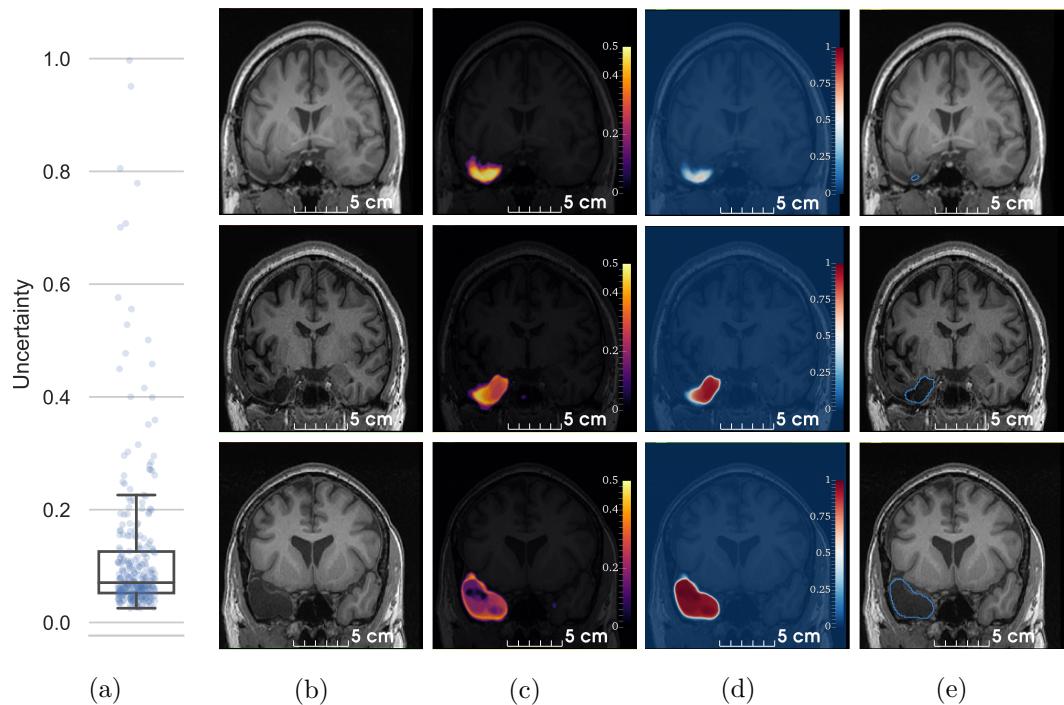


Figure 4.10: Generating reliable pseudolabels for semi-supervised learning. Image-level uncertainty for the 297 unlabeled postoperative images in EPISURG (a). Unlabeled postoperative  $T_1w$  MRI (b). Voxel-wise uncertainty (c) estimated as the standard deviation of the probabilities across all Monte Carlo iterations. The mean prediction (d) is thresholded at 0.5 to generate the pseudolabel (e). Image-level uncertainties for the three cases are 0.805 (top), 0.195 (middle) and 0.025 (bottom).

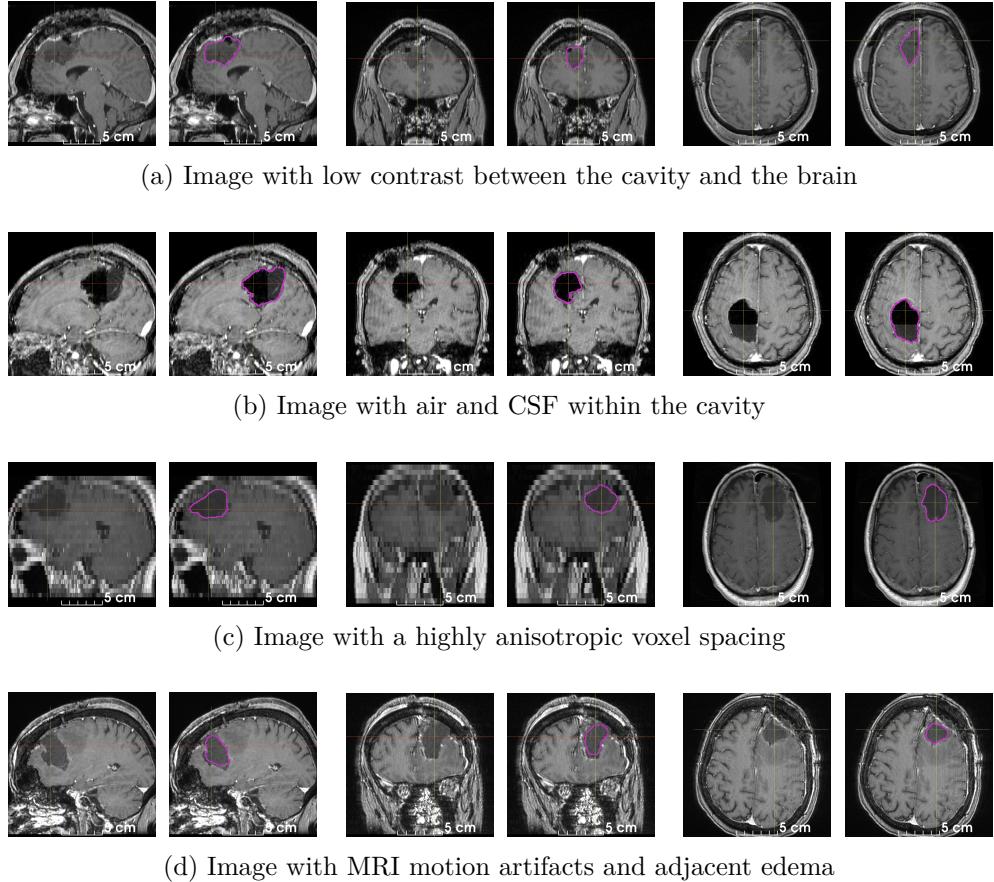


Figure 4.11: Qualitative evaluation of the self-supervised model on a dataset of postoperative brain tumor  $T_1$ wCE MRI. The model is robust to multiple challenging scenarios: low contrast between the cavity and the brain (a), air and CSF within the RC (b), highly anisotropic voxel spacing (c), motion artifacts and edema (d), and a different modality than used for training (all). Note that these images are from a different institution, modality and pathology than the datasets used for quantitative evaluation. Ground-truth labels are not shown as manual annotations are not available.

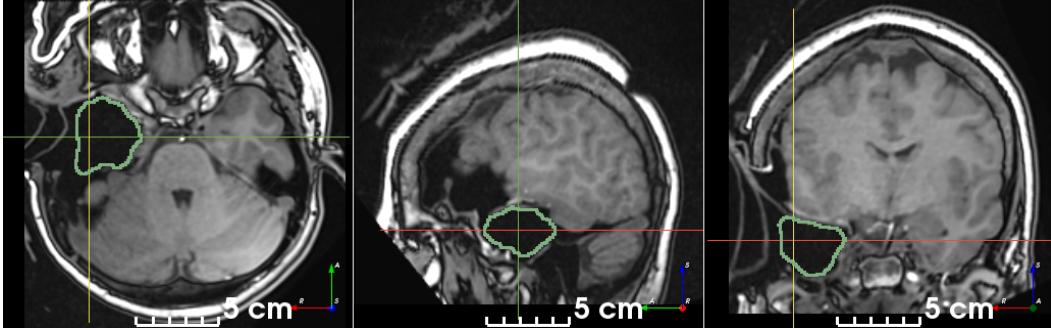


Figure 4.12: Qualitative result on an intraoperative MRI. The baseline model correctly discarded regions filled with air or CSF outside of the RC.

#### 4.3.4.5 Qualitative evaluation on intraoperative image

We used our baseline model to segment the RC on one intraoperative MRI from our institution. Despite the large domain shift between the training dataset and the intraoperative image, which includes a retracted skin flap and a missing bone flap, the model was able to correctly estimate the RC, discarding similar regions filled with CSF or air (Fig. 4.12).

## 4.4 Discussion

We addressed the challenge of segmenting postoperative brain RCs from  $T_1$ w MRI without annotated data. We developed a self-supervised learning strategy to train without manually annotated data, and a method to simulate RCs from preoperative MRI to generate training data. Our novel approach is conceptually simple, easy to implement, and relies on clinical knowledge about postoperative phenomena. The resection simulation is computationally efficient ( $< 1$  s), so it can run during training as part of a data augmentation pipeline. It is compatible with the TorchIO framework [Pérez-García 21e] (Chapter 5) to leverage other data augmentation techniques during training, enabling our model to have a robust performance across MRI of variable quality.

Modeling a realistic cavity shape is important (Section 4.3.4.1). Our model

generalizes well to clinical data from different institutions and pathologies, including epilepsy and glioma. Models may be easily fine-tuned using small annotated clinical datasets to improve performance. Moreover, our resection simulation and learning strategy may be extended to train with arbitrary modalities, or synthetic modalities generated from brain parcellations [Billot 20a]. Therefore, our strategy can be adopted by institutions with a large amount of unlabeled data, while fine-tuning and testing on a smaller labeled dataset.

Poor segmentation performance is often due to very small cavities, where the cavity was not detected, and large brain shift or subdural edema, where regions were incorrectly segmented. The former issue may be overcome by training with a distribution of cavity volumes which oversamples small resections. The latter can be addressed by extending our method to simulate displacement with biomechanical models or nonlinear deformations of the brain [Granados 21].

The baseline model performance improved by leveraging unlabeled postoperative images for semi-supervised learning, but remained lower than inter-rater variability [Pérez-García 20e]. We believe that a setting with a smaller training dataset might benefit further from the semi-supervised approach. However, we did not perform an extensive assessment of our semi-supervised approach as this is out of the scope of this paper.

We showed that our model correctly segmented an intraoperative image, respecting imaginary boundaries between brain and skull, suggesting a good inductive bias of human neuroanatomy. Qualitative results and execution time, which is in the order of milliseconds, suggest that our method could be used intraoperatively, for image guidance during resection or to improve registration with preoperative images by masking the cost function using the RC segmentation [Brett 01]. Segmenting the RC may also be used to study potential damage to white matter tracts postoperatively [Winston 12]. Our method could be easily adapted to simulate other lesions for self-supervised training, such as cerebral microbleeds [Cuadrado-Godia 18], narrow

---

**88    Chapter 4. Segmentation of postoperative brain resection cavities**

and snake-shaped RCs typical of disconnective surgeries [Mohamed 11], or RCs with residual tumor [Meier 17].

As part of this work, we curated and released EPISURG, an MRI dataset with annotations from three independent raters. EPISURG could serve as a benchmark dataset for quantitative analysis of pre- and postoperative imaging of open resection for epilepsy treatment. To the best of our knowledge, this is the first open annotated database of post-resection MRI for epilepsy patients.

# TorchIO: a library for medical image preprocessing and augmentation in deep learning

---

## Table of contents

<b>5.1</b>	<b>Introduction</b>	<b>89</b>
5.1.1	Motivation	91
5.1.2	Related work	96
<b>5.2</b>	<b>Methods</b>	<b>98</b>
5.2.1	Data	99
5.2.2	Transforms	103
<b>5.3</b>	<b>Results</b>	<b>111</b>
5.3.1	Code availability	111
5.3.2	Usage examples	112
<b>5.4</b>	<b>Discussion</b>	<b>114</b>

**Foreword** This chapter is an *in extenso* reproduction of:

- **Fernando Pérez-García**, Rachel Sparks and Sébastien Ourselin. *TorchIO: A Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning*. Computer Methods and Programs in Biomedicine, volume 208, page 106236, September 2021

## 5.1 Introduction

Recently, deep learning has become a ubiquitous research approach for solving image understanding and analysis problems.

Convolutional neural networks (CNNs) have become the state of the art for many medical imaging tasks including segmentation [Cicek 16], classification [Lu 18], re-

construction [Chen 18] and registration [Shan 18]. Many of the network architectures and techniques have been adopted from computer vision.

Compared to 2D red-green-blue (RGB) images typically used in computer vision, processing of medical images such as magnetic resonance image (MRI), ultrasound (US) or computerized tomography (CT) presents different challenges. These include a lack of labels for large datasets, high computational costs (as the data is typically volumetric), and the use of metadata to describe the physical size and position of voxels.

Open-source frameworks for training CNNs with medical images have been built on top of TensorFlow [Abadi 16, Pawlowski 17, Gibson 18]. Recently, the popularity of PyTorch [Paszke 19] has increased among researchers due to its improved usability compared to TensorFlow [He 19], driving the need for open-source tools compatible with PyTorch. To reduce duplication of effort among research groups, improve experimental reproducibility and encourage open-science practices, we have developed TorchIO: an open-source Python library for efficient loading, preprocessing, augmentation, and patch-based sampling of medical images designed to be integrated into deep learning workflows.

TorchIO is a compact and modular library that can be seamlessly used alongside higher-level deep learning frameworks for medical imaging, such as the Medical Open Network for AI (MONAI). It removes the need for researchers to code their own preprocessing pipelines from scratch, which might be error-prone due to the complexity of medical image representations. Instead, it allows researchers to focus on their experiments, supporting experiment reproducibility and traceability of their work, and standardization of the methods used to process medical images for deep learning.

### 5.1.1 Motivation

The nature of medical images makes it difficult to rely on a typical computer-vision pipeline for neural network training. In Section 5.1.1.1, we describe challenges related to medical images that need to be overcome when designing deep learning workflows. In Section 5.1.1.2, we justify the choice of PyTorch as the main deep learning framework dependency of TorchIO.

#### 5.1.1.1 Challenges in medical image processing for deep learning

In practice, multiple challenges must be addressed when developing deep learning algorithms for medical images: 1) handling metadata related to physical position and size, 2) lack of large labeled datasets, 3) high computational costs due to data multidimensionality and 4) lack of consensus for best normalization practices. These challenges are very common in medical imaging and require certain features that may not be implemented in more general-purpose image processing frameworks such as Albumentations [Buslaev 20] or TorchVision [Paszke 19].

**Metadata** In computer vision, picture elements, or *pixels*, which are assumed to be square, have a spatial relationship that comprises proximity and depth according to both the arrangement of objects in the scene and camera placement. In comparison, medical images are reconstructed such that the location of volume elements, or cuboid-shaped *voxels*, encodes a meaningful 3D spatial relationship. In simple terms, for 2D natural images, pixel vicinity does not necessarily indicate spatial correspondence, while for medical images spatial correspondence between nearby voxels can often be assumed.

Metadata, which encodes the physical size, spacing, and orientation of voxels, determines spatial relationships between voxels [Larobina 14]. This information can provide meaningful context when performing medical image processing, and is often implicitly or explicitly used in medical imaging software. Furthermore, metadata is

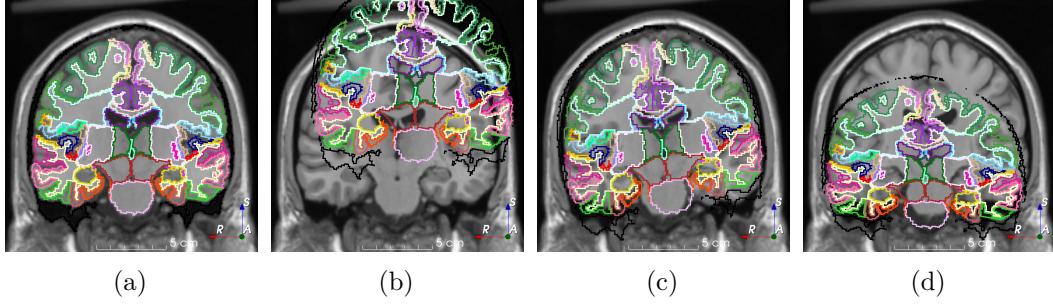


Figure 5.1: Demonstration of the importance of spatial metadata in medical image processing. The size of both the MRI and the segmentation is  $181 \times 181$ . When spatial metadata is taken into account (a), images are correctly superimposed (only the borders of each region are shown for clarity purposes). Images are incorrectly superimposed if (b) origin, (c) orientation or (d) spacing are ignored.

often used to determine correspondence between images as well as voxels within an image. For example, registration algorithms for medical images typically work with physical coordinates rather than voxel indices.

Fig. 5.1 shows the superposition of an MRI and a corresponding brain parcellation [Cardoso 15] with the same size ( $181 \times 181$ ) but different origin, spacing and orientation. A naïve user would assume that, given that the superimposition looks correct and both images have the same size, they are ready for training. However, the visualization is correct only because 3D Slicer [Fedorov 12], the software used for visualization, is aware of the spatial metadata of the images. As CNNs generally do not take spatial metadata into account, training using these images without preprocessing would lead to poor results.

Medical images are typically stored in specialized formats such as Data Imaging and Communications in Medicine (DICOM) or Neuroimaging Informatics Technology Initiative (NIfTI) [Larobina 14], and commonly read and processed by medical imaging frameworks such as SimpleITK [Lowekamp 13] or NiBabel [Brett 20].

**Limited training data** Deep learning methods typically require large amounts of annotated data, which are often scarce in clinical scenarios due to concerns over patient privacy, the financial and time burden associated with collecting data as part

of a clinical trial, and the need for annotations from highly-trained and experienced raters. Data augmentation techniques can be used to increase the size of the training dataset artificially by applying different transformations to each training instance while preserving the relationship to annotations.

Data augmentation performed in computer vision typically aims to simulate variations in camera properties, field of view (FOV), or perspective. Traditional data augmentation operations applied in computer vision include geometrical transforms such as random rotation or zoom, color-space transforms such as random channel swapping or kernel filtering such as random Gaussian blurring. Data augmentation is usually performed on the fly, i.e., every time an image is loaded from disk during training.

Several computer vision libraries supporting data augmentation have appeared recently, such as Albumentations [Buslaev 20], or `imgaug` [Jung 20]. PyTorch also includes some computer vision transforms, mostly implemented as Pillow wrappers [wiredfool 16]. However, none of these libraries support reading or transformations for 3D images. Furthermore, medical images are almost always greyscale, therefore colour-space transforms are not applicable. Additionally, cropping and scaling are more challenging to apply to medical images without affecting the spatial relationships of the data. Metadata should usually be considered when applying these transformations to medical images.

In medical imaging, the purpose of data augmentation is designed to simulate anatomical variations and scanner artifacts. Anatomical variation and sample position can be simulated using spatial transforms such as elastic deformation, lateral flipping, or affine transformations. Some artifacts are unique to specific medical image modalities. For example, ghosting artifacts will be present in MRI if the patient moves during acquisition, and metallic implants often produce streak artifacts in CT. Simulation of these artifacts can be useful when performing augmentation on medical images.

**Computational costs** The number of pixels in 2D images used in deep learning is rarely larger than one million. For example, the input size of several popular image classification models is  $224 \times 224 \times 3 = 150\,528$  pixels (588 KiB if 32 bits per pixel are used). In contrast, 3D medical images often contain hundreds of millions of voxels, and downsampling might not be acceptable when small details should be preserved. For example, the size of a high-resolution lung CT-scan used for quantifying chronic obstructive pulmonary disease damage in a research setting, with spacing  $0.66 \times 0.66 \times 0.30$  mm, is  $512 \times 512 \times 1069 = 280\,231\,936$  voxels (1.04 GiB if 32 bits per voxel are used).

In computer vision applications, images used for training are grouped in batches whose size is often in the order of hundreds [Krizhevsky 12] or even thousands [Chen 20] of training instances, depending on the available graphics processing unit (GPU) memory. In medical image applications, batches rarely contain more than one [Cicek 16] or two [Milletari 16] training instances due to their larger memory footprint compared to natural images. This reduces the utility of techniques such as batch normalization, which rely on batches being large enough to estimate dataset variance appropriately [Ioffe 15]. Moreover, large image size and small batches result in longer training time, hindering the experimental cycle that is necessary for hyperparameter optimization. In cases where GPU memory is limited and the network architecture is large, it is possible that not even the entirety of a single volume can be processed during a training iteration. To overcome this challenge, it is common in medical imaging to train using subsets of the image, or image *patches*, randomly extracted from the volumes.

Networks can be trained with 2D slices extracted from 3D volumes, aggregating the inference results to generate a 3D volume [Lucena 19]. This can be seen as a specific case of patch-based training, where the size of the patches along a dimension is one. Other methods extract volumetric patches for training, that are often cubes, if the voxel spacing is isotropic [Li 17], or cuboids adapted to the anisotropic spacing

of the training images [Nikolov 18].

**Transfer learning and normalization** One can pre-train a network on a large dataset of natural images such as ImageNet [Deng 09], which contains more than 14 million labeled images, and fine-tune on a custom, much smaller target dataset. This is a typical use of transfer learning in computer vision [Weiss 16]. The literature has reported mixed results using transfer learning to apply models pretrained on natural images to medical images [Cheplygina 19, Raghu 19].

In computer vision, best practice is to normalize each training instance before training, using statistics computed from the whole training dataset [Krizhevsky 12]. Preprocessing of medical images is often performed on a per-image basis, and best practice is to take into account the bimodal nature of medical images (i.e., that an image has a background and a foreground).

Medical image voxel intensity values can be encoded with different data types and intensity ranges, and the meaning of a specific value can vary between different modalities, sequence acquisitions, or scanners. Therefore, intensity normalization methods for medical images often involve more complex parameterization of intensities than those used for natural images [Nyúl 99].

### 5.1.1.2 Deep learning frameworks

There are currently two major generic deep learning frameworks: TensorFlow [Abadi 16] and PyTorch [Paszke 19], primarily maintained by Google and Facebook, respectively. Although TensorFlow has traditionally been the primary choice for both research and industry, PyTorch has recently seen a substantial increase in popularity, especially among the research community [He 19].

PyTorch is often preferred by the research community as it is *pythonic*, i.e., its design, usage, and application programming interface (API) follow the conventions of plain Python. Moreover, the API for tensor operations follows a similar paradigm to the one for NumPy multidimensional arrays, which is the primary array programming

library for the Python language [van der Walt 11]. In contrast, for TensorFlow, researchers need to become familiar with new design elements such as sessions, placeholders, feed dictionaries, gradient tapes and static graphs. In PyTorch, objects are standard Python classes and variables, and a dynamic graph makes debugging intuitive and familiar to anyone already using Python. These differences have decreased with the recent release of TensorFlow 2, whose eager mode makes usage reminiscent of Python.

TorchIO was designed to be in the style of PyTorch and uses several of its tools to reduce the barrier to learning how to use TorchIO for those researchers already familiar with PyTorch.

### 5.1.2 Related work

NiftyNet [Gibson 18] and the Deep Learning Toolkit (DLTK) [Pawlowski 17] are deep learning frameworks designed explicitly for medical image processing using the TensorFlow 1 platform. Both of them are no longer being actively maintained. They provide implementations of some popular network architectures such as U-Net [Cicek 16], and can be used to train 3D CNNs for different tasks. For example, NiftyNet was used to train a 3D residual network for brain parcellation [Li 17], and DLTK was used to perform multi-organ segmentation on CT and MRI [Valindria 18].

The `medicatorch` library [Perone 18] closely follows the PyTorch design, and provides some functionalities for preprocessing, augmentation and training of medical images. However, it does not leverage the power of specialized medical image processing libraries, such as SimpleITK [Lowekamp 13], to process volumetric images. Similar to DLTK, this library has not seen much activity since 2018.

The `batchgenerators` library [Isensee 20], used within the popular medical segmentation framework nn-UNet [Isensee 21], includes custom dataset and data loader classes for multithreaded loading of 3D medical images, implemented before data loaders were available in PyTorch. In the usage examples from GitHub,

preprocessing is applied to the whole dataset before training. Then, spatial data augmentation is performed at the volume level, from which one patch is extracted and intensity augmentation is performed at the patch level. In this approach, only one patch is extracted per volume, diminishing the efficiency of training pipelines. Transforms in `batchgenerators` are mostly implemented using NumPy [van der Walt 11] and SciPy [Virtanen 20].

More recently, a few PyTorch-based libraries for deep learning and medical images have appeared. There are two other libraries, developed in parallel to TorchIO, focused on data preprocessing and augmentation. `Rising`<sup>1</sup> is a library for data augmentation entirely written in PyTorch, which allows for gradients to be propagated through the transformations and perform all computations on the GPU. However, this means specialized medical imaging libraries such as SimpleITK cannot be used. `pymia` [Jungo 21] provides features for data handling (loading, preprocessing, sampling) and evaluation. It is compatible with TorchIO transforms, which are typically leveraged for data augmentation, as their data handling is more focused on preprocessing. `pymia` can be easily integrated into either PyTorch or TensorFlow pipelines. It was recently used to assess the suitability of evaluation metrics for medical image segmentation [Kofler 21].

MONAI [Ma 21] and Eisen [Mancolo 20] are PyTorch-based frameworks for deep learning workflows with medical images. Similar to NiftyNet and DLTK, they include implementation of network architectures, transforms, and higher-level features to perform training and inference. For example, MONAI was recently used for brain segmentation on fetal MRI [Ranzini 21]. As these packages are solving a large problem, i.e., that of workflow in deep learning for medical images, they do not contain all of the data augmentation transforms present in TorchIO. However, it is important to note that an end user does not need to select only one open-source package, as TorchIO transforms are compatible with both Eisen and MONAI.

TorchIO is a library that specializes in preprocessing and augmentation using

---

<sup>1</sup><https://github.com/PhoenixDL/rising>

PyTorch, focusing on ease of use for researchers. This is achieved by providing a PyTorch-like API, comprehensive documentation with many usage examples, and tutorials showcasing different features, and by actively addressing feature requests and bug reports from the many users that have already adopted TorchIO. This is in contrast with other modern libraries released after TorchIO such as MONAI, which aims to deliver a larger umbrella of functionalities including federated learning or active learning, but may have slower development and deployment.

## 5.2 Methods

We developed TorchIO, a Python library that focuses on data loading and augmentation of medical images in the context of deep learning.

TorchIO is a unified library to load and augment data that makes explicit use of medical image properties, and is flexible enough to be used for different loading workflows. It can accelerate research by avoiding the need to code a processing pipeline for medical images from scratch.

In contrast with Eisen or MONAI, we do not implement network architectures, loss functions or training workflows. This is to limit the scope of the library and to enforce modularity between training of neural networks and preprocessing and data augmentation.

Following the PyTorch philosophy [Paszke 19], we designed TorchIO with an emphasis on simplicity and usability while reusing PyTorch classes and infrastructure where possible. Although we designed TorchIO following PyTorch style, the library could also be used with other deep learning platforms such as TensorFlow or Keras<sup>2</sup>.

TorchIO makes use of open-source medical imaging software platforms. Packages were selected to reduce the number of required external dependencies and the need to re-implement basic medical imaging processing operations (image loading, resampling, etc.).

---

<sup>2</sup><https://keras.io>

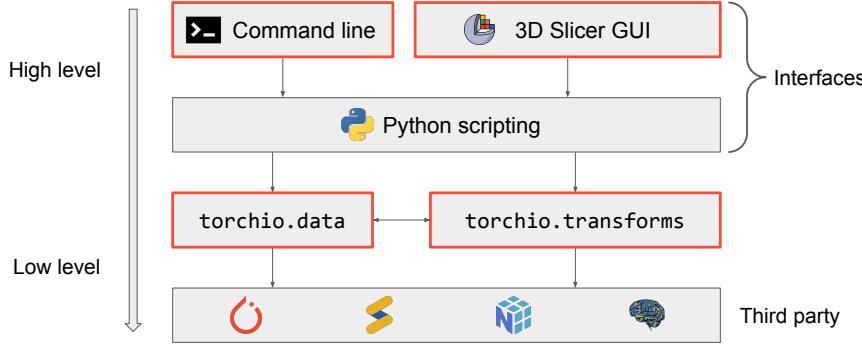


Figure 5.2: General diagram of TorchIO, its dependencies and its interfaces. Boxes with a red border (—) represent elements implemented in TorchIO. Logos indicate lower-level Python libraries used by TorchIO. : NiBabel [Brett 20]; : SimpleITK [Lowekamp 13]; : NumPy [van der Walt 11]; : PyTorch [Paszke 19].

TorchIO features are divided into two categories: data structures and input/output (`torchio.data`), and transforms for preprocessing and augmentation (`torchio.transforms`). Fig. 5.2 represents a diagram of the codebase and the different interfaces to the library.

### 5.2.1 Data

#### 5.2.1.1 Input/Output

TorchIO uses the medical imaging libraries NiBabel and SimpleITK to read and write images. Dependency on both is necessary to ensure broad support of image formats. For instance, NiBabel does not support reading Portable Network Graphics (PNG) files, while SimpleITK does not support some neuroimaging-specific formats.

TorchIO supports up to 4D images, i.e., 2D or 3D single-channel or multi-channel data such as X-rays, RGB histological slides, microscopy stacks, multispectral images, CT-scans functional MRI (fMRI) and diffusion MRI (dMRI).

#### 5.2.1.2 Data structures

**Image** The `Image` class, representing one medical image, stores a 4D tensor, whose voxels encode, e.g., signal intensity or segmentation labels, and the corresponding

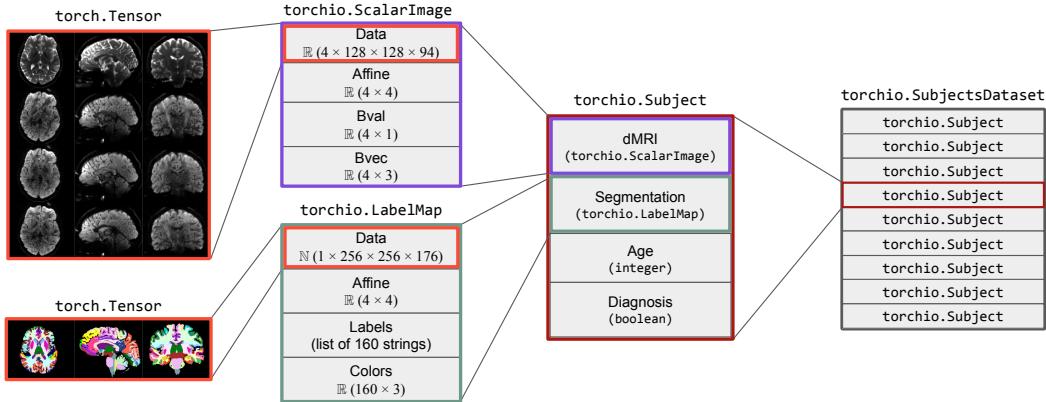


Figure 5.3: Usage example of `ScalarImage`, `LabelMap`, `Subject` and `SubjectsDataset`. The images store a 4D dMRI and a brain parcellation, and other related metadata.

affine transform, typically a rigid (Euclidean) transform, to convert voxel indices to world coordinates in millimeters. Arbitrary fields such as acquisition parameters may also be stored.

Subclasses are used to indicate specific types of images, such as `ScalarImage` and `LabelMap`, which are used to store, e.g., CT scans and segmentations, respectively.

An instance of `Image` can be created using a filepath, a PyTorch tensor, or a NumPy array. This class uses lazy loading, i.e., the data is not loaded from disk at instantiation time. Instead, the data is only loaded when needed for an operation (e.g., if a transform is applied to the image).

Fig. 5.3 shows two instances of `Image`. The instance of `ScalarImage` contains a 4D tensor representing a dMRI, which contains four 3D volumes (one per gradient direction), and the associated affine matrix. Additionally, it stores the strength and direction for each of the four gradients. The instance of `LabelMap` contains a brain parcellation of the same subject, the associated affine matrix, and the name and color of each brain structure.

**Subject** The `Subject` class stores instances of `Image` associated to a subject, e.g., a human or a mouse. As in the `Image` class, `Subject` can store arbitrary fields such

as age, diagnosis or ethnicity.

**Subjects dataset** The `SubjectsDataset` inherits from the PyTorch `Dataset`. It contains the list of subjects and optionally a transform to be applied to each subject after loading. When `SubjectsDataset` is queried for a specific subject, the corresponding set of images are loaded, a transform is applied to the images and the instance of `Subject` is returned.

For parallel loading, a PyTorch `DataLoader` may be used. This loader spawns multiple processes, each of which contains a shallow copy of the `SubjectsDataset`. Each copy is queried for a different subject, therefore loading and transforming is applied to different subjects in parallel on the central processing unit (CPU) (Fig. 5.4a).

An example of subclassing `SubjectsDataset` is `torchio.datasets.IXI`, which may be used to download the Information eXtraction from Images (IXI) dataset<sup>3</sup>.

### 5.2.1.3 Patch-based training

Memory limitations often require training and inference steps to be performed using image subvolumes or *patches* instead of the whole volumes, as explained in Section 5.1.1.1. In this section, we describe how TorchIO implements patch-based training via image sampling and queueing.

**Samplers** A sampler takes as input an instance of `Subject` and returns a version of it whose images have a reduced FOV, i.e., the new images are subvolumes, also called windows or *patches*. For this, a `PatchSampler` may be used.

Different criteria may be used to select the center voxel of each output patch. A `UniformSampler` selects a voxel as the center at random with all voxels having an equal probability of being selected. A `WeightedSampler` selects the patch center according to a probability distribution image defined over all voxels, which is passed

---

<sup>3</sup><https://brain-development.org/ixi-dataset/>

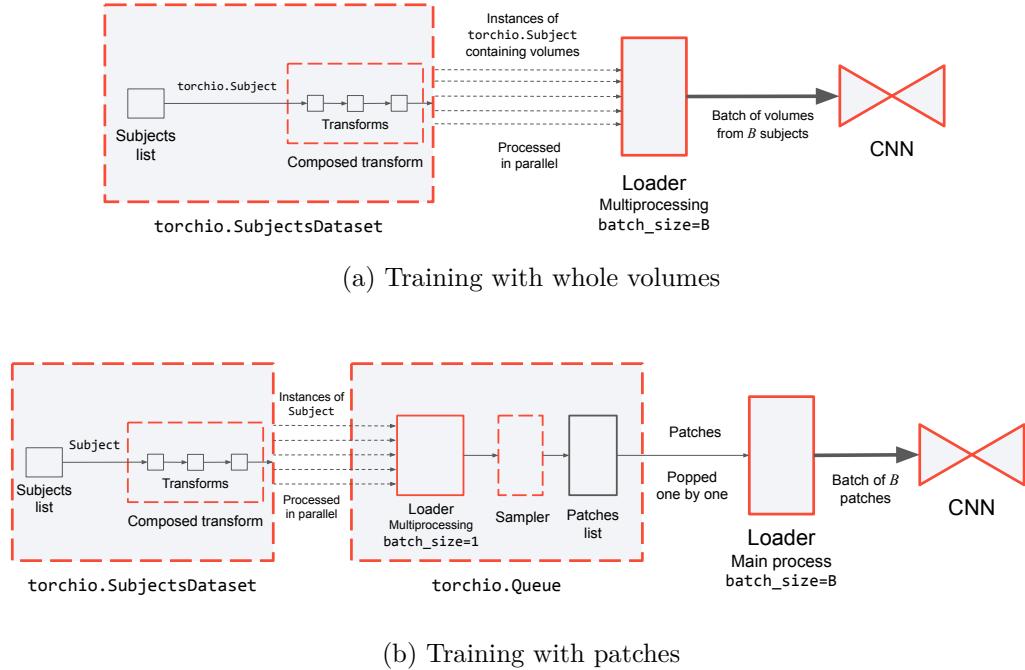


Figure 5.4: Diagram of data pipelines for training with whole volumes (top) and patches (bottom). Boxes with a red border represent PyTorch classes (—) or TorchIO classes that inherit from PyTorch classes (---).

as input to the sampler.

At testing time, images are sampled such that a dense inference can be performed on the input volume. A `GridSampler` can be used to sample patches such that the center voxel is selected using a set stride. In this way, sampling over the entire volume is ensured. The potentially-overlapping inferred patches can be passed to a `GridAggregator` that builds the resulting volume patch by patch (or batch by batch).

**Queue** A training iteration (i.e., forward and backward pass) performed on a GPU is usually faster than loading, preprocessing, augmenting, and cropping a volume on a CPU. Most preprocessing operations could be performed using a GPU, but these devices are typically reserved for training the CNN so that the batch size and input tensor can be as large as possible. Therefore, it is beneficial to prepare (i.e., load, preprocess and augment) the volumes using multiprocessing CPU techniques

in parallel with the forward-backward passes of a training iteration.

Once a volume is appropriately prepared, it is computationally beneficial to sample multiple patches from a volume rather than having to prepare the same volume each time a patch needs to be extracted. The sampled patches are then stored in a buffer or *queue* until the next training iteration, at which point they are loaded onto the GPU to perform an optimization iteration. For this, TorchIO provides the `Queue` class, which inherits from the PyTorch `Dataset` (Fig. 5.4b). In this queueing system, samplers behave as generators that yield patches from volumes contained in the `SubjectsDataset`.

The end of a training epoch is defined as the moment after which patches from all subjects have been used for training. At the beginning of each training epoch, the subjects list in the `SubjectsDataset` is shuffled, as is typically done in machine learning pipelines to increase variance of training instances during model optimization. A PyTorch loader queries the datasets copied in each subprocess, which load and process the volumes in parallel on the CPU. A PyTorch loader begins by shallow-copying the dataset to each subprocess. Each worker subprocess loads and applies image transforms to the volumes in parallel. A patches list is filled with patches extracted by the sampler, and the queue is shuffled once it has reached a specified maximum length so that batches are composed of patches from different subjects. The internal data loader continues querying the `SubjectsDataset` using multiprocessing. The patches list, when emptied, is refilled with new patches. A second data loader, external to the queue, may be used to collate batches of patches stored in the queue, which are passed to the neural network.

### 5.2.2 Transforms

The transforms API was designed to be similar to the PyTorch `torchvision.transforms` module. TorchIO includes augmentations such as random affine transformation (Fig. 5.5e) or random blur (Fig. 5.5b), but they are implemented using medical

imaging libraries [Lowekamp 13, Brett 20] to take into account specific properties of medical images, namely their size, resolution, location, and orientation (see Section 5.1.1.1). Table 5.1 shows transforms implemented in TorchIO v0.18.0 and their main corresponding library dependencies.

Table 5.1: Transforms included in TorchIO v0.18.0. Logos indicate the main library used to process the images. : NiBabel [Brett 20]; : SimpleITK [Lowekamp 13]; : NumPy [van der Walt 11]; : PyTorch [Paszke 19].

	Spatial	Intensity
Preprocessing	ToCanonical  Resample  Crop  Pad  CropOrPad 	HistogramStandardization  [Nyúl 99] RescaleIntensity  ZNormalization 
Augmentation	RandomAffine  RandomElasticDeformation  RandomFlip 	RandomMotion  [Shaw 19] RandomBiasField  [Sudre 17] RandomGhosting  RandomSpike  [Shaw 20] RandomBlur  RandomGamma  RandomNoise  RandomSwap  [Chen 19] RandomLabelsToImage  [Billot 20a] RandomAnisotropy  [Billot 20b]

Transforms are designed to be flexible regarding input and output types. Following a duck typing approach, they can take as input PyTorch tensors, SimpleITK images, NumPy arrays, Pillow images, Python dictionaries, and instances of `Subject` and `Image`, and will return an output of the same type.

TorchIO transforms can be classified into either spatial and intensity transforms, or preprocessing and augmentation transforms (Table 5.1). All are subclasses of the `Transform` base class. Spatial transforms and intensity transforms are related to the `SpatialTransform` and `IntensityTransform` classes, respectively. Transforms whose parameters are randomly chosen are subclasses of `RandomTransform`.

Instances of `SpatialTransform` typically modify the image bounds or spacing, and often need to resample the image using interpolation. They are applied to

all image types. Instances of `IntensityTransform` do not modify the position of voxels, only their values, and they are only applied to instances of `ScalarImage`. For example, if a `RandomNoise` transform (which is a subclass of `IntensityTransform`) receives as input a `Subject` with a `ScalarImage` representing a CT scan and a `LabelMap` representing a segmentation, it will add noise to only the CT scan. On the other hand, if a `RandomAffine` transform (which is a subclass of `SpatialTransform`) receives the same input, the same affine transformation will be applied to both images, with nearest-neighbor interpolation always used to interpolate `LabelMap` objects.

### 5.2.2.1 Preprocessing

Preprocessing transforms are necessary to ensure spatial and intensity uniformity of training instances.

Spatial preprocessing is important as CNNs do not generally take into account metadata related to medical images (see Section 5.1.1.1), therefore it is necessary to ensure that voxels across images have similar spatial location and relationships before training. Spatial preprocessing transforms typically used in medical imaging include resampling (e.g., to make voxel spacing isotropic for all training samples) and reorientation (e.g., to orient all training samples in the same way). For example, the `Resample` transform can be used to fix the issue presented in Fig. 5.1.

Intensity normalization is generally beneficial for optimization of neural networks. TorchIO provides intensity normalization techniques including min-max scaling or standardization<sup>4</sup>, which are computed using pure PyTorch. A binary image, such as a mask representing the foreground or structures of interest, can be used to define the set of voxels to be taken into account when computing statistics for intensity normalization. We also provide a method for MRI histogram standardization [Nyúl 00], computed using NumPy, which may be used to overcome the differences in

---

<sup>4</sup>In this context, standardization refers to correcting voxel intensity values to have zero mean and unit variance.

intensity distributions between images acquired using different scanners or sequences.

### 5.2.2.2 Augmentation

TorchIO includes spatial augmentation transforms such as random flipping using PyTorch and random affine and elastic deformation transforms using SimpleITK. Intensity augmentation transforms include random Gaussian blur using a SimpleITK filter (Fig. 5.5b) and addition of random Gaussian noise using pure PyTorch (Fig. 5.5d). All augmentation transforms are subclasses of `RandomTransform`.

Although current domain-specific data augmentation transforms available in TorchIO are mostly related to MRI, we encourage users to contribute physics-based data augmentation techniques for US or CT [Omigbodun 19].

We provide several MRI-specific augmentation transforms related to  $k$ -space, which are described below. An MR image is usually reconstructed as the magnitude of the inverse Fourier transform of the  $k$ -space signal, which is populated with the signals generated by the sample as a response to a radio-frequency electromagnetic pulse. These signals are modulated using coils that create gradients of the magnetic field inside the scanner. Artifacts are created by using  $k$ -space transforms to perturb the Fourier space and generate corresponding intensity artifacts in image space. The forward and inverse Fourier transforms are computed using the Fast Fourier Transform (FFT) algorithm implemented in NumPy.

**Random  $k$ -space spike artifact** Gradients applied at a very high duty cycle may produce bad data points, or noise spikes, in  $k$ -space [Zhuo 06]. These points in  $k$ -space generate a spike artifact, also known as Herringbone, crisscross or corduroy artifact, which manifests as uniformly-separated stripes in image space, as shown in Fig. 5.5i. This type of data augmentation has recently been used to estimate uncertainty through a heteroscedastic noise model [Shaw 20].

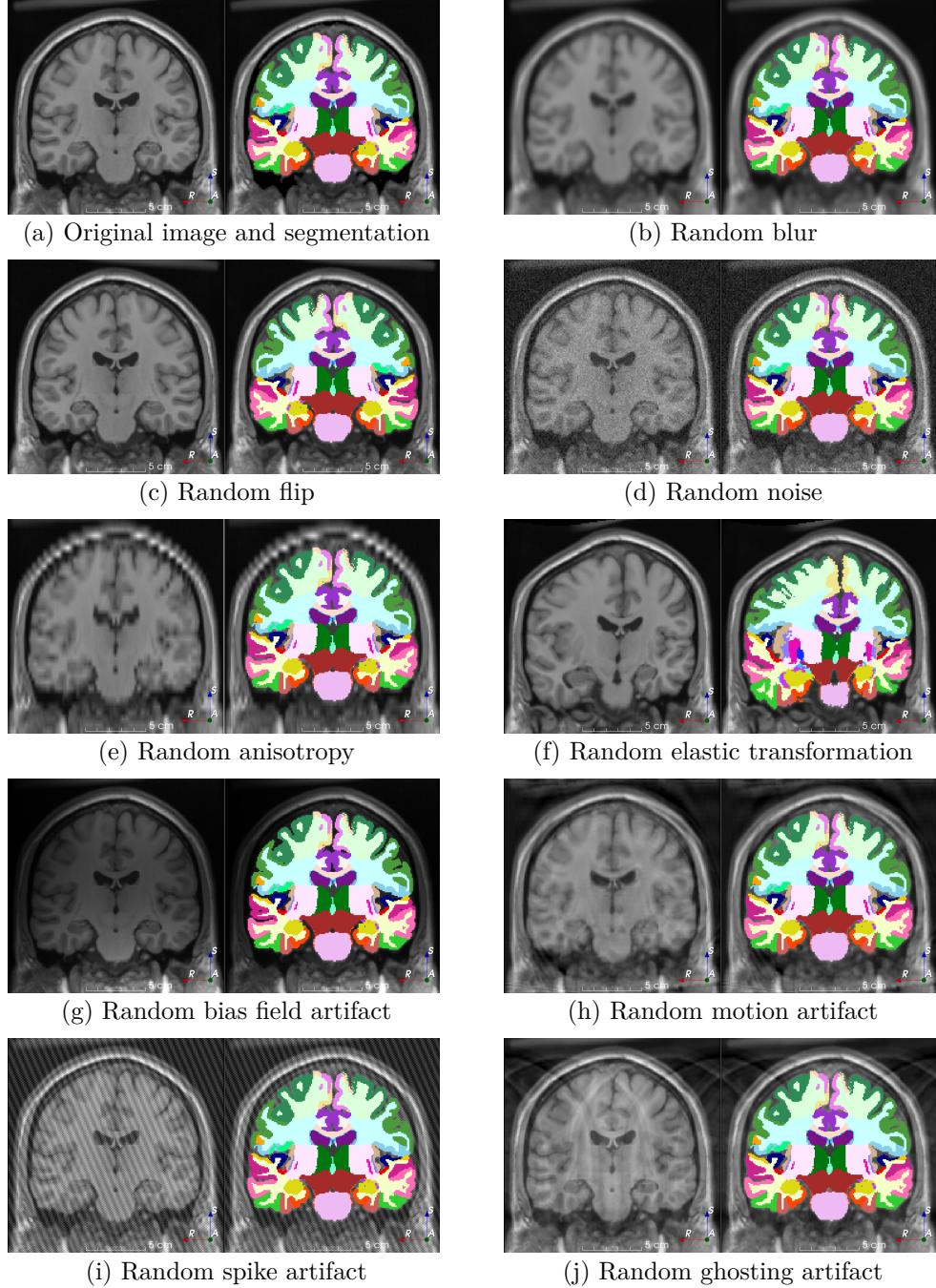


Figure 5.5: A selection of data augmentation techniques available in TorchIO v0.18.0. Each example is presented as a pair of images composed of the transformed image and a corresponding transformed label map. Note that all screenshots are from a 2D coronal slice of the transformed 3D images. The MRI corresponds to the Montreal Neurological Institute (MNI) Colin 27 average brain [Holmes 98], which can be downloaded using `torchio.datasets.Colin27`. Label maps were generated using an automated brain parcellation algorithm [Cardoso 15].

**Random  $k$ -space motion artifact** The  $k$ -space is often populated line by line, and the sample in the scanner is assumed to remain static. If a patient moves during the MRI acquisition, motion artifacts will appear in the reconstructed image. We implemented a method to simulate random motion artifacts (Fig. 5.5h) that has been used successfully for data augmentation to model uncertainty and improve segmentation [Shaw 19].

**Random  $k$ -space ghosting artifact** Organs motion such as respiration or cardiac pulsation may generate ghosting artifacts along the phase-encoding direction [Zhuo 06] (see Fig. 5.5j). We simulate this phenomenon by removing every  $n$ th plane of the  $k$ -space along one direction to generate  $n$  ghosts along that dimension, while keeping the center of  $k$ -space intact.

**Random bias field artifact** Inhomogeneity of the static magnetic field in the MRI scanner produces intensity artifacts of very low spatial frequency along the entirety of the image. These artifacts can be simulated using polynomial basis functions [Van Leemput 99], as shown in Fig. 5.5g.

### 5.2.2.3 Composability

All transforms can be composed in a linear fashion, as in the PyTorch `torchvision` library, or building a directed acyclic graph (DAG) using the `OneOf` transform (as in [Buslaev 20]). For example, a user might want to apply a random spatial augmentation transform to 50% of the samples using either an affine or an elastic transform, but they want the affine transform to be applied to 80% of the augmented images, as the execution time is faster. Then, they might want to rescale the volume intensity for all images to be between 0 and 1. Figure 5.6 shows a graph representing the transform composition. This transform composition can be implemented with just three statements:

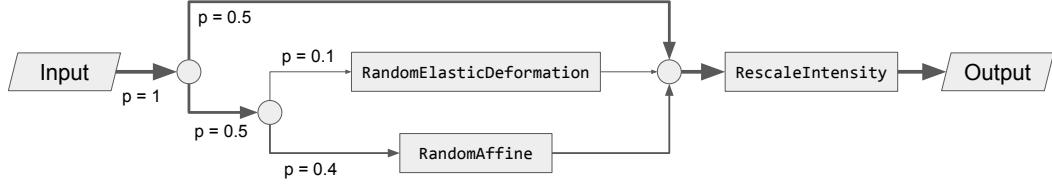


Figure 5.6: Graph representation of the composed transform described in Section 5.2.2.3.

---

```

import torchio as tio
spatial_transforms = {
    tio.RandomElasticDeformation(): 0.2,
    tio.RandomAffine(): 0.8,
}
transform = tio.Compose([
    tio.OneOf(spatial_transforms, p=0.5),
    tio.RescaleIntensity((0, 1)),
])

```

---

`Compose` and `OneOf` are implemented as TorchIO transforms.

#### 5.2.2.4 Extensibility

The `Lambda` transform can be passed an arbitrary callable object, which allows the user to augment the library with custom transforms without having a deep understanding of the underlying code.

Additionally, more complex transforms can be developed. For example, we implemented a TorchIO transform to simulate brain resection cavities (RCs) from preoperative MRIs within a self-supervised learning pipeline [Pérez-García 20e]. The `RandomLabelsToImage` transform may be used to simulate an image from a tissue segmentation. It can be composed with `RandomAnisotropy` to train neural networks agnostic to image contrast and resolution [Billot 20a, Billot 20b, Iglesias 20].

### 5.2.2.5 Reproducibility and traceability

To promote open science principles, we designed TorchIO to support experiment reproducibility and traceability.

All transforms support receiving Python primitives as arguments, which makes TorchIO suitable to be used with a configuration file associated to a specific experiment.

A history of all applied transforms and their computed random parameters is saved in the transform output so that the path in the DAG and the parameters used can be traced and reproduced. Furthermore, the `Subject` class includes a method to compose the transforms history into a single transform that may be used to reproduce the exact result (Section 5.2.2.3).

### 5.2.2.6 Invertibility

Inverting transforms is especially useful in scenarios where one needs to apply some transformation, infer a segmentation on the transformed data and then apply the inverse transformation to bring the inference into the original image space. The `Subject` class includes a method to invert the transformations applied. It does this by first inverting all transforms that are invertible, discarding the ones that are not. Then, it composes the invertible transforms into a single transform.

Transforms invertibility is most commonly applied to test-time augmentation [Moshkov 20] or estimation of aleatoric uncertainty [Wang 19a] in the context of image segmentation.

## 5.3 Results

### 5.3.1 Code availability

All the code for TorchIO is available on GitHub<sup>5</sup>. We follow the semantic versioning system [Preston-Werner 20] to tag and release our library. Releases are published on the Zenodo data repository<sup>6</sup> to allow users to cite the specific version of the package they used in their experiments. The version described in this paper is v0.18.0 [Pérez-García 20a]. Detailed API documentation is hosted on Read the Docs (Fig. 5.7) and comprehensive Jupyter notebook tutorials are hosted on Google Colaboratory, where users can run examples online. The library can be installed with a single line of code on Windows, macOS or Linux using the Pip Installs Packages (PIP) package manager: `pip install torchio`.

TorchIO has a strong community of users, with more than 1200 stars on GitHub and more than 3000 Python Package Index (PyPI) downloads per week<sup>7</sup> as of December 2021.

#### 5.3.1.1 Additional interfaces

The provided command-line interface (CLI) tool `torchio-transform` allows users to apply a transform to an image file without using Python. This tool can be used to visualize only the preprocessing and data augmentation pipelines and aid in experimental design for a given application. It can also be used in shell scripts to preprocess and augment datasets in cases where large storage is available and on-the-fly loading needs to be faster.

Additionally, we provide a graphical user interface (GUI) implemented as a Python scripted module within the *TorchIO* extension available in 3D Slicer [Fedorov 12]<sup>8</sup>. It can be used to visualize the effect of the transforms parameters without any coding

---

<sup>5</sup><https://github.com/fepegar/torchio>

<sup>6</sup><https://zenodo.org/>

<sup>7</sup><https://pypistats.org/packages/torchio>

<sup>8</sup><https://github.com/fepegar/SlicerTorchIO>

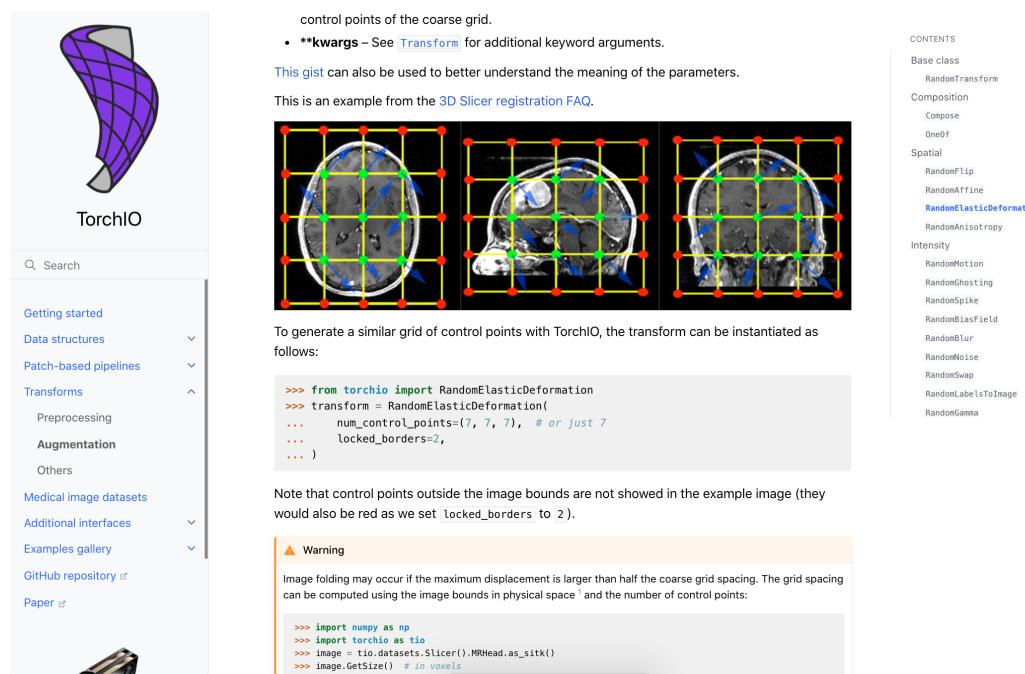


Figure 5.7: Screenshot of the documentation for TorchIO, available at <http://torchio.rtfd.io/>.

(Fig. 5.8). As with the CLI tool, users can experimentally assess preprocessing and data augmentation before network training to ensure the preprocessing pipeline is suitable for a given application.

### 5.3.2 Usage examples

In this section, we briefly describe the implementations of two medical image computing papers from the literature, pointing out the TorchIO features that could be used to replicate their experiments.

#### 5.3.2.1 Super-resolution and synthesis of MRI

In [Iglesias 20], a method is proposed to simulate high-resolution  $T_1$ -weighted MRIs from images of different modalities and resolutions.

First, brain regions are segmented on publicly available datasets of brain MRI. During training, an MRI (`ScalarImage`) and the corresponding segmentation (`LabelMap`) corresponding to a specific subject (`Subject`) are sampled from the

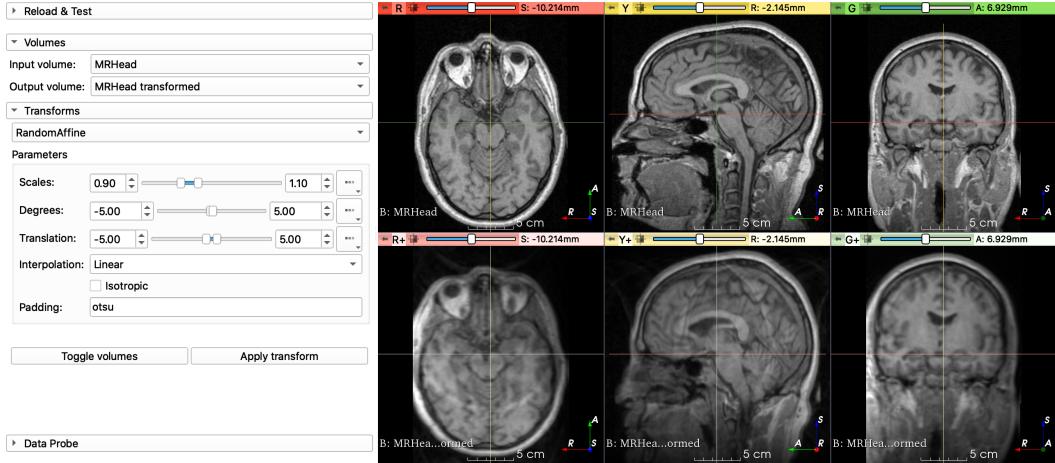


Figure 5.8: GUI for TorchIO, implemented as a 3D Slicer extension. In this example, the applied transforms are `RandomBiasField`, `RandomGhosting`, `RandomMotion`, `RandomAffine` and `RandomElasticDeformation`.

training dataset (`SubjectsDataset`). Next, the same spatial augmentation transform is applied to both images by composing an affine transform (`RandomAffine`) and a nonlinear diffeomorphic transform (`RandomElasticDeformation`). Then, a Gaussian mixture model (GMM) conditioned on the labels is sampled at each voxel location to simulate an MRI of arbitrary contrast (`RandomLabelsToImage`) [Billot 20a]. Finally, multiple degrading phenomena are simulated on the synthetic image: variability in the coordinate frames (`RandomAffine`), bias field inhomogeneities (`RandomBiasField`), partial-volume effects due to a large slice thickness during acquisition [Billot 20b] (`RandomAnisotropy`), registration errors (`RandomAffine`), and resampling artifacts (`Resample`).

### 5.3.2.2 Adaptive sampling for segmentation of CT scans

In [Berger 18], CT scans that are too large to fit on a GPU are segmented using patch-based training with weighted sampling of patches. Discrepancies between labels and predictions are used to create error maps and patches are preferentially sampled from voxels with larger error.

During training, a CT scan (`ScalarImage`) and its corresponding segmentation

(`LabelMap`) from a subject (`Subject`) are loaded and the same augmentation is performed to both by applying random rotations and scaling (`RandomAffine`). Then, voxel intensities are clipped to  $[-1000, 1000]$  (`RescaleIntensity`) and divided by a constant factor representing the standard deviation of the dataset (can be implemented with `Lambda`). As the CT scans are too large to fit in the GPU, patch-based training is used (`Queue`). To obtain high-resolution predictions and a large receptive field simultaneously, two patches of similar size but different FOV are generated from each sampled patch: a context patch generated by downsampling the original patch (`Resample`) and a full-resolution patch with a smaller FOV (`CropOrPad`). At the end of each epoch, error maps for each subject (`Subject`) are computed as the difference between the labels and predictions. The error maps are used in the following epoch to sample patches with large errors more often (`WeightedSampler`). At inference time, a sliding window (`GridSampler`) is used to predict the segmentation patch by patch, and patches are aggregated to build the prediction for the whole input volume (`GridAggregator`).

## 5.4 Discussion

We have presented TorchIO, a new library to efficiently load, preprocess, augment and sample medical imaging data during the training of CNNs. It is designed in the style of the deep learning framework PyTorch to provide medical imaging specific preprocessing and data augmentation algorithms.

The main motivation for developing TorchIO as an open-source toolkit is to help researchers standardize medical image processing pipelines and allow them to focus on the deep learning experiments. It also encourages good open-science practices, as it supports experiment reproducibility and is version-controlled so that the software can be cited precisely.

The library is compatible with other higher-level deep learning frameworks for medical imaging such as MONAI. For example, users can benefit from TorchIO’s

MRI transforms and patch-based sampling while using MONAI’s networks, losses, training pipelines and evaluation metrics.

The main limitation of TorchIO is that most transforms are not differentiable. The reason is that PyTorch tensors stored in TorchIO data structures must be converted to SimpleITK images or NumPy arrays within most transforms, making them not compatible with PyTorch’s automatic differentiation engine. However, compatibility between PyTorch and ITK has recently been improved, partly thanks to the appearance of the MONAI project [McCormick 21]. Therefore, TorchIO might provide differentiable transforms in the future, which could be used to implement, e.g., spatial transformer networks for image registration [Lee 19]. Another limitation is that many more transforms that are MRI-specific exist than for other imaging modalities such as CT or US. This is in part due to more users working on MRI applications and requesting MRI-specific transforms. However, we welcome contributions for other modalities as well.

In the future, we will work on extending the preprocessing and augmentation transforms to different medical imaging modalities such as CT or US, and improving compatibility with related works. The source code, as well as examples and documentation, are made publicly available online, on GitHub. We welcome feedback, feature requests, and contributions to the library, either by creating issues on the GitHub repository or by emailing the authors.

DRAFT – January 19, 2022

## CHAPTER 6

# Discussion

---

Epilepsy is a complex disorder with a complicated treatment pathway. Symptoms are strongly heterogeneous and may sometimes be interpreted differently by teams in different institutions. Decisions are often taken subjectively, based on clinicians' experience, who have seen a finite number of cases in their careers. Data-driven techniques, such as machine learning, may be used to take objective decisions based on large amounts of retrospective data. However, large datasets are rare in clinical settings, hindering the performance of approaches reliant on vast amounts of data.

The overall goal of this thesis is to identify elements in the current clinical pathway for the treatment of epilepsy that could be supported by data-driven computational methods, even when relatively small amounts of data are available.

Deep learning, the main set of techniques utilized in this thesis, has seen great success in the last decade, mostly thanks to improvements in hardware, such as graphics processing units (GPUs), and the vast amount of data generated every day. However, large datasets are typically not readily available in clinical settings because of privacy concerns and expensive annotation.

In this thesis, we leverage techniques such as transfer learning, self-supervised learning and semi-supervised learning to overcome these low-data regimes.

The specific directions for future work for the different chapters were described in Sections 2.4, 3.4, 4.4 and 5.4.

In the next section, I summarize the contributions presented in each chapter.

## 6.1 Contributions

### 6.1.1 Classification of epileptic seizure videos

Chapter 2 introduces Generalized Epileptic Seizure classification from video-Telemetry Using REcurrent convolutional neural networkS (GESTURES) our open-source framework for automatic classification of seizures from videos. I present a novel method combining convolutional and recurrent neural networks to model seizures of arbitrary duration. Our deep learning approach is robust to occlusions by bed linens and clinical staff, differences in illumination and pose, and poor video quality caused by compression artifacts or details out of focus [Pérez-García 21c]. The code is available on GitHub<sup>1</sup>.

Due to privacy concerns, we were not able to share the video data used in the study. However, a derived dataset of per-frame feature vectors with no identifiable patient data has been made open-access and it is freely available for download at the UCL Research Data Repository [Pérez-García 21b].

### 6.1.2 Visualization of the epileptogenic zone on brain images

Chapter 3 describes a piece of software developed in collaboration with neurologists Ali Alim-Marvasti and Gloria Romagnoli. My contributions to this project are 1) a 3D Slicer module [Fedorov 12] that reads the output of the querying tool and generates a 3D visualization on a parcellated brain magnetic resonance image (MRI), where the brightness associated to each brain structure is proportional to the probability of the epileptogenic zone (EZ) being in the structure; 2) the software engineering aspects of the project: a Pip Installs Packages (PIP)-installable Python package for the querying tool, including continuous integration (CI), and an application programming interface (API) to access the Python package from 3D Slicer or from the command line; and 3) the implementation of the online demo, which does not

---

<sup>1</sup><https://github.com/fepellar/gestures-miccai-2021>

require installing 3D Slicer<sup>2</sup>. The database and code are freely available on GitHub<sup>3</sup>. Our Semiology Visualization Tool (SVT) may be used to better understand the relation between the seizure onset zone and the observed seizure semiologies, and to perform an objective planning of intracranial EEG (iEEG) electrodes implantation.

### 6.1.3 Segmentation of postoperative brain resection cavities

Chapter 4 presents our framework for resective surgery quantification. I first describe our method to simulate resection cavities on normal (preoperative)  $T_1$ -weighted ( $T_{1w}$ ). The PIP-installable Python package `resector` is available on GitHub<sup>4</sup> and has recently been used in the context of brain tumor segmentation by researchers in China and the UK [Zhang 21].

The resection simulator was used to train a cavity segmentation model without manual annotations, obtaining a performance comparable to human inter-rater variability [Pérez-García 20e]. A PIP-installable command-line interface (CLI) tool to segment resection cavities, `resseg`, is available online<sup>5</sup>. A convenient 3D Slicer module that uses `resseg` is available as a graphical user interface (GUI) for users without coding experience<sup>6</sup>. A 3D Slicer extension for the deep learning framework PyTorch was created in the context of this project<sup>7</sup>. It can be installed from the built-in 3D Slicer Extensions Manager.

In the context of this project, we curated the EPISURG dataset, comprising 699 MRIs from 430 patients who underwent epilepsy surgery at the National Hospital for Neurology and Neurosurgery (NHNN) between 1990 and 2018, including 200 manual annotations from three different human raters (133 of which were performed by myself). To the best of our knowledge, EPISURG is the first open annotated database of post-resection MRI for epilepsy patients. EPISURG is an open-access

---

<sup>2</sup><https://github.com/fepegar/SVT-web>

<sup>3</sup><https://github.com/fepegar/Semiology-Visualisation-Tool>

<sup>4</sup><https://github.com/fepegar/resector>

<sup>5</sup><https://github.com/fepegar/resseg>

<sup>6</sup><https://github.com/fepegar/SlicerParcellation#brain-resection-cavity-segmentation>

<sup>7</sup><https://github.com/fepegar/SlicerPyTorch>

dataset and can be freely downloaded from the UCL Research Data Repository [Pérez-García 20d]. A 3D Slicer extension to download and visualize EPISURG is available on GitHub<sup>8</sup>.

We collaborated with hospitals in Milan, Paris and Marseille to validate our segmentation framework using heterogeneous clinical data [Pérez-García 21a]. The code is available on GitHub<sup>9</sup>.

#### 6.1.4 TorchIO: a software library for medical image processing

Chapter 5 describes our open-source Python library TorchIO [Pérez-García 21e], which was initially developed in the context of the work presented in Chapter 4. The documentation<sup>10</sup> and code<sup>11</sup> are available online. A 3D Slicer extension to use TorchIO without the need to code can be installed from the Extensions Manager<sup>12</sup>.

---

<sup>8</sup><https://github.com/fepegar/SlicerEPISURG>

<sup>9</sup><https://github.com/fepegar/ressseg-ijcars>

<sup>10</sup><http://torchio.rtfd.io/>

<sup>11</sup><https://github.com/fepegar/torchio>

<sup>12</sup><https://github.com/fepegar/SlicerTorchIO>

# Bibliography

---

- [Abadi 16] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. arXiv:1603.04467 [cs], March 2016. arXiv: 1603.04467. 78, 90, 95
- [Ahmedt-Aristizabal 17] David Ahmedt-Aristizabal, Clinton Fookes, Sasha Dionisio, Kien Nguyen, João Paulo S. Cunha and Sridha Sridharan. *Automated analysis of seizure semiology and brain electrical activity in presurgery evaluation of epilepsy: A focused survey*. Epilepsia, volume 58, number 11, pages 1817–1831, 2017. 34
- [Ahmedt-Aristizabal 18a] David Ahmedt-Aristizabal, Clinton Fookes, Simon Denman, Kien Nguyen, Tharindu Fernando, Sridha Sridharan and Sasha Dionisio. *A hierarchical multimodal system for motion analysis in patients with epilepsy*. Epilepsy & Behavior, volume 87, pages 46–58, October 2018. 35
- [Ahmedt-Aristizabal 18b] David Ahmedt-Aristizabal, Clinton Fookes, Kien Nguyen, Simon Denman, Sridha Sridharan and Sasha Dionisio. *Deep facial analysis: A new phase I epilepsy evaluation using computer vision*. Epilepsy & Behavior, volume 82, pages 17–24, May 2018. 35
- [Ahmedt-Aristizabal 18c] David Ahmedt-Aristizabal, Kien Nguyen, Simon Denman, Sridha Sridharan, Sasha Dionisio and Clinton Fookes. *Deep*

- Motion Analysis for Epileptic Seizure Classification.* In 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pages 3578–3581, July 2018. ISSN: 1557-170X. 35
- [Alexandre 15] Veriano Alexandre, Blanca Mercedes, Luc Valton, Louis Maillard, Fabrice Bartolomei, William Szurhaj, Edouard Hirsch, Cécile Marchal, Francine Chassoux, Jérôme Petit, Arielle Crespel, Anca Nica, Vincent Navarro, Philippe Kahane, Bertrand De Toffol, Pierre Thomas, Sarah Rosenberg, Marie Denuelle, Jacques Jonas, Philippe Ryvlin, Sylvain Rheims and REPO2MSE study group. *Risk factors of postictal generalized EEG suppression in generalized convulsive seizures.* Neurology, volume 85, number 18, pages 1598–1603, November 2015. 26
- [Alim-Marvasti 21a] Ali Alim-Marvasti, Gloria Romagnoli, Karan Dahele, Hadi Modarres, **Fernando Pérez-García**, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson, Fahmida Chowdhury, Beate Diehl and John S. Duncan. *Probabilistic Landscape of Seizure Semiology Localising Values.* Under submission (Brain Communications), 2021. 49, 56
- [Alim-Marvasti 21b] Ali Alim-Marvasti, **Fernando Pérez-García**, Karan Dahele, Gloria Romagnoli, Beate Diehl, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson and John S. Duncan. *Machine Learning for Localizing Epileptogenic-Zone in the Temporal Lobe: Quantifying the Value of Multimodal Clinical-Semiology and Imaging Concordance.* Frontiers in Digital Health, volume 3, page 8, 2021.
- [Alim-Marvasti 21c] Ali Alim-Marvasti, **Fernando Pérez-García**, Gloria Romagnoli, Peter N. Taylor, Vejay Vakharia, Rachel Sparks, Sébastien Ourselin, Matthew J. Clarkson and John S. Duncan. *Mapping Epileptic Symptomatology to Cortical Epileptogenicity in MNI Space: Seizure Semiology-to-Brain Visualisation Tool.* Under submission (NeuroImage: Clinical), 2021. 49, 53, 54, 56
- [Berger 18] Lorenz Berger, Hyde Eoin, M. Jorge Cardoso and Sébastien Ourselin. *An Adaptive Sampling Scheme to Efficiently Train Fully Convolutional Networks for Semantic Segmentation.* In

- Mark Nixon, Sasan Mahmoodi and Reyer Zwiggelaar, editors, *Medical Image Understanding and Analysis*, Communications in Computer and Information Science, pages 277–286, Cham, 2018. Springer International Publishing. 113
- [Bien 09] Christian G. Bien, Miriam Szinay, Jan Wagner, Hans Clusmann, Albert J. Becker and Horst Urbach. *Characteristics and Surgical Outcomes of Patients With Refractory Magnetic Resonance Imaging–Negative Epilepsies*. Archives of Neurology, volume 66, number 12, pages 1491–1499, December 2009. 27
- [Billot 20a] Benjamin Billot, Douglas N. Greve, Koen Van Leemput, Bruce Fischl, Juan Eugenio Iglesias and Adrian Dalca. *A Learning Strategy for Contrast-agnostic MRI Segmentation*. In Medical Imaging with Deep Learning, pages 75–93. PMLR, September 2020. ISSN: 2640-3498. 87, 104, 109, 113
- [Billot 20b] Benjamin Billot, Eleanor Robinson, Adrian V. Dalca and Juan Eugenio Iglesias. *Partial Volume Segmentation of Brain MRI Scans of Any Resolution and Contrast*. In Medical Image Computing and Computer Assisted Intervention – MICCAI 2020, Lecture Notes in Computer Science, pages 177–187, Cham, 2020. Springer International Publishing. 79, 104, 109, 113
- [Brett 01] Matthew Brett, Alexander P. Leff, Chris Rorden and John Ashburner. *Spatial Normalization of Brain Images with Focal Lesions Using Cost Function Masking*. NeuroImage, volume 14, number 2, pages 486–500, August 2001. 87
- [Brett 20] Matthew Brett, Christopher J. Markiewicz, Michael Hanke, Marc-Alexandre Côté, Ben Cipollini, Paul McCarthy, Christopher P. Cheng, Yaroslav O. Halchenko, Michiel Cottaar, Satrajit Ghosh, Eric Larson, Demian Wassermann, Stephan Gerhard, Gregory R. Lee, Hao-Ting Wang, Erik Kastman, Ariel Rokem, Cindee Madison, Félix C. Morency, Brendan Moloney, Mathias Goncalves, Cameron Riddell, Christopher Burns, Jarrod Millman, Alexandre Gramfort, Jaakkko Leppäkangas, Ross Markello, Jasper J.F. van den Bosch, Robert D. Vincent, Henry

Braun, Krish Subramaniam, Dorota Jarecka, Krzysztof J. Gorgolewski, Pradeep Reddy Raamana, B. Nolan Nichols, Eric M. Baker, Soichi Hayashi, Basile Pinsard, Christian Haselgrove, Mark Hymers, Oscar Esteban, Serge Koudoro, Nikolaas N. Oosterhof, Bago Amirbekian, Ian Nimmo-Smith, Ly Nguyen, Samir Reddigari, Samuel St-Jean, Egor Panfilov, Eleftherios Garyfallidis, Gael Varoquaux, Jakub Kaczmarzyk, Jon Haitz Legarreta, Kevin S. Hahn, Oliver P. Hinds, Bennet Fauber, Jean-Baptiste Poline, Jon Stutters, Kesshi Jordan, Matthew Cieslak, Miguel Estevan Moreno, Valentin Haenel, Yannick Schwartz, Benjamin C Darwin, Bertrand Thirion, Dimitri Papadopoulos Orfanos, **Fernando Pérez-García**, Igor Solovey, Ivan Gonzalez, Jath Palasubramaniam, Justin Lecher, Katrin Leinweber, Konstantinos Raktivan, Peter Fischer, Philippe Gervais, Syam Gadde, Thomas Ballinger, Thomas Roos, Venkateswara Reddy Reddam and freec84. *nipy/nibabel: 3.0.1*, January 2020. 92, 99, 104

- [Buslaev 20] Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin and Alexandr A. Kalinin. *Albumentations: Fast and Flexible Image Augmentations*. Information, volume 11, number 2, page 125, February 2020. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. 91, 93, 108
- [Bussonnier 18] Matthias Bussonnier, J. Forde, Jeremy Freeman, B. Granger, T. Head, Chris Holdgraf, Kyle Kelley, Gladys Nalvarte, Andrew Osherooff, M. Pacer, Yuvi Panda, **Fernando Pérez**, B. Ragan-Kelley and Carol Willing. *Binder 2.0 - Reproducible, interactive, sharable environments for science at scale*. 2018. 52
- [Cardoso 15] M. Jorge Cardoso, Marc Modat, Robin Wolz, Andrew Melbourne, David Cash, Daniel Rueckert and Sébastien Ourselin. *Geodesic Information Flows: Spatially-Variant Graphs and Their Application to Segmentation and Fusion*. IEEE Trans Med Imaging, volume 34, number 9, pages 1976–1988, September 2015. 52, 57, 59, 72, 92, 107

- [Carreira 17] J. Carreira and A. Zisserman. *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset*. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4724–4733, July 2017. ISSN: 1063-6919. 35
- [Chen 15] Kanglin Chen, Alexander Derksen, Stefan Heldmann, Marc Hallmann and Benjamin Berkels. *Deformable Image Registration with Automatic Non-Correspondence Detection*. In Jean-François Aujol, Mila Nikolova and Nicolas Papadakis, editors, Scale Space and Variational Methods in Computer Vision, Lecture Notes in Computer Science, pages 360–371, Cham, 2015. Springer International Publishing. 62
- [Chen 17] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy and Alan L. Yuille. *DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*. arXiv:1606.00915 [cs], May 2017. arXiv: 1606.00915. 78
- [Chen 18] Feiyu Chen, Valentina Taviani, Itzik Malkiel, Joseph Y. Cheng, Jonathan I. Tamir, Jamil Shaikh, Stephanie T. Chang, Christopher J. Hardy, John M. Pauly and Shreyas S. Vasanawala. *Variable-Density Single-Shot Fast Spin-Echo MRI with Deep Learning Reconstruction by Using Variational Networks*. Radiology, volume 289, number 2, pages 366–373, July 2018. 90
- [Chen 19] Liang Chen, Paul Bentley, Kensaku Mori, Kazunari Misawa, Michitaka Fujiwara and Daniel Rueckert. *Self-supervised learning for medical image analysis using image context restoration*. Medical Image Analysis, volume 58, page 101539, December 2019. 63, 104
- [Chen 20] Ting Chen, Simon Kornblith, Mohammad Norouzi and Geoffrey Hinton. *A Simple Framework for Contrastive Learning of Visual Representations*. In International Conference on Machine Learning, pages 1597–1607. PMLR, November 2020. ISSN: 2640-3498. 94
- [Cheplygina 19] Veronika Cheplygina. *Cats or CAT scans: Transfer learning from natural or medical image source data sets?* Current

- Opinion in Biomedical Engineering, volume 9, pages 21–27, March 2019. 95
- [Chitphakdithai 10] Nicha Chitphakdithai and James S. Duncan. *Non-rigid Registration with Missing Correspondences in Preoperative and Postresection Brain Images*. In Tianzi Jiang, Nassir Navab, Josien P. W. Pluim and Max A. Viergever, editors, MICCAI 2010, Lecture Notes in Computer Science, pages 367–374, Berlin, Heidelberg, 2010. Springer. 62
- [Cicek 16] Ozgun Cicek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox and Olaf Ronneberger. *3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation*. In Sebastien Ourselin, Leo Joskowicz, Mert R. Sabuncu, Gozde Unal and William Wells, editors, Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016, Lecture Notes in Computer Science, pages 424–432, Cham, 2016. Springer International Publishing. 77, 89, 94, 96
- [Cuadrado-Godia 18] Elisa Cuadrado-Godia, Pratistha Dwivedi, Sanjiv Sharma, Angel Ois Santiago, Jaume Roquer Gonzalez, Mercedes Balcells, John Laird, Monika Turk, Harman S. Suri, Andrew Nicolaides, Luca Saba, Narendra N. Khanna and Jasjit S. Suri. *Cerebral Small Vessel Disease: A Review Focusing on Pathophysiology, Biomarkers, and Machine Learning Strategies*. Journal of Stroke, volume 20, number 3, pages 302–320, September 2018. 87
- [Cunha 03] J. P. S. Cunha, C. Vollmar, Z. Li, J. Fernandes, B. Feddersen and S. Noachtar. *Movement quantification during epileptic seizures: a new technical contribution to the evaluation of seizure semiology*. In Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No.03CH37439), volume 1, pages 671–673 Vol.1, September 2003. ISSN: 1094-687X. 34
- [Cunha 16] João Paulo Silva Cunha, Hugo Miguel Pereira Choupina, Ana Patrícia Rocha, José Maria Fernandes, Felix Achilles, Anna Mira Loesch, Christian Vollmar, Elisabeth Hartl and Soheyl Noachtar. *NeuroKinect: A Novel Low-Cost 3Dvideo-EEG*

- System for Epileptic Seizure Motion Quantification.* PLOS ONE, volume 11, number 1, page e0145669, January 2016. 34
- [Deng 09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Li Fei-Fei. *ImageNet: A large-scale hierarchical image database.* In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 248–255, June 2009. ISSN: 1063-6919. 95
- [Desikan 06] Rahul S. Desikan, Florent Ségonne, Bruce Fischl, Brian T. Quinn, Bradford C. Dickerson, Deborah Blacker, Randy L. Buckner, Anders M. Dale, R. Paul Maguire, Bradley T. Hyman, Marilyn S. Albert and Ronald J. Killiany. *An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest.* NeuroImage, volume 31, number 3, pages 968–980, July 2006. 58
- [Devinsky 16] Orrin Devinsky, Dale C. Hesdorffer, David J. Thurman, Samden Lhatoo and George Richerson. *Sudden unexpected death in epilepsy: epidemiology, mechanisms, and prevention.* The Lancet. Neurology, volume 15, number 10, pages 1075–1088, September 2016. 26
- [Dorent 21] Reuben Dorent, Thomas Booth, Wenqi Li, Carole H. Sudre, Sina Kafriabadi, Jorge Cardoso, Sébastien Ourselin and Tom Vercauteren. *Learning joint segmentation of tissues and brain lesions from task-specific hetero-modal domain-shifted datasets.* Medical Image Analysis, volume 67, page 101862, January 2021. 63
- [Duncan 16] John S Duncan, Gavin P Winston, Matthias J Koepp and Sébastien Ourselin. *Brain imaging in the assessment for epilepsy surgery.* The Lancet Neurology, volume 15, number 4, pages 420–433, April 2016. 27, 37
- [Engel 16] Jerome Engel. *What can we do for people with drug-resistant epilepsy?* Neurology, volume 87, number 23, pages 2483–2489, December 2016. 48
- [Ermiş 20] Ekin Ermiş, Alain Jungo, Robert Poel, Marcela Blatti-Moreno, Raphael Meier, Urspeter Knecht, Daniel M. Aebersold, Michael K. Fix, Peter Manser, Mauricio Reyes and Evelyn

- Herrmann. *Fully automated brain resection cavity delineation for radiation target volume definition in glioblastoma patients using deep learning.* Radiation Oncology, volume 15, number 1, page 100, May 2020. 62, 63
- [Fedorov 12] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, John Buatti, Stephen Aylward, James V. Miller, Steve Pieper and Ron Kikinis. *3D Slicer as an Image Computing Platform for the Quantitative Imaging Network.* Magnetic resonance imaging, volume 30, number 9, pages 1323–1341, November 2012. 47, 52, 76, 92, 111, 118
- [Fiest 17] Kirsten M. Fiest, Khara M. Sauro, Samuel Wiebe, Scott B. Patten, Churl-Su Kwon, Jonathan Dykeman, Tamara Pringsheim, Diane L. Lorenzetti and Nathalie Jetté. *Prevalence and incidence of epilepsy: A systematic review and meta-analysis of international studies.* Neurology, volume 88, number 3, pages 296–303, January 2017. 25
- [Fisher 17] Robert S. Fisher, J. Helen Cross, Jacqueline A. French, Norimichi Higurashi, Edouard Hirsch, Floor E. Jansen, Lieven Lagae, Solomon L. Moshé, Jukka Peltola, Eliane Roulet Perez, Ingrid E. Scheffer and Sameer M. Zuberi. *Operational classification of seizure types by the International League Against Epilepsy: Position Paper of the ILAE Commission for Classification and Terminology.* Epilepsia, volume 58, number 4, pages 522–530, 2017. 33, 34
- [Fonov 09] Vladimir Fonov, Alan Evans, Robert McKinstry, C.R. Almlie and Louis Collins. *Unbiased nonlinear average age-appropriate brain templates from birth to adulthood.* Neuroimage, volume 47, July 2009. 52, 77
- [Fonov 11] Vladimir Fonov, Alan C. Evans, Kelly Botteron, C. Robert Almlie, Robert C. McKinstry and D. Louis Collins. *Unbiased average age-appropriate atlases for pediatric studies.* NeuroImage, volume 54, number 1, pages 313–327, January 2011. 77

- [Forsgren 05] Lars Forsgren, W. Allen Hauser, Elias Olafsson, J. W. a. S. Sander, Matti Sillanpää and Torbjörn Tomson. *Mortality of epilepsy in developed countries: a review*. Epilepsia, volume 46 Suppl 11, pages 18–27, 2005. 25
- [Ghadiyaram 19] Deepti Ghadiyaram, Matt Feiszli, Du Tran, Xuetong Yan, Heng Wang and Dhruv Mahajan. *Large-scale weakly-supervised pre-training for video action recognition*. arXiv:1905.00561 [cs], May 2019. arXiv: 1905.00561. 35, 36, 40
- [Gibson 18] Eli Gibson, Wenqi Li, Carole Sudre, Lucas Fidon, Dzhoshkun I. Shakir, Guotai Wang, Zach Eaton-Rosen, Robert Gray, Tom Doel, Yipeng Hu, Tom Whyntie, Parashkev Nachev, Marc Modat, Dean C. Barratt, Sébastien Ourselin, M. Jorge Cardoso and Tom Vercauteren. *NiftyNet: a deep-learning platform for medical imaging*. Computer Methods and Programs in Biomedicine, volume 158, pages 113–122, May 2018. 90, 96
- [Granados 21] Alejandro Granados, **Fernando Pérez-García**, Martin Schweiger, Vejay Vakharia, Sjoerd B. Vos, Anna Miserocchi, Andrew W. McEvoy, John S. Duncan, Rachel Sparks and Sébastien Ourselin. *A generative model of hyperelastic strain energy density functions for multiple tissue brain deformation*. International Journal of Computer Assisted Radiology and Surgery, volume 16, number 1, pages 141–150, January 2021. 87
- [Greff 17] Klaus Greff, Aaron Klein, Martin Chovanec, Frank Hutter and Jürgen Schmidhuber. *The Sacred Infrastructure for Computational Research*. Proceedings of the 16th Python in Science Conference, pages 49–56, 2017. 78
- [Gudbjartsson 95] Hákon Gudbjartsson and Samuel Patz. *The Rician Distribution of Noisy MRI Data*. Magnetic resonance in medicine, volume 34, number 6, pages 910–914, December 1995. 72
- [Havaei 17] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin and Hugo Larochelle. *Brain tumor segmentation with Deep Neural Networks*. Medical Image Analysis, volume 35, pages 18–31, January 2017. 29

- [He 15] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification.* arXiv:1502.01852 [cs], February 2015. 78
- [He 19] Horace He. *The State of Machine Learning Frameworks in 2019*, October 2019. 90, 95
- [Holmes 98] C. J. Holmes, R. Hoge, L. Collins, R. Woods, A. W. Toga and A. C. Evans. *Enhancement of MR images using registration for signal averaging.* Journal of Computer Assisted Tomography, volume 22, number 2, pages 324–333, April 1998. 107
- [Hutchinson 20] M. Hutchinson, S. Samsi, W. Arcand, D. Bestor, B. Bergeron, C. Byun, M. Houle, M. Hubbell, M. Jones, J. Kepner, A. Kirby, P. Michaleas, L. Milechin, J. Mullen, A. Prout, A. Rosa, A. Reuther, C. Yee and V. Gadepally. *Accuracy and Performance Comparison of Video Action Recognition Approaches.* In 2020 IEEE High Performance Extreme Computing Conference (HPEC), pages 1–8, September 2020. ISSN: 2643-1971. 40
- [Iglesias 11] Juan Eugenio Iglesias, Cheng-Yi Liu, Paul M. Thompson and Zhuowen Tu. *Robust brain extraction across datasets and comparison with publicly available methods.* IEEE Trans Med Imaging, volume 30, number 9, pages 1617–1634, September 2011. 77
- [Iglesias 20] Juan Eugenio Iglesias, Benjamin Billot, Yael Balbastre, Azadeh Tabari, John Conklin, Daniel Alexander, Polina Goliland, Brian Edlow and Bruce Fischl. *Joint super-resolution and synthesis of 1 mm isotropic MP-RAGE volumes from clinical MRI exams with scans of different orientation, resolution and contrast.* arXiv preprint arXiv:2012.13340, 2020. 109, 112
- [Ioffe 15] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.* In International Conference on Machine Learning, pages 448–456. PMLR, June 2015. ISSN: 1938-7228. 78, 94
- [Isensee 20] Fabian Isensee, Paul Jäger, Jakob Wasserthal, David Zimmerer, Jens Petersen, Simon Kohl, Justus Schock, Andre

- Klein, Tobias Roß, Sebastian Wirkert, Peter Neher, Stefan Dinkelacker, Gregor Köhler and Klaus Maier-Hein. *batchgenerators - a python framework for data augmentation*, January 2020. 96
- [Isensee 21] Fabian Isensee, Paul F. Jaeger, Simon A. A. Kohl, Jens Petersen and Klaus H. Maier-Hein. *nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation*. Nature Methods, volume 18, number 2, pages 203–211, February 2021. Number: 2 Publisher: Nature Publishing Group. 96
- [Jack 08] Clifford R. Jack, Matt A. Bernstein, Nick C. Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J. Britson, Jennifer L Whitwell, Chadwick Ward, Anders M. Dale, Joel P. Felmlee, Jeffrey L. Gunter, Derek L. G. Hill, Ron Killiany, Norbert Schuff, Sabrina Fox-Bosetti, Chen Lin, Colin Studholme, Charles S. DeCarli, Gunnar Krueger, Heidi A. Ward, Gregory J. Metzger, Katherine T. Scott, Richard Mallozzi, Daniel Blezek, Joshua Levy, Josef P. Debbins, Adam S. Fleisher, Marilyn Albert, Robert Green, George Bartzokis, Gary Glover, John Mugler and Michael W. Weiner. *The Alzheimer’s Disease Neuroimaging Initiative (ADNI): MRI methods*. Journal of magnetic resonance imaging: JMRI, volume 27, number 4, pages 685–691, April 2008. 74
- [Jenssen 06] Sigmund Jenssen, Edward J. Gracely and Michael R. Sperling. *How Long Do Most Seizures Last? A Systematic Comparison of Seizures Recorded in the Epilepsy Monitoring Unit*. Epilepsia, volume 47, number 9, pages 1499–1503, 2006. 35, 37
- [Jha 21] Ashwani Jha, Cheongeun Oh, Dale Hesdorffer, Beate Diehl, Sasha Devore, Martin J. Brodie, Torbjörn Tomson, Josemir W. Sander, Thaddeus S. Walczak and Orrin Devinsky. *Sudden Unexpected Death in Epilepsy: A Personalized Prediction Tool*. Neurology, volume 96, number 21, pages e2627–e2638, May 2021. Publisher: Wolters Kluwer Health, Inc. on behalf of the American Academy of Neurology Section: Article. 26

- [Jobst 15] Barbara C. Jobst and Gregory D. Cascino. *Resective epilepsy surgery for drug-resistant focal epilepsy: a review.* JAMA, volume 313, number 3, pages 285–293, January 2015. 27, 28, 29, 62
- [Jung 20] Alexander B. Jung, Kentaro Wada, Jon Crall, Satoshi Tanaka, Jake Graving, Christoph Reinders, Sarthak Yadav, Joy Banerjee, Gábor Vecsei, Adam Kraft, Zheng Rui, Jirka Borovec, Christian Vallentin, Semen Zhydenko, Kilian Pfeiffer, Ben Cook, Ismael Fernández, François-Michel De Rainville, Chi-Hung Weng, Abner Ayala-Acevedo, Raphael Meudec, Matias Laporte and others. *imgaug*, 2020. 93
- [Jungo 20] Alain Jungo, Fabian Balsiger and Mauricio Reyes. *Analyzing the Quality and Challenges of Uncertainty Estimations for Brain Tumor Segmentation.* Frontiers in Neuroscience, volume 14, 2020. Publisher: Frontiers. 68
- [Jungo 21] Alain Jungo, Olivier Scheidegger, Mauricio Reyes and Fabian Balsiger. *pymia: A Python package for data handling and evaluation in deep learning-based medical image analysis.* Computer Methods and Programs in Biomedicine, volume 198, page 105796, January 2021. 97
- [Kabat 12] Joanna Kabat and Przemysław Król. *Focal cortical dysplasia – review.* Polish Journal of Radiology, volume 77, number 2, pages 35–43, 2012. 27
- [Karácsony 20] T. Karácsony, A. M. Loesch-Biffar, C. Vollmar, S. Noachtar and J. P. S. Cunha. *A Deep Learning Architecture for Epileptic Seizure Classification Based on Object and Action Recognition.* In ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4117–4121, May 2020. ISSN: 2379-190X. 35
- [Kendall 17] Alex Kendall and Yarin Gal. *What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?* arXiv:1703.04977 [cs], October 2017. arXiv: 1703.04977. 68
- [Kingma 14] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* arXiv:1412.6980 [cs], December 2014. 78

- [Kofler 21] Florian Kofler, Ivan Ezhov, Fabian Isensee, Fabian Balsiger, Christoph Berger, Maximilian Koerner, Johannes Paetzold, Hongwei Li, Suprosanna Shit, Richard McKinley, Spyridon Bakas, Claus Zimmer, Donna Ankerst, Jan Kirschke, Benedikt Wiestler and Bjoern H. Menze. *Are we using appropriate segmentation metrics? Identifying correlates of human expert perception for CNN training beyond rolling the DICE coefficient.* arXiv:2103.06205 [cs, eess], March 2021. arXiv: 2103.06205. 97
- [Krizhevsky 12] Alex Krizhevsky, Ilya Sutskever and Geoffrey E. Hinton. *ImageNet Classification with Deep Convolutional Neural Networks.* In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, NIPS'12, pages 1097–1105, USA, 2012. Curran Associates Inc. 94, 95
- [LaMontagne 19] Pamela J. LaMontagne, Tammie LS Benzinger, John C. Morris, Sarah Keefe, Russ Hornbeck, Chengjie Xiong, Elizabeth Grant, Jason Hassenstab, Krista Moulder, Andrei G. Vlassenko, Marcus E. Raichle, Carlos Cruchaga and Daniel Marcus. *OASIS-3: Longitudinal Neuroimaging, Clinical, and Cognitive Dataset for Normal Aging and Alzheimer Disease.* medRxiv, page 2019.12.13.19014902, December 2019. Publisher: Cold Spring Harbor Laboratory Press. 74
- [Larobina 14] Michele Larobina and Loredana Murino. *Medical Image File Formats.* Journal of Digital Imaging, volume 27, number 2, pages 200–206, April 2014. 91, 92
- [Lee 19] Matthew C. H. Lee, Ozan Oktay, Andreas Schuh, Michiel Schaap and Ben Glocker. *Image-and-Spatial Transformer Networks for Structure-Guided Image Registration.* In Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap and Ali Khan, editors, Medical Image Computing and Computer Assisted Intervention – MICCAI 2019, Lecture Notes in Computer Science, pages 337–345, Cham, 2019. Springer International Publishing. 115
- [Li, Z. 02] Li, Z., A. M. da Silva and J. P. S. Cunha. *Movement quantification in epileptic seizures: a new approach to video-EEG*

- analysis.* IEEE Transactions on Biomedical Engineering, volume 49, number 6, pages 565–573, June 2002. Conference Name: IEEE Transactions on Biomedical Engineering. 34
- [Li 17] Wenqi Li, Guotai Wang, Lucas Fidon, Sebastien Ourselin, M. Jorge Cardoso and Tom Vercauteren. *On the Compactness, Efficiency, and Representation of 3D Convolutional Networks: Brain Parcellation as a Pretext Task.* In Marc Niethammer, Martin Styner, Stephen Aylward, Hongtu Zhu, Ipek Oguz, Pew-Thian Yap and Dinggang Shen, editors, Information Processing in Medical Imaging, Lecture Notes in Computer Science, pages 348–360, Cham, 2017. Springer International Publishing. 59, 94, 96
- [Lorensen 87] William E. Lorensen and Harvey E. Cline. *Marching cubes: A high resolution 3D surface construction algorithm.* In Proceedings of the 14th annual conference on Computer graphics and interactive techniques, SIGGRAPH ’87, pages 163–169, New York, NY, USA, August 1987. Association for Computing Machinery. 53
- [Loshchilov 19] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization.* arXiv:1711.05101 [cs, math], January 2019. arXiv: 1711.05101. 40, 78
- [Lowekamp 13] Bradley C. Lowekamp, David T. Chen, Luis Ibáñez and Daniel Blezek. *The Design of SimpleITK.* Frontiers in Neuroinformatics, volume 7, page 45, 2013. 78, 92, 96, 99, 104
- [Lu 18] Donghuan Lu, Karteek Popuri, Gavin Weiguang Ding, Rakesh Balachandar, Mirza Faisal Beg and Alzheimer’s Disease Neuroimaging Initiative. *Multimodal and Multiscale Deep Neural Networks for the Early Diagnosis of Alzheimer’s Disease using structural MR and FDG-PET images.* Scientific Reports, volume 8, number 1, page 5697, 2018. 89
- [Lucena 19] Oeslle Lucena, Roberto Souza, Letícia Rittner, Richard Frayne and Roberto Lotufo. *Convolutional neural networks for skull-stripping in brain MR imaging using silver standard masks.* Artificial Intelligence in Medicine, volume 98, pages 48–58, 2019. 94

- [Ma 21] Nic Ma, Wenqi Li, Richard Brown, Yiheng Wang, Benjamin Gorman, Behrooz, Hans Johnson, Isaac Yang, Eric Kerfoot, Yiwen Li, Mohammad Adil, Yuan-Ting Hsieh, charliebudd, Arpit Aggarwal, Cameron Trentz, adam aji, Ben Murray, Gagan Daroach, Petru-Daniel Tudosiu, myron, Mark Graham, Balamurali, Christian Baker, Jan Sellner, Lucas Fidon, Alex Powers, Guy Leroy, Alxaline and Daniel Schulz. *Project-MONAI/MONAI: 0.5.0*, April 2021. 97
- [Maia 19] P. Maia, E. Hartl, C. Vollmar, S. Noachtar and J. P. Silva Cunha. *Epileptic seizure classification using the NeuroMov database*. In 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), pages 1–4, February 2019. 35
- [Mancolo 20] Frank Mancolo. *Eisen: a python package for solid deep learning*. arXiv:2004.02747 [cs, eess], March 2020. arXiv: 2004.02747. 97
- [Matzkin 20] Franco Matzkin, Virginia Newcombe, Susan Stevenson, Aneesh Khetani, Tom Newman, Richard Digby, Andrew Stevens, Ben Glocke and Enzo Ferrante. *Self-supervised Skull Reconstruction in Brain CT Images with Decompressive Craniectomy*. In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu and Leo Joskowicz, editors, MICCAI 2020, Lecture Notes in Computer Science, pages 390–399, Cham, 2020. Springer International Publishing. 63
- [McCormick 14] Matthew Michael McCormick, Xiaoxiao Liu, Luis Ibanez, Julien Jomier and Charles Marion. *ITK: enabling reproducible research and open science*. Frontiers in Neuroinformatics, volume 8, 2014. 52
- [McCormick 21] Matt McCormick, Dženan Zukić and Stephen Aylward on. *ITK 5.2 Release Candidate 3 available for testing*, March 2021. 115
- [Meier 17] Raphael Meier, Nicole Porz, Urs Peter Knecht, Tina Loosli, Philippe Schucht, Jürgen Beck, Johannes Slotboom, Roland Wiest and Mauricio Reyes. *Automatic estimation of extent of*

- resection and residual tumor volume of patients with glioblastoma.* Journal of Neurosurgery, volume 127, number 4, pages 798–806, October 2017. 62, 63, 88
- [Mercier 12] Laurence Mercier, Rolando F. Del Maestro, Kevin Petrecca, David Araujo, Claire Haegelen and D. Louis Collins. *Online database of clinical MR and ultrasound images of brain tumors.* Medical Physics, volume 39, number 6, pages 3253–3261, June 2012. 76, 83
- [Milletari 16] Fausto Milletari, Nassir Navab and Seyed-Ahmad Ahmadi. *V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation.* In 2016 Fourth International Conference on 3D Vision (3DV), pages 565–571. IEEE Computer Society, October 2016. 78, 94
- [Modat 14] Marc Modat, David M. Cash, Pankaj Daga, Gavin P. Winston, John S. Duncan and Sébastien Ourselin. *Global image registration using a symmetric block-matching approach.* Journal of Medical Imaging, volume 1, number 2, July 2014. 77
- [Mohamed 11] Ahmad R. Mohamed, Jeremy L. Freeman, Virginia Maixner, Catherine A. Bailey, Jacquie A. Wrennall and A. Simon Harvey. *Temporoparietooccipital disconnection in children with intractable epilepsy: Clinical article.* Journal of Neurosurgery: Pediatrics, volume 7, number 6, pages 660–670, June 2011. Publisher: American Association of Neurological Surgeons Section: Journal of Neurosurgery: Pediatrics. 88
- [Moshkov 20] Nikita Moshkov, Botond Mathe, Attila Kertesz-Farkas, Reka Hollandi and Peter Horvath. *Test-time augmentation for deep learning-based cell segmentation on microscopy images.* Scientific Reports, volume 10, number 1, page 5068, March 2020. Number: 1 Publisher: Nature Publishing Group. 110
- [Nashef 97] L. Nashef. *Sudden unexpected death in epilepsy: terminology and definitions.* Epilepsia, volume 38, number 11 Suppl, pages S6–8, November 1997. 26
- [Nashef 12] Lina Nashef, Elson L. So, Philippe Ryvlin and Torbjörn Tomson. *Unifying the definitions of sudden unexpected death in*

- [NICE 12] NICE. Epilepsies: diagnosis and management. National Institute for Health and Care Excellence, 2012. 25
- [Nikolov 18] Stanislav Nikolov, Sam Blackwell, Ruheena Mendes, Jeffrey De Fauw, Clemens Meyer, Cían Hughes, Harry Askham, Bernardino Romera-Paredes, Alan Karthikesalingam, Carlton Chu, Dawn Carnell, Cheng Boon, Derek D’Souza, Syed Ali Moinuddin, Kevin Sullivan, DeepMind Radiographer Consortium, Hugh Montgomery, Geraint Rees, Ricky Sharma, Mustafa Suleyman, Trevor Back, Joseph R. Ledsam and Olaf Ronneberger. *Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy*. arXiv:1809.04430 [physics, stat], September 2018. arXiv: 1809.04430. 95
- [Nowell 15] Mark Nowell, Roman Rodionov, Gergely Zombori, Rachel Sparks, Gavin Winston, Jane Kinghorn, Beate Diehl, Tim Wehner, Anna Miserocchi, Andrew W. McEvoy, Sebastien Ourselin and John Duncan. *Utility of 3D multimodality imaging in the implantation of intracranial electrodes in epilepsy*. Epilepsia, volume 56, number 3, pages 403–413, 2015. \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/epi.12924>. 49
- [Nowell 17] Mark Nowell, Rachel Sparks, Gergely Zombori, Anna Miserocchi, Roman Rodionov, Beate Diehl, Tim Wehner, Mark White, Sebastien Ourselin, Andrew McEvoy and John Duncan. *Resection planning in extratemporal epilepsy surgery using 3D multimodality imaging and intraoperative MRI*. British Journal of Neurosurgery, volume 31, number 4, pages 468–470, July 2017. 49
- [Nuñez 18] Jamie R. Nuñez, Christopher R. Anderton and Ryan S. Renslow. *Optimizing colormaps with consideration for color vision deficiency to enable accurate interpretation of scientific data*. PLOS ONE, volume 13, number 7, page e0199239, August 2018. Publisher: Public Library of Science. 56

- [Ny  l 99] L. G. Ny  l and J. K. Udupa. *On standardizing the MR image intensity scale*. Magnetic Resonance in Medicine, volume 42, number 6, pages 1072–1081, December 1999. 95, 104
- [Ny  l 00] L. G. Ny  l, J. K. Udupa and X. Zhang. *New variants of a method of MRI scale standardization*. IEEE transactions on medical imaging, volume 19, number 2, pages 143–150, February 2000. 79, 105
- [O'Dwyer 07] Rebecca O'Dwyer, Joao P. Silva Cunha, Christian Vollmar, Cordula Mauerer, Berend Feddersen, Richard C. Burgess, Alois Ebner and Soheyl Noachtar. *Lateralizing significance of quantitative analysis of head movements before secondary generalization of seizures of patients with temporal lobe epilepsy*. Epilepsia, volume 48, number 3, pages 524–530, March 2007. 34
- [Omigbodun 19] Akinyinka O. Omigbodun, Frederic Noo, Michael McNitt-Gray, William Hsu and Scott S. Hsieh. *The effects of physics-based data augmentation on the generalizability of deep neural networks: Demonstration on nodule false-positive reduction*. Medical Physics, volume 46, number 10, pages 4563–4574, 2019. 106
- [Ourselin 00] S. Ourselin, A. Roche, S. Prima and N. Ayache. *Block Matching: A General Framework to Improve Robustness of Rigid Registration of Medical Images*. In Scott L. Delp, Anthony M. DiGoia and Branislav Jaramaz, editors, Medical Image Computing and Computer-Assisted Intervention – MICCAI 2000, Lecture Notes in Computer Science, pages 557–566, Berlin, Heidelberg, 2000. Springer. 57
- [Paszke 19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai and Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alch  -Buc, E. Fox and R. Garnett, editors, Advances in Neural

- Information Processing Systems 32, pages 8024–8035. Curran Associates, Inc., 2019. 77, 90, 91, 95, 98, 99, 104
- [Pawlowski 17] Nick Pawlowski, Sofia Ira Ktena, Matthew C. H. Lee, Bernhard Kainz, Daniel Rueckert, Ben Glocker and Martin Rajchl. *DLTK: State of the Art Reference Implementations for Deep Learning on Medical Images*. arXiv:1711.06853 [cs], November 2017. arXiv: 1711.06853. 90, 96
- [Perlin 02] Ken Perlin. *Improving noise*. ACM Transactions on Graphics (TOG), volume 21, number 3, pages 681–682, July 2002. 69
- [Perone 18] Christian S. Perone, cclauss, Elvis Saravia, Pedro Lemos Ballester and MohitTare. *perone/medicaltorch: Release v0.2*, November 2018. 96
- [Pezeshk 17] Aria Pezeshk, Nicholas Petrick, Weijie Chen and Berkman Sahiner. *Seamless lesion insertion for data augmentation in CAD training*. IEEE Trans Med Imaging, volume 36, number 4, pages 1005–1015, April 2017. 63
- [Pinter 19] Csaba Pinter, Andras Lasso and Gabor Fichtinger. *Poly-morph segmentation representation for medical image computing*. Computer Methods and Programs in Biomedicine, volume 171, pages 19–26, April 2019. 53
- [Preston-Werner 20] Tom Preston-Werner. *Semantic Versioning 2.0.0*, 2020. Library Catalog: semver.org. 111
- [Radosavovic 17] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari and Kaiming He. *Data Distillation: Towards Omni-Supervised Learning*. arXiv:1712.04440 [cs], December 2017. arXiv: 1712.04440. 67
- [Raghu 19] Maithra Raghu, Chiyuan Zhang, Jon Kleinberg and Samy Bengio. *Transfusion: Understanding Transfer Learning for Medical Imaging*. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019. 95
- [Ranzini 21] Marta B. M. Ranzini, Lucas Fidon, Sébastien Ourselin, Marc Modat and Tom Vercauteren. *MONAI<sub>fbs</sub>: MONAI-based fetal*

- brain MRI deep learning segmentation.* arXiv:2103.13314 [cs, eess], March 2021. arXiv: 2103.13314. 97
- [Rosenow 01] Felix Rosenow and Hans Lüders. *Presurgical evaluation of epilepsy.* Brain, volume 124, number 9, pages 1683–1700, September 2001. 27, 48, 62
- [Ryvlin 13] Philippe Ryvlin, Lina Nashef, Samden D. Lhatoo, Lisa M. Bateman, Jonathan Bird, Andrew Bleasel, Paul Boon, Arielle Crespel, Barbara A. Dworetzky, Hans Høgenhaven, Holger Lerche, Louis Maillard, Michael P. Malter, Cecile Marchal, Jagarlapudi M. K. Murthy, Michael Nitsche, Ekaterina Pataraia, Terje Rabben, Sylvain Rheims, Bernard Sadzot, Andreas Schulze-Bonhage, Masud Seyal, Elson L. So, Mark Spitz, Anna Szucs, Meng Tan, James X. Tao and Torbjörn Tomson. *Incidence and mechanisms of cardiorespiratory arrests in epilepsy monitoring units (MORTEMUS): a retrospective study.* The Lancet. Neurology, volume 12, number 10, pages 966–977, October 2013. 34
- [Schroeder 06] Will Schroeder, Ken Martin and Bill Lorensen. *Visualization Toolkit: An Object-Oriented Approach to 3D Graphics,* 4th Edition. Kitware, Clifton Park, NY, 4th edition edition, December 2006. 52, 78
- [Shan 18] Siyuan Shan, Wen Yan, Xiaoqing Guo, Eric I.-Chao Chang, Yubo Fan and Yan Xu. *Unsupervised End-to-end Learning for Deformable Medical Image Registration.* arXiv:1711.08608 [cs], January 2018. arXiv: 1711.08608. 90
- [Shaw 19] Richard Shaw, Carole Sudre, Sébastien Ourselin and M. Jorge Cardoso. *MRI k-Space Motion Artefact Augmentation: Model Robustness and Task-Specific Uncertainty.* In International Conference on Medical Imaging with Deep Learning, pages 427–436, May 2019. 79, 104, 108
- [Shaw 20] Richard Shaw, Carole H. Sudre, Sébastien Ourselin and M. Jorge Cardoso. *A Heteroscedastic Uncertainty Model for Decoupling Sources of MRI Image Quality.* In Medical Imaging with Deep Learning, pages 733–742. PMLR, September 2020. ISSN: 2640-3498. 79, 104, 106

- [Simonyan 14] Karen Simonyan and Andrew Zisserman. *Two-stream convolutional networks for action recognition in videos*. In Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1, NIPS'14, pages 568–576, Cambridge, MA, USA, December 2014. MIT Press. 35
- [So 08] Elson L. So. *What is known about the mechanisms underlying SUDEP?* Epilepsia, volume 49 Suppl 9, pages 93–98, December 2008. 26
- [Sparks 17] Rachel Sparks, Gergely Zombori, Roman Rodionov, Mark Nowell, Sjoerd B. Vos, Maria A. Zuluaga, Beate Diehl, Tim Wehner, Anna Miserocchi, Andrew W. McEvoy, John S. Duncan and Sébastien Ourselin. *Automated multiple trajectory planning algorithm for the placement of stereo-electroencephalography (SEEG) electrodes in epilepsy treatment*. International Journal of Computer Assisted Radiology and Surgery, volume 12, number 1, pages 123–136, January 2017. 49
- [Sudre 17] Carole H. Sudre, M. Jorge Cardoso and Sébastien Ourselin. *Longitudinal segmentation of age-related white matter hyper-intensities*. Medical Image Analysis, volume 38, pages 50–64, May 2017. 79, 104
- [Sutula 03] Thomas P. Sutula, Joshua Hagen and Asla Pitkänen. *Do epileptic seizures damage the brain?* Current Opinion in Neurology, volume 16, number 2, pages 189–195, April 2003. 25
- [Pérez-García 19] **Fernando Pérez-García.** *fepegar/highresnet: PyTorch implementation of HighRes3DNet*, November 2019. 59
- [Pérez-García 20a] **Fernando Pérez-García.** *fepegar/torchio: TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning*, November 2020. 111
- [Pérez-García 20b] **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation*. In International League Against Epilepsy

- (ILAE) British Branch Virtual Annual Scientific Conference, 2020.
- [Pérez-García 20c] **Fernando Pérez-García**, Ali Alim-Marvasti, Gloria Romagnoli, Matthew J. Clarkson, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Towards Objective Targeting of Intracranial Electroencephalography Using Data-Driven Semiology-Brain Visualisation*. In American Epilepsy Society (AES) Annual Meeting, 2020.
- [Pérez-García 20d] **Fernando Pérez-García**, Roman Rodionov, Ali Alim-Marvasti, Rachel Sparks, John Duncan and Sébastien Ourselin. *EPISURG: a dataset of postoperative magnetic resonance images (MRI) for quantitative analysis of resection neurosurgery for refractory epilepsy*, December 2020. 7, 76, 120
- [Pérez-García 20e] **Fernando Pérez-García**, Roman Rodionov, Ali Alim-Marvasti, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *Simulation of Brain Resection for Cavity Segmentation Using Self-supervised and Semi-supervised Learning*. In MICCAI 2020, Lecture Notes in Computer Science, pages 115–125, Cham, 2020. Springer International Publishing. 64, 76, 81, 87, 109, 119
- [Pérez-García 21a] **Fernando Pérez-García**, Reuben Dorent, Michele Rizzi, Francesco Cardinale, Valerio Frazzini, Vincent Navarro, Caroline Essert, Irène Ollivier, Tom Vercauteren, Rachel Sparks, John S. Duncan and Sébastien Ourselin. *A self-supervised learning strategy for postoperative brain cavity segmentation simulating resections*. International Journal of Computer Assisted Radiology and Surgery, June 2021. 7, 120
- [Pérez-García 21b] **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Data to support the paper "Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures"*, July 2021. 41, 118
- [Pérez-García 21c] **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures*. In Medical Image Computing and Computer Assisted

- Intervention – MICCAI 2021, Lecture Notes in Computer Science, pages 334–344, Cham, 2021. Springer International Publishing. 118
- [Pérez-García 21d] **Fernando Pérez-García**, Catherine Scott, Rachel Sparks, Beate Diehl and Sébastien Ourselin. *Transfer Learning of Deep Spatiotemporal Networks to Model Arbitrarily Long Videos of Seizures*. In American Epilepsy Society (AES) Annual Meeting, 2021.
- [Pérez-García 21e] **Fernando Pérez-García**, Rachel Sparks and Sébastien Ourselin. *TorchIO: A Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning*. Computer Methods and Programs in Biomedicine, volume 208, page 106236, September 2021. 31, 79, 86, 120
- [Thom 14] Maria Thom. *Review: Hippocampal sclerosis in epilepsy: a neuropathology review*. Neuropathology and Applied Neurobiology, volume 40, number 5, pages 520–543, August 2014. 27
- [Tufenkjian 12] Krikor Tufenkjian and Hans O. Lüders. *Seizure Semiology: Its Value and Limitations in Localizing the Epileptogenic Zone*. Journal of Clinical Neurology (Seoul, Korea), volume 8, number 4, pages 243–250, December 2012. 26, 34, 49
- [Vakharia 19] Vejay N. Vakharia, Rachel E. Sparks, Kuo Li, Aidan G. O’Keeffe, **Fernando Pérez-García**, Lucas G. S. França, Andrew L. Ko, Chengyuan Wu, Joshua P. Aronson, Brett E. Youngerman, Ashwini Sharan, Guy McKhann, Sébastien Ourselin and John S. Duncan. *Multicenter validation of automated trajectories for selective laser amygdalohippocampectomy*. Epilepsia, volume 60, number 9, pages 1949–1959, 2019.
- [Valindria 18] Vanya V. Valindria, Nick Pawlowski, Martin Rajchl, Ioannis Lavdas, Eric O. Aboagye, Andrea G. Rockall, Daniel Rueckert and Ben Glocker. *Multi-modal Learning from Unpaired Images: Application to Multi-organ Segmentation in CT and MRI*. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 547–556, March 2018. 96

- [van der Walt 11] Stefan van der Walt, S. Chris Colbert and Gael Varoquaux. *The NumPy Array: A Structure for Efficient Numerical Computation.* Computing in Science Engineering, volume 13, number 2, pages 22–30, March 2011. Conference Name: Computing in Science Engineering. 78, 96, 97, 99, 104
- [Van Leemput 99] K. Van Leemput, F. Maes, D. Vandermeulen and P. Suetens. *Automated model-based tissue classification of MR images of the brain.* IEEE transactions on medical imaging, volume 18, number 10, pages 897–908, October 1999. 108
- [Venturini 20] Lorenzo Venturini, Aris T. Papageorghiou, J. Alison Noble and Ana I. L. Namburete. *Uncertainty Estimates as Data Selection Criteria to Boost Omni-Supervised Learning.* In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu and Leo Joskowicz, editors, Medical Image Computing and Computer Assisted Intervention – MICCAI 2020, Lecture Notes in Computer Science, pages 689–698, Cham, 2020. Springer International Publishing. 64, 68
- [Vilella 21] Laura Vilella, Nuria Lacuey, Johnson P. Hampson, Liang Zhu, Shirin Omidi, Manuela Ochoa-Urrea, Shiqiang Tao, M. R. Sandhya Rani, Rup K. Sainju, Daniel Friedman, Maromi Nei, Kingman Strohl, Catherine Scott, Luke Allen, Brian K. Gehlbach, Norma J. Hupp, Jaison S. Hampson, Nassim Shafabadi, Xiuhe Zhao, Victoria Reick-Mitrisin, Stephan Schuele, Jennifer Ogren, Ronald M. Harper, Beate Diehl, Lisa M. Bateman, Orrin Devinsky, George B. Richerson, Philippe Ryvlin, Guo-Qiang Zhang and Samden D. Lhatoo. *Association of Periictal Brainstem Posturing With Seizure Severity and Breathing Compromise in Patients With Generalized Convulsive Seizures.* Neurology, volume 96, number 3, pages e352–e365, January 2021. 26
- [Virtanen 20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric

- Jones, Robert Kern, Eric Larson, C. J. Carey, Ilhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa and Paul van Mulbregt. *SciPy 1.0: fundamental algorithms for scientific computing in Python*. Nature Methods, pages 1–12, February 2020. 97
- [Wang 19a] Guotai Wang, Wenqi Li, Michael Aertsen, Jan Deprest, Sébastien Ourselin and Tom Vercauteren. *Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks*. Neurocomputing, volume 338, pages 34–45, April 2019. 67, 68, 110
- [Wang 19b] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang and L. Van Gool. *Temporal Segment Networks for Action Recognition in Videos*. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 41, number 11, pages 2740–2755, November 2019. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. 36
- [Weiss 16] Karl Weiss, Taghi M. Khoshgoftaar and DingDing Wang. *A survey of transfer learning*. Journal of Big Data, volume 3, number 1, page 9, May 2016. 95
- [Winston 12] Gavin P. Winston, Pankaj Daga, Jason Stretton, Marc Modat, Mark R. Symms, Andrew W. McEvoy, Sébastien Ourselin and John S. Duncan. *Optic radiation tractography and vision in anterior temporal lobe resection*. Annals of Neurology, volume 71, number 3, pages 334–341, March 2012. 87
- [wiredfool 16] wiredfool, Alex Clark, Hugo, Andrew Murray, Alexander Karpinsky, Christoph Gohlke, Brian Crowell, David Schmidt, Alastair Houghton, Steve Johnson, Sandro Mani, Josh Ware, David Caro, Steve Kossouho, Eric W. Brown, Antony Lee, Mikhail Korobov, Michał Górný, Esteban Santana Santana, Nicolas Pieuchot, Oliver Tonnhofer, Michael Brown, Benoit Pierre, Joaquín Cuenca Abela, Lars Jørgen Solberg, Felipe Reyes, Alexey Buzanov, Yifu Yu, eliempje and Fredrik Tolf. *Pillow: 3.1.0*, January 2016. 93

- [Zagoruyko 17] Sergey Zagoruyko and Nikos Komodakis. *Wide Residual Networks*. arXiv:1605.07146 [cs], June 2017. arXiv: 1605.07146. 41
- [Zhang 21] Xiaoman Zhang, Weidi Xie, Chaoqin Huang, Ya Zhang and Yanfeng Wang. *Self-supervised Tumor Segmentation through Layer Decomposition*. arXiv:2109.03230 [cs], September 2021. arXiv: 2109.03230. 7, 119
- [Zhu 14] Liangjia Zhu, Ivan Kolesov, Yi Gao, Ron Kikinis and Allen Tannenbaum. *An Effective Interactive Medical Image Segmentation Method using Fast GrowCut*. In Int Conf Med Image Comput Comput Assist Interv. Workshop on Interactive Methods, volume 17, 2014. 76
- [Zhuo 06] Jiachen Zhuo and Rao P. Gullapalli. *MR Artifacts, Safety, and Quality Control*. RadioGraphics, volume 26, number 1, pages 275–297, January 2006. 106, 108
- [Zwillinger 99] Daniel Zwillinger and Stephen Kokoska. CRC Standard Probability and Statistics Tables and Formulae. CRC Press, December 1999. Google-Books-ID: tB3RVEZ0UIMC. 68