



UNIVERSITÀ
DEGLI STUDI
FIRENZE

PhD in
Atomic and Molecular Photonics
CYCLE XXXV

COORDINATOR
Prof. Diederik Wiersma

**Deep Learning Neuronal Mapping in
Fluorescence Microscopy Imaging of the
Human Brain**

Academic Discipline (SSD) FIS/03

Supervisor:
Prof. Francesco Saverio Pavone

Coordinator:
Prof. Diederik Wiersma

Doctoral Candidate:
Filippo Maria Castelli

Years 2019/2023



UNIVERSITÀ
DEGLI STUDI
FIRENZE

PhD in
Atomic and Molecular Photonics

CYCLE XXXV

COORDINATOR
Prof. Diederik Wiersma

**Deep Learning Neuronal Mapping in
Fluorescence Microscopy Imaging of the
Human Brain**

Academic Discipline (SSD) FIS/03

Supervisor:
Prof. Francesco Saverio Pavone

Coordinator:
Prof. Diederik Wiersma

Doctoral Candidate:
Filippo Maria Castelli

Years 2019/2023

Abstract

The Human Brain is, by far, the most complex system known to man and the mysteries of its inner workings are concealed behind the intricate relationships between functional and topological organization of its neuronal structures. Quantitative studies of this target at the microscopic scale require both high-resolution imaging of large brain tissue samples, and ways of automatically analyzing the massive quantities of data created in the process. Fluorescence Microscopy has the imaging capability needed to cover the scale gap between the micron scale, on which individual neurons are defined, and the centimeter scale of brain functional areas. This potential is still partially unexpressed due to engineering challenges in automated processing and analysis of such massive data-streams that, to this day, have not been entirely solved. This work proposes a methodological framework for the exploration of extended areas of Human Brain, by means of Two-Photon Confocal Microscopy and Light-Sheet Fluorescence Microscopy Imaging, exploiting Deep Learning models and careful data-flow design to map large quantities of individual neurons across vast volumetric extensions.

Contents

1	Introduction	11
1.1	Goal of this work	12
1.2	A Scale Perspective	13
1.3	Roadmap	15
2	M.L.	19
2.1	Learning Algorithms	20
2.2	Artificial Neural Networks	22
2.3	CNNs	23
2.3.1	CNN Properties	23
2.4	CNNs in Segmentation	25
2.4.1	Classical ML Approaches	28
2.4.2	Convolutional Neural Networks	29
2.4.3	Convolutional Interpretation of Sliding Windows	30
2.4.4	Fully Convolutional Networks	31
2.4.5	Convolutional Networks with Graphical Models	32
2.4.6	Encoder-Decoder Networks	33
2.4.7	Instance Segmentation	35
3	Experimental and Computational Methods	45
3.1	Physical Principles of Fluorescence Microscopy	45

3.1.1	The Jablonski Diagram	46
3.2	TTPFM	48
3.2.1	Two-Photon Excitation and its Advantages in Mi- croscopy	48
3.2.2	2PE Confocal Setup	50
3.3	LightSheet Microscopy	52
3.4	Optical Clearing	54
3.4.1	Clearing Methods	54
3.4.2	Applicability to Fluorescence Microscopy	57
3.5	Data Processing Pipeline	58
3.5.1	Stitching Datasets	61
3.5.2	Semantic Segmentation Framework: NeuroSeg- menter	62
3.6	Dataset Sharing	66
3.6.1	Extending the BIDS Standard: Microscopy-BIDS	66
3.6.2	BIDS-Microscopy Example Dataset	67
3.6.3	The <i>pyometiff</i> library	69
4	TTPFM Brain Imaging	83
4.1	Introduction	84
4.2	Methods	86
4.2.1	Biological Samples	87
4.2.2	Tissue Clearing and Labeling	87
4.2.3	TTPFM Setup	88
4.2.4	Frame Stitching and Alignment	90
4.2.5	CNN-based Segmentation	91
4.2.6	Ground Truth Collection and Model Training	93
4.2.7	From 2D Heatmaps to 3D Polygon Meshes	94
4.3	Results	95
4.3.1	Whole-Sample Imaging	95

4.3.2	3D Visualization	95
4.3.3	Neuronal Distributions Analysis	97
4.3.4	Performance Assessment over Whole Stacks	100
4.4	Conclusions	102
4.4.1	Choice of CNN Architecture	102
4.4.2	Grouping Artifacts	102
4.4.3	Conclusions	104
5	Lightsheet Microscopy Broca Area Investigation	111
5.1	Introduction	112
5.2	Methods	114
5.2.1	Tissue Clearing and Labeling Biological Samples	114
5.2.2	LSFM Setup	115
5.2.3	Acquisition Data Management	119
5.2.4	Deep Learning-enabled Segmentation	120
5.2.5	Computational Infrastructure for Data Processing	127
5.2.6	Data Visualization	128
5.3	Results	129
5.3.1	Dataset Acquisition, Stitching and Reslicing	129
5.3.2	Semantic Segmentation	133
5.3.3	Instance Segmentation	136
5.4	Discussion	139
5.4.1	Neuronal Counting in the Broca Area	139
5.4.2	Model Performance	143
5.4.3	Architecture Selection	145
5.4.4	Potential Limitations of CNNs	146
5.5	Conclusions	148
5.5.1	Data and Software Availability	149
6	Conclusions	157
6.1	Future Perspectives	160

6.1.1	Multi-channel and Multi-target	160
6.1.2	Multi-View	161
6.1.3	Multi-Modal Cell Atlases	161

List of Tables

3.1	Data Size Estimations for SPIM Brain Acquisitions	60
4.1	Neuronal Counts for Cortical Layer	101
5.1	Data Augmentation Transforms	124
5.2	Dataset Digital Footprint	130
5.3	Semantic Segmentation Metrics	135
5.4	Neuronal Population per Cortical Layer	142

List of Figures

2.1	The Machine Learning Paradigm	21
2.2	Rosenblatt’s Perceptron Functional Scheme	22
2.3	Sparse Connectivity and Parameter Sharing	24
2.4	Semantic Segmentation and Instance Segmentation	26
2.5	Classical vs DL-based Segmentation	28
2.6	The FCN Architecture (from Long et al. [23])	32
2.7	The UNet Architecture (from Ronneberger et al. [35])	34
3.1	The Jablonski Diagram	47
3.2	Two-Photon Excitation	49
3.3	Data Processing Pipeline	59
3.4	<i>NeuroSegmenter</i> working diagram	71
4.1	SWITCH/TDE Clearing	88
4.2	Human Cortical Samples	89
4.3	The 2.5D Approach	91
4.4	TPFM CNN Architecture	92
4.5	Whole-Sample 3D Reconstruction	96
4.6	Whole Sample 3D Reconstructions	98
4.7	Distribution maps of counting, mean volume and volumetric density	99
4.8	Manual Segmentation of the Gray Matter	100

4.9	Segmented Stack	100
4.10	Grouping Artifacts	103
5.1	Individual Slice from the Broca Area Sample	115
5.2	The di2-CLSFM Setup	116
5.3	3D Schematics of the di2-CLSFM apparatus	118
5.4	The SPIMlab Control Software	119
5.5	NEUROresUnet	121
5.6	Instance Segmentation Process	125
5.7	Processing Architecture	127
5.8	Wide-scale reconstruction of Broca Area slices	132
5.9	Semantic Segmentation of Neurons in LSFM	134
5.10	ROC and PR Curves	136
5.11	Neuronal Density Distribution in Broca Area Slices	138
5.12	Soma Diameter Distributions in Broca Area Slices	140
5.13	Cortical Layer Segmentation in the Broca Area	141

Chapter 1

Introduction

The human brain is possibly the most complex object known to man, and its study one of the largest collective efforts of the scientific community, engaging scientists and experts from almost every research field. The brain appears as an highly organized multi-scale network-like structure where both spatial and topological complexity arise in a fascinating interconnection of its functional and morphological features. At every spatial scale a seemingly infinite amount of detail seem to emerge, from the intricacies of synapses to large-scale connections between different brain areas, and a variety of different technologies have to be employed to study them, from Electron Microscopy to Magnetic Resonance Imaging.

LENS - European Laboratory for NonLinear Spectroscopy - and its Biophotonics Group places themselves in the first line of the the collective effort towards a deeper characterization of the human brain by providing innovative techniques of tissue exploration based on Fluorescence Microscopy in the context of large international projects like the Human Brain Project by the European Commission, or the BRAIN

Initiative by the National Institutes of Health of the United States of America. Creating large imaging datasets from human brain tissue is a collective effort that involves many people with multiple and differentiated skills: preparing samples before imaging requires deep knowledge of all the biological aspects regarding the subject and expertise in tissue clearing and transformation techniques, skills in the physics engineering areas are participating in designing, realizing and operating the imaging instruments, the technical challenges tied to the creation of data transfer and processing flows, along with the implementation of suitable hardware infrastructures to handle them require computer engineering skills which are different from the statistical and mathematical competences needed to design and deploy interpretation models of the collected data. My activity, as a PhD student at LENS in the Biophotonics Group, saw my direct involvement in the last three of the mentioned areas, with particular focus on designing and deploying the computational infrastructure, algorithms and models to automate the production, processing and analysis of entire areas of the human brain.

1.1 Goal of this work

The aim of this work is to offer a technical framework on which any Fluorescence Microscopy laboratory can easily build their own pipeline to convert large amounts of human brain imaging into 3D reconstructions of all its neuronal bodies, allowing the researcher to easily respond to quantitative questions like *"how many neurons are in this area?"*, *"how are they distributed?"* or *"what shape do they have?"*. The key element in answering these questions is represented by automated deep-learning-based analysis of the microscopy data. Convolutional Neural Networks offer the mean to translate raw tissue acquisitions from a quantitative imaging space to a semantically-aware domain, this is known technically

as a *semantic segmentation* problem and finding a technologically feasible and scalable way to perform it on arbitrarily large datasets is one of the main focuses of this PhD thesis.

1.2 Framing the Problem: a Scale Perspective

The neuronal structures in our brain are expressed across multiple physical scales, ranging from the nanometric dimensions of single synapses, to the micrometric scales of neuronal somas, to long scale connections between different areas of the organ in the centimeter range. Different imaging techniques are used to cover these different scales: synaptic connections are usually observed with Electron Microscopy, large scale tractography uses Magnetic Resonance Imaging - MRI (in combination with Diffusion Tensor Imaging - DTI) and Computer Tomography, while the micrometer scale is best covered by optical techniques such as Fluorescence Microscopy - FM or Optical Coherence Tomography - OCT. We, at LENS, use Fluorescence Microscopy in its variants (in this work we present data acquired in Two Photon Confocal Microscopy and Light Sheet Fluorescence Microscopy) to tackle the ambitious challenge of covering the entire mesoscopic scale from micrometers (10^{-6}m) to centimeter (10^{-2}m): four entire orders of magnitude. Fluorescence Microscopy imaging and creation of detailed atlases of entire mouse brains [4, 9, 10] has been an accessible target for LENS and for the general neuroscience community in the past years, extending the same level of analysis to human subjects is an ambitious challenge that has not yet been solved. Optimizing tissue transformation and clearing techniques to overcome the biological differences between mice and human tissue is just a piece of the puzzle (and not necessarily easier to solve than the others). From a scale perspective alone the two challenges appears totally different, the reader could get a feel of the involved data sizes by looking at

table 3.1: imaging an entire mouse brain, that typically has a physical size in the order of 1 cm^3 , depending on the chosen resolution and number of imaging channels can occupy from as little as 2 GB of disk space to 6 TB, these numbers change drastically for human brains, sized 1500 cm^3 , for which the low boundary of 3 TB corresponding to $10\text{ }\mu\text{m}$ resolution monochrome imaging is significantly more accessible than the approximate 9 PB ($9 \times 10^6\text{ GB}$) required for $1\text{ }\mu\text{m}$ resolution imaging in three channels. For reference, the setup described in chapter 5 is capable of sub-micrometric imaging in four independent channels.

Obtaining single-neuron resolution volumetric multi-spectral images of extended areas in the human brain, and building the computer infrastructure to properly compress and store them is an immense technical challenge on its own, but a feat that would ultimately reveal itself as a white elephant if we didn't complement it with a reliable and scalable process to extract high level information about the neuronal distributions. My mission at LENS was not only to find a way to analyze the massive quantities of data produced by imaging extended brain sections, but to do it in a way that would scale to entire brains and potentially to multiple subjects.

Having success in the effort towards the creation of massive neuronal atlases would pose the basis for a Big Data platform for future researchers to explore. Creating computer-interrogable atlases of our neural structures and untangling their inherent complexity has the potential to unlock knowledge about the human mind that would otherwise be hidden behind a wall of analytical inaccessibility and is untapped at the current day. This work aims at presenting a minimal yet functional contribution towards this monumental vision.

1.3 A Roadmap for the Next Chapters

One of my main contributions to the Biophotonics@LENS group is represented by the study and development of semantic and instance segmentation approaches to tackle the problem of making sense of the large quantities of data we produce, for this reason the first chapter of this work is dedicated to a general overview of the statistical and computational approaches representing the state of the art in solving the problem of automatical neuron detection and mapping at the current day.

Moving to the third chapter, in the first part we briefly present the various experimental and computational methods involved in the Fluorescence Imaging process, starting from a primer on the physical principles of fluorescence microscopy, we then introduce the specific technologies we used (Two Photon Confocal Microscopy and Lightsheet Fluorescence Microscopy) as well as the optical clearing and tissue transformation methods needed to obtain imaging of raw samples. The second part of the chapter is dedicated to the data elaboration pipeline: here we describe the computational approaches we used to create complete imaging volumes from single acquisitions using *ZetaStitcher*[8], an internally developed software capable of stitching and alignment of our large volumetric datasets. We then introduce *NeuroSegmenter*[2], a semantic segmentation framework based on TensorFlow that I specifically developed for microscopy applications in our group and offers an user interface combining accessibility requirements for users without programming expertise - aimed to a general microscopist figure who needs to train and evaluate models in autonomy - to complete flexibility and customizability for the operating ML specialist.

Next we talk about the issues revolving around sharing this kind of huge and complex data with the rest of neuroscience community: the bleeding edge novelty of our datasets has rendered manifest the inadequacy

of the current data standards, I've been directly involved in a collaboration for the definition of an extension of one of the current standards for sharing neuroimaging data, the Brain Imaging Data Structure Microscopy Extension Proposal[1]. The proposed extension has since been positively integrated in the BIDS standard and the microscopy data we provided is currently used as a reference dataset for microscopy data in the BIDS[5] format. The chapter also covers a Python library that I've developed, *pyometiff*[3], providing a full and up-to-date Python programming interface towards the OME-TIFF imaging specification and that is currently seeing application in the general neuroscience community [6, 7]. Fourth and fifth chapters explore two different experimental applications, in chapter 4 we use Two Photon Fluorescence Microscopy to analyze four different cortical tissue samples from different areas of the brain, in chapter 5 we use LightSheet Fluorescence Microscopy on an entire human Broca Area: the general pipeline presented in chapter 3 is adapted to the specific needs of the specific application, two different and unique segmentation models, responding to different imaging and data requirements.

Finally, in chapter 6 we conclude this work by making remarks on future perspectives.

Bibliography

- [1] Marie-Hélène Bourget et al. “Microscopy-BIDS: An Extension to the Brain Imaging Data Structure for Microscopy Data”. In: *Frontiers in Neuroscience* 16 (Apr. 2022), p. 871228. ISSN: 1662-4548. DOI: 10.3389/fnins.2022.871228. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9063519/> (visited on 05/18/2022).
- [2] Filippo Castelli. *neurosegmenter*. Dec. 2022. URL: <https://github.com/filippocastelli/neurosegmenter> (visited on 12/20/2022).
- [3] Filippo Castelli and Giacomo Mazzamuto. *pyometiff*. Jan. 2023. DOI: 10.5281/zenodo.7520378. URL: <https://doi.org/10.5281/zenodo.7520378>.
- [4] Antonino Paolo Di Giovanna et al. “Whole-Brain Vasculature Reconstruction at the Single Capillary Level”. En. In: *Scientific Reports* 8.1 (Aug. 2018), p. 12573. ISSN: 2045-2322. DOI: 10.1038/s41598-018-30533-3. URL: <https://www.nature.com/articles/s41598-018-30533-3> (visited on 05/10/2019).
- [5] Krzysztof J. Gorgolewski et al. “The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments”. en. In: *Scientific Data* 3.1 (June 2016), p. 160044. ISSN: 2052-4463. DOI: 10.1038/sdata.2016.44. URL: <https://www.nature.com/articles/sdata201644> (visited on 01/10/2023).

- [6] GradinaruLab/useqfish_analysis. URL: https://github.com/GradinaruLab/useqfish_analysis (visited on 01/24/2023).
- [7] Min Jang et al. *Spatial transcriptomics for profiling the tropism of viral vectors in tissues*. en. Jan. 2023. DOI: 10.21203/rs.3.rs-1486912/v1. URL: <https://www.researchsquare.com> (visited on 01/24/2023).
- [8] Giacomo Mazzamuto. *lens-biophotonics/ZetaStitcher*. Oct. 2022. URL: <https://github.com/lens-biophotonics/ZetaStitcher> (visited on 11/21/2022).
- [9] L. Silvestri et al. “Confocal light sheet microscopy: micron-scale neuroanatomy of the entire mouse brain”. EN. In: *Optics Express* 20.18 (Aug. 2012), pp. 20582–20598. ISSN: 1094-4087. DOI: 10.1364/OE.20.020582. URL: <https://opg.optica.org/oe/abstract.cfm?uri=oe-20-18-20582> (visited on 03/18/2023).
- [10] Ludovico Silvestri et al. “Towards a Full Volumetric Atlas of Cell-specific Neuronal Spatial Organization in the Entire Mouse Brain”. EN. In: *Biophotonics Congress: Biomedical Optics Congress 2018 (Microscopy/Translational/Brain/OTS) (2018), paper JTU3A.62*. Optical Society of America, Apr. 2018, JTU3A.62. DOI: 10.1364/TRANSLATIONAL.2018.JTu3A.62. URL: <https://www.osapublishing.org/abstract.cfm?uri=BRAIN-2018-JTu3A.62> (visited on 06/29/2019).

Chapter 2

Machine Learning and CNNs

Automating the process of detecting and mapping individual neurons across Petabyte-sized (thousands of TeraBytes) microscopy imaging of entire human brains is an engineering challenge that can only be tackled with advanced mathematical and computational tools. Classical computational approaches to image analysis, based on the imperative definition of selection rules to semantically separate objects of interest from the background are unable to perform reliably on large quantities of complex data. The inherent visual complexity of the subject poses an insurmountable obstacle to the explicit formulation of an exhaustive mathematical model representing how a neuron should look in every imaging situation across an entire brain, for this reason the solution to the problem of neuronal mapping has to be searched in an inverted perspective in which the selection rules are not defined by a human operator, but are directly inferred from the organizational structure of the data itself. This kind of approaches are known collectively under the name of *machine learning*.

Needing a tool that could ultimately allow us to individually track the tens of billions of neurons which are present in the human brain, we've chosen to rely on Convolutional Neural Networks, a particular algorithm that falls under the *machine learning* general category.

The first part of the chapter introduces convolutional neural networks as a general architecture, while the second part is a review of state of the art in automated strategies for neuron segmentation.

2.1 Learning Algorithms

The first definitions of what *machine learning* is can be attributed to artificial intelligence pioneer Arthur Samuel, who popularized the term in 1959 [38]. By paraphrasing his words we can say that

a *machine learning* algorithm uses data to make predictions or decisions without being explicitly being programmed to do so.

This definition, although limited, is enough to place learning algorithms to a conceptual distance from classical algorithms: while the classical paradigm sees a human expert (a programmer) setting a series of instructions or rules by which the machine has to operate on the input data to obtain answers, the goal of a learning algorithm is, conversely, to obtain the rules by which the data is transformed to the desired outputs, these rules can then be applied to new unseen data to produce answers.

In Figure 2.1 we see a graphical depiction of this paradigm inversion: the human designer is no longer responsible for providing algorithmic selection rules to make sense of the data, but, conversely has to provide sufficient data for a general algorithm to extract the underlying rules.

An infinite plethora of machine learning models exist, but a minimal

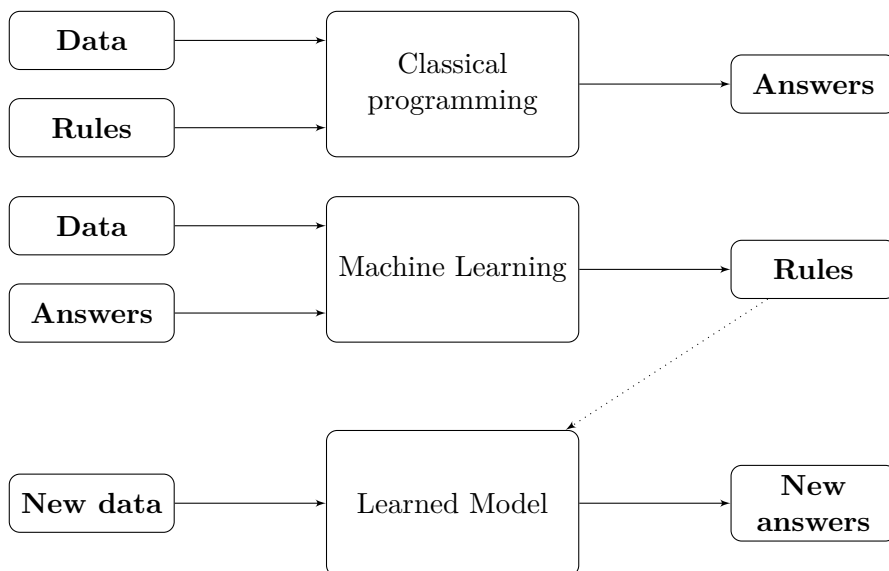


Figure 2.1: Diagram of the machine learning paradigm : While classical algorithms use pre-defined rules to extract answers from input data, Machine learning models use data and the desired outcomes to extract the rules by which new data can be processed to obtain answers.

taxonomy can be defined based on how they obtain answers and the kind of data that has to be provided.

The goal of *unsupervised* algorithms is to find some kind of structural property in the given data in order to solve tasks, which might be *clustering* the data itself (dividing the dataset in groups of samples - "clusters" - that share the same property) or *density estimation* of the probability distribution that generated the data in the first place. In all these cases the data itself is fed to the algorithm, with and no additional annotation on the datapoints is involved. The other category of machine learning algorithms is represented by *supervised* algorithms, to which Convolutional Neural Networks most of the Deep Learning area pertain. In this case each datapoint is coupled with a *label* representing

its belonging class: this extra information is needed if the algorithm's goal is to create an implicit mapping between features of the data points and the space in which labels are defined in order to make predictions of class membership on new and unseen examples.

2.2 Artificial Neural Networks

Among the *supervised* models, there's a interesting class of machine learning models that takes the name of *Artificial Neural Network* or *ANNs* which serves as the basis for Convolutional Neural Networks. The term *artificial neural networks* refers to the fact that their working is based on the functioning of biological neurons in the visual cortex. Theoretical foundations of ANNs can be traced back to early 40s works on neurophysiological models [16, 27] that led, in the late 50s, to the first classifier model based on those premises: Rosenblatt's Perceptron [36].

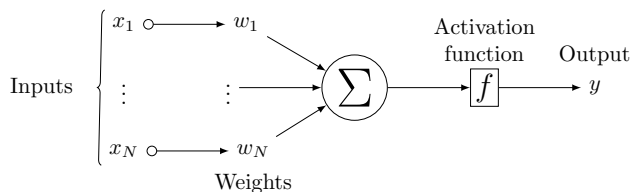


Figure 2.2: Rosenblatt's Perceptron Functional Scheme

The idea behind the Perceptron, schematically represented in Fig 2.2, was simple but effective: similarly to how a real neuron receives inputs from synapses and propagates conditional output along its axon, the Perceptron has a number of inputs $\vec{x} = (x_1, \dots, x_d)$ and produces responses as a thresholded linear combination of those with weights $\vec{w} = (w_1, \dots, w_d)$. Single units can be stacked together using non-linear

activation functions to obtain Multi Layer Perceptrons (MLPs), (any linear combination of a linear classifier like the Perceptron wouldn't benefit from unit stacking): despite their simplicity good classification performances can be obtained by MLPs and they can still be found in modern networks as final classification layers.

2.3 Convolutional Neural Networks

Convolutional Neural Networks [19] define a type of Artificial Neural Network-like model that makes use of the *convolution* operation where MLP layers would use direct matrix multiplication. In the MLP case the inputs of each neuron layer is obtained as a matrix multiplication of the outputs of the previous layer with a weight matrix, in CNNs the inputs of a layer are generally a result of a convolution operation with a weight *kernel* matrix.

2.3.1 CNN Properties

Convolutional Neural Networks have been extremely successful in many fields but particularly excel computer vision and speech recognition, and more generally in all cases where the data has a regular structure and is defined in a grid-like manner. This is surely the case of 2D, 3D and multimodal images where the data is inherently defined in a matrix-like structure, but can also be the case of time-serieses which can be seen as monodimensional grid structures.

There are three direct advantages that stem from the nature of convolution operations itself. The first one is *sparse connectivity*: in MLPs the inputs of each neuron are dependent on the outputs of each neuron of the previous layer, in CNNs this dependence is limited only to a small portion, determined by the width of the convolution filter. A neuron in CNNs will typically only have local influence on the next layer, mean-

ing that they're particularly fit for detecting local feature. Moreover, with geometrically fewer connections than MLPs, less computational power and storage space will be needed operate and store the model. Computational gains with respect to fully-connected MLPs are actually extremely significant and allow for far deeper networks with a fraction of the computational cost.

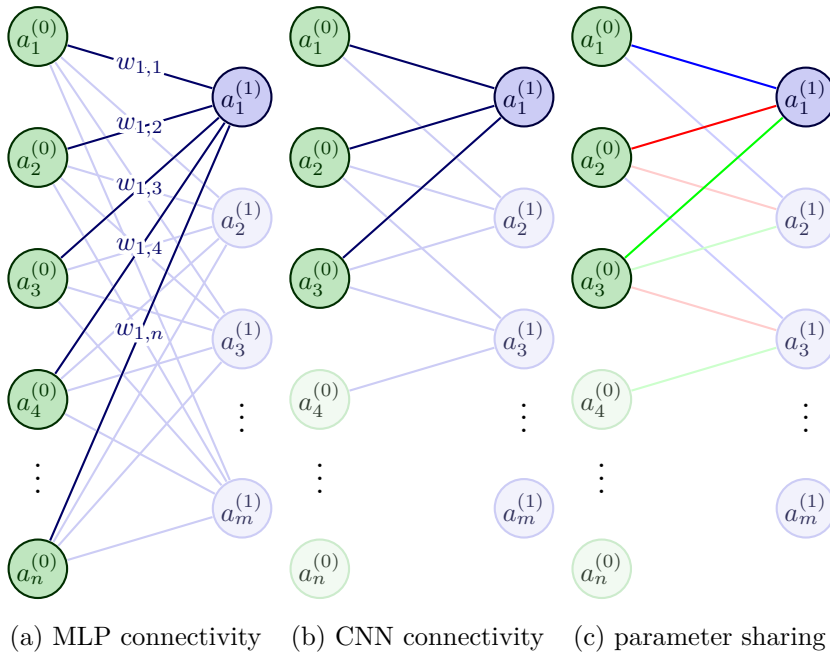


Figure 2.3: Sparse Connectivity, Parameter Sharing: in the first case the inputs of neuron $a_1^{(1)}$ are determined by the outputs of all the neurons of previous layer and the weights $w_{1,1} \dots w_{1,n}$, in the second example $a_1^{(1)}$ only depends on the outputs of a small portion of neurons determined by the size of the convolution kernel. In the third panel we have a visual representation of parameter sharing: connections with the same color share the same parameters in the convolution operation.

A second advantage of CNNs is *parameter sharing*: in MLPs each

neuron of the inputs is connected to every neuron in the outputs with a unique weight value. In CNNs we only have a limited number of convolution kernel parameters which is used across the entire layer: in other terms, the Toeplitz matrix representing the discrete convolution has a number of tied weights and the number of learnable parameters corresponds exactly to the dimensionality of the convolution kernel.

A consequence of sharing weights in convolution is the *translational equivariance* property: if $f(\vec{x})$ is the layer function and $t()$ is a translation operation, we have that $f(t(\vec{x})) = t(f(\vec{x}))$. If a convolution operation specializes in finding certain local features in a small pixel neighborhood, applying it to the entire input layer results in a 2D map of said feature, this is useful in deep networks where low level features are typically extracted in the lowest layers.

2.4 CNNs in Cell Detection and Segmentation

The brief introduction the about the working principles of Convolutional Neural Network models serves as a functional basis for us to talk about the kind of segmentation models we've developed to solve the challenge of automating the creation of whole-brain cell censuses.

Fluorescence Microscopy Imaging is not able - by its own - to give quantitative information about the distribution and shape of neuronal structures. Stereological analysis of neuronal distributions based on manual counting / segmentation are extremely lengthy and their extrapolations on large volumes rely on sample uniformity assumptions that are often not entirely met.

The need for automated ways of analyzing large quantities of imaging data has led in the years to the adoption of numerous approaches, differentiated by their specific goal. *Cell detection* is the task of roughly

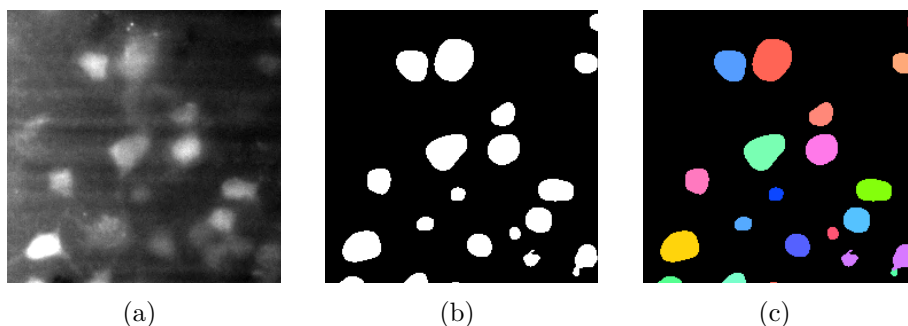


Figure 2.4: Semantic Segmentation vs Instance Segmentation: Semantic segmentation is the task of determining a belonging class for every pixel of the image (white for neuron, black for background) instance segmentation adds the difficulty of determining not only to which class the pixel belongs to, but also to the specific instance of that class it pertains to.

positioning individual nuclei inside a three-dimensional volume, *semantic segmentation* aims at a per-pixel classification of all the points of the imaging dataset, without discerning individual cells but only belonging classes (neuron / background) and *instance segmentation* represents the combination of the two, aiming at tracing a label mask for each individual instance of a detected object. The other criteria of differentiation between methods is represented by their operating principles: we can generally distinguish between the two macro-categories of classical image algorithms or machine learning models.

Many of the commonly used methods in microscopy imaging analysis employ combinations of classical image processing algorithms such as watershed segmentation, thresholding and level-set methods [24, 25, 32, 48], however these schemes have historically proven to suffer from performance drawbacks in imaging cases with low signal-to-noise ratio or blurred cell separation. In many instances these cell detection schemes

require manual tuning of parameters regulating expected cell size, nucleus shape and cell density [45]. Acceptable results are generally obtainable only for small datasets: this class of approaches inevitably falls short on scalability expectations to large samples. Moreover, the assumptions in image structure made in classical image processing methods are not always met in real life scenarios. Two instances of this could be the fact that thresholding methods like the Otsu method [31] assume bimodality of intensity value histograms or the expectation of clear separability of boundaries of many region growing approaches: these conditions are virtually never universally met in large microscopy datasets.

The continuous evolution of the Machine Learning and Deep Learning panorama has made available a large number of cell detection and segmentation models that outperform classical imaging methods [4]. The passage from a classical image processing paradigm to Deep Learning-based approaches has enabled the creation of many software packages which are readily available to the neuroscience and bio-imaging communities at large in the form of standalone solutions such as Ilastik [3, 42] and CellProfiler [6] or plugins for the Fiji/ImageJ platform [39], which usually require manual labeling of a few representative images from the experiment and produce inference results on the rest of the dataset. The viability and user-friendliness of these applications had a significant role in encouraging widespread adoption of CNN-based methods outside the machine-learning / computer-vision specialist niche, allowing the general life-science community to familiarize with them.

The performance gap between classical Machine Learning methods and Deep Learning-based ones - and consequently their almost-exclusive adoption in current-day image analysis - might be better explained by reviewing the fundamental differences between the two.

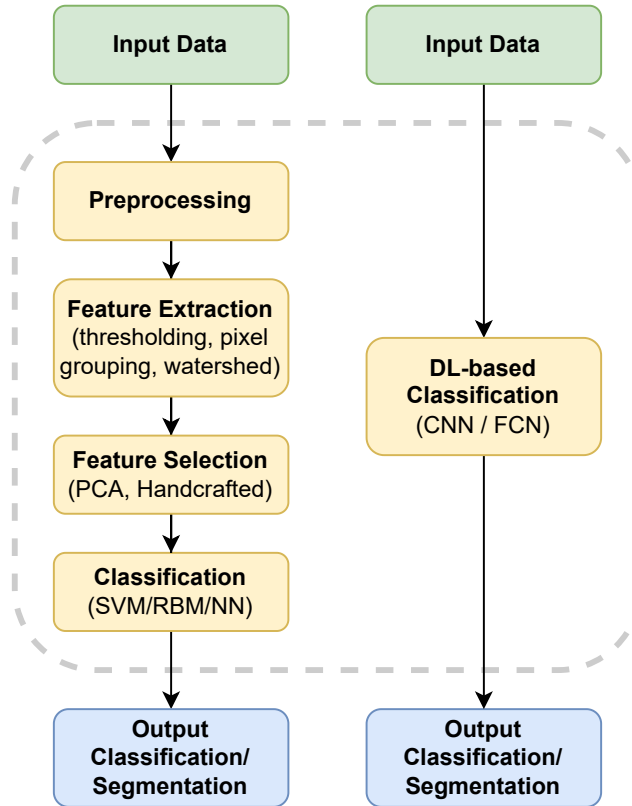


Figure 2.5: Classical ML-based Semantic Segmentation vs DL-based Segmentation: classical semantic segmentation are organized on a general workflow where a feature extraction step is needed before classification. In deep CNNs there’s no need for ”manual” feature extraction as the first layers of the network optimize automatically to act as feature extraction stages without a need for manual tuning.

2.4.1 Classical ML Approaches

Classical Machine Learning-based image segmentation approaches adopt a roughly standardized general schema: raw input data is usually split into overlapping ROIs, elaborated with a pre-processing stage which

can be represented by noise-filtering schemes, image normalization, and contrast-enhancement steps. A feature extraction and selection stage follows, where the relevant features of the image are identified and summarized, either using statistical methods like Principal Components Analysis - PCA or by application-specific handcrafted combinations of the extracted features. An infinite variety of designs in feature extraction methods has been proposed, ranging from simple mathematical operations as as applying a threshold to extremely more complex methods based on pixel grouping, morphological elaboration, watershed, level-set, and trainable models. The last step of the elaboration chain is usually a form of trainable classifier such as a Support Vector Machine SVM, a Restricted Boltzmann Machine RBM or an Artificial Neural Network ANN, which uses the extracted features to determine the class of the central pixel of the selected ROI. Parallel application of this scheme to a set of ROIs covering the entire image allows for segmentation of whole imaging planes: this approach is also referred to as a *sliding window* approach, as the input of the model are serially defined by a selection window sliding onto the original data. The resulting matrix of class predictions can be used directly as a segmented version of the original image.

2.4.2 Convolutional Neural Networks

Convolutional Neural Network extend the concept of Artificial Neural Networks to include convolutional learnable filters, pooling layers and nonlinear activations along with the *fully connected* organization schemes of ANNs. A CNN classifier, in its most recognizable form, is comprised of a series of convolutional filters and pooling layers which reduce the inputs to a minimal vector representation - much like the feature extraction and selection step in classical machine learning protocols - and a series of fully connected layers terminated by an activation function

takes the role of a classification stage.

This kind of unified approach rendered obsolete the need for complicated handcrafted feature extractors, as the lower layers of CNNs themselves implicitly act as highly efficient feature extractors during training [18]. The learned internal representations of the inputs before the fully connected stages tend to be much more reliable predictors of the output classes than any equivalently handcrafted feature. Furthermore, if the network has enough layers and representational capability, even basic preprocessing of the input data becomes implicitly learnable: a single Deep Learning model is capable to directly infer output classes from the raw data itself without the need for the definition of multiple intermediate overly tunable steps, while achieving better performance than classical machine learning methods.

2.4.3 Convolutional Interpretation of Sliding Windows

Interestingly, the CNN model approach described above still results in a *sliding window* scheme where the inputs are divided into overlapping patches and the network is trained to express class membership of their central pixel. Naive application of this approach - ex. by deploying an instance of the same model for each classification patch - comes with great computational overhead due to the unavoidable redundancy of information on the overlapping input regions. As observed in [41], the formal equivalence between fully connected layers of d units and convolutions with a $1 \times 1 \times d$ kernel can be used to efficiently evaluate responses over extended inputs [22]. By acknowledging this fact the sliding windows problem can be reformulated by replacing every fully connected layer with its 1×1 kernel equivalent. A CNN modified in this fashion is said to be operating in *fully convolutional* mode: this network doesn't just output a single classification value corresponding to the center of a patch, but arbitrarily sized inputs can easily be converted to segmen-

tation maps through the series of convolutions and pooling filters. If the rest of the network is comprised by only convolutional and pooling filters, the output dimensions are determined by their number and type: this introduces the concept of *native stride* of the classifier as the stride at which a sliding window sized as the inputs of the original network, should ideally be moved across the image plane in order to produce outputs with the same shape as those produced in fully convolutional operation. Acknowledging the existence of an intrinsic stride of the network, it's still possible to compensate it by artificially up-sampling the input image using bi-linear or nearest neighbor interpolation [26].

2.4.4 Fully Convolutional Networks

Traditional CNN models classifiers estimate the probability that the central pixel of a patch belongs to a specific class, however, the progressive compression and refinement of information along the series of convolution filters can result in loss of local spatial information. Moreover, high-range spatial correlations between features larger than the network's receptive field are completely lost in this kind of formulation.

An alternative approach to sliding windows is represented by *fully convolutional networks*. FCNs, introduced by Long et al. [23] employ transposed convolutions as feature upsampling stages, so that they're able to directly produce segmentation maps with the same size as the inputs instead of single probability values. This work was the first to propose an end-to-end fully convolutional training scheme for a segmentation network, but is still limited by loss of long-range spatial information due to the spatial compression of the feature maps before upsampling.

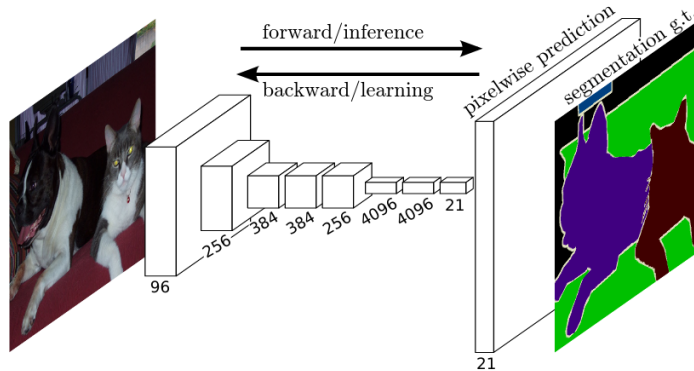


Figure 2.6: The FCN Architecture (from Long et al. [23]): the FCN model employs transposed convolutions for up-scaling spatially compressed feature maps to obtain segmentation maps of the the same size as the inputs.

2.4.5 Convolutional Networks with Graphical Models

Another solution to the loss of spatial context resulting from both sliding window approaches and the original FCN proposed by Long et al. [23] is to couple probabilistic graphical models such as Markov Random Fields MRFs or Conditional Random Fields CRFs to the outputs of a normal CNN classifier: it was shown by Chen et al. [8] that responses from the final layer of a CNN classifier, while not localized enough to be directly usable as a semantic segmentation map, still retained enough spatial information to train a CRF to retrieve accurate segmentations. Other works employ various levels of integration with different training schemes between CNNs and CRFs for semantic segmentation [8, 21, 40, 47]. This kind of approach produces better results than using sliding window classifiers alone, but can't still account entirely for the spatial information loss which is intrinsic to the network architecture.

2.4.6 Encoder-Decoder Networks

The need for retrieval of spatial information in segmentation networks pushed towards the development of strategies to decompress high-level internal representations without excessive losses in localization. Most DL-based segmentation models now employ an *encoder-decoder* architecture where semantically relevant information is extracted in an *encoding* path made of convolutional and padding layers, and spatial information is reconstructed in a *decoding* path made of deconvolution (transposed convolutions) and *unpooling* layers. Both transposed convolution and unpooling layers functionally serve as upscaling methods which take low resolution feature maps and output higher resolution tensors. Unpooling can be viewed as an approximate inverse of the pooling operation, which creates an high resolution version of the inputs by placing its values in fixed locations of the outputs, the remaining pixels can then be filled with zeros to produce what's called a *bed of nails* unpooling operation or assigned values using a nearest-neighbor policy. Transposed convolutions are upscaling filters obtained by transposing a convolution Toeplitz matrix, literature can refer to them as *upconvolutions* or *deconvolution* layers.¹

An initial model by Noh et al [30] used the VGG-16 network as encoder and a decoding path made of deconvolution and unpooling layers, this work was later refined by the SegNet model [2] by binding the pooling indices in the encoding path to the unpooling indices in the decoder, removing the necessity of learning them from scratch.

The U-Net [35] model - represented in 2.7 from the original paper - is considered one of the most influential models in the biomedical image

¹The latter name can be misleading because it might suggest inverse convolution, which is not the function of these layers do: the general inspiration for this name is to be found in the fact that deconvolutional layers act in the "opposite" way of convolutions in terms of tensor shape transformations.

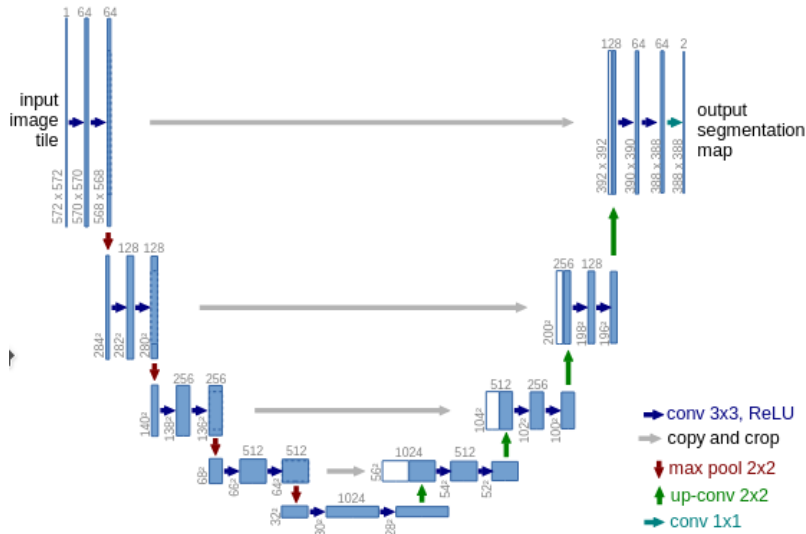


Figure 2.7: The UNet Architecture (from Ronneberger et al. [35]):U-Net introduces horizontal skip connections to solve the problem of preserving high-frequency spatial information retrieval, by concatenating the outputs of convolutional blocks in the encoding path to symmetrical positions in the decoding path, spatial information is propagated to produce detailed segmentation maps.

segmentation field. It introduces skip-connections between symmetrical parts of the contraction and expansion paths (tensor concatenation of the inputs of an upsampling stage with the outputs of the corresponding block in the contracting path to allow direct signal propagation in the network without introducing computational complexity. Various models were built on top of the U-Net architecture, among them are the extension of U-Net to 3D images [9] by the same authors, the integration of residual learning [46] within encoding blocks and nested U-Net architectures [49]. V-Net [28] was introduced roughly at the same time of U-Net and exploits the same encoder-decoder structure with horizontal

skip connections and applies it to 3D medical image segmentation. Later in this work we'll introduce a model for semantic segmentation of Light Sheet Microscopy images that we based on the general U-Net architecture, introducing additional elements from residual learning and adding dynamical module sizing capabilities.

2.4.7 Instance Segmentation

Two main approaches can be distinguished in solving the cell segmentation problem: a first top-down class of methods performs cell detection first and then refines an initial rough estimation of the cell shape using object detection networks like YOLO [33, 44], R-CNN [12, 34] to predict bounding boxes for the detected objects, which are then refined via semantic segmentation of pixels inside the bounding box, e.g. MASK-RCNN [15, 17, 25]. Most of these methods use non-maximum-suppression to avoid multiple detections of the same object if there's significant bounding box overlap. A limiting factor to the efficacy of this approach is the possible inadequacy of axis-aligned bounding boxes to correctly describe neuronal shapes. A second class of approaches, to which the methods presented in this work pertain to, exploits an alternative *bottom-up* perspective where semantic segmentation is performed on the inputs to produce per-pixel (or per-voxel) classification maps with learned classifiers, such as random forest classifiers [43], SVMs, Convolutional Neural Networks or semantic segmentation-specific networks like U-Net [4, 5, 7, 9, 10, 13, 26, 35], and subsequently apply pixel grouping strategies to distinguish between different object instances.

Bibliography

- [1] Charu C. Aggarwal. *Neural Networks and Deep Learning*. Cham: Springer, 2018. ISBN: 978-3-319-94462-3. DOI: 10.1007/978-3-319-94463-0.
- [2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation*. Oct. 2016. DOI: 10.48550/arXiv.1511.00561. URL: <http://arxiv.org/abs/1511.00561> (visited on 12/06/2022).
- [3] Stuart Berg et al. “ilastik: interactive machine learning for (bio)image analysis”. In: *Nature Methods* 16.12 (2019), pp. 1226–1232. ISSN: 1548-7105. URL: <https://doi.org/10.1038/s41592-019-0582-9>.
- [4] Juan C. Caicedo et al. “Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images”. en. In: *Cytometry Part A* 95.9 (2019), pp. 952–965. ISSN: 1552-4930. DOI: 10.1002/cyto.a.23863. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cyto.a.23863> (visited on 12/02/2022).
- [5] Yue Cao et al. “DenseUNet: densely connected UNet for electron microscopy image segmentation”. en. In: *IET Image Processing* 14.12 (2020), pp. 2682–2689. ISSN: 1751-9667. DOI: 10.1049/iet-ipr.2019.1527. URL: <https://ietresearch.onlinelibrary>.

- wiley.com/doi/abs/10.1049/iet-ipr.2019.1527 (visited on 09/06/2021).
- [6] Anne E. Carpenter et al. “CellProfiler: image analysis software for identifying and quantifying cell phenotypes”. In: *Genome Biology* 7.10 (Oct. 2006), R100. ISSN: 1474-760X. DOI: 10.1186/gb-2006-7-10-r100. URL: <https://doi.org/10.1186/gb-2006-7-10-r100> (visited on 12/02/2022).
- [7] Hao Chen et al. *DCAN: Deep Contour-Aware Networks for Accurate Gland Segmentation*. Apr. 2016. DOI: 10.48550/arXiv.1604.02677. URL: <http://arxiv.org/abs/1604.02677> (visited on 12/02/2022).
- [8] Liang-Chieh Chen et al. *Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs*. June 2016. DOI: 10.48550/arXiv.1412.7062. URL: <http://arxiv.org/abs/1412.7062> (visited on 12/06/2022).
- [9] Özgün Çiçek et al. “3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation”. en. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*. Ed. by Sebastien Ourselin et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 424–432. ISBN: 978-3-319-46723-8. DOI: 10.1007/978-3-319-46723-8_49.
- [10] Yuxin Cui et al. “A Deep Learning Algorithm for One-step Contour Aware Nuclei Segmentation of Histopathological Images”. In: *Medical & Biological Engineering & Computing* 57.9 (Sept. 2019), pp. 2027–2043. ISSN: 0140-0118, 1741-0444. DOI: 10.1007/s11517-019-02008-8. URL: <http://arxiv.org/abs/1803.02786> (visited on 05/05/2022).

- [11] David A. van Dyk and Xiao-Li Meng. “The Art of Data Augmentation”. In: *Journal of Computational and Graphical Statistics* 10.1 (2001), pp. 1–50. ISSN: 1061-8600. URL: <https://www.jstor.org/stable/1391021> (visited on 10/21/2022).
- [12] Ross Girshick et al. *Rich feature hierarchies for accurate object detection and semantic segmentation*. Oct. 2014. DOI: 10.48550/arXiv.1311.2524. URL: <http://arxiv.org/abs/1311.2524> (visited on 12/02/2022).
- [13] William J. Godinez et al. “A multi-scale convolutional neural network for phenotyping high-content cellular images”. eng. In: *Bioinformatics (Oxford, England)* 33.13 (July 2017), pp. 2010–2019. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btx069.
- [14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. en. MIT Press, Nov. 2016. ISBN: 978-0-262-03561-3.
- [15] Kaiming He et al. *Mask R-CNN*. Jan. 2018. DOI: 10.48550/arXiv.1703.06870. URL: <http://arxiv.org/abs/1703.06870> (visited on 12/02/2022).
- [16] D. O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. en. Psychology Press, Apr. 2005. ISBN: 978-1-135-63190-1.
- [17] Jeremiah W. Johnson. “Adapting Mask-RCNN for Automatic Nucleus Segmentation”. In: *arXiv:1805.00500 [cs]* 944 (2020). DOI: 10.1007/978-3-030-17798-0. URL: <http://arxiv.org/abs/1805.00500> (visited on 09/08/2020).
- [18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. Vol. 25. Curran Associates, Inc., 2012. URL: <https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html> (visited on 01/19/2023).

- [19] Y. LeCun et al. “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Computation* 1.4 (Dec. 1989), pp. 541–551. ISSN: 0899-7667. DOI: 10.1162/neco.1989.1.4.541.
- [20] Yann A. LeCun et al. “Efficient BackProp”. en. In: *Neural Networks: Tricks of the Trade: Second Edition*. Ed. by Grégoire Montavon, Geneviève B. Orr, and Klaus-Robert Müller. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 9–48. ISBN: 978-3-642-35289-8. DOI: 10.1007/978-3-642-35289-8_3. URL: https://doi.org/10.1007/978-3-642-35289-8_3 (visited on 08/07/2019).
- [21] Guosheng Lin et al. *Efficient piecewise training of deep structured models for semantic segmentation*. June 2016. DOI: 10.48550/arXiv.1504.01013. URL: <http://arxiv.org/abs/1504.01013> (visited on 12/06/2022).
- [22] Min Lin, Qiang Chen, and Shuicheng Yan. “Network In Network”. In: *arXiv:1312.4400 [cs]* (Dec. 2013). URL: <http://arxiv.org/abs/1312.4400> (visited on 08/17/2019).
- [23] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully Convolutional Networks for Semantic Segmentation”. en. In: (), p. 10.
- [24] R.A. Lotufo, A.X. Falcao, and F.A. Zampiroli. “IFT-Watershed from gray-scale marker”. In: *Proceedings. XV Brazilian Symposium on Computer Graphics and Image Processing*. Oct. 2002, pp. 146–152. DOI: 10.1109/SIBGRA.2002.1167137.
- [25] Martin Maška et al. “Segmentation and shape tracking of whole fluorescent cells based on the Chan-Vese model”. eng. In: *IEEE transactions on medical imaging* 32.6 (June 2013), pp. 995–1006. ISSN: 1558-254X. DOI: 10.1109/TMI.2013.2243463.

- [26] G. Mazzamuto et al. “Automatic Segmentation of Neurons in 3D Samples of Human Brain Cortex”. In: *Applications of Evolutionary Computation*. Ed. by Kevin Sim and Paul Kaufmann. Cham: Springer International Publishing, 2018, pp. 78–85. ISBN: 978-3-319-77538-8.
- [27] Warren S. McCulloch and Walter Pitts. “A logical calculus of the ideas immanent in nervous activity”. en. In: *The bulletin of mathematical biophysics* 5.4 (Dec. 1943), pp. 115–133. ISSN: 1522-9602. DOI: 10.1007/BF02478259. URL: <https://doi.org/10.1007/BF02478259> (visited on 06/18/2019).
- [28] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. “V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation”. In: *arXiv:1606.04797 [cs]* (June 2016). URL: <http://arxiv.org/abs/1606.04797> (visited on 07/03/2019).
- [29] Kevin P. Murphy. *Machine learning : a probabilistic perspective*. Cambridge, Mass. [u.a.]: MIT Press, 2013. ISBN: 978-0-262-01802-9 0-262-01802-0. URL: https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr_1_2?ie=UTF8&qid=1336857747&sr=8-2.
- [30] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. *Learning Deconvolution Network for Semantic Segmentation*. May 2015. DOI: 10.48550/arXiv.1505.04366. URL: <http://arxiv.org/abs/1505.04366> (visited on 12/06/2022).
- [31] Nobuyuki Otsu. “A Threshold Selection Method from Gray-Level Histograms”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (Jan. 1979), pp. 62–66. ISSN: 2168-2909. DOI: 10.1109/TSMC.1979.4310076.

- [32] Dirk Padfield, Jens Rittscher, and Badrinath Roysam. “Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis”. eng. In: *Medical Image Analysis* 15.4 (Aug. 2011), pp. 650–668. ISSN: 1361-8423. DOI: 10.1016/j.media.2010.07.006.
- [33] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. May 2016. DOI: 10.48550/arXiv.1506.02640. URL: <http://arxiv.org/abs/1506.02640> (visited on 12/02/2022).
- [34] Shaoqing Ren et al. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. Jan. 2016. DOI: 10.48550/arXiv.1506.01497. URL: <http://arxiv.org/abs/1506.01497> (visited on 12/02/2022).
- [35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv:1505.04597 [cs]* (May 2015). URL: <http://arxiv.org/abs/1505.04597> (visited on 05/13/2019).
- [36] F. Rosenblatt. “The perceptron: a probabilistic model for information storage and organization in the brain”. eng. In: *Psychological Review* 65.6 (Nov. 1958), pp. 386–408. ISSN: 0033-295X.
- [37] D. E. Rumelhart, J. L. McClelland, and PDP Research Group, eds. *Parallel Distributed Processing. Volume 1: Foundations*. Cambridge, MA: MIT Press, 1986.
- [38] Arthur L. Samuel. “Some studies in machine learning using the game of Checkers”. In: *Ibm Journal of Research and Development* (1959), pp. 71–105.
- [39] Johannes Schindelin et al. “Fiji: an open-source platform for biological-image analysis”. en. In: *Nature Methods* 9.7 (July 2012), pp. 676–682. ISSN: 1548-7105. DOI: 10.1038/nmeth.2019. URL: <https://www.nature.com/articles/nmeth.2019> (visited on 12/02/2022).

- [40] Alexander G. Schwing and Raquel Urtasun. *Fully Connected Deep Structured Networks*. Mar. 2015. DOI: 10.48550/arXiv.1503.02351. URL: <http://arxiv.org/abs/1503.02351> (visited on 12/06/2022).
- [41] Pierre Sermanet et al. “OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks”. In: *arXiv:1312.6229 [cs]* (Dec. 2013). URL: <http://arxiv.org/abs/1312.6229> (visited on 08/13/2019).
- [42] Christoph Sommer et al. “Ilastik: Interactive learning and segmentation toolkit”. In: *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. Mar. 2011, pp. 230–233. DOI: 10.1109/ISBI.2011.5872394.
- [43] Vladimír Ulman et al. “An objective comparison of cell-tracking algorithms”. In: *Nature Methods* 14 (2017), p. 1141. URL: <http://dx.doi.org/10.1038/nmeth.4473>.
- [44] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. July 2022. DOI: 10.48550/arXiv.2207.02696. URL: <http://arxiv.org/abs/2207.02696> (visited on 12/02/2022).
- [45] Fuyong Xing and Lin Yang. “Robust Nucleus/Cell Detection and Segmentation in Digital Pathology and Microscopy Images: A Comprehensive Review”. eng. In: *IEEE reviews in biomedical engineering* 9 (2016), pp. 234–263. ISSN: 1941-1189. DOI: 10.1109/RBME.2016.2515127.
- [46] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. “Road Extraction by Deep Residual U-Net”. en. In: *IEEE Geoscience and Remote Sensing Letters* 15.5 (May 2018), pp. 749–753. ISSN: 1545-

- 598X, 1558-0571. DOI: 10.1109/LGRS.2018.2802944. URL: <http://arxiv.org/abs/1711.10684> (visited on 08/25/2019).
- [47] Shuai Zheng et al. “Conditional Random Fields as Recurrent Neural Networks”. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015, pp. 1529–1537. DOI: 10.1109/ICCV.2015.179. URL: <http://arxiv.org/abs/1502.03240> (visited on 12/06/2022).
- [48] Xiaobo Zhou et al. “A novel cell segmentation method and cell phase identification using Markov model”. eng. In: *IEEE transactions on information technology in biomedicine: a publication of the IEEE Engineering in Medicine and Biology Society* 13.2 (Mar. 2009), pp. 152–157. ISSN: 1558-0032. DOI: 10.1109/TITB.2008.2007098.
- [49] Zongwei Zhou et al. *UNet++: A Nested U-Net Architecture for Medical Image Segmentation*. July 2018. DOI: 10.48550/arXiv.1807.10165. URL: <http://arxiv.org/abs/1807.10165> (visited on 12/07/2022).

Chapter 3

Experimental and Computational Methods for High-Res Volumetric Mapping of Large Brain Samples

3.1 Physical Principles of Fluorescence Microscopy

The microscopy techniques we're using to produce imaging of brain areas rely on the physical phenomenon of fluorescence. In this section we're giving a brief introduction to the main physical principles in Fluorescence Microscopy.

Fluorescent molecules have the property of being able to absorb photons at a certain energy and emit photons at a lower energy: this feature is particularly useful for microscopy applications as it enables the ob-

ervation of signal from interest molecules in a different frequency range than the one of the illumination source.

The Jablonski Diagram, in Figure 3.1 is a useful tool for visualizing the electronic states of a molecule and the main transition processes between them, which are photon absorption, internal conversion, fluorescence, intersystem crossing and phosphorescence.

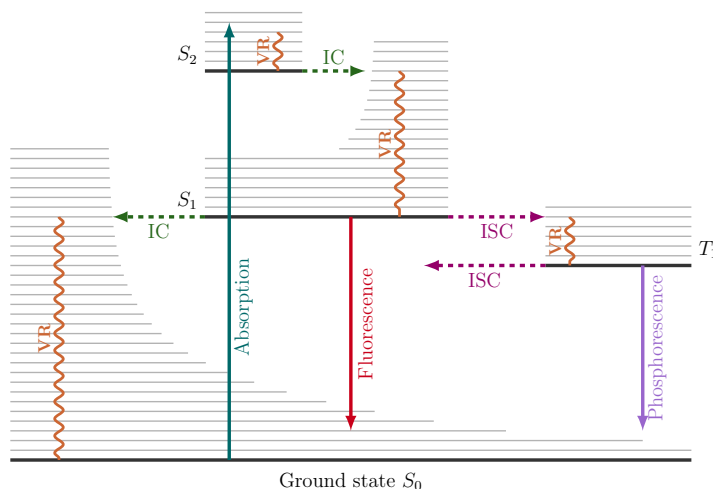
3.1.1 The Jablonski Diagram


The singlet states are denoted as S_n , where S_0 is the ground singlet state, while the triplet states are T_n ; each of these electronic states has associated vibrational levels.


The absorption of a photon (represented with a green vertical arrow) causes an electron from the molecule ground state S_0 to be promoted to a new orbital with the same spin multiplicity, typically the first excited state S_1 , in a timescale of 10^{-15} s. From here different processes can occur resulting emission of a photon - in the case of *radiative processes* - or not - in this case we are talking about a *nonradiative* process.

Internal conversion refers to a nonradiative transition between two states at same multiplicity. This process is followed by vibrational relaxation towards the lowest vibrational state of the target electronic state, excess energy is dissipated in the form of heat in the surrounding environment. In cases where the absorption event promoted the electron to a vibrational state with larger energy than the first electronic state, internal conversion is preceded by vibrational relaxation towards its first vibrational level. This process occurs in 10^{-12} s or less, whereas fluorescence processes have timescales of 10^{-10} s to 10^{-7} s, therefore internal conversions are generally complete before fluorescence.

Because excited singlet states S_n with $n > 1$ generally relax very rapidly to the lowest vibrational level of the first excited state S_1 via non-



 Internal Conversion, $S_i \rightarrow S_j$ non radiative transition.

 InterSystem Crossing, $S_i \rightarrow T_j$ non radiative transition.

 Vibrational Relaxation.

Figure 3.1: The Jablonski Diagram is a schematical representation of a molecular system's electronic and vibrational configurations, along with the possible transition processes connecting them. Thick and thinner lines denote, respectively the system's electronic and vibrational states. Radiative transitions, indicated by straight vertical arrows, involve the absorption or emission of a photon. Vibrational relaxation of an excited vibrational state to its ground state is noted with wavy lines, horizontal green dotted lines represent internal conversion, i.e. the coupling of a vibrational state of an electronic excited state to another state with same multiplicity. Horizontal purple dotted lines indicate inter-system crossing i.e. a transition to a state with different spin multiplicity (singlet to triplet or triplet to singlet).

radiative processes, fluorescence - defined as the radiative spin-allowed relaxation to a lower level - can typically only be observed between S_1 and S_0 , therefore its characteristics are not dependent on the excitation wavelength. The fluorescence photon is emitted at lower energy than the absorbed photon due to the energy loss in vibrational relaxation processes.

A concurrent pathway for fluorescence and internal conversion is *intersystem crossing*: this refers to the nonradiative transition between isoenergetic states with different spin multiplicity, i.e. from singlet to triplet states. This kind of transition is, in principle, forbidden by selection rules but spin-orbit coupling can be large enough to make it relevant. After intersystem crossing further radiative relaxation to the S_0 state may happen in a *phosphorescence* event: again, the transition from a triplet state to S_0 is only possible through spin-orbit coupling terms, resulting in very low radiative rates. Phosphorescence spectra are located at higher wavelengths than fluorescence because the T_1 state is located at a lower energy than S_1 . The lifetimes of triplet states are extremely long - anywhere from 10^{-6} s to the order of seconds - and during this time collisions with other molecules can favor intersystem crossing and nonradiative vibrational relaxation to S_0 .

3.2 Two-Photon Confocal Microscopy - TPFM

3.2.1 Two-Photon Excitation and its Advantages in Microscopy

Multi-photon excitation is a non-linear process involving the simultaneous absorption of two or more photons in a single event, in particular, *two-photon excitation* processes (noted as *2PE* as opposed to *1PE* single photon excitation) require the cooperation of two low-energy pho-

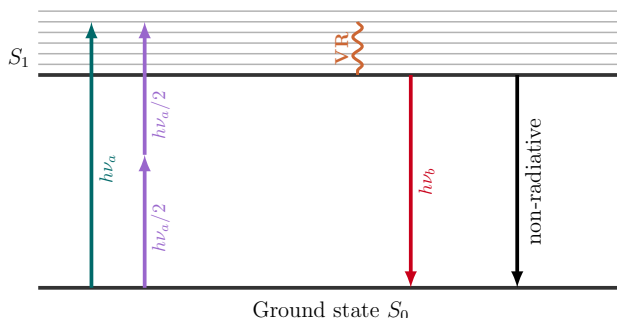


Figure 3.2: Two-Photon Excitation: this simplified version of the Jablonski diagram depicts the simultaneous absorption of two photons to produce a single excitation event.

tons to cause an higher-energy electronic transition: in figure 3.2 we have a simplified version of the Jablonski diagram where 2PE events are represented by two consecutive arrows, the same transition to a vibrational state of S_1 can be achieved by absorption of a single photon of energy $h\nu_a$ or the simultaneous absorption of two photons of energy $\frac{h\nu_a}{2}$, in either case the fluorescence emission would be a single photon of energy $h\nu_b$ equal to the energy gap between S_0 and S_1 .

The two-photon excitation process, originally predicted by Maria Goppert-Mayer in 1931 [22], follows different selection rules than the single-photon case, since the interaction of the two photons with the fluorophore needs to be almost simultaneous (in a timeframe of 10^{-16} s), the absorption rate depends quadratically on the light intensity instead of linearly (hence the non-linearity). This is actually a favorable property for optical imaging. Let's take the case of a focused beam passing inside a photo-luminescent medium: focusing the beam reduces its size and locally increases the light intensity near the focal point, having the absorption rate depending on the squared intensity limits the excited areas of the sample to a small locality around the focal point, avoiding

the excitation of out-of-plane targets, thus increasing the overall contrast of the image.

In confocal setups the signal is generated by the entire depth of the sample, independently of the position of the actual focal plane, and a pinhole is used as a spatial filter to reject out-of-focus background. With Two-Photon Microscopy only the in-focus portion of the sample emits actually absorbs radiation and emits signal, removing the need for a pinhole filter in the light path. Smaller exposed areas also translate to reduced photobleaching (photochemical alterations of fluorophores such that they lose their fluorescence properties) and photodamage (biological damage to the tissues due to electromagnetic interaction), whereas in single-photon microscopy the entire thickness of the sample would have been subject to the same detrimental effects even if data was only collected from the focal plane. Other advantages over single-photon techniques are represented by the use of low IR wavelengths that better penetrate the tissue compared to visible frequencies used in single-photon microscopy, because of the reduced scattering absorption by endogenous chromophores.

3.2.2 2PE Confocal Setup

A 2PE setup is generally similar to a confocal microscope and consists of three main elements: a laser illumination source, a scanning system and a detector. The main differences between a confocal setup and a two-photon setup are the absence of a pinhole for spatial filtering and the illumination source.

Illumination Sources for 2PE

The possibility of 2PE absorption events was originally postulated by Goppert-Mayer in 1931, but experimental observations didn't come until 1961 - 30 years later - after the invention of laser sources: the best arc lamps light sources at the time, focused through the best objective lenses, would only have produced - in the best-case scenario - a single 2PE event every few minutes. According to her original theory, the intensities needed for reliably observe 2PE events would have been six orders of magnitude higher than those needed for the same number of single photon events: this kind of intensity wasn't only impossible to achieve at the time but would have also precluded every application to biological samples. It is the invention of ultrashort mode-locked lasers that provide high peak power while maintaining sufficiently low average emission that finally enabled, in 1990 a practical application of 2PE to imaging of biological samples [17]. *Mode-locked* lasers produce pulses of light with extremely short duration, this is achieved by inducing a fixed phase relationship between the standing wave frequencies (modes) of the laser's resonant cavity so that a train of pulses can be obtained by means of constructive interference. The resulting light train is characterized by a repetition rate dictated by the distance between the two cavity mirrors

$$\Delta\nu = \frac{c}{2L} \quad (3.1)$$

where c is the speed of light and L is the distance between the mirrors.

The temporal confinement provided by the laser pulse, alone, is not enough to allow for a number of 2PE events suitable for microscopy imaging: spatial confinement of the excitation photons is also needed for obtaining suitably high densities. This is achieved using a high NA (numerical aperture) lens that focuses the laser beam to a spot of roughly

300nm in diameter, yielding a 10^5 -fold greater photon density than what is seen with a 1mm laser beam diameter.

3.3 Light Sheet Fluorescence Microscopy

Light-Sheet Microscopy bridges the gap between the small scale, high-resolution imaging domain in hundreds-of-micrometer scales, which is usually covered by confocal microscopy techniques, and the centimeter scope which can be explored tomographically at relatively low resolution. Light-Sheet microscopy applications can achieve cellular resolution on samples in the millimeter-to-centimeter scale in times that are negligible with respect to point scanning methods, posing themselves as the go-to solution for fast high-resolution imaging of extended specimens.

Light-Sheet Microscopy setups feature two different orthogonal optical axes: the first provides illumination with a laser light sheet and the other is used for wide-field detection of the fluorescence signal. Illumination with a planarly collimated source ensures optical sectioning, as fluorescence is only generated in the exposed plane. Moreover, the physical dimensions of the light sheet can be adjusted in thickness and horizontal extent to obtain the desired levels of optical sectioning and FOV coverage. As the fluorescence signal is generated simultaneously across the entire FOV, signal is rapidly collected in a parallel way using camera sensors: since there's no scanning involved, the pixel resolution of the obtained images is only limited by the number of pixels of the used sensor.

A clear advantage of Light-Sheet microscopy over confocal techniques is the heavily reduced photo-bleaching: while in confocal microscopy it is in fact necessary to illuminate the entire sample volume to image a single plane, in Light-Sheet microscopy light exposition - and, conse-

quently, photo-damage and photo-bleaching - is limited to the imaged plane itself. The acquisition of z-stacks in confocal setups would require the sample to be illuminated as many times as the number of images in the stack, multiplying the above described effects, while this problem is simply nonexistent in Light-Sheet setups.

Illumination Sources

Planar illumination of the sample can be achieved by using cylindrical optics (SPIM [28]): in this setup light converges at the position of the sample and diverges away from it, in such a way that the intersection area between the sample and the beam is approximately planar. Different optics could be used to adapt the physical size of the light sheet to accommodate for different kind of subjects, choosing thinner planes ($\sim 1 \mu\text{m}$) for small specimens, and thicker ones ($\sim 5 \mu\text{m}$ to $10 \mu\text{m}$) for large specimens where absorption and non-uniformity of the illumination plane might be an issue, however a lower limit to the thickness of the illumination sheet is imposed by diffraction. An alternative way to produce light-sheet illumination is by using a linearly collimated beam and scanning it horizontally, this is the main concept behind DLSSM (Digital Scanned Light-Sheet Microscopy) [32] and derived methods. The inherent benefits of this kind of approach are increased uniformity of the illumination intensity profile, as well as the ability to digitally control the height of the light sheet itself. The scanned nature of this approach implies that, at a given time, only a fraction of the total image is actually illuminated: this means that local light intensities need to be significantly higher (~ 300 fold) with respect to the SPIM case in order to achieve comparable fluorescence levels across the whole image. The higher intensities involved in DLSSM methods can result in increased degradation of the sample, posing a trade-off between these effects and

the increased illumination uniformity.

3.4 Optical Clearing

3.4.1 Clearing Methods

Both Two-Photon Confocal Microscopy and Light-Sheet microscopy allow extensive investigation of the three-dimensional structure of biological samples: these technologies enable volumetric imaging by means of optical sectioning, intrinsically derived from the localized excitation area in the case of TPFM or from planarly collimated illumination in the case of LSM but, on the other hand, the effectiveness of application of these techniques to real-world samples is hampered by the natural opaqueness of biological specimens. Generally speaking, opaqueness of samples is a consequence of high scattering of their biological components, rather than absorption, which characterizes only a subset of tissues such as blood, muscles and liver. A biological sample can be naively modeled as constituted by molecules with high refractive indexes (mostly lipids and proteins), estimated to have a refractive index $n_{\text{dry}} = 1.50$ [25] immersed in a low refractive-index medium (water) with a refractive index $n_{\text{water}} = 1.33$: the significant difference between the two results in multiple scattering events and, ultimately, opacity.

Optical clearing methods pursue the objective of rendering the sample optically transparent by matching as closely as possible the refractive index n_{dry} of the biological components to that of the surrounding medium. This can be done either by substituting water with other mediums with different refractive indexes or by modifying n_{dry} , either removing some of the scattering components, such as lipids, or by altering their optical properties. Depending on the particular clearing protocol (more than 20 distinct methods have been proposed during the

years), the final refractive indexes of the cleared samples n_{clear} can range anywhere from 1.38 [25] to 1.56 [3].

Different clearing methods can be identified by their working principle. The first category of protocols is based the substitution of the water medium with organic solvent clearing agents with refractive indexes matching n_{dry} . The BABB protocol [18] uses a mixture of benzyl benzoate and benzyl alcohol as clearing agent, since benzyl benzoate is insoluble in water, the sample is first dehydrated in ethanol. While ethanol removes water, also some loss of lipids can happen, the result is a fairly homogeneous sample, as proteins have a refractive index greater than 1.5 (higher than both water and lipids) with high refractive index. However, this has been observed to have a detrimental effect on fluorescent proteins, resulting in the almost complete loss of the fluorescence properties. As a solution to this problem the BABB method has been modified to use peroxide-free tetrahydrofuran (THF) to dehydrate the tissue and peroxide-free dibenzyl ether (DBE) as clearing agent ($n = 1.56$) [3, 44] [19], which has shown better success at maintaining fluorescence properties of the cleared tissue. Solvent-based clearing protocols are known to be robust and work in a variety of tissue types, however, the fact that many solvents are toxic, and their ability to dissolve glues used in the construction of objective lenses, along with fluorescence quenching and significant shrinkage of the tissue that occurs during dehydration (sometimes up to 50% [3]), all contribute to reduce the effective practicability of these methods.

A second class of protocols aims at preserving fluorescence by keeping the sample in an aqueous environment so that fluorescent proteins whose properties are dependent on their hydration state are not excessively affected by the quenching effects of non-aqueous clearing agents.

Some of these methods use high concentrations of sugars such as sucrose [31] ($n = 1.44$) or fructose ($n = 1.48$) [15, 27]. Clearing solutions obtained via this method have the main drawback of high viscosity, to solve this issue, procedures have been proposed where sugar solutions are substituted with water-soluble compounds like 2,2'-thiodiethanol (TDE) [1, 15, 49] ($n = 1.42$). There are also proprietary solutions whose complete compositions are not disclosed like FocusClear™[38](by Cel-Explorer Labs, Taiwan)($n = 1.45$). Refractive indexes obtainable by water-based solutions, though, are generally characterized by a maximum value of $n = 1.48$ and depending on the actual n_{dry} refractive index of the dry tissue, the use of aqueous clearing agents might be limiting. The problem can be tackled from a double perspective, both by optically matching the clearing medium and by acting on the n_{dry} , either by removal of lipids with a non-hydrophobic solvent or lowering the effective refractive index of proteins. The first work to make use this mechanism was Scale [25], where an extensive incubation with a water-soluble detergent (Triton™X-100) was used for lipid removal, while urea was used to denature and hyper-hydrate the proteins, reducing the overall refractive index of the sample in a glycerol medium down to $n = 1.38$. CUBIC [50] uses a similar urea-based hyper-hydration strategy, in this case immersion in a high-refractive index sucrose solution was used as an additional step to expedite the clearing process $n = 1.48$.

The removal of lipidic content does indeed provide smaller refractive indexes, but the detergents used to achieve it often cause significant losses in proteins. Additionally, very hydrophilic substances can cause protein denaturation, hence hampering IHC-useful epitopes. Another category of protocols for tissue clearing addresses these issues by stabilizing the proteic content by means of cross-linking proteins to a gel mesh before lipid removal. This fourth category goes under the name of

tissue transforming protocols. The CLARITY [14] model, which paved the way to other works in the category, uses paraformaldehyde to cross-link proteins and nucleic acids to a polyacrylamide gel, which is then treated to sodium dodecyl sulfate (SDS) for lipid removal, the sample is then immersed in FocusClear™ to make it transparent. In general it's possible to speed up the lipidic removal step down to days by using electrophoresis instead of weeks of passive incubation in SDS.

3.4.2 Applicability to Fluorescence Microscopy

Two-Photon Confocal Microscopy

TPFM setups are characterized by fine optical sectioning capabilities which are reached by using high NA objectives, in fact the degree of reachable optical sectioning is a function of the numerical aperture of the focusing objective. As a consequence, the emergence of spherical aberration effects represents a limiting factor for TPFM, which needs objectives which are specifically corrected for the refractive index of employed immersion medium. Commercial availability of clearing medium-specific objective lenses is one of the binding criteria in the choice of a protocol for TPFM.

Thanks to the nonlinear nature of 2PE excitation, diffused illumination radiation does not significantly contribute to background signal, as a result TPFM tends to perform better in deep tissue when compared to single-photon excitation methods and LSFM, so that extreme transparency of the cleared tissue is not as binding as in other applications. However, TPFM images are obtained using a scanning system (usually galvanometric mirrors) and imaging of large samples can span on significant amounts of time: the clearing method has to assure stability of the mechanical, optical and fluorescent properties of the sample across the whole temporal extension of the acquisition, which can reach several

weeks depending on the sample.

Light Sheet Microscopy

When compared to point-scanning TPFM, the wide-field detection scheme of Light Sheet Microscopy offers faster volumetric imaging. The planar illumination required for optical sectioning can either be provided by cylindrical optics or by scanning of a single illumination line. The linear nature of excitation in LSFM makes it more susceptible to diffusion than TPFM. Samples that are to be cleared for LSFM need to be as transparent as possible in order to minimize scattering events and improve image quality. In general, organic solvents provide optimal results in terms of tissue transparency but their application is again limited by commercial availability of specific-medium optics.

3.5 Data Processing Pipeline

Working with the large data-streams originated by Fluorescence Microscopy Imaging poses a series of technical challenges which are mostly unencountered in the vast majority of computer-driven applications. Stream sizes from multichannel Light-Sheet Microscopy are about three orders of magnitude higher than those involved in conventional Confocal Microscopy and can easily reach multiple Gbit s^{-1} and extremely intensive CPU and GPU loads are expected: in such configurations accurate planning and management of the data flows needs to be taken into account as any step of the elaboration path risks to become a significant bottleneck in the overall process.

In table 3.1 we offer a reference for the estimated data sizes of mouse, monkey and human brain acquisitions: it can be easily seen that data processing challenges emerging from the sheer physical size of human brain specimens require significantly different solution strategies than

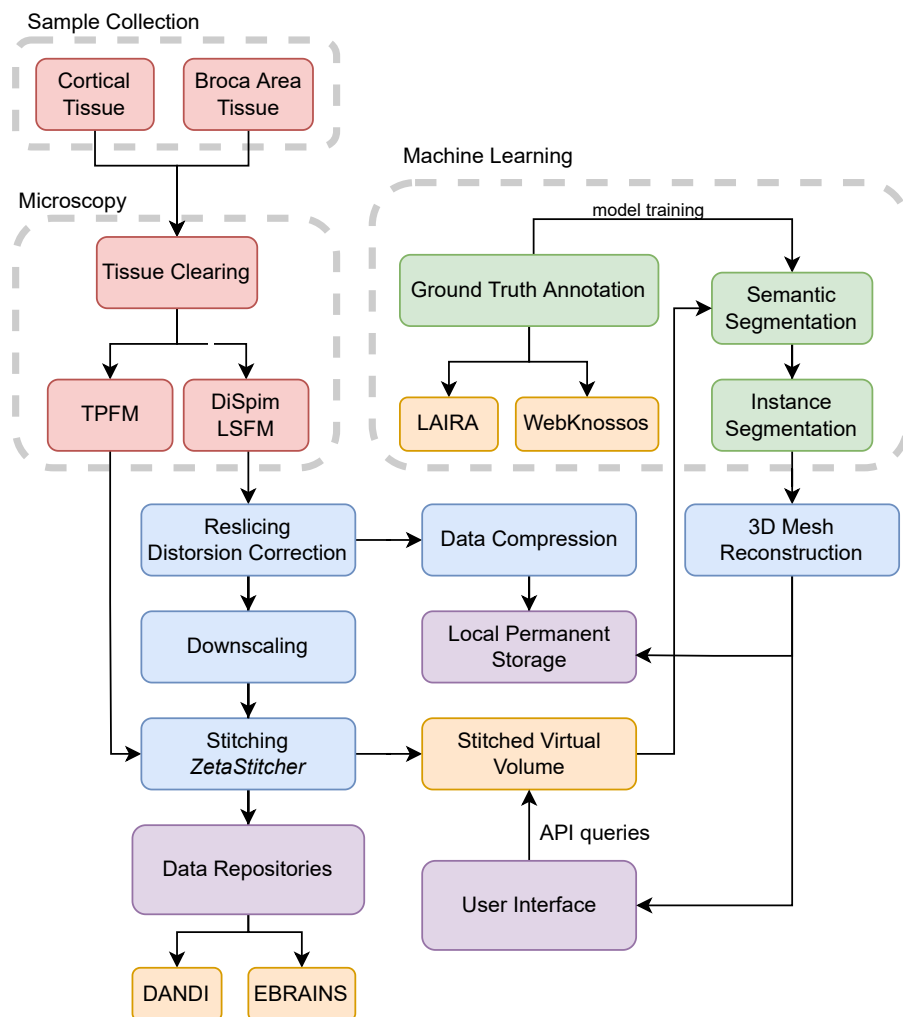


Figure 3.3: Data Processing Pipeline: this representation summarizes the steps involved in our unified pipeline for acquisition, analysis, processing, storage and sharing of microscopy data.

the animal case. Assuming an isotropic resolution of just $10\ \mu\text{m}$ per pixel, acquisition of a $1\ \text{cm}^3$ volume of tissue results in $\frac{1\ \text{cm}^3}{(10\ \mu\text{m})^3} = 10^9$

Configuration	1 cm ³ (Mouse Brain)	100 cm ³ (Monkey Brain)	1500 cm ³ (Human Brain)
1 channel, 10 μ m resolution	2 GB	200 GB	3 TB
3 channels, 10 μ m resolution	6 GB	600 GB	9 TB
1 channel, 1 μ m resolution	2 TB	200 TB	3 PB
3 channels, 1 μ m resolution	6 TB	600 TB	9 PB

Table 3.1: Data Size Estimations for SPIM Brain Acquisitions

voxels, which, when represented by 16bit floats, end up taking 2 GB of disk space. Increasing spatial acquisition resolutions to 1 μ m with three different channel streams, in the case of the acquisition of a mouse brain, results in total storage needs that could be satisfied with a single consumer-level HDD, while acquiring a whole human brain at the same resolution would require multiple Petabyte-sized infrastructures and hardware investments in the order of hundreds of thousands of euros (as of 2022).

The delicate orchestration of data flows to create a reproducible processing model is, in fact, the nodal point of this work. The need for a scalable architecture with reduced bottlenecks has driven the choices of every single algorithmic component of the pipeline: in some cases heavier algorithms with promising accuracy performances have been discarded in favor of faster methods which enabled the overall computational process to be executed within timing limitations.

The soft target that was held in designing this pipeline was to reach a computational setup that would run the needed processing in roughly the same time that was needed to acquire biological samples, so that physical process of microscopy acquisition of the biological samples wouldn't be limited by computing times and data transfers.

3.5.1 Stitching Datasets

Extended specimens cannot be microscopically acquired in their entirety in a single acquisition as the their physical extent is generally incompatible with the microscope’s field of view: it’s usually needed to mount the sample on one or more moving digitally-controlled stages while multiple overlapping views of the subject are serially recorded and the complete sample is reconstructed from the single acquisitions.

While it’s important to have accurate positioning of the traslational stages, a number of factors can contribute to discrepancy between the recorded position of the stage to the actual coordinates in the sample reference, such as the stages having higher positioning error than the spatial resolution or limited temporal stability of the sample itself. Simply positioning the acquisition stacks to their nominal stage coordinates in the sample space more often than not results in visual distortion of the reconstructed acquisition. This issue can be solved with an *a posteriori* approach, where the nominal positions of the acquisition stacks are used to determine pairs of adjacent stacks and the actual relative displacements are calculated by maximizing the cross-correlation of the overlapping sections.

To perform tile registration we use a custom developed software package, *ZetaStitcher* [40], which first calculates the relative displacement between pairs of images, then computes an optimal placement of the stacks in the global image space. The partial overlap between stacks is managed via blending the two different acquisitions in the overlapping area. Other microscopy and fluorescence microscopy-specific image registration methods exist in literature, such as [26], which is based on the *ImageJ / Fiji* software package, [54], built on *Insight ToolKit - ITK*, [12]

One of the most interesting features of *Zetastitcher* is an API interface for virtualized access to the aligned volume, which can be generated

on the fly by retrieving the optimized image registration parameters and stack positions in the image space, without ever needing to generate a global image volume, which could be extremely impractical to store and, ultimately, redundant: the imaging stacks can be individually stored in a compressed format and portions of the overall stitched volume can be accessed querying a specially constructed *VirtualFusedVolume* class in *ZetaStitcher*, which can decompress and fuse the relevant image sections on-the-fly.

3.5.2 Semantic Segmentation Framework: NeuroSegmenter

Approaching a segmentation task from the perspective of a microscopist can be a less-than-pleasant experience. Users without an extensive coding background often turn to pre-packaged solutions like Ilastik [4] or ImageJ-Fiji plugins which offer a coding-free experience, but the need for user interaction limits their applicability in real-time pipelines with continuous data production. The more experienced programmer who wants full control on the segmentation models can turn to various deep learning framework solutions like *TensorFlow* [39], *PyTorch* [43] or *JAX* [8] which, on the other hand, require some experience both in generic *Python* programming and in the concepts and coding paradigms defined by the libraries themselves (graph-based programming, autograd-style differentiation and GPU acceleration in general) but grant the advantage of extreme flexibility in every aspect of model and pipeline definition. The general user in microscopy application is often in a middle position between these two poles and can feel the limitations of a pre-packaged solution but don't have the level of coding expertise required to fully manage custom training and inference loops. High-level interfaces and libraries like *Keras* [13] (the official high-level functional library of *TensorFlow*) are now bundled in the main deep learning frameworks but, while extremely useful in speeding up the prototyping and production of

large deep learning pipelines, often can't lower the difficulty bar enough for the microscopist with occasional scripting ability.

When starting a new ML project, a ML specialist usually has to set up a lot of boilerplate code: this includes basic data processing pipelines (preprocessing, data streaming and data augmentation), custom training and inference loops and the definition and implementation of a deep learning model. Next he has to tune it properly and, optionally, set up an experiment tracking system to monitor model training. Usability of such solutions by non-expert personnel entirely depend on the presence of a simple user interface which has to be created ad-hoc from the ML specialist, adjustments to the model often require coding experience, making the continuous presence of a specialist in the loop an essential element to the working of the pipeline.

During the course of my PhD I developed a solution - *NeuroSegmenter* - to target this middle usebase, which is characterized by general coding literacy but have limited proficiency in the specifics of Deep Learning. *NeuroSegmenter* relies on *TensorFlow* and *Keras* as back-end and aims at providing an efficient, yet entirely customizable deep learning segmentation experience, flexible enough for a machine learning specialist to set it up yet remaining easy to use enough for microscopists to operate and tune autonomously without necessarily having to interact with the code.

From a software design perspective, three guiding principles behind the development of *NeuroSegmenter* are modularity, extendibility and ease of use. Specifically, the module exposes a number of independently implemented components which can be extended for the specific application. A *DataGenManager* component is responsible for the network input pipeline, loading the input data, preprocessing and exposing gen-

erators for data augmentation which will be used during training. By default, integration with *ZetaStitcher* is provided, but any data format and input pipeline can be implemented as an extension of the *DataGenManager* component. The actual CNN model is defined by a *ModelManager* component, the use of a configurable component enables dynamic parametrical definition of the network architecture without interacting with the actual code. The loss function, along with the optimizer and their parameters are managed by a *OptimizerManager* component decoupled and independent from the actual model definition, while a *TrackingManager* component handles metric logging and integration with remote experiment tracking services: compatibility with Weights&Biases [6] is provided out of the box but custom callbacks can be defined for integration with any other service. Inference over large volumes of data in a tiled fashion is managed by *TileBasedPredict* so that the end user, independently of the size of the network's receptive field, can directly receive segmentation results with the same dimensionality as the inputs without worrying about implementing their own reconstruction pipeline. Lastly, if ground truth data is provided, model performance is measured by a *PerformanceEvaluator* component.

The whole training and inference processes with every single of the above mentioned components are configured by the end user with a single human-readable *YAML* configuration file, model definition can take just a few lines, here follows a configuration example:

```
1 # training_config.yml
2 model_cfg:
3     model: "neuroresunet2d"
4     input_shape: [256, 256]
5     unet_depth: 2
6     base_filters: 16
```

```
7     batch_normalization: true
8     transposed_convolution: false
9     residual_preactivation: true
10
11 dataset_cfg:
12     dataset_path: /mnt/ssd1/datasets/broca_spim/
13     mode: "stack"
14     n_channels: 1
15     positive_class_value: 1
16     negative_class_value: 255
17
18 training_cfg:
19     mode: "2d"
20     epochs: 100
21     batch_size: 10
22     loss: "binary_crossentropy"
23     track_metrics:
24         - "accuracy"
25         - "jaccard_index"
26         - "dice_coefficient"
27 optimizer_cfg:
28     optimizer: "adam"
29     learning_rate: 0.00067
30     beta_1: 0.9
31     beta_2: 0.999
32     epsilon: 0.000001
33     amsgrad: false
```

Once defined the configuration file, the training process can easily be started by the end user with a single command

```
castelli@localhost:~$ neuroseg_train train_config.yml
```

NeuroSegmenter is available on GitHub for download [10] as a python package and a *Docker* image for easy and repeatable deploy on production machines.

3.6 Sharing Datasets with the Neuroscience Community

Data availability and the ability to allow the general scientific community to conduct independent analyses on our datasets is an essential priority of every international project we might contribute to. Solving the difficulties related to number of generally underestimated factors including the sheer size of the acquired datasets, data compression, the complexity related to using custom data formats and metadata inclusion, can be properly considered a research question in itself. Our research efforts in this direction, in the context of sharing the datasets we're generating, resulted firstly in my involvement in the collaborative definition of an extension of the Brain Imaging Data Structure - BIDS standard specification to specifically handle microscopy data [23], which was positively repected and integrated into the standard [Bourget, Kametsky, Ghosh, Mazzamuto, Castelli et al. 2022] [7], and in a published python library, *pyometiff* [Castelli, 2022] [11], to read and write fully standard-compliant OME-TIFF [5, 21, 37] image data that contain all the metadata fields specified in the BIDS-31 standard extension (as well as every other metdatata field in the OME-TIFF specification).

3.6.1 Extending the BIDS Standard: Microscopy-BIDS

The Brain Imaging Data Structure (BIDS) is a technical specification for the organization, sharing and archival of neuroimaging data modeled to

interoperability and reusability principles. The neuroscientific community has yet to settle on an univocally defined data sharing standard and multiple independent specifications have emerged, such as the Essential Metadata for 3D BRAIN Microscopy [45] or the Recommended Metadata for Biological Images (REMBI) [46]. To the current date, the Brain Imaging Data Structure (BIDS) standard is a well-established specification for sharing and organizing neuroscientific imaging data. Initially developed for magnetic resonance imaging (MRI) datasets, it has been extended to imaging modalities such as positron emission tomography [33], quantitative MRI(qMRI) [30] and magnetoencephalography [42], yet a specific extension aimed at 2D and 3D microscopy application was still missing.

The Microscopy-BIDS extension proposal that I contributed to [7] extends the BIDS specification of filename formats with the addition of microscopy-specific entities such as the biological *sample*, the used *stain* and the specific data *chunk* for acquisitions spanning across different files (pre-stitching) as well as a standardized identifier for the used microscopy modality (2PE for two-photon excitation microscopy, SPIM for selective plane illumination microscopy, BF for brightfield microscopy etc.).

3.6.2 BIDS-Microscopy Example Dataset

The BEP031 extension proposal of the BIDS specification, officially incorporated into the BIDS specification since release 1.7.0 in February 2022, was complemented with two example datasets to be used as reference: the first one relative to Scanning Electron Microscopy acquisitions of ten spinal chord samples with myelin manual segmentations by Zaimi et al. [51, 52], the other is a Broca Area sample acquired using the setup

described in chapter 5 and data acquisition pipeline, as well as manual segmentations of neuronal somas[41].

Below we can see an example of the file hierarchy of the dataset. One of the most useful contribution of the Microscopy-BIDS proposal is the extension of the naming standard by introducing a `micr` data type specific to microscopy, the filenames now feature microscopy-specific entities such as `sample-<sample>`, `stain-<stain>` and `chunk-<chunk>`. Other specifications introduced by Microscopy-BIDS are a `samples.tsv` file describing sample attributes and new additional columns in the `participants.tsv` sidecar file which now also describes animal metadata such as its species and specific strain.

```
└─ data_broca-area_spim
  └─ rawdata
    └─ dataset_description.json
    └─ participants.tsv
    └─ samples.tsv
    └─ sub-I46
      └─ ses-SPIM
        └─ micr
          └─ sub-I46_ses-SPIM_sample-BrocaAreaS07_stain-
              GAD67_SPIM.json
          └─ sub-I46_ses-SPIM_sample-BrocaAreaS07_stain-
              GAD67_chunk-00_SPIM.ome.tif
          └─ sub-I46_ses-SPIM_sample-BrocaAreaS07_stain-
              NeuN_SPIM.json
          └─ sub-I46_ses-SPIM_sample-BrocaAreaS07_stain-
              NeuN_chunk-00_SPIM.ome.tif
```

3.6.3 The *pyometiff* library

Microscopy-BIDS also defines standardized file formats for storing raw microscopy data, which were selected to accommodate both 2D and 3D datasets as well as both lossless and lossy compression. Three file formats are supported by Microscopy-BIDS: non-tiled 2D data should use PNG and TIFF file formats, which benefits from wide compatibility with image visualizers and editors, while large resolution whole-slide imaging and tiled 3D data should be stored using the OME-TIFF [5] file format, developed by the Open Microscopy Environment Consortium (OME). OME-TIFF enables for storage of multi-dimensional 5D tiled data and includes metadata in a XML header (OME-XML specification) included in the OME-TIFF file. OME-TIFF data can be read and wrote using the *Bio-formats* library [37], which comes already integrated into the Fiji-ImageJ [47] distribution. Other specialized toolboxes that are able to process OME-TIFF files exist, such as *libvips* [36], *tiff file* [20] and *apeer-ometiff-library* [2], the latter two implemented in Python.

Even if the Bio-formats bundle offers a complete implementation of OME-TIFF format conversions, its reliance on the Java platform makes it cumbersome to integrate in Python-based workflows. The pre-existing Python packages (*tiff file* and *apeer-ometiff-library*) at the time of writing proved to offer either limited compatibility (*tiff file* is an astounding general tool for TIFF file handling but lacks extensive implementation of the OME-TIFF standard) or outdated standard definitions (*apeer-ometiff-library*). I personally introduced a new Python library, *pyometiff* [11], aimed at offering a complete tool for handling OME-TIFF files, updated to the latest OME-XML specifications. The library handles compilation and parsing of OME-XML headers, as well as automatically handling imaging stack dimension conversions, while using *tiff file* as a backend for low level I/O (not for handling metadata, which is entirely man-

aged by pyometiff). The tool is distributed on the PyPi repository and can effortlessly be installed and integrated in any Python workflow. It's currently seeing activity and preliminary adoption by the microscopy community [24, 29] from which it receives useful feedback to steer its continuous development, it can be considered as an open project and an incentive to the adoption of Microscopy-BIDS among microscopists who are familiar with Python in their workflows.

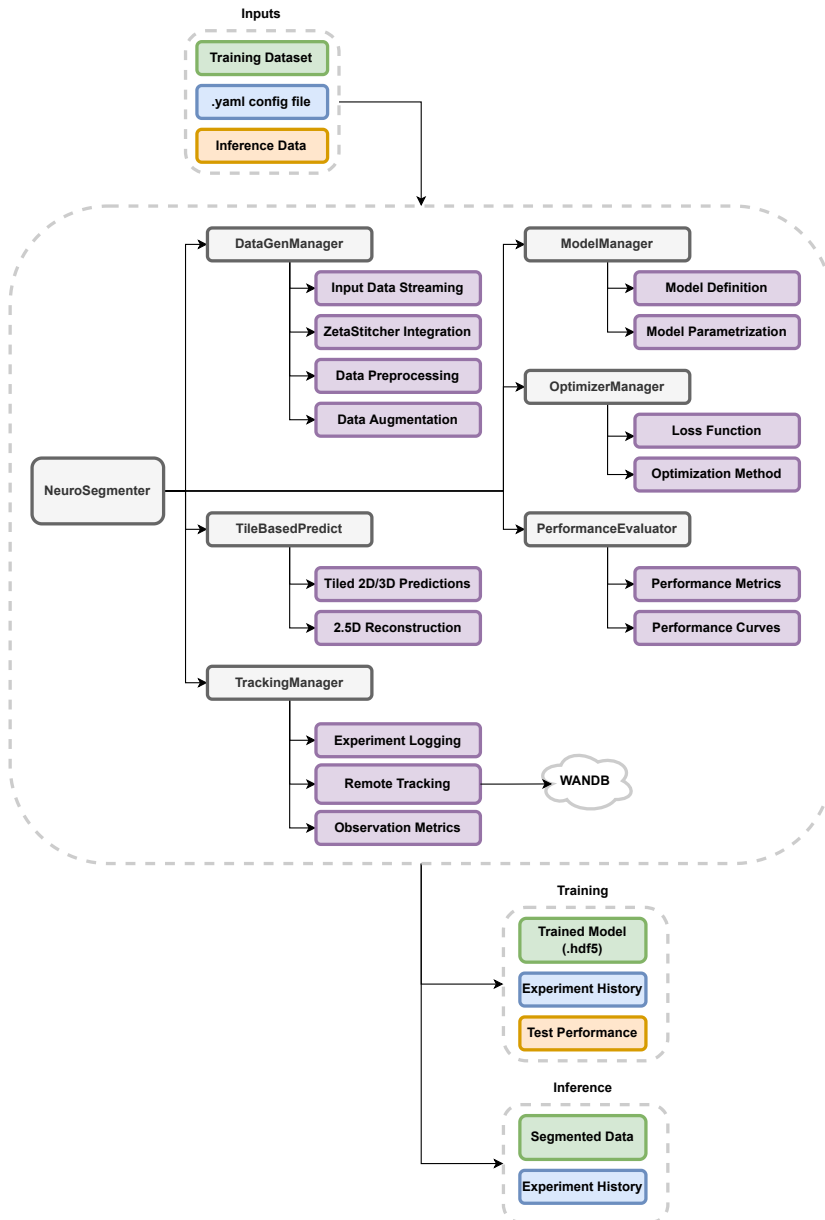


Figure 3.4: *NeuroSegmenter* working diagram: schematic representation of the different components exposed by the *neurosegmenter* module.

Bibliography

- [1] Yuka Aoyagi et al. “A rapid optical clearing protocol using 2,2'-thiodiethanol for microscopic observation of fixed mouse brain”. eng. In: *PloS One* 10.1 (2015), e0116280. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0116280.
- [2] *apeer-ometiff-library*. Dec. 2022. URL: <https://github.com/apeer-micro/apeer-ometiff-library> (visited on 01/10/2023).
- [3] Klaus Becker et al. “Chemical Clearing and Dehydration of GFP Expressing Mouse Brains”. In: *PLoS ONE* 7.3 (Mar. 2012). ISSN: 1932-6203. DOI: 10.1371/journal.pone.0033916. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3316521/> (visited on 05/09/2019).
- [4] Stuart Berg et al. “ilastik: interactive machine learning for (bio)image analysis”. In: *Nature Methods* 16.12 (2019), pp. 1226–1232. ISSN: 1548-7105. URL: <https://doi.org/10.1038/s41592-019-0582-9>.
- [5] Sébastien Besson et al. “Bringing Open Data to Whole Slide Imaging”. en. In: *Digital Pathology*. Ed. by Constantino Carlos Reyes-Aldasoro et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2019, pp. 3–10. ISBN: 978-3-030-23937-4. DOI: 10.1007/978-3-030-23937-4_1.

-
- [6] Lukas Biewald. *Experiment Tracking with Weights and Biases*. 2020. URL: <https://www.wandb.com/>.
- [7] Marie-Hélène Bourget et al. “Microscopy-BIDS: An Extension to the Brain Imaging Data Structure for Microscopy Data”. In: *Frontiers in Neuroscience* 16 (Apr. 2022), p. 871228. ISSN: 1662-4548. DOI: 10.3389/fnins.2022.871228. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9063519/> (visited on 05/18/2022).
- [8] James Bradbury et al. *JAX: composable transformations of Python+NumPy programs*. 2018. URL: <http://github.com/google/jax>.
- [9] Alessandro Bria and Giulio Iannello. “TeraStitcher - A tool for fast automatic 3D-stitching of teravoxel-sized microscopy images”. In: *BMC Bioinformatics* 13.1 (Nov. 2012), p. 316. ISSN: 1471-2105. DOI: 10.1186/1471-2105-13-316. URL: <https://doi.org/10.1186/1471-2105-13-316> (visited on 12/12/2022).
- [10] Filippo Castelli. *neurosegmenter*. Dec. 2022. URL: <https://github.com/filippocastelli/neurosegmenter> (visited on 12/20/2022).
- [11] Filippo Castelli and Giacomo Mazzamuto. *pyometiff*. Jan. 2023. DOI: 10.5281/zenodo.7520378. URL: <https://doi.org/10.5281/zenodo.7520378>.
- [12] Joe Chalfoun et al. “MIST: Accurate and Scalable Microscopy Image Stitching Tool with Stage Modeling and Error Minimization”. en. In: *Scientific Reports* 7.1 (July 2017), p. 4988. ISSN: 2045-2322. DOI: 10.1038/s41598-017-04567-y. URL: <https://www.nature.com/articles/s41598-017-04567-y> (visited on 12/12/2022).
- [13] François Chollet. *Keras*. 2015. URL: <https://github.com/fchollet/keras>.

- [14] Kwanghun Chung et al. “Structural and molecular interrogation of intact biological systems”. en. In: *Nature* 497.7449 (May 2013), pp. 332–337. ISSN: 1476-4687. DOI: 10.1038/nature12107. URL: <https://www.nature.com/articles/nature12107> (visited on 05/10/2019).
- [15] Irene Costantini et al. “A versatile clearing agent for multi-modal brain imaging”. en. In: *Scientific Reports* 5 (May 2015), p. 9808. ISSN: 2045-2322. DOI: 10.1038/srep09808. URL: <https://www.nature.com/articles/srep09808> (visited on 05/07/2019).
- [16] Irene Costantini et al. “In-vivo and ex-vivo optical clearing methods for biological tissues: review”. EN. In: *Biomedical Optics Express* 10.10 (Oct. 2019), pp. 5251–5267. ISSN: 2156-7085. DOI: 10.1364/BOE.10.005251. URL: <https://www.osapublishing.org/boe/abstract.cfm?uri=boe-10-10-5251> (visited on 09/08/2020).
- [17] W. Denk, J. H. Strickler, and W. W. Webb. “Two-photon laser scanning fluorescence microscopy”. eng. In: *Science (New York, N.Y.)* 248.4951 (Apr. 1990), pp. 73–76. ISSN: 0036-8075. DOI: 10.1126/science.2321027.
- [18] Hans-Ulrich Dodt et al. “Ultramicroscopy: three-dimensional visualization of neuronal networks in the whole mouse brain”. en. In: *Nature Methods* 4.4 (Apr. 2007), pp. 331–336. ISSN: 1548-7105. DOI: 10.1038/nmeth1036. URL: <https://www.nature.com/articles/nmeth1036> (visited on 11/24/2022).
- [19] Ali Ertürk et al. “Three-dimensional imaging of solvent-cleared organs using 3DISCO”. en. In: *Nature Protocols* 7.11 (Nov. 2012), pp. 1983–1995. ISSN: 1750-2799. DOI: 10.1038/nprot.2012.119. URL: <https://www.nature.com/articles/nprot.2012.119> (visited on 11/24/2022).

- [20] Christoph Gohlke. *cgohlke/tiffifile: v2022.5.4*. July 2022. DOI: 10.5281/zenodo.6795861. URL: <https://zenodo.org/record/6795861> (visited on 01/10/2023).
- [21] Ilya G. Goldberg et al. “The Open Microscopy Environment (OME) Data Model and XML file: open tools for informatics and quantitative analysis in biological imaging”. In: *Genome Biology* 6.5 (May 2005), R47. ISSN: 1474-760X. DOI: 10.1186/gb-2005-6-5-r47. URL: <https://doi.org/10.1186/gb-2005-6-5-r47> (visited on 01/10/2023).
- [22] Maria Göppert-Mayer. “Über Elementarakte mit zwei Quantensprüngen”. In: *Annalen der Physik* 401 (Jan. 1931), pp. 273–294. ISSN: 0003-3804. DOI: 10.1002/andp.19314010303. URL: <https://ui.adsabs.harvard.edu/abs/1931AnP...401..273G> (visited on 11/16/2022).
- [23] Krzysztof J. Gorgolewski et al. “The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments”. en. In: *Scientific Data* 3.1 (June 2016), p. 160044. ISSN: 2052-4463. DOI: 10.1038/sdata.2016.44. URL: <https://www.nature.com/articles/sdata201644> (visited on 01/10/2023).
- [24] *GradinaruLab/useqfish_analysis*. URL: https://github.com/GradinaruLab/useqfish_analysis (visited on 01/24/2023).
- [25] Hiroshi Hama et al. “Scale: a chemical approach for fluorescence imaging and reconstruction of transparent mouse brain”. en. In: *Nature Neuroscience* 14.11 (Nov. 2011), pp. 1481–1488. ISSN: 1546-1726. DOI: 10.1038/nn.2928. URL: <https://www.nature.com/articles/nn.2928> (visited on 11/23/2022).
- [26] David Hörl et al. “BigStitcher: reconstructing high-resolution image datasets of cleared and expanded samples”. en. In: *Nature Methods* 16.9 (Sept. 2019), pp. 870–874. ISSN: 1548-7105. DOI:

- 10.1038/s41592-019-0501-0. URL: <https://www.nature.com/articles/s41592-019-0501-0> (visited on 12/12/2022).
- [27] Bing Hou et al. “Scalable and DiI-compatible optical clearance of the mammalian brain”. In: *Frontiers in Neuroanatomy* 9 (2015). ISSN: 1662-5129. URL: <https://www.frontiersin.org/articles/10.3389/fnana.2015.00019> (visited on 11/28/2022).
- [28] Jan Huisken and Didier Y. R. Stainier. “Selective plane illumination microscopy techniques in developmental biology”. In: *Development (Cambridge, England)* 136.12 (June 2009), pp. 1963–1975. ISSN: 0950-1991. DOI: 10.1242/dev.022426. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2685720/> (visited on 05/13/2019).
- [29] Min Jang et al. *Spatial transcriptomics for profiling the tropism of viral vectors in tissues*. en. Jan. 2023. DOI: 10.21203/rs.3.rs-1486912/v1. URL: <https://www.researchsquare.com> (visited on 01/24/2023).
- [30] Agah Karakuzu et al. “qMRI-BIDS: An extension to the brain imaging data structure for quantitative magnetic resonance imaging data”. en. In: *Scientific Data* 9.1 (Aug. 2022), p. 517. ISSN: 2052-4463. DOI: 10.1038/s41597-022-01571-4. URL: <https://www.nature.com/articles/s41597-022-01571-4> (visited on 01/10/2023).
- [31] Meng-Tsen Ke, Satoshi Fujimoto, and Takeshi Imai. “SeeDB: a simple and morphology-preserving optical clearing agent for neuronal circuit reconstruction”. eng. In: *Nature Neuroscience* 16.8 (Aug. 2013), pp. 1154–1161. ISSN: 1546-1726. DOI: 10.1038/nn.3447.

- [32] Philipp J. Keller et al. “Digital scanned laser light-sheet fluorescence microscopy (DSLM) of zebrafish and *Drosophila* embryonic development”. eng. In: *Cold Spring Harbor Protocols* 2011.10 (Oct. 2011), pp. 1235–1243. ISSN: 1559-6095. DOI: 10.1101/pdb.prot065839.
- [33] Gitte M. Knudsen et al. “Guidelines for the content and format of PET brain data in publications and archives: A consensus paper”. eng. In: *Journal of Cerebral Blood Flow and Metabolism: Official Journal of the International Society of Cerebral Blood Flow and Metabolism* 40.8 (Aug. 2020), pp. 1576–1585. ISSN: 1559-7016. DOI: 10.1177/0271678X20905433.
- [34] Ulrich Kubitscheck, ed. *Fluorescence Microscopy: From Principles to Biological Applications*. English. 2 edition. Weinheim: Wiley-Blackwell, June 2017. ISBN: 978-3-527-33837-5.
- [35] Joseph R. Lakowicz. *Principles of Fluorescence Spectroscopy*. en. 3rd ed. Springer US, 2006. ISBN: 978-0-387-31278-1. URL: <https://www.springer.com/it/book/9780387312781> (visited on 04/13/2019).
- [36] *libvips*. URL: <https://www.libvips.org/> (visited on 01/10/2023).
- [37] Melissa Linkert et al. “Metadata matters: access to image data in the real world”. In: *Journal of Cell Biology* 189.5 (May 2010), pp. 777–782. ISSN: 0021-9525. DOI: 10.1083/jcb.201004104. URL: <https://doi.org/10.1083/jcb.201004104> (visited on 01/10/2023).
- [38] Yee-Chien Liu and Ann-Shyn Chiang. “High-resolution confocal imaging and three-dimensional rendering”. eng. In: *Methods (San Diego, Calif.)* 30.1 (May 2003), pp. 86–93. ISSN: 1046-2023. DOI: 10.1016/s1046-2023(03)00010-0.

- [39] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. URL: <https://www.tensorflow.org/>.
- [40] Giacomo Mazzamuto. *lens-biophotonics/ZetaStitcher*. Oct. 2022. URL: <https://github.com/lens-biophotonics/ZetaStitcher> (visited on 11/21/2022).
- [41] Giacomo Mazzamuto et al. *Broca's Area Light-Sheet Microscopy*. Sept. 2021. DOI: 10.5281/zenodo.5541821. URL: <https://zenodo.org/record/5541821> (visited on 01/10/2023).
- [42] Guiomar Niso et al. "MEG-BIDS, the brain imaging data structure extended to magnetoencephalography". en. In: *Scientific Data* 5.1 (June 2018), p. 180110. ISSN: 2052-4463. DOI: 10.1038/sdata.2018.110. URL: <https://www.nature.com/articles/sdata2018110> (visited on 01/10/2023).
- [43] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. URL: <https://papers.nips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html> (visited on 12/20/2022).
- [44] Douglas S. Richardson and Jeff W. Lichtman. "Clarifying Tissue Clearing". In: *Cell* 162.2 (July 2015), pp. 246–257. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.06.067. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4537058/> (visited on 11/25/2022).
- [45] Alexander J. Ropelewski et al. *Essential Metadata for 3D BRAIN Microscopy*. May 2021. DOI: 10.48550/arXiv.2105.09158. URL: <http://arxiv.org/abs/2105.09158> (visited on 01/10/2023).

- [46] Ugis Sarkans et al. “REMBI: Recommended Metadata for Biological Images—enabling reuse of microscopy data in biology”. en. In: *Nature Methods* 18.12 (Dec. 2021), pp. 1418–1422. ISSN: 1548-7105. DOI: 10.1038/s41592-021-01166-8. URL: <https://www.nature.com/articles/s41592-021-01166-8> (visited on 01/10/2023).
- [47] Johannes Schindelin et al. “Fiji: an open-source platform for biological-image analysis”. en. In: *Nature Methods* 9.7 (July 2012), pp. 676–682. ISSN: 1548-7105. DOI: 10.1038/nmeth.2019. URL: <https://www.nature.com/articles/nmeth.2019> (visited on 12/02/2022).
- [48] Ludovico Silvestri et al. “Clearing of fixed tissue: a review from a microscopist’s perspective”. In: *Journal of Biomedical Optics* 21.8 (Mar. 2016), p. 081205. ISSN: 1083-3668, 1560-2281. DOI: 10.1117/1.JBO.21.8.081205. URL: <https://www.spiedigitallibrary.org/journals/Journal-of-Biomedical-Optics/volume-21/issue-8/081205/Clearing-of-fixed-tissue--a-review-from-a-microscopists/10.1117/1.JBO.21.8.081205.short> (visited on 05/09/2019).
- [49] Thorsten Staudt et al. “2,2’-thiodiethanol: a new water soluble mounting medium for high resolution optical microscopy”. eng. In: *Microscopy Research and Technique* 70.1 (Jan. 2007), pp. 1–9. ISSN: 1059-910X. DOI: 10.1002/jemt.20396.
- [50] Etsuo A. Susaki et al. “Whole-brain imaging with single-cell resolution using chemical cocktails and computational analysis”. eng. In: *Cell* 157.3 (Apr. 2014), pp. 726–739. ISSN: 1097-4172. DOI: 10.1016/j.cell.2014.03.042.
- [51] Aldo Zaimi et al. “AxonDeepSeg: automatic axon and myelin segmentation from microscopy data using convolutional neural networks”. en. In: *Scientific Reports* 8.1 (Feb. 2018), p. 3816. ISSN: 2045-2322. DOI: 10.1038/s41598-018-22181-4. URL: <https://www.nature.com/articles/s41598-018-22181-4>

- [//www.nature.com/articles/s41598-018-22181-4](https://www.nature.com/articles/s41598-018-22181-4) (visited on 01/10/2023).
- [52] Aldo Zaimi et al. *data_axondeepseg_sem*. Nov. 2021. DOI: 10.5281/zenodo.5744487. URL: <https://zenodo.org/record/5744487> (visited on 01/10/2023).
- [53] Yanjing Zhan et al. “Organic solvent-based tissue clearing techniques and their applications”. en. In: *Journal of Biophotonics* 14.6 (2021), e202000413. ISSN: 1864-0648. DOI: 10.1002/jbio.202000413. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/jbio.202000413> (visited on 11/24/2022).
- [54] Dženan Zukić et al. “ITKMontage: A Software Module for Image Stitching”. en. In: *Integrating Materials and Manufacturing Innovation* 10.1 (Mar. 2021), pp. 115–124. ISSN: 2193-9772. DOI: 10.1007/s40192-021-00202-x. URL: <https://doi.org/10.1007/s40192-021-00202-x> (visited on 12/12/2022).

Chapter 4

Two-Photon Microscopy Analysis of the Human Cortical Samples

Studying the human brain on a quantitative basis poses challenges in every step of the process: from the choice of microscopy technique, to finding a suitable process to clear mesoscopic tissue samples, to engineering an efficient data transformation pipeline for acquisition, automated elaboration and storage of large quantities of collected data. As of today, a complete turn-key transformation process to obtain 3D spatial distributions of neurons from raw samples is not uniquely defined and every lab that is interested in making such extensive and quantitative kind of analysis has to come up with their own pipeline, depending both on their microscopy imaging, biology, machine learning and computer science expertises and the availability of suitable imaging apparatuses, elaboration and storage hardware.

In this chapter we propose an end-to-end pipeline for clearing, imag-

ing with Two-Photon Confocal Microscopy, data transformation, Deep-Learning-enabled automated neuronal mapping and visualization that allowed us to image, label, and classify neuronal distributions in 100 mm³ sized human cortical samples.

4.1 Introduction

The combination of clearing methodologies and microscopy TPFM imaging advancements has enabled volumetric reconstruction of large tissue samples by means of optical sectioning: tissue clearing is crucial for deep antigen and light penetration in the sample while Two-Photon Microscopy provides both granular spatial differentiation of the features and optimal contrast. Human tissue presents a series of challenges compared to other targets, such as variability of post-mortem fixation conditions, presence of blood inside the vessels, auto-fluorescence signal caused by lipofuscin-type pigments and the need of exogenous labeling: out of the plethora of tissue clearing protocols, tissue transformation techniques such as CLARITY [9] have proved the most success in achieving uniform and reliable transparency of this kind of samples. Approaches based on organic solvents such as iDISCO [27] were also adapted to human tissue [6, 14, 19] but the need for specific sample preparations, such as fresh-frozen samples or in-situ controlled perfusion fixation, render those methods potentially inapplicable to a more general case such as ours. A novel clearing method, SWITCH/TDE, based on the SWITCH tissue transformation protocol [26], was implemented to reliably clear and stain human brain samples.

A consideration that needs to be addressed is that the advantages we've seen lately in tissue clearing and imaging have not always been followed - in the microscopy and brain study fields - by on-par innovation

on the front of large-scale data analysis and management. In order to scale up processing capabilities of significant amounts of data produced by 3D anatomical reconstructions - that we're able to obtain by the combination of clearing techniques with high-resolution optical methods - high-throughput computational approaches are needed. Solutions in this regard need to find a Goldilocks zone defined by the concurrent needs of predictive accuracy, computational scalability and access to a sufficient amount of manual annotations for *supervised* machine learning methods that need ground truth data to be trained on.

The vast majority of approaches to cell segmentation in multi-photon microscopy we can find in literature originate from research efforts in the study of time-dependent activity of limited populations of neurons, typically in in-vivo animal targets, using two-photon calcium imaging. These research applications to functional tracing of neurons in two-photon microscopy are more-than-tangentially related to our challenge of structural mapping, as in many cases the techniques used to perform semantic segmentation of the neurons at a fixed time-point are directly applicable to static frames like the ones we produce. The main difference, though, is that we're interested in segmentation stability across an extreme variety of imaging conditions whereas time-dependent signal analysis in a limited spatial domain comparatively requires much less generality with respect to image quality and illumination conditions, making most of these approaches impractical for our purposes. Many of these methods employ classical image processing strategies such as combinations of convolutional and morphological filters [13] or adaptive thresholding [30]: the relative ease of implementation of these methods, along with their generally low computational complexity comes at the compromise of a limited spatial generalization reach and vulnerability to noise or imaging artifacts [1]. More sophisticated solutions are based on super-

vised machine learning frameworks. Many of these exploit classical ML paradigms such as SVMs [15], gaussian mixtures in combination with region growing algorithms [5] and boost classifiers [32], but way more success and popularity was encountered by DL solutions based on CNNs [33] and on fully-convolutional networks like U-Net [28] [16] [3].

We used Two-Photon Confocal Microscopy to image mesoscopic volumes at sub-micron resolution, the large multi-channel data streams produced by the microscope were first aligned and combined in three dimensions for whole-sample reconstruction using a custom tool called *ZetaStitcher* [25, 31]. The aligned volume is then automatically analyzed with the aid of a custom-designed Convolutional Neural Network model to retrieve an estimate of the volumetric distribution of neuronal somas. The choice of a relatively light-weight network architecture over heavier and deeper CNN models enables fast and on-the-fly computation of inference maps, while limiting the the same time the needs for extensive manually annotated datasets. The final step of the transformation pipeline is an *instance segmentation* algorithm which is able to reconstruct three-dimensional surface meshes of individual neuronal somas from the *semantic segmentation* probability maps obtained with the CNN model.

4.2 Methods

The wide variety of different competences and expertises in the Biophotonics Group at LENS allowed us to propose custom methodological solutions for each step of the pipeline, from the tissue clearing and labeling protocols, to the Two-Photon Microscopy apparatus, to the CNN model architecture and 3D mesh reconstruction and visualization pipelines. In this section we're presenting a quick review of the various methodologies

involved in our analysis.

4.2.1 Biological Samples

We focused our studies on four different samples of human cortical tissue from different-aged subjects (pediatric, adult and elderly), three obtained from biopsies collected during surgical interventions to cure epilepsy and one collected post-mortem. The first two samples pertain to the left prefrontal cortex of, respectively, an adult patient (Male, 67 years old) and an elderly subject (Female, 99 years old). The remaining two are a dysplastic brain sample from the left temporo-occipital cortex of a 29 year old male, operated to treat drug-resistant epilepsy due to focal cortical dysplasia Type IIa, and a dysplastic sample from the left temporo-parietal cortex of an 8 year old male operated to treat drug-resistant epilepsy due to hemimegalencephaly.

The four specimens were stored in formalin for, respectively, 6 months, 1 year, 4 years and 5 years before the study.

4.2.2 Tissue Clearing and Labeling

The reconstruction of mesoscopic volumes of human brain samples at sub-micron resolution via Two-Photon Fluorescence Microscopy requires both uniformly high transparency and homogeneous staining.

A modified version of the SWITCH tissue transformation protocol [26] combined with 2,2'-thiodiethanol (TDE) clearing [10] was developed as a method to reliably and uniformly labeling and clearing tissue from different regions, subjects and fixation conditions. The combination of these two techniques (SWITCH and TDE clearing) enabled tissue imaging with TPFM up to 1 mm depths of small fluorescent molecules like SYTOXTMGreen, while antibodies could be imaged consistently in 500 μm thick slices as shown in Figure 4.1 (c) and (d).

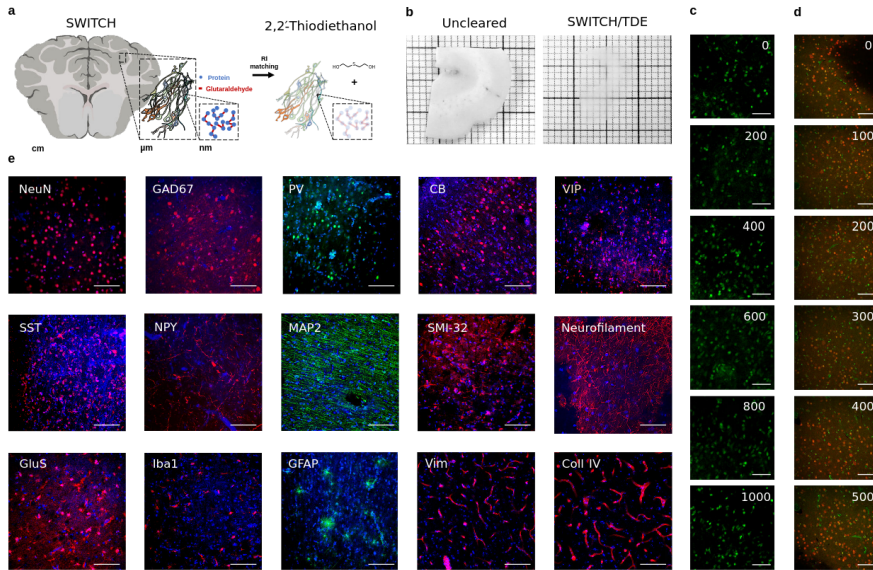


Figure 4.1: SWITCH/TDE CLearing: In (a) and (b) are, respectively, schematic representations of the SWITCH transformation protocol and the TDE clearing method. (c) column depicts imaging of SYTOX™Green at varying depths ranging from 0 μm to 1000 μm , the scale bar is 100 μm . (d) shows NeuN acquisitions at different depths, from 0 μm to 500 μm , scale bar 50 μm . In (e) we have a series of representative acquisitions of cleared tissue after immuno-staining with various antibodies and DAPI.

After clearing with the SWITCH/TDE method, the four samples were immuno-labeled with the Neuron-specific Nuclear protein (NeuN) to detect neuronal somas and with DAPI for nuclear staining.

4.2.3 Two-Photon Confocal Microscopy Setup

We used a mode-locked Ti:Sapphire laser (Chameleon by Coherent, USA) light source, characterized by 120 fs pulse width and 80 MHz repetition rate, operating at 800 nm frequency, which was coupled to a

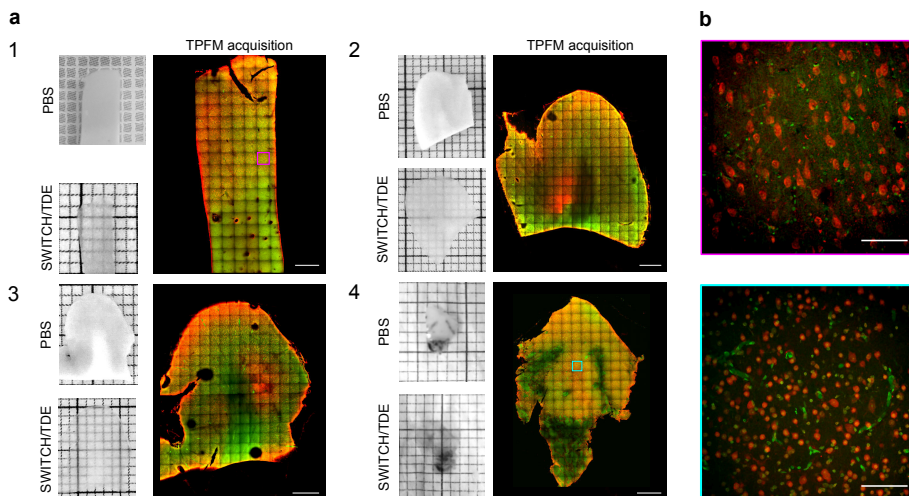


Figure 4.2: Human Cortical Samples: (a) The four imaged specimens: on the left samples are shown before and after the SWITCH/TDE clearing protocol, on the right a representative slice (200 μm depth) of the reconstructed TPFM imaged volume is depicted, scale bar = 100 μm . Red channel represents the NeuN antibody staining, green represents DAPI staining.

(b) Magnified portions of specimen 1 (magenta) and specimen 2 (cyan) showing the native resolution of the acquisition.

custom-made scanning system based on two galvanometric mirrors (LSKGG4/M by Thorlabs, CA, USA). Focusing was achieved via a refractive index tunable 25 \times objective lens (LD LCI Plan-Apochromat 25 \times /0.88 Imm Corr DIC M27 by Zeiss, Germany). Planar movement of the sample was achieved via a closed-loop piezoelectric stage (U-780 PILine XY Stage System, 135 \times 85mm travel range by Physik Instrumente, Germany) and a vertical stage was used for displacement along the z-axis (ND72Z2LAW PIFOC objective scanning system, 2 mm travel range, by Physik Instrumente, Germany). Fluorescence signal was detected by two independent GaAsP photomultiplier modules (H7422 by Hamamatsu Photon-

ics, NJ, USA). Emission filters of (440 ± 40) nm, (530 ± 55) nm and (612 ± 25) nm were used to detect signal from, respectively, DAPI, SYTOXTMGreen / Alexa 488 and Alexa Fluor®568. Custom control software for the instrument was written in LabView (National Instruments, TX, USA).

Our system was able to perform z-stack imaging for a depth of (500 ± 100) μm of overlapping adjacent regions ($40 \mu\text{m}$ overlap) with $0.88 \times 0.88 \times 2 \mu\text{m}$ resolution. The acquisition were characterized by a field of view of $450 \mu\text{m}$ and resulted in 512×512 px images, the dwell time between each acquisition was $5 \mu\text{s}$, meaning that each image could be acquired in 1.31 s and a whole $500 \mu\text{m}$ deep stack could be completed in 5.46 min, therefore 11 h to 30 h of continuous imaging were needed for each of the four samples.

4.2.4 Frame Stitching and Alignment

Mesoscopic acquisitions of large samples in TPFM can take several days, during this time temperature changes and evaporation of the mounting medium can lead to micron-scale distortion. For this reason, instead of simply tiling together the acquisitions using the nominal positioning of the motorized stages, the stacks were fused together using *ZetaStitcher* [24] to compute optimal alignment and correct small scale distortions that the sample might have undergone during the microscopy session. The software achieves this by computing cross-correlation maps at different depths for every pair of adjacent image stacks: the optimal relative displacements are determined by using a global optimization algorithm to maximize cross-correlations of individual pairs. After finding optimal positioning of the imaging stacks, whole acquisition planes can be either saved as a single TIFF file, or accessed through a custom API provided by *ZetaStitcher* that allows us to perform queries to the whole fused acquisition without the need of duplicating already existing data.

4.2.5 CNN-based Neuronal Segmentation

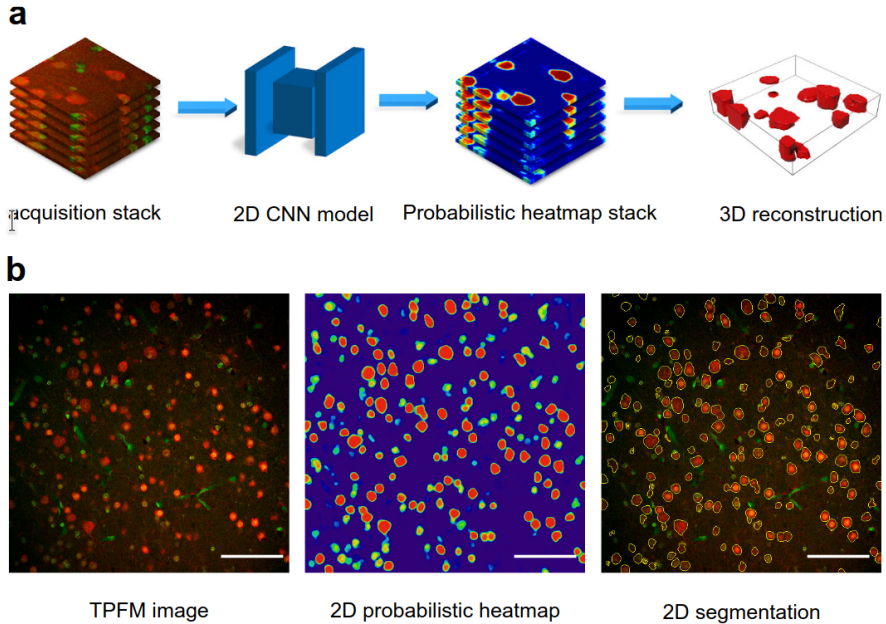


Figure 4.3: The 2.5D Approach: Panel (a) summarizes the working principles of the 2.5D inference approach, individual images from an acquisition stack are independently processed by our CNN inference model and the resulting probability maps are re-combined into a single volumetric stack, we then compute individual neuronal meshes as iso-surfaces of this probabilistic field. (b) visualizes the differences between, respectively, the raw images, the probabilistic heatmaps, and the original images after overlaying the segmentation shapes. Scale bar is 100 μm .

A feed-forward Convolutional Neural Network model was used for pixel-based segmentation of the multichannel images we reconstructed, using information from the red and green channel (respectively NeuN-marked neuronal somas and tissue autofluorescence). The model assigns to each pixel the probability of belonging to either the *neuron* or the *background*

class. Our CNN model effectively transforms the multichannel source image into a grayscale one, which we denoted as *probability heatmap*. 2D images are processed independently by the neural network and the resulting heatmaps are reassembled back into a volumetric stack: we refer to this procedure as a 2.5D approach, where we use the normalized recombination of bi-dimensional heatmaps as a proxy to the actual volumetric probability distribution of neuronal soma presence. Instance semantic segmentation is then performed on the heatmap volume by applying a statistical acceptance threshold of 0.5 in order to extract the three-dimensional surfaces of each uniquely identified neuron.

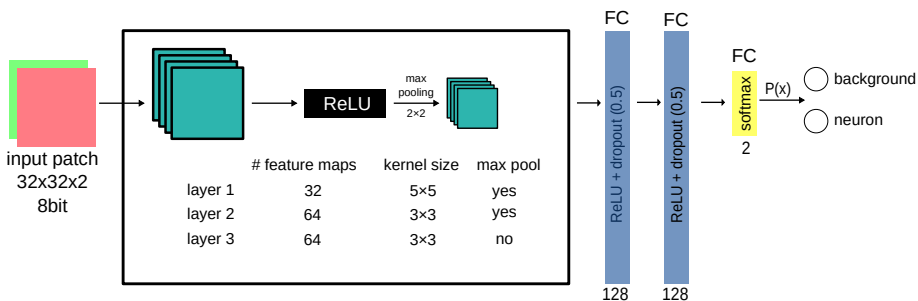


Figure 4.4: CNN Architecture for Two-Photon Microscopy Segmentation: The CNN model for fast patch-based classification converts $32 \times 32 \times 2$ texture patches to a binary classification value representing the probability of neuronal presence at its central pixel. The architecture consists of a feature-extraction module, composed of three ReLU-activated convolutional layers, the feature maps of the first two of which are spatially compressed by MaxPooling layers, the feature extraction module is followed by a Fully Connected classification stage which is terminated by a softmax activation function.

The feed-forward network architecture, summarized in Figure 4.4, expands upon of a previous work [23] and can be considered a generally lightweight model. In this application $32 \times 32 \times 2$ patches are extracted from the stitched imaging volume and fed to the the network after a

pre-processing step consisting of a single 5×5 gaussian kernel filtering stage with $\sigma = 3$. This operation empirically replicates blurring caused by resampling of small patches during data augmentation in training phase. The neural network model consists of a feature-extracting stage made of three convolutional layers, two of which are followed by Max-Pooling downsampling, classification is then achieved via three a Fully Connected layers, the last of which makes use of a Binary SoftMax activation function.

The CNN model defined in Figure 4.4 classifies the central pixel of each input by exploiting the visual pattern of its local neighborhood. Efficient inference on input data larger than 32×32 pixels is possible by operating the network in *fully convolutional* mode, i.e. exploiting the formal equivalence between fully connected layers and convolutions with 1×1 kernels [20] to make the network effectively agnostic to the specific shape of the inputs. Our network, which was trained to classify 32×32 wide patches, after substituting its fully connected layers with equivalent convolutions, can effectively accept arbitrarily sized inputs. This method allows us to produce probability heatmaps of entire stack frames instead of small patches in a fast and efficient manner.

4.2.6 Ground Truth Collection and Model Training

Segmentation ground truth data was annotated by two distinct operators on LAIRA[®] (by Bioretics, Italy), a web-based collaborative application for image annotation. [17]. The network was incrementally trained following an Active Learning paradigm [29] against a number of positive (neuronal presence) and negative (background) random samples from the four imaged subjects to improve inter-specimen statistical representation. The final dataset consists of 112 images sized 512×512 px, corresponding to a $(450 \times 450)\mu\text{m}^2$ imaging area, for a total of 7312 individually segmented neurons. 14 independent images (1505 neurons)

were used to validate the network, and further 14 images (1208 objects) to test it. All the annotated ground truth data is also shared in *Ximage* annotation format [34] on the EBRAINS platform [12]. Model definition, along with training and inference has been performed on the ALIQUIS[®] software ecosystem[2] (by Bioretics, Italy), which exploits TensorFlow [22] as computational backend.

4.2.7 From 2D Heatmaps to 3D Polygon Meshes

The CNN model converts 2D frames from the acquisition volume to probability heatmaps, these two-dimensional representations are reassembled back into a 3D stack to obtain an estimate of the three-dimensional probability distribution of neuronal soma presence. The obtained volume is post-processed via application of a 5×5 median filter and a gray-scale morphological opening with a 3×3 structuring element, this step serves as a false-positive reduction strategy.

We use the resulting volumetric stack as a proxy to the real 3D probability distribution, this means that we consider the isosurfaces of this field, corresponding to a 0.5 statistical threshold, to be representative of the physical boundaries of neuronal somas. To calculate these isosurfaces we use a custom variant of the Marching Cubes algorithm [21] [18], followed by custom additional topological fixes on the identified objects to ensure that every soma is represented by a 2-manifold watertight mesh.

This approach allows us to retrieve a three-dimensional vectorial reconstruction of the segmented objects in the entire z-stack in an efficient way using C++ implemented CGAL libraries for fast batch-based computation of the neuronal soma boundaries.

4.3 Results

The methodological framework we established in the last section allowed us to perform tissue clearing, TPFM imaging, CNN-based semantic segmentation and, lastly, three-dimensional separation and identification of individual somas. We now have access to individual representations of the neuronal inside the four analyzed samples. Visualization and analysis of this data required innovative solutions, tailored to its unique characteristics.

4.3.1 Whole-Sample Imaging

We used the ZetaStitcher tool to automatically stitch and align hundreds of TPFM imaging stacks to obtain whole plane reconstructions of the four samples. After alignment and fusion the resulting datasets are sized, respectively, 19 GB, 50 GB, 57 GB and 94 GB. In the top half of Figure 4.5 we show a complete wide-scale reconstruction of a slice from the first sample: fusion of different stacks is can be noticed by the vignetting effects in each independent acquisition.

4.3.2 3D Reconstruction and Visualization

Despite the latest advancement in 3D rendering tools and hardware, visualizing large amounts of individually defined meshes remains an extremely difficult task to solve. The standard way to render extremely high amounts of individual meshes is to generate them dynamically from a few defining parameters using GPU shaders: instead of visualizing the exact vertex positions of the neurons, a list of neuron positions and volumes can be passed directly to the GPU, which executes a custom code (shader) for direct parallel computation of likely meshes, these are then to be rendered in 3D space to form the final image. This process is many orders of magnitude faster than individually rendering exact meshes but

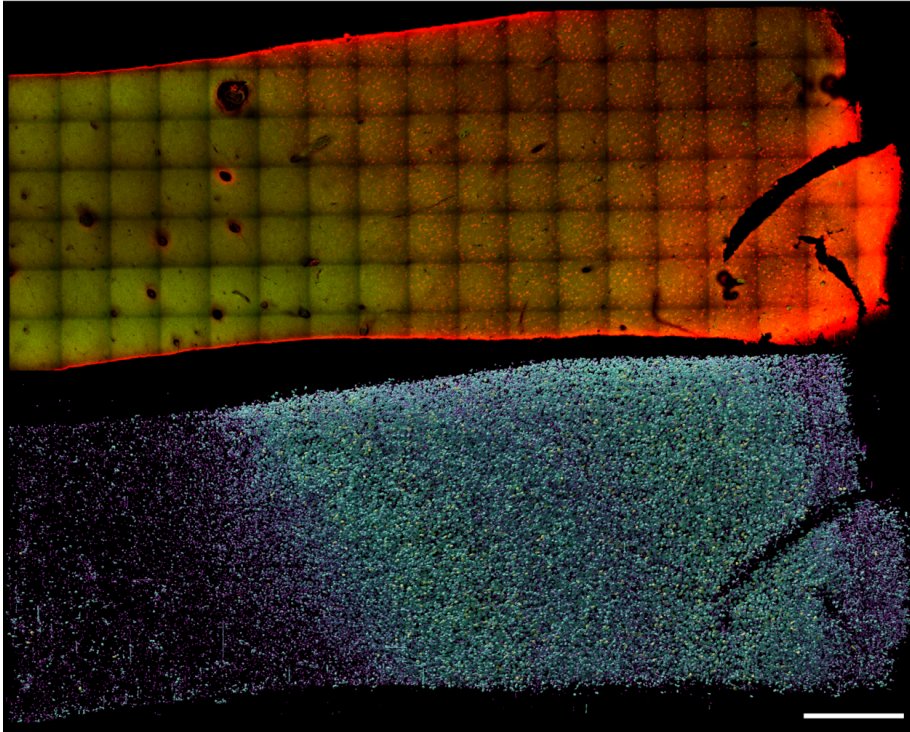


Figure 4.5: 3D Reconstruction of a whole cortical tissue sample: This image offers a visual comparison between a representative slice of the imaging plane, reconstructed from individual stack acquisitions using ZetaStitcher, and the three-dimensional reconstruction of neuronal somas detected with our CNN approach after mesh computation. The second visualization was realized with the Blender [4] render engine in combination with our own custom software extensions [7]. Scale bar = 100 μm .

the resulting image, while it might be useful as visualization tool on a large scale, wouldn't be physically representative of the biological sample at the microscopic level.

Wanting to obtain biologically accurate representations of the segmented objects and not just close approximations, we decided to pursue the road of direct mesh rendering.

We were able to use the Blender rendering engine [4] and its programming extension API, to perform full mesh-accurate renderings of the four specimens using custom code extensions, as shown in figure 4.6. We developed and relased on Github a custom Blender addon, called *SelectByVolume* [7], for visual differentiation of volumetric classes of neurons. This tool enabled us to render three-dimensional meshes of individual neruons in different colors depending on their computed internal volume, this characterization on its own is enough to visually appreciate the anatomical architecture of the six cortical layers as an emergent property of the reconstructed volume.

4.3.3 Neuronal Distributions Analysis

We characterized the structural organization of the analyzed cortical samples by calculating their cell density, mean volume and volumetric density distributions. To achieve this we subdivided the sample space in $10\ \mu\text{m}$ binnings and sampled these three properties inside each spatial division. Distribution profiles were then smoothed using a $100\times 100\times 100\ \mu\text{m}$ volumetrical moving average, the resulting maps are then plotted in Figure 4.7. With the exception of the dysplastic sample number 4, in which cortical layer organization is disrupted, layer organization can be visually appreciated. In particular volume profiles show peaks in layers 3 and 6, while the neuronal density has peaks in layers 2 and 4.

We also calculated, for each layer of the cortex, the total amount of neuronal cells, the total volume occupied by neurons, their numerical

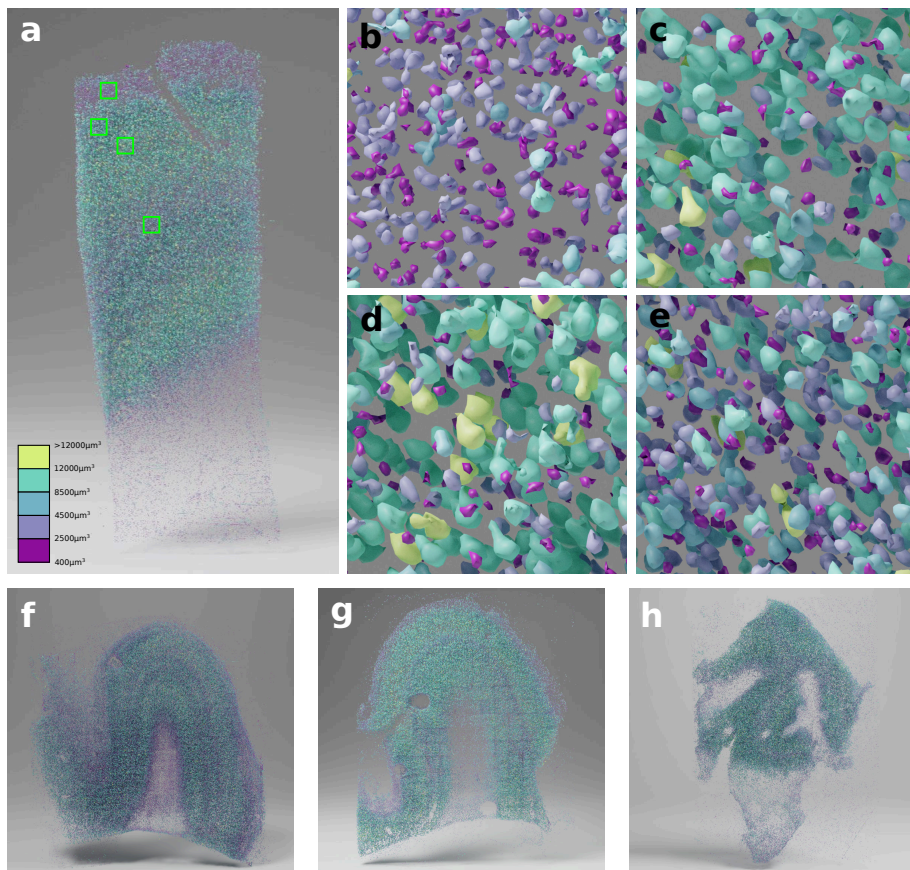


Figure 4.6: Three-dimensional details of Whole Sample Reconstructions: Panels a, f, g, h show the 3D renderings of 3D neuronal meshes from, respectively, specimens 1, 2, 3 and 4. Panels b,c,d,e represent magnified views of panel a, highlighting different neuronal sizes and densities in different cortical layers. Colors are assigned to different volumetric classes using the colormap described in bottom left of panel a.

density, mean volume and the fraction of volume in the layer occupied by neurons. In order to do that, gray matter and individual layers were manually segmented for each specimen, excluding large blood vessels,

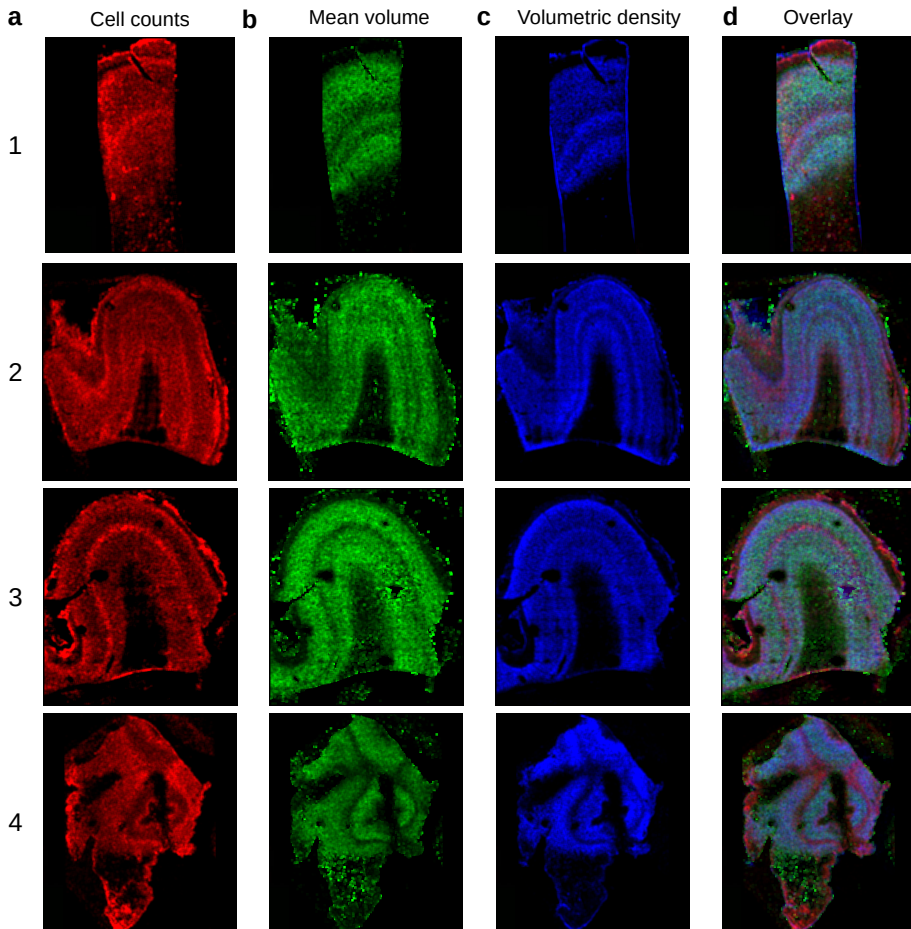


Figure 4.7: Representative distribution maps of the cell counting (a), mean volume (b), volumetric density (c) and an overlay of the three maps (d) at the middle plane of each specimen.

tissue holes/breakages and imaging artifacts. Numerical results are reported in Table 4.1.

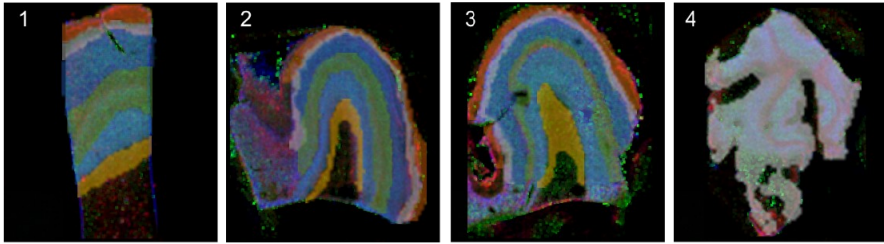


Figure 4.8: Manual Segmentation of the gray matter: cortical layers were individually segmented for sample 1, 2 and 3. Sample 4 shows a disruption of the structural organization of the cortex, making layer classification nonsensical, in that case only gray matter was segmented. Every image of the reconstructed stack was segmented in this way to obtain a volumetric mask.

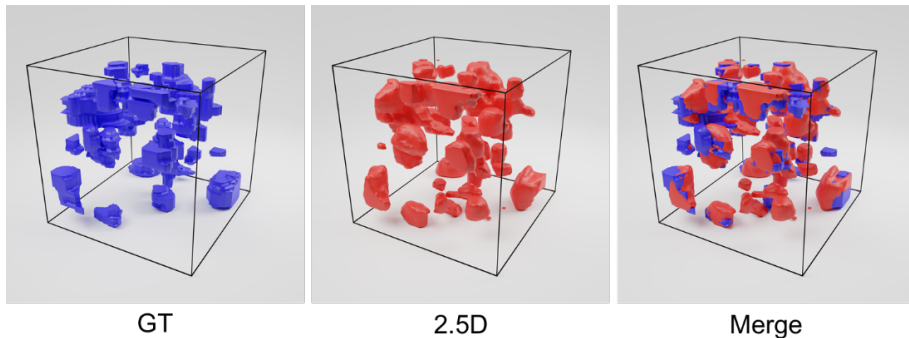


Figure 4.9: 3D Representation of the neurons in a manually annotated stack (in blue) used for whole-stack performance assessment, along with the our CNN predictions on the same stack (red) and visual overlap of the two.

4.3.4 Performance Assessment over Whole Stacks

The performances of the CNN were statistically assessed against four $(100 \times 100 \times 100) \mu m^3$ representative stacks, one for each analyzed specimen, along with a single $(100 \times 100 \times 450) \mu m^3$ stack to determine

	Layer	N. Neurons	Tissue Volume mm^{-3}	Density $10^3 mm^{-3}$	Mean Volume μm^3	Filling Fraction
Specimen 1						
	1	47329	2.880	17	2603	1.38%
	2	11580	0.486	33	2916	3.87%
	3	40684	1.758	27	4441	5.02%
	4	47433	1.814	31	3549	4.90%
	5	23290	1.300	24	4358	4.68%
	6	13843	0.769	22	2263	1.99%
	tot.	184159	9.007	24	3569	21.8%
Specimen 2						
	1	178432	11.533	16	2550	1.5%
	2	193783	11.724	17	2567	1.8%
	3	219289	12.789	18	2922	2.3%
	4	97771	2.413	51	2730	6.5%
	5	78402	2.199	43	3324	6.4%
	6	41167	1.054	47	1989	3.8%
	tot.	808844	41.712	20	2740	22.4%
Specimen 3						
	1	34997	1.646	21	2105	1.8%
	2	36697	1.432	26	3412	5.3%
	3	66012	3.464	19	4627	5.7%
	4	46979	1.804	26	3524	5.8%
	5	55617	2.387	23	4584	6.4%
	6	16912	5.656	3	3410	1.9%
	tot.	257214	16.389	16	3853	26.9%
Specimen 4						
	tot.	177286	12.694	14	3008	4.2%

Table 4.1: Number, mean volume and neuronal density in the six layers of the cortex, calculated using our automated analysis pipeline.

accuracy along the full specimen depth. Each stack was independently manually annotated by an operator and automatically segmented using our 2.5D approach, resulting in a total of 442 detected objects over the 474 neurons annotated in the four ground truth stacks. The volumetric evaluation of the neurons is characterized by a precision and a recall rate, respectively equal to $(97 \pm 7)\%$ and $(96 \pm 3)\%$. Figure 4.9 gives a visual depiction of a manually labeled stack, next to the inference predictions of the model.

4.4 Discussion and Conclusions

4.4.1 Choice of CNN Architecture

4.2.6 The model architecture in 4.4 is a relatively lightweight model based on local recognition of 32×32 wide textures. The state of the art in biomedical and life-science segmentation networks is dominated by significantly heavier models with large receptive fields that are able to directly perform full-image segmentation, but their use comes with two main drawbacks. Firstly, the geometrically higher number of learnable parameters require large amounts of annotation ground truth data which, for microscopy applications, is difficult and often prohibitively expensive to obtain, then we have that model complexity often poses additional difficulty in scaling to large samples as additional computational resources and time are required. Considering the significant costs associated with obtaining manual annotations in specialized microscopy fields due to the high specialization grade needed in performing accurate histological manual segmentations of the data when compared to annotation costs in other less-specialized applications, and also considering the large computational resources that state of the art models would require to run large quantities of microscopy data, we favored a simpler model to potentially more powerful and elaborated alternatives, deeming this approach easier to replicate on large scales. The chosen model architecture could be exhaustively trained with limited ground truth annotation quantities while at the same time offering excellent execution times.

4.4.2 Grouping Artifacts

While the presented method, based on the on-the-fly computation of neuronal meshes using Marching Cubes, offers advantages on scalability and enables reconstruction of extended areas, it also suffers from inher-

ent limitations, namely the presence of grouping effects that sometimes can appear when neuronal somas that are too close to each other are identified as a single unit. This effect has limited impact on high quality images, such as those obtained with Two-Photon Confocal Microscopy where spatial features of the neuronal somas are easily distinguishable, but can become significative when imaging conditions degrade or when imaging highly packed areas of the cortex.



Figure 4.10: Neuron Grouping Artifacts: objects that are too close together can sometimes be mistaken for single entities. From left to right we have the ground truth annotations, the network’s prediction and an overlap of the two.

A possible way to counteract this effect is by exploiting geometrical properties of the conjoined meshes. Neuronal aggregates of two and more neurons are easily identifiable by their geometrical properties which depart from the approximated star-convexity that characterizes somas in their general appearance. After geometrical filtering and classification of the aggregates, individual items can be separated by tracing cut planes in correspondence of their maximum convexity. The low appearance of this kind of artifact in our data didn’t justify the extra computational overhead associated to this geometrical analysis and processing step but its implementation is considered for future applications.

4.4.3 Conclusions

We introduced an end-to-end pipeline for cell-resolution investigation of the cortex that addresses the most critical challenges in human brain imaging (i.e. sample transformation and labeling, high-resolution whole-sample reconstruction), automatic neuronal segmentation and data visualization. We used the SWITCH/TDE approach to prepare four volumetric samples from different areas of the cortex, which are then labeled with anti-NeuN antibody and DAPI. Two-Photon Fluorescence Microscopy was used to retrieve high resolution volumetric images at sub-micron resolution and a custom software package - ZetaStitcher - was used for whole-sample reconstruction from large amounts of imaging data (ranging from tens of Gigabytes to Terabytes). We performed automated cell analysis on the data using a custom 2.5D machine learning approach which enabled us to conduct assessments on mean volume and cell counting and obtained individual geometrical mesh representation for each neuron in the imaged volume.

Despite the proposed innovation, there are a number of aspects that would need to be considered in order to obtain a faster, more scalable and high-throughput process. In primis a faster optical technique, such as Lightsheet Fluorescence Microscopy could be used to extend the analysis to larger samples. Heavier segmentation networks could be used to reduce classification errors, while a more refined instance separation methodology could reduce instance grouping artifacts.

In the next chapter we'll answer each of these issues in the context of a much wider-scale application using Light-Sheet Microscopy Imaging, in combination with a more advanced Deep Learning-based custom semantic segmentation and instance segmentation approach.

Bibliography

- [1] Waseem Abbas and David Masip. “Computational Methods for Neuron Segmentation in Two-Photon Calcium Imaging Data: A Survey”. en. In: *Applied Sciences* 12.14 (Jan. 2022), p. 6876. ISSN: 2076-3417. DOI: 10.3390/app12146876. URL: <https://www.mdpi.com/2076-3417/12/14/6876> (visited on 03/17/2023).
- [2] *Aliquis . Bioretics srl*. URL: <https://www.bioretics.com/aliquis> (visited on 08/19/2019).
- [3] Yijun Bao et al. “Segmentation of neurons from fluorescence calcium recordings beyond real time”. en. In: *Nature Machine Intelligence* 3.7 (July 2021), pp. 590–600. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00342-x. URL: <https://www.nature.com/articles/s42256-021-00342-x> (visited on 03/17/2023).
- [4] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Institute, Amsterdam: Blender Foundation. URL: <http://www.blender.org>.
- [5] Alejandro Luis Callara et al. “A Smart Region-Growing Algorithm for Single-Neuron Segmentation From Confocal and 2-Photon Datasets”. In: *Frontiers in Neuroinformatics* 14 (2020). ISSN: 1662-5196. URL: <https://www.frontiersin.org/articles/10.3389/fninf.2020.00009> (visited on 03/17/2023).

- [6] Filippo Casoni et al. “Development of the neurons controlling fertility in humans: new insights from 3D imaging and transparent fetal brains”. eng. In: *Development (Cambridge, England)* 143.21 (Nov. 2016), pp. 3969–3981. ISSN: 1477-9129. DOI: 10.1242/dev.139444.
- [7] Filippo Castelli. *SelectByVolume*. Aug. 2021. URL: <https://github.com/filippocastelli/SelectByVolume> (visited on 11/29/2022).
- [8] Filippo M. Castelli et al. “Semantic Segmentation of Neuronal Bodies in Fluorescence Microscopy Using a 2D+3D CNN Training Strategy with Sparsely Annotated Data”. en. In: *Machine Learning, Optimization, and Data Science*. Ed. by Giuseppe Nicosia et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, pp. 95–99. ISBN: 978-3-030-64583-0. DOI: 10.1007/978-3-030-64583-0_10.
- [9] Kwanghun Chung et al. “Structural and molecular interrogation of intact biological systems”. en. In: *Nature* 497.7449 (May 2013), pp. 332–337. ISSN: 1476-4687. DOI: 10.1038/nature12107. URL: <https://www.nature.com/articles/nature12107> (visited on 05/10/2019).
- [10] Irene Costantini et al. “A versatile clearing agent for multi-modal brain imaging”. en. In: *Scientific Reports* 5 (May 2015), p. 9808. ISSN: 2045-2322. DOI: 10.1038/srep09808. URL: <https://www.nature.com/articles/srep09808> (visited on 05/07/2019).
- [11] Irene Costantini et al. “Large-scale, cell-resolution volumetric mapping allows layer-specific investigation of human brain cytoarchitecture”. EN. In: *Biomedical Optics Express* 12.6 (June 2021), pp. 3684–3699. ISSN: 2156-7085. DOI: 10.1364/BOE.415555. URL: <https://www.osapublishing.org/boe/abstract.cfm?uri=boe-12-6-3684> (visited on 10/15/2021).

- [12] *EBRAINS - A key enabler to advance brain science*. en. URL: <https://ebrains.eu/> (visited on 11/22/2022).
- [13] Jiangheng Guan et al. “NeuroSeg: automated cell detection and segmentation for in vivo two-photon Ca²⁺ imaging data”. en. In: *Brain Structure and Function* 223.1 (Jan. 2018), pp. 519–533. ISSN: 1863-2661. DOI: 10.1007/s00429-017-1545-5. URL: <https://doi.org/10.1007/s00429-017-1545-5> (visited on 03/17/2023).
- [14] Sven Hildebrand et al. “Scalable Labeling for Cytoarchitectonic Characterization of Large Optically Cleared Human Neocortex Samples”. en. In: *Scientific Reports* 9.1 (July 2019), p. 10880. ISSN: 2045-2322. DOI: 10.1038/s41598-019-47336-9. URL: <https://www.nature.com/articles/s41598-019-47336-9> (visited on 12/01/2022).
- [15] Cihan Bilge Kayasandik and Demetrio Labate. “Improved detection of soma location and morphology in fluorescence microscopy images of neurons”. en. In: *Journal of Neuroscience Methods* 274 (Dec. 2016), pp. 61–70. ISSN: 0165-0270. DOI: 10.1016/j.jneumeth.2016.09.007. URL: <https://www.sciencedirect.com/science/article/pii/S0165027016302102> (visited on 03/17/2023).
- [16] Aleksander Klibisz et al. “Fast, Simple Calcium Imaging Segmentation with Fully Convolutional Networks”. en. In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Ed. by M. Jorge Cardoso et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 285–293. ISBN: 978-3-319-67558-9. DOI: 10.1007/978-3-319-67558-9_33.
- [17] *LAIRA - "Laira is an AI-based Research Assistant" - Bioretics srl*. URL: <https://laira.bioretics.com/> (visited on 08/19/2019).

- [18] Thomas Lewiner et al. “Efficient Implementation of Marching Cubes’ Cases with Topological Guarantees”. en. In: *Journal of Graphics Tools* 8.2 (Jan. 2003), pp. 1–15. ISSN: 1086-7651. DOI: 10.1080/10867651.2003.10487582. URL: <http://www.tandfonline.com/doi/abs/10.1080/10867651.2003.10487582> (visited on 08/19/2019).
- [19] Thomas Liebmann et al. “Three-Dimensional Study of Alzheimer’s Disease Hallmarks Using the iDISCO Clearing Method”. eng. In: *Cell Reports* 16.4 (July 2016), pp. 1138–1152. ISSN: 2211-1247. DOI: 10.1016/j.celrep.2016.06.060.
- [20] Min Lin, Qiang Chen, and Shuicheng Yan. “Network In Network”. In: *arXiv:1312.4400 [cs]* (Dec. 2013). URL: <http://arxiv.org/abs/1312.4400> (visited on 08/17/2019).
- [21] 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. New York, NY, USA: ACM, 1987, pp. 163–169. ISBN: 0-89791-227-6. DOI: 10.1145/37401.37422. URL: <http://doi.acm.org/10.1145/37401.37422>.
- [22] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. URL: <https://www.tensorflow.org/>.
- [23] G. Mazzamuto et al. “Automatic Segmentation of Neurons in 3D Samples of Human Brain Cortex”. In: *Applications of Evolutionary Computation*. Ed. by Kevin Sim and Paul Kaufmann. Cham: Springer International Publishing, 2018, pp. 78–85. ISBN: 978-3-319-77538-8.

- [24] Giacomo Mazzamuto. *lens-biophotonics/ZetaStitcher*. Oct. 2022. URL: <https://github.com/lens-biophotonics/ZetaStitcher> (visited on 11/21/2022).
- [25] Giacomo Mazzamuto et al. *Software Tools for Efficient Processing of High-Resolution 3D Images of Macroscopic Brain Samples*. URL: <https://www.osapublishing.org/abstract.cfm?uri=Microscopy-2018-JTh3A.64> (visited on 08/19/2019).
- [26] Evan Murray et al. “Simple, scalable proteomic imaging for high-dimensional profiling of intact systems”. In: *Cell* 163.6 (Dec. 2015), pp. 1500–1514. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.11.025. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5275966/> (visited on 05/13/2019).
- [27] Nicolas Renier et al. “iDISCO: A Simple, Rapid Method to Immunolabel Large Tissue Samples for Volume Imaging”. English. In: *Cell* 159.4 (Nov. 2014), pp. 896–910. ISSN: 0092-8674, 1097-4172. DOI: 10.1016/j.cell.2014.10.010. URL: [https://www.cell.com/cell/abstract/S0092-8674\(14\)01297-5](https://www.cell.com/cell/abstract/S0092-8674(14)01297-5) (visited on 12/01/2022).
- [28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv:1505.04597 [cs]* (May 2015). URL: <http://arxiv.org/abs/1505.04597> (visited on 05/13/2019).
- [29] Burr Settles. *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin–Madison, 2009. URL: <http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf>.
- [30] Simon P. Shen et al. “Automatic Cell Segmentation by Adaptive Thresholding (ACSAT) for Large-Scale Calcium Imaging Datasets”. In: *eNeuro* 5.5 (Sept. 2018). ISSN: 2373-2822. DOI: 10.1523/

- ENEURO.0056-18.2018. URL: <https://www.eneuro.org/content/5/5/ENEURO.0056-18.2018> (visited on 03/17/2023).
- [31] Ludovico Silvestri et al. “Towards a Full Volumetric Atlas of Cell-specific Neuronal Spatial Organization in the Entire Mouse Brain”. EN. In: *Biophotonics Congress: Biomedical Optics Congress 2018 (Microscopy/Translational/Brain/OTS) (2018)*, paper JTU3A.62. Optical Society of America, Apr. 2018, JTU3A.62. DOI: 10.1364/TRANSLATIONAL.2018.JTU3A.62. URL: <https://www.osapublishing.org/abstract.cfm?uri=BRAIN-2018-JTU3A.62> (visited on 06/29/2019).
- [32] Ilya Valmianski et al. “Automatic Identification of Fluorescently Labeled Brain Cells for Rapid Functional Imaging”. In: *Journal of Neurophysiology* 104.3 (Sept. 2010), pp. 1803–1811. ISSN: 0022-3077. DOI: 10.1152/jn.00484.2010. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2944673/> (visited on 03/17/2023).
- [33] Yangzhen Wang et al. “Efficient implementation of convolutional neural networks in the data processing of two-photon in vivo imaging”. In: *Bioinformatics* 35.17 (Sept. 2019), pp. 3208–3210. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btz055. URL: <https://doi.org/10.1093/bioinformatics/btz055> (visited on 03/17/2023).
- [34] *XImage*. Mar. 2020. URL: <https://github.com/bioretics/ximage> (visited on 11/22/2022).

Chapter 5

LightSheet Microscopy Investigation of the Human Broca Area

In the previous chapter we demonstrated how Two Photon Confocal Microscopy can be used in combination with software-based sample reconstruction and CNN-enabled automated segmentation to produce three-dimensional reconstructions of entire samples, we now want to replicate the same pipeline on a larger scale using Light-Sheet Microscopy.

In this chapter we introduce a custom Light-Sheet Microscopy setup with a Dual-View Inverted SPIM geometry with extremely high data throughput: a first set of high-level challenges is represented by finding engineering solutions for managing, storing and processing the enormous data flows produced by our instrument. Automated CNN-based processing of this data is a challenge on its own as not only the data flows are geometrically scaled, but visual complexity of the target itself is considerably increased.

The vast majority of applications of Light-Sheet Fluorescence Mi-

croscopy to neuronal mapping involves targets that are significantly smaller in volumetric extension than the ones we're trying to image, such as entire drosophilae [2, 6], zebrafish specimens [1, 7, 10, 11] or mice brains [4, 5, 8, 9]: scaling the same technology to large scale mapping of neurons in the human brain is, to our knowledge, a mostly unpaved road.

We have developed our own solutions for every aspect of this unique and ambitious challenge, from the image acquisition and processing software to the semantic and instance segmentation models, all maintaining a constant focus on making each part of the pipeline modular and ready to be scaled to even larger targets.

By the end of this chapter we prove to be able to obtain a database of individual neurons, complete with their position and geometrical characterization in the volumetric domain from multi-terabyte sized acquisitions, unlocking high-level insight and measurement capabilities on extended biological samples.

5.1 Introduction

The Light-Sheet Fluorescence Microscopy data we're presenting in this chapter is part of a collaborative effort [8] (in the context of BRAIN Initiative Cell Census Network project) to construct multi-modal atlases of the human cerebral cortex which uses Magnetical Resonance Imaging (MRI) for large-scale referencing, LightSheet Fluorescence Microscopy (LSFM) for 3D imaging at cellular resolution and Optical Coherence Tomography (OCT) for meso-scale registration of LSFM samples to the MRI reference. The perspective of obtaining queryable neuronal maps, through the collaborative contribution of multiple institutions and laboratories operating different imaging technologies, heavily relies on our capacity of imaging and processing the data at its highest available res-

olution.

If we wish to move our investigation interests from relatively small samples to entire brain areas - and eventually to entire brains - we can reasonably expect that the tools and procedures we used in the first case might not be sufficient or performing well enough under significantly different conditions. The first limit we encounter is represented by the scanning nature of the Two Photon Confocal Microscopy apparatus we used in the last chapter: the amount of time we would need to scan multiple volumes in the cubic centimeter scale with TPFM would make such an effort extremely impractical. Light Sheet Microscopy (LSFM) would seem to be the perfect candidate for covering the micrometer-to-centimeter mesoscopic scale gap that TPFM can't reasonably reach. With many advantages - such as extremely high imaging speed and the possibility of acquiring multiple views of the same sample simultaneously - also come many possible downsides: the ability of generating data at a much faster rate implies the technical challenge of handling massive data streams, the different geometry involved in image generation also determines the possibility of uneven illumination and the emergence of visual artifacts that are unique to LSFM such as horizontal striping due to shadowing by dense objects, and complicated setup geometries imply nontrivial geometrical corrections of the acquired data.

A simple transposition of the processing methods used in the last chapter to larger samples wouldn't suffice in tackling what is, from many points of view, a much more complex objective: the increased visual complexity of LSFM images calls for more sophisticated segmentation models, the larger data sizes involved need optimized and heavily parallelized GPU-accelerated data processing pipelines and even the data visualization approaches can't be directly transposed to the new targets. In this chapter we first present the diSPIM apparatus we used to image an entire human Broca Area (Brodmann's areas 44 and 45) at sub-

micron resolution, we then present a CNN semantic segmentation model - NEUROresUNet - we developed for semantic segmentation of the neuronal somas in the biological samples, followed by an instance segmentation pipeline used for mapping individual neurons in the imaging volume.

5.2 Methods

5.2.1 Tissue Clearing and Labeling Biological Samples

The individual sample used in this work is an entire human Broca Area (Brodmann's areas 44 and 45). The postmortem specimen was obtained from a 79 year old male subject with no known neurologic and psychiatric illnesses.

48 500 μm thick slices from the samples were treated with the SHORT [23] protocol, a modified version of the SWITCH/TDE tissue transformation method used in the last chapter, combining the SWITCH [21] technique with the TDE clearing method[9].

After clearing, the sample was immunolabeled with multiple stainings by incubating with primary antibodies against NeuN (Merck ABN91 chicken) and calretinin(CR) (Proteintech 12278-1-AP rabbit). The samples were then incubated with the secondary antibodies conjugated with different dyes (Anti-Chicken IgY, AF 647 Abcam ab150171, Anti-Rabbit IgG AF 568, Abcam ab175470, Goat Anti-Chicken IgY AF 488 Abcam ab150169, Donkey Anti-Rabbit IgG AF 647 Abcam ab150075, Donkey Anti-Rabbit IgG AF 488 Abcam ab150073).

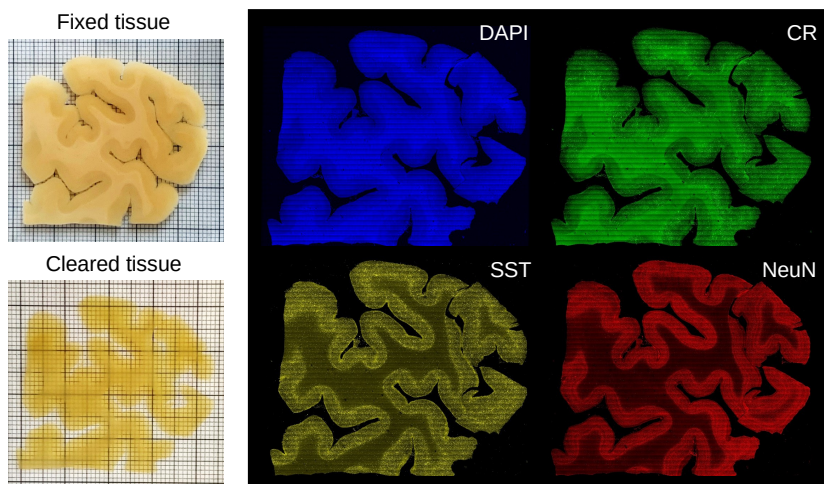


Figure 5.1: Individual Slice from the Broca Area sample: the images on the left, from top to bottom, represent a $500\ \mu\text{m}$ thick slice of the Broca Area sample respectively after tissue fixation and after optical clearing. The thick divisions of the grid represent 1 mm spacings, individual slices have a physical extent of approximately $1\ \text{cm}^2$. The distinction between *gray matter*, primarily consisting of neuronal cell bodies (somas) and *white matter* areas, mainly composed of myelinated axons appearing white due to high lipidic content, can be appreciated visually. After optical clearing, the gray matter areas are index-matched to the immersion medium and no longer visible, white matter is still partially noticeable due to its high fat composition. On the right we represented wide-scale reconstructions of the whole samples imaged with four different wavelengths expressing different stainings: DAPI, Calretinin, Somatostatin, and NeuN. The four channels can be acquired simultaneously by our LSFM apparatus.

5.2.2 Dual-View Selective Plane Illumination Light-Sheet (DiSPIM) Setup

Imaging of the 48 individually sliced samples from the Broca Area specimen was obtained with a custom-built dual-view inverted confocal light

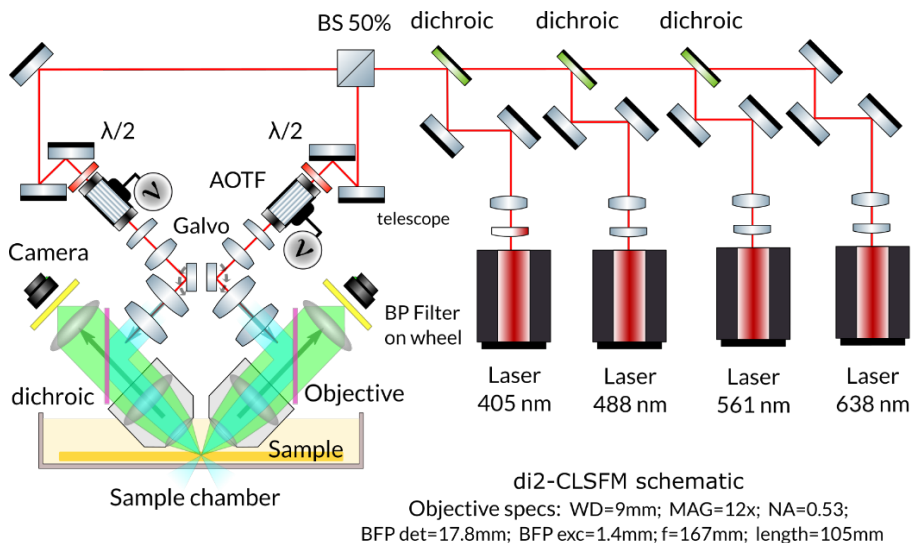


Figure 5.2: The di2-CLSFM apparatus: our custom Dual-view Inverted Light-Sheet Fluorescence Microscope [11] based on the SPIM geometry has been designed for simultaneous acquisition of four individual channels at $1\ \mu\text{m}$ isotropic resolution. The apparatus supports imaging of $1\ \text{mm}$ -thick samples with high data throughput ($1\ \text{Gbit s}^{-1}$ for each channel), up to $0.1\ \text{mm}^3\ \text{s}^{-1}$ volumetric rates. The two objectives, placed orthogonally, act alternatively as detection and illumination paths, producing a scanning light-sheet controlled by synchronized galvanometric mirrors. Illumination frequency is controlled by two acousto-optic filters (AOTF). Each detection path is equipped with a five sets of band-pass (BP) filters mounted on a motorized wheel to selectively block reflections and a high resolution sensor producing 2048×2048 pixel data streams at 16 bit depth.

sheet microscope [24], depicted in Figure 5.2. The setup features two identical objectives (LaVision Biotec LVMI-Fluor 12x PLAN, with 12x magnification, NA 0.53, WD 8.5-11mm), being used alternatively as detection and illumination paths, placed orthogonally to each other in such a way that their FOVs overlap at the center. The two objectives

are inclined at 45° relatively to the sample holder plane to allow for maximal lateral sample size without interfering with the microscope itself. Four different laser sources were used for illumination (Cobolt MLD 405nm/100mW, MLD 488nm/60mW, DPL 561nm/100mW, MLD 638nm/180mW), the width of each Gaussian beam was regulated using a dedicated telescope, before combining them with three dichroic mirrors. The combined beam is split with a 50 – 50% beam splitter and conveyed to the two identical excitation/detection pathways of the light-sheet setup. Two acousto-optical filters (AA Optoelectronics AOTFnC-400.650-TN) were used to modulate each beam in intensity, timing and transmitted wavelength. The light-sheets were realized by scanning each beam with a galvo mirror (Cambridge Technology 6220H) followed by scanning lenses (Edmund Optics #45-353, fl=100mm, achromat) to convert the angular displacement from the galvo mirrors into horizontal displacement. Lastly, the beams were directed to the objectives through an excitation tube lens (Edmund Optics #45-159, fl=200mm, achromat). Both objectives were placed on a motorized stage (PI L-509.14AD00) to adjust the position of the focal plane.

The samples were held on a 3-axis motorized stage system (two PI M-531.DDG for xy displacement, a PI L-310.2ASD for the vertical axis, for a combined motion range within a $30 \times 30 \times 2.5 \text{ cm}^3$ with less than $1 \mu\text{m}$ repeatability) with a custom quartz sample holder, inserted into a plastic tray filled with the refractive index matching solution. The samples were scanned in the horizontal axis at a 45 frames per second rate, corresponding to a volumetric scanning rate of 0.5 cm^3 per hour. On the detection path, fluorescence signal collected by the objective is geometrically diverted from the illumination path with a multi-band dichroic beam splitter (Semrock Di03-R405/488/561/635-t3-55x75) and directed with a detection tube lens (Edmund Optics #45-179, fl=200mm, achromat) to a sCMOS sensor (Hamamatsu OrcaFlash4.0 v3) operating

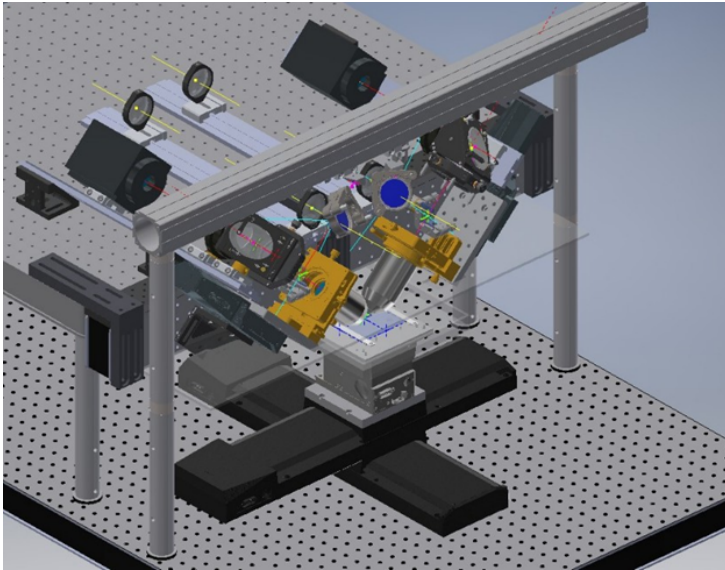


Figure 5.3: 3D Schematics of the di2-CLSFM apparatus: this view allows us to appreciate the 3-axis motorized system composed of a vertical displacement stage combined with two linear stages. The overall volumetric extent of the apparatus is characterized a relatively small planar footprint.

in confocal detection mode with the rolling shutter sweep synchronized with the galvo scan of the digital light-sheet using a PCIe control card (National Instruments PCIe-6363). Lastly, five sets of band-pass filters were mounted on a motorized filter wheel (Thorlabs FW102C) to image selectively differently labeled cells and structures and blocking any residual illumination light.

On the control hardware side, microscope operation and data acquisition were handled by a workstation running custom control software (written in C++ and Qt), particular attention was devoted to multi-threaded performance to make it sustain a data rate of approximately 800 MB/s at 47 fps (2048×2048 pixel 16 bit monochrome acquisitions)

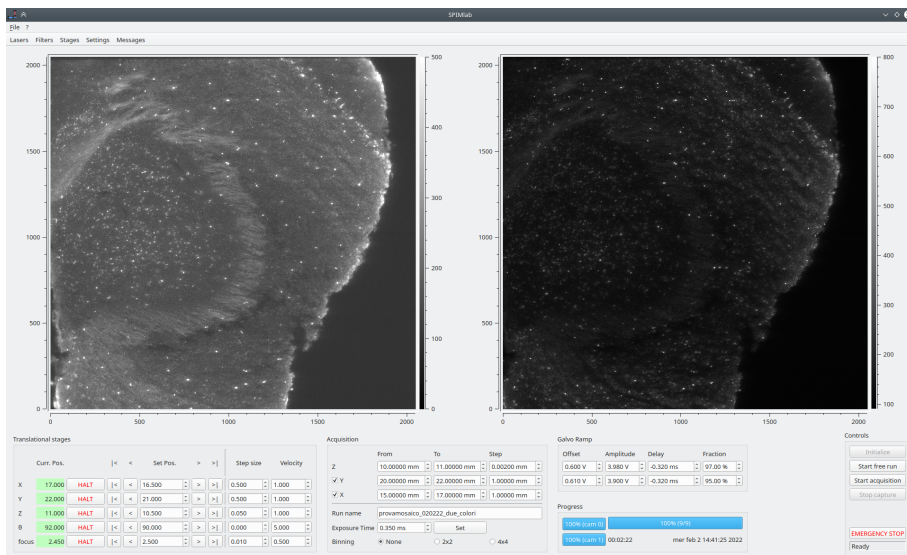


Figure 5.4: The SPIMlab Control Software: the di2-CLSFM microscope is controlled by a custom interface, SPIMlab [18], developed by LENS and CNR-INO. The microscopist operator is able to use a single interface for both controlling the hardware interface and starting acquisitions. More than 14000 lines of heavily parallelized and optimized C++ and Qt code make this tool capable of storing and displaying in real-time the massive data streams produced by di2-CLSFM apparatus.

writing directly to a 16 TB locally installed SSD RAID.

Samples were acquired in overlapping stacks by continuous translation of the x axis through the microscope’s field of view, at the end of each stack the sample was moved along the y axis by 1 mm before repeating the procedure to obtain contiguous acquisitions with a 100 μm overlap.

5.2.3 Acquisition Data Management

Geometrical reconstruction of entire slices was done by first applying affine transforms described in ?? to compensate for microscope geometry which features 45° inclined imaging of the horizontally translating

sample, this allows us to express the acquired stacks in the correct sample space. Acquisition stacks, featuring a lateral overlap of 100 μm are combined using *ZetaStitcher* [19] to obtain an overall representation of the whole sample. The stacks can optionally be resliced to 3.6 μm isotropic resolution (which corresponds to the axial resolution of the apparatus) for visualization purpose or for large-scale analysis. Illumination intensity homogeneity can be obtained by averaging the intensity profile along the direction of propagation for each fluorescence band in order to mitigate illumination intensity artifacts occurring across the transversal sample extension and to improve general readability of the whole stack.

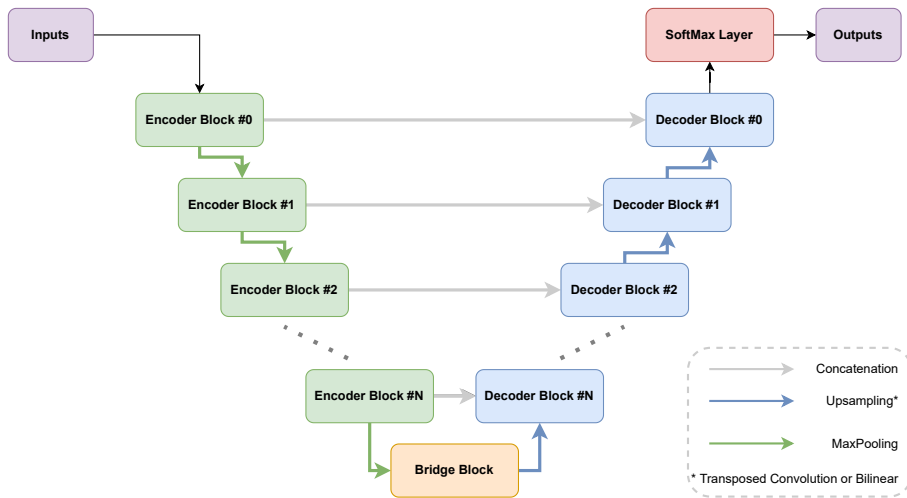
Sharing and long-term storage of this data is possible by lossy JPEG2000 compression which allows for 1:20 compression factors with minimal information loss.

5.2.4 Deep Learning-enabled Segmentation

Four slices from the whole imaged dataset were selected for quantitative analysis at full imaging resolution. Neuronal soma presence probability stacks of this data were obtained using a custom-designed model - NEUROresUNet - which was trained using the internally developed *NeuroSegmenter*[4] high-level deep learning framework based on *TensorFlow*.

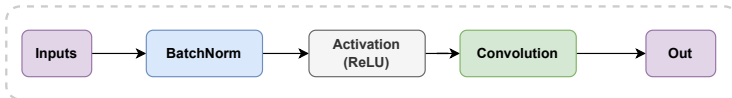
The NEUROresUNet Model

We developed a custom CNN model - NEUROresUNet - for semantic segmentation of the neuronal somas in our fluorescence microscopy imaging. The general architecture is based on a U-Net style [26] backbone with encoder and decoder paths linked by skip connections in their symmetrical blocks. Differently from the original UNet model, the func-

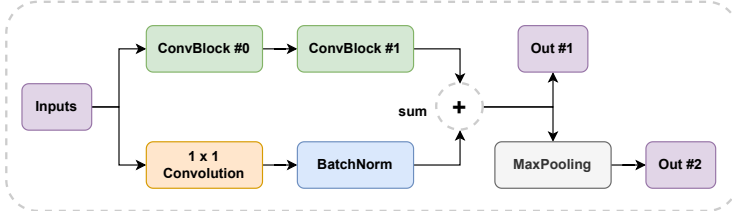


(a) large scale architecture diagram

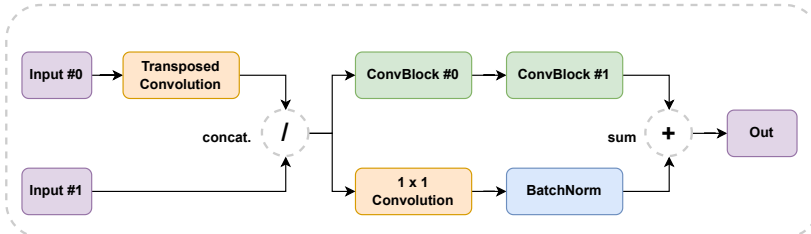
ConvBlock



Encoder Block



Decoder Block



(b) block details

Figure 5.5: The NEUROresUNet Architecture

tional blocks of both the decoder and decoder blocks implement residual learning schemes (summation of the convolutional block’s outputs to its own inputs) rather than a simple sequence of discrete convolutions. The introduction of residual units helps to tackle an accuracy degradation problem that emerges with network depth in very deep models [13, 14, 29]: in large networks it can often be posited that optimal solutions for convolutional blocks in its deepest sections are functionally very close to identity mappings, which are not easily approximable by combinations of non-linearly activated convolutions. Gradient-descent based optimization of this kind of target is better manageable by reformulating the whole block as a residual unit in which the inputs of the block are summed to its own outputs (after adjusting the relative dimensionalities with a 1×1 convolution layer): in this perspective, mostly-linear optimal solutions for the convolutional blocks can be searched as perturbative corrections of the identity mapping rather than having to be optimized directly as non-linearly activated convolutions.

One of the main features of NEUROresUNet lies in its dynamic functional implementation. The whole model is generated at runtime in an entirely parametrized fashion so that the user can size the network to his specific needs, choosing the number of levels of the architecture (the number of encoder / decoder block levels, i.e the depth of the network), a base number of convolutional filters for each layer in the encoder / decoder block and two different decoding schemes: one which employs transposed convolutions and one with a deterministic upsampling followed by discrete convolution. This particular implementation style enables for architectural optimization in addition to traditional hyperparameter tuning.

The network’s encoding and decoding paths are composed of modu-

larly defined *Encoder* and *Decoder* blocks, both of which use an uniquely defined *ConvBlock*, composed of a Batch Normalization layer, an activation function and, lastly, a convolution filter. The number of filters of each convolution layer in a *ConvBlock* is calculated as $(f, 2f)$ where $f = 2^d * b$, d is the depth level of the block and b is an adjustable parameter for regulating the network’s representational capacity. Both the *Encoder Block* and *Decoder Block* are defined as residual units with two *ConvBlock* units, a residual path with a 1×1 convolution for shape matching and a Batch Normalization layer. Encoding blocks are serially connected via Max-Pooling of the outputs, while, symmetrically, decoders use Transposed Convolutions to upsample compressed feature maps (which can optionally be substituted with deterministic bi-linear up-sampling followed by a discrete convolution, this strategy can be employed to avoid checkerboard artifacts that can emerge when using Transposed Convolutions) and skip connections are realized by concatenation of the upscaled inputs with outputs from the corresponding encoder block. The deepest part of the network is represented by the *Bridge Block*, which is formally equivalent to an *Encoder* block where the second set of outputs is not connected to any other part of the network.

Model Training

The model was trained on the customly-developed *NeuroSegmenter* framework on 205 manually annotated stacks, uniformly extracted from the entire acquisition area, sized $256 \times 256 \times 32$ px, equivalent to $133.1 \times 133.1 \times 115.2 \mu\text{m}^3$ volumes in sample space. Manual semantic annotation of these stacks was performed by four different operators, for a total annotation time of approximately 226 hours on the WebKnossos online collaborative annotation platform. A 250 – 40 – 20 split was chosen for

Transform	Description	Parameters	Prob.
Spatial	grid-based elastic deformation, image rotation, scaling and resampling	$d_{\text{deform}} = [0.9, 1.0]$ $\theta_{\text{rot}} = [-\pi, +\pi]$ $d_{\text{scale}} = [0.95, 1.05]$	0.15
Additive Brightness	additive brightness transform, brightness factor sampled from (μ, σ) Gaussian Distribution	$\mu = 0$ $\sigma = 0.1$	0.15
Multiplicative Brightness	multiplicative brightness transform	$b_{\text{multi}} = [0.9, 1.1]$	0.15
Mirror	random mirroring of the image on x or y axis		0.15
Gamma	gamma transformation $V_{\text{out}} = V_{\text{in}}^\gamma$	$\gamma = [0.9, 1.1]$	0.15
Rot90	90° rotation of the image		0.15
Gaussian Noise	addition of Gaussian noise from a $(0, \sigma_{\text{noise}})$ distribution	$\sigma_{\text{noise}} = [0, 510^{-5}]$	0.15
Gaussian Blur	gaussian blur convolutional filter with σ_{blur} variance	$\sigma_{\text{blur}} = [1, 1.5]$	0.15

Table 5.1: Data Augmentation Transforms

Data Augmentation Transforms: A number of data augmentation transformations is sequentially and non-deterministically applied in the input data pipeline, the high number of different imaging transformations helps the model learn correct semantical mapping in a variety of difficult imaging conditions.

training, test and validation data.

Architectural tuning of the *NEUROresUNet* model on the validation data resulted in the choice 4 depth levels and a base parameter for the number of convolutional filters in the blocks of 12. Data augmentation was managed with the *batchgenerators* [16] volumetric data augmentation engine, which was interfaced in *NeuroSegmenter*. Table 5.1 lists the transformations involved in the data augmentation pipeline, which was

executed at runtime during training and includes geometrical transforms (grid-based elastic deformations, rotations, mirroring and scaling), color transformations (gamma, linear brightness transforms) noise addition (gaussian noise) and convolution filterings (gaussian blur). For each training batch, each transformation was optionally chosen in random order with a probability of selection equal to 0.15, parameters for each transform were sampled for each data batch from the intervals reported in table 5.1.

Instance Segmentation and Data Aggregation

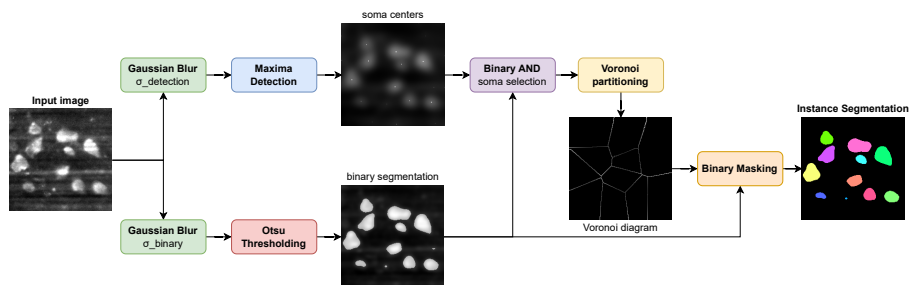


Figure 5.6: Instance Segmentation Process

The purpose of the instance segmentation step is to transform the volumetric maps of neuronal soma presence probability into individually defined neuronal instances. The approach used in chapter 4 based on map thresholding, morphological operations and 3D mesh computation with a variant of the Marching Cubes algorithm was not easily applicable to the Broca Area light-sheet microscopy data for a number of reasons: firstly, the presence of heavier visual artifacts due to striping and uneven illumination of the sample favored the emergence of grouping artifacts that weren't explicitly countered by that methodology, moreover, processing significantly larger amounts of data called for faster and GPU

accelerated strategies in order to keep processing times within affordable limits (weeks of multi-machine time computing instead of months or potentially years).

The entire inference volume, stored as individually compressed stacks, was accessed and scanned using *ZetaStitcher* in partially overlapping sections, each section is then processed to correct the geometrical distortion due to the inclined diSPIM geometry before the instance separation algorithm summarized in 5.6. The approximate positions of soma centers are found by first applying a gaussian filter to smooth intensity profiles and then searching for maximal intensity voxels. Binary segmentation of the somas is obtained by first applying another gaussian smoothing filter and then thresholding using the Otsu method [22], this binary map is then used as a filter to exclude spurious maxima that might erroneously selected by the first step, the remaining soma centers are then used to construct a 3D Voronoi tessellation diagram which, ultimately, serves to separate individual object instances in the binary segmentation map. Each individual object, now represented by a group of voxels sharing the same label, is then characterized by an individually defined UUID4 label, the *position* of the center of its soma in real sample coordinates, its *volume* and a set of shape descriptors like its *equivalent diameter area*, the *length* of its *major and minor axes*, the coordinates and a shape of its enclosing bounding box.

CPU and GPU memory sizes impose upper limits on the dimensions of each individually processed volumetric chunk, this means that neuronal instances which happen to be defined on two consecutive stacks are inevitably counted twice if no compensation strategy is employed: to reduce this effect, contiguous stacks are selected to include a partial overlap. In these areas the same objects often appear differently labeled under the two segmentation maps: to align these objects a bipartite graph is constructed, each node is defined by an individual label and

connectivity between edges is set to be proportional to the Jaccard Index of the two individual volumetric segmentations. Maximum bipartite graph matching is used to determine pairs of individually segmented objects that are present in both stacks, this information is then used to correct the database and uniform duplicated labels.

5.2.5 Computational Infrastructure for Data Processing

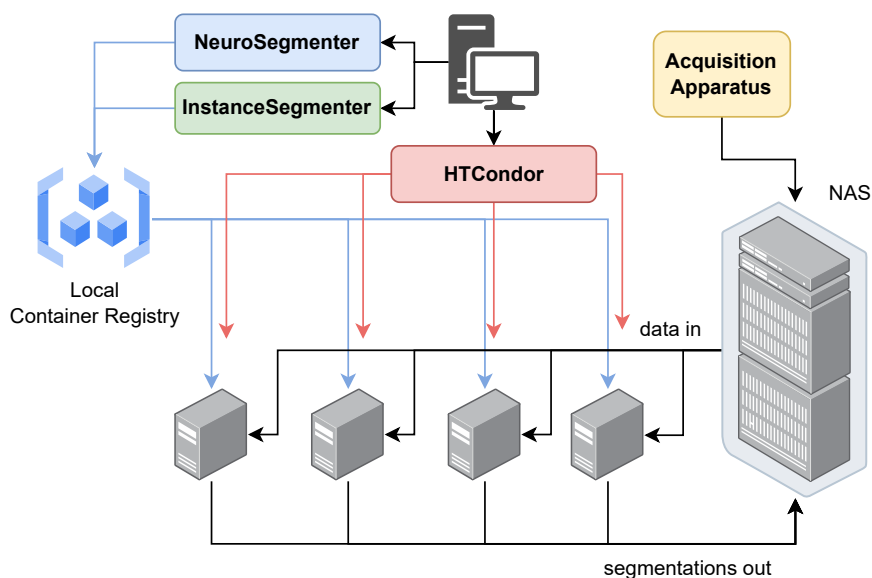


Figure 5.7: Processing Architecture: semantic segmentation and instance segmentation steps are implemented as two independent docker containers, which are served from a locally hosted container registry to every machine of the cluster, segmentation jobs are submitted to a local HTCondor manager which executes them on individual machines of the cluster. Both the acquisition data and the segmentation outputs reside on a local network attached storage.

Every design choice in the processing pipeline has prioritized computational scalability over theoretical performance in order to obtain a reproducible workflow that could satisfy execution time requirements (we aimed for processing times in the same scale as imaging acquisitions), modularity and replicability on similarly sized hardware.

The semantic segmentation engine and the following instance segmentation pipeline were implemented as independent Docker [20] container images. Process containerization comes with the inherent advantage of dependency decoupling between different parts of the pipeline. The actual code was executed on a custom-built computational cluster made from 4 worker Xeon servers, equipped with, respectively, an NVidia Tesla P100, An NVidia RTX 3090 and eight NVidia RTX 2080Ti (these two were distributed across two machines). Job execution on cluster machines was scheduled via HTCondor [28]. The user would issue jobs using a job descriptor file defining one or multiple data processing requests, referring to the acquisition data position on the internal Network Attached Storage system and to the specific version of the docker pipeline container revision used for processing, the HTCondor scheduler would then manage the job queue and distribute load across the cluster machines. Each machine would request the needed container image versions to a Container Registry Server within our network, which hosts all processing software revisions. This modular design allows us to execute different processing jobs on the same machine in a totally transparent manner, while containerization avoids all dependency conflicts. Both the acquisition data and the processed results reside on 0.5 PB sized expandable NAS connected to the single machines via a 10Gb network.

5.2.6 Data Visualization

Mesh generation of each individual object using a Marching Cube algorithm is time expensive but feasible by employing a large amount of

computational resources, on the other hand rendering this many individually defined meshes, where each vertex and edge are explicitly defined in sample space, using commercially available software like Blender, is technically challenging if not straight-up impossible in a naive problem formulation. Several techniques could be employed to reduce problem complexity, such as using GPU shaders to procedurally generate individual meshes or culling visually occluded neuronal instances. Ultimately, transposing the same data visualization approach used in the case of TPFM-imaged tissue proved to be too unpractical for the large amounts of data involved in this application.

Large scale visualizations of whole samples were obtained using by projection of the sample space into bi-dimensional spatial bins, represented as individual pixels in the output image. Pixel intensity is then assigned either proportionally to the number of objects in the spatial bin sharing a given property, or to the mean value of that property over the objects in the bin, these two kind of plots can also be combined by using false colors to represent property values and modulating their alpha intensity proportionally to object counts.

5.3 Results

5.3.1 Dataset Acquisition, Stitching and Reslicing

After tissue transformation and clearing, the 48 slices of the Broca Area sample were individually acquired using the di2-CLSFM apparatus. The 3-axis stage system was used to produce partially overlapping acquisition stacks of approximately 8500 2048×2048 16 bit images in four different channels (two per camera), amounting to about 71.3 GB per stack and per channel. The number of stacks needed to cover a slice ranges from 40 to 65 and depends on its specific physical dimensions.

Without any compression strategy, a monochromatic acquisition of a single slice would generally occupy 2.9 TB to 4.3 TB of disk space, resulting in an approximated 507 TB needed for the entire four-channel dataset. It's only through data compression that we're able to store this data for subsequent analysis. We're using lossy JPEG2000 compression the acquisition data to deflate our dataset down to a cumulative size of 25 TB. The choice of a lossy algorithm allows us to keep images in their $0.52 \times 0.52 \times 3.6 \mu\text{m}$ original spatial resolution (xy-plane resolution is obtained optically, while z-axis resolution anisotropy is due to horizontal movement of the positioning stage) while dramatically reducing data footprints with negligible loss in visual clarity. Any following automated analysis is performed on data which has not undergone spatial resolution compression or downsampling. Lossless compression is also applied to the already lossy-compressed data, these reductions are not as important as the lossy one but are useful to further fownsize the overall footprint of the dataset.

Number of files	Raw Data ($0.52 \times 0.52 \times 3.6 \mu\text{m}$)	JPEG2000 Compression ($0.52 \times 0.52 \times 3.6 \mu\text{m}$)
8602	507 TB	25 TB
Reslicing ($3.6 \mu\text{m}$ isotropic)	Reslicing, lossless compression ($3.6 \mu\text{m}$ isotropic)	
10.4 TB	4.2 TB	

Table 5.2: Broca Area dataset digital footprints: the size footprint of the uncompressed version of this dataset is too high to be stored locally, data compression (lossless and lossy) enables significative reduction of storage needs, reslicing to a lower resolution also reduces file sizes and can be used for dataset sharing.

After acquisition, compression and storage, individual stacks are then aligned together using the *ZetaStitcher* [19] tool which looks for

optimal relative displacement of the overlaps by maximizing their cross-correlation. This operation results in a virtual *stitched* dataset which is not duplicated on disk: only the optimal displacements are stored and the entire volume can be accessed on the fly using ZetaStitcher's API. After receiving a query for a volume or a specific part of it the tool autonomously handles decompression of the individual stacks involved in the query and fuses overlapped areas using a specific weighting function.

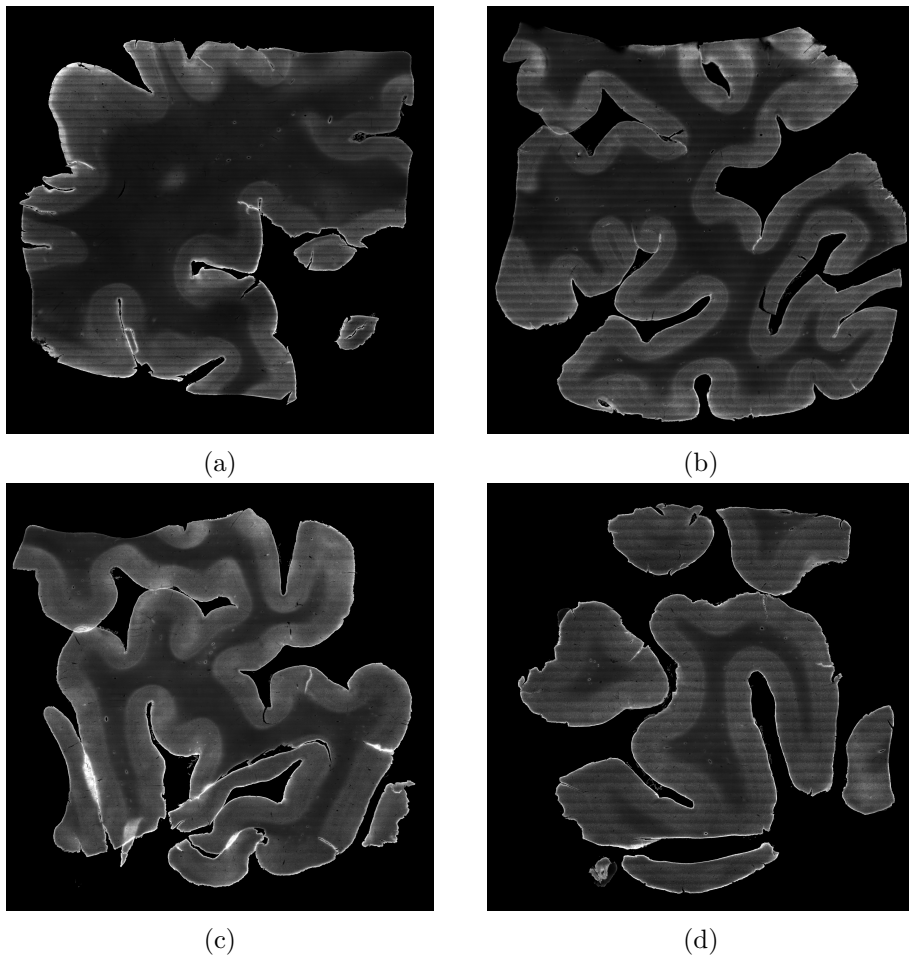


Figure 5.8: Wide-scale reconstruction of Broca Area slices: samples (a), (b), (c) and (d) are, respectively slice 6, 18, 30 and 42 from the 48-slice Broca Area sample, imaged in the NeuN channel. Individual acquisitions are aligned and stitched using *ZetaStitcher*, each slice is represented by approximately 4.72 TBs of data (up to 18.89 TB for multichannel stacks) and can be aligned in less than a minute thanks to heavy parallelization of the module. After the alignment process, optimal relative displacements between stacks are computed by *ZetaStitcher*, whole stitched volumes can be accessed through an API and calculated on the fly starting directly from compressed acquisition stacks to avoid over-redundancy and wasteful disk space occupation.

In figure 5.8 we have reconstructed four different slices of the same Broca Area sample starting from individual acquisitions. Illumination attenuation inside each light-sheet acquisition is the cause of the slightly visible horizontal artifacts, while overall readability of the final images is not impaired, this effect serves as a visualization tool for how the stacks were combined into a single imaging volume.

Lastly, isotropic resolution views were created via reslicing of the original data, these versions of the dataset are useful for inter-modality registration or general consultation and are characterized by significantly more manageable digital footprints. Without any compression scheme, just reslicing the dataset to $3.6\ \mu\text{m}$ isotropic resolution occupies 10.4 TB of disk space, this can be easily compressed to 4.2 TB using additional lossless compression, achieving the the same volume occupation of a single acquisition stack in the original resolution.

5.3.2 Semantic Segmentation

Semantic segmentation of neuronal somas was performed on four entire slices from the dataset, using the NEUROresUNet model trained on manual annotations of the NeuN channel. Of the 48 individual slices of the dataset, slice 6, 18, 30 and 42 were chosen to represent the entire sample space as faithfully as possible. The *NeuroSegmenter* framework introduced earlier in this thesis was used for model definition, training and parallelized inference on individual acquisition stacks using Docker process containerization and HTCCondor to distribute the workload across multiple machines and GPUs in our computational cluster. To create an extensive neuron soma presence probability map of the entire analyzed dataset we used ZetaStitcher on probability stacks, with the same alignment parameters we optimized for acquisition stitching.

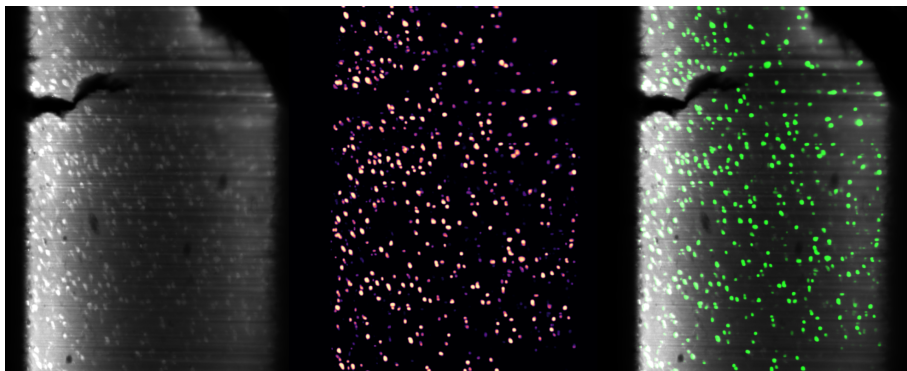


Figure 5.9: Semantic Segmentation of Neuronal Somas in LightSheet Fluorescence Microscopy using NEUROresUNet: the leftmost image is a depiction of a raw light-sheet acquisition from a 500 μm -thick NeuN-stained slice. The typical artifacts of Light-Sheet Microscopy are noticeable, namely the directional light attenuation and horizontal striping artifacts, the visual features of the neuronal somas are defined across the entire dynamic range, making the interpretation of images from this modality particularly challenging for non-experts. The middle image represents the corresponding probability map of neuronal soma presence, as produced by NEUROresUNet, overlaid on the original acquisition in the rightmost one, the model is able to work across the dynamic range and accurately detects low contrast objects that would be hardly noticeable even by a human expert.

Figure 5.9 offers a visual representation of the segmentation inference on a single 2048×2048 acquisition frame. The network outputs high neuronal presence probability in correspondence of the detected objects, just from visual inspection it's possible to notice detection consistency across the wide range of image conditions that might be present inside an acquisition field of view. A single image can contain the horizontal striping artifacts that are typical to Light-Sheet Microscopy, light attenuation and near-plane light-sheet thickness effects, our CNN model is satisfactorily resilient with respect to the range of imaging conditions

that characterizes this modality.

The network we implemented has a 256×256 receptive field and outputs probability maps of the same pixel dimensions. In order to perform inference on entire frames we averaged multiple network responses from partially overlapping inputs tiles with a radial weighting window. The computational overhead caused by repeated inference over the same areas is balanced by the heavy reduction of border artifacts caused by padded convolutions. We included automated model-agnostic inference on arbitrarily sized 2D and 3D datasets using partially overlapping input tiles as part of the NeuroSegmenter package: the dimensions of the overlapping areas, type of averaging window and multi-tile fusion strategy are all parameters that the end user can tune to balance computational overhead and desired output accuracy.

F1 Score	True Positive Rate (TPR)	True Negative Rate (TNR)
.641	.585	.989
Area Under ROC Curve	Area Under PR Curve	Accuracy
.976	.715	.971

Table 5.3: Semantic segmentation metrics over the test dataset: voxel-wise performance estimates are given for the F1 score (Dice coefficient), True Positive and True Negative Rates, Accuracy, as well as the integrals under the Receiver Operator Characteristic and Precision-Recall Curves.

Numerical pixel-wise semantic segmentation performance metrics of the model have been evaluated over a test dataset comprised of $37 \ 32 \times 256 \times 256$ image stacks, randomly sampled from the full extent of all 48 Broca Area slices, results are reported in table 5.3. Precision-Recall and Receiver Operator Characteristic curves have also been computed and plotted in figure 5.10.

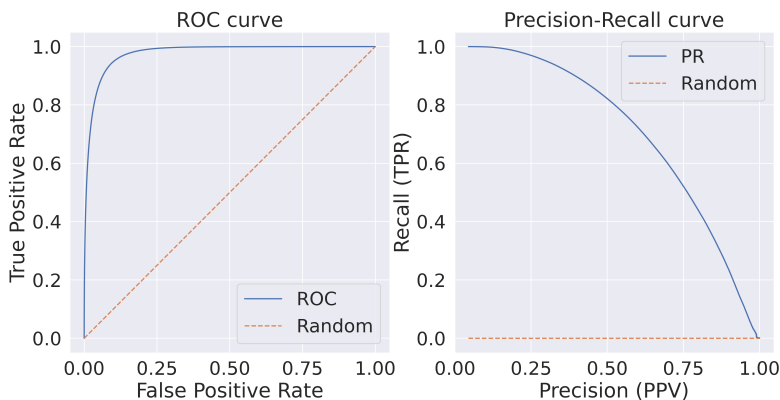


Figure 5.10: Response Operator Characteristic and Precision-Recall Curves evaluated on the test Datasets: PR and ROC curves calculated varying the classification threshold over the continuous neuron soma presence probability heatmaps, dotted lines represent the random classifier references.

5.3.3 Instance Segmentation

In Figure 5.11 we have presented whole-slice acquisition and reconstruction using the ZetaStitcher alignment tool of four slices from the Broca Area sample. We’ve obtained neuronal soma presence probability maps with the NEUROresUNet model, these maps are then processed using the instance segmentation pipeline introduced in 5.2.4 to separate individual neural somas from the global probability map. In this case, similarly to the last chapter, we’ve also applied the 2.5D scheme described earlier, that basically consists in treating the aligned volumetric stack of independent 2D probability maps (obtained with our 2D segmentation network) as a statistical proxy to an underlying volumetric neuronal presence probability distribution: the separation of different items inside the 2.5D probability heatmap operates on a volumetric level and not on single images.

Each neuronal soma detected inside the imaging volume (more than 6×10^6 objects are present in each slice) is individually measured, characterized and stored in a database with a unique identifier. We are able to link each of the individually detected neurons to its position in real space, its pixel coordinates inside the volume, its volume and linear dimensions and general descriptors of its shape such as the major and minor axis lengths.

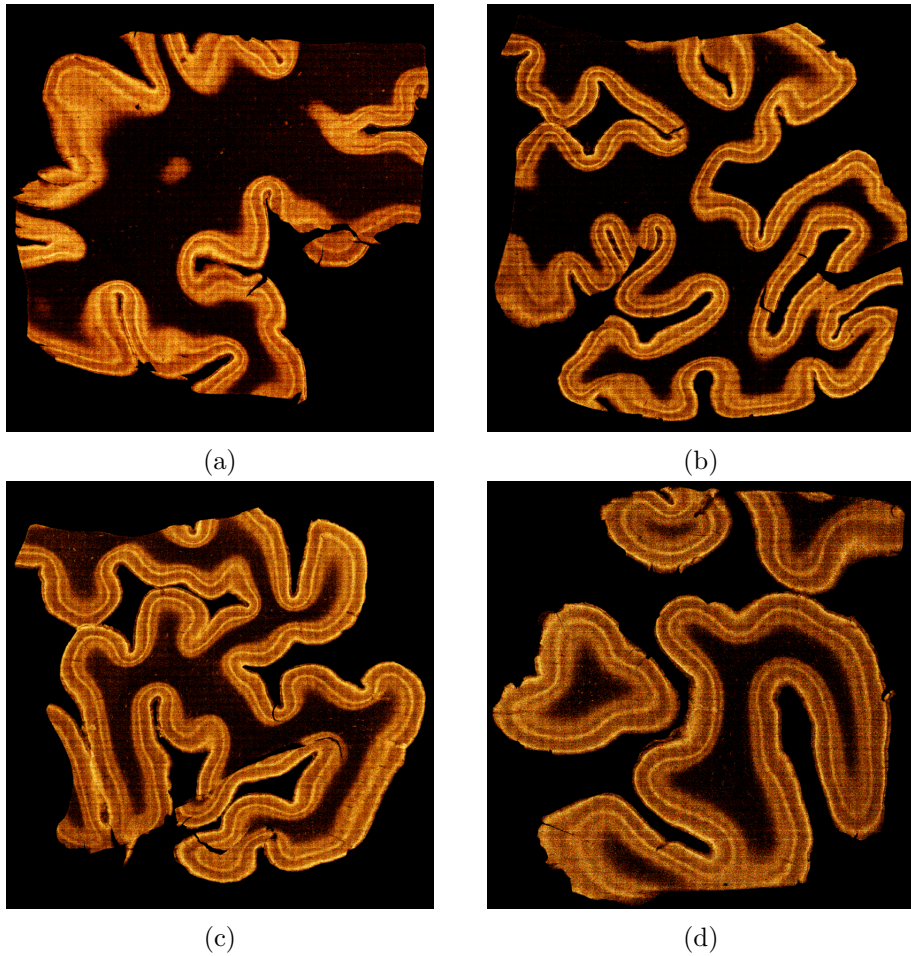


Figure 5.11: Neuronal Density Distributions in Broca Area Slices: figures (a), (b), (c) and (d) represent the volumetric density distributions of detected objects in the three-dimensional space of the samples. Each pixel of these images is linked to a volumetrical bin in sample space, pixel color displacement over a *fire* colormap (black to red to yellow) represents the relative density of detected neuronal somas. Layer structuring which can be seen faintly in figure 5.8 is evident just from a rapid analysis of soma numerical density. The number of individual objects represented in each of these figures is over 6×10^6 .

Figure 5.11 depicts the distribution of segmented neuronal somas across the four entire slices, each pixel in the image is representative of a spatial bin in sample space, and its color value across a red-to-yellow colormap is proportional to the relative neuronal density of detected objects in that specific bin division. In order to correctly evaluate neuronal distributions, spurious activations outside the sample volume are filtered with a manually traced mask.

We're also able to produce visualization of different distributions than just numerical density: in figure 5.12 we've represented the distribution of soma diameter inside the four samples. Instead of directly measuring linear diameters of the objects, which might be difficult and computationally expensive due to resolution anisotropy and non-trivial geometry of the detected neurons, we've calculated diameters starting from the direct voxel measurement of their volume under spherical approximation.

5.4 Discussion

5.4.1 Neuronal Counting in the Broca Area

The queryable maps obtained after instance separation enable us to perform quantitative inference of quantities that are generally inferrable only from manual stereology and histological analysis. Moreover, when queries from our database are combined with histological expertise in the definition of masks, we're able to perform deep characterizations and directly measuring histological quantities with minimal effort.

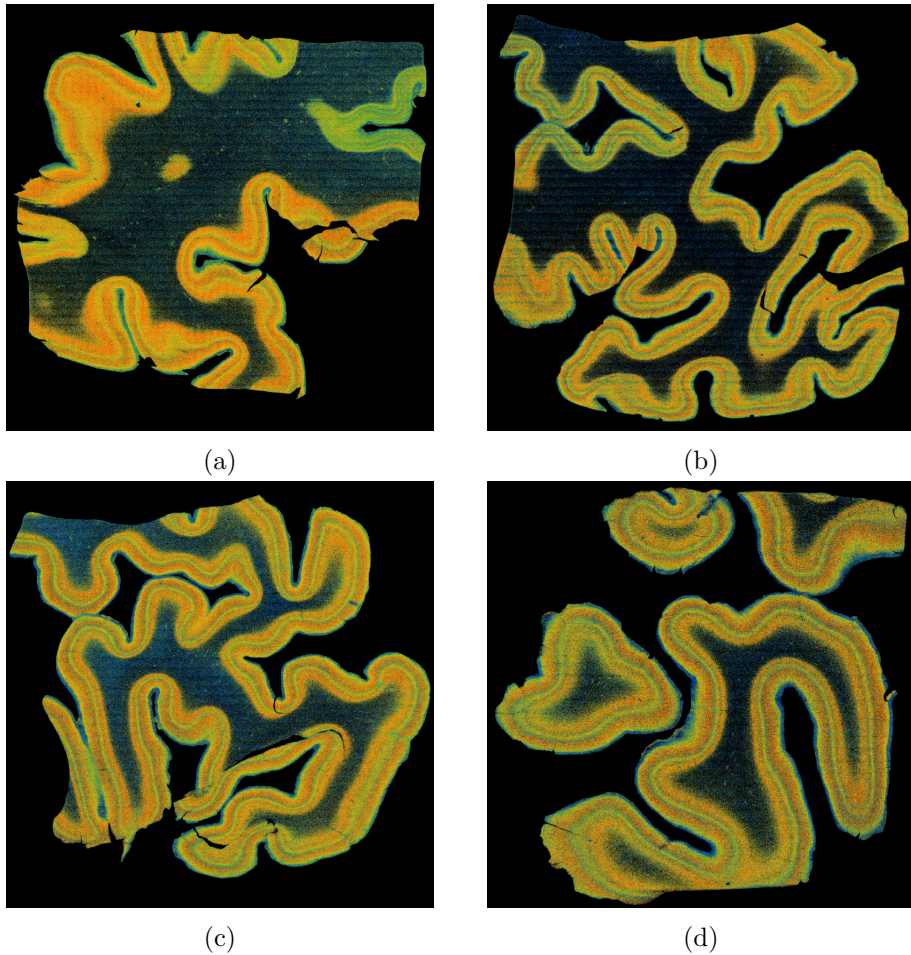


Figure 5.12: Soma Diameter Distributions in Broca Area Slices: figures (a), (b), (c) and (d) are a visualization of the distribution of the soma diameters inside the sample, colors are chosen in a *rainbow* colormap (blue to green to red). Cortical layer stratification is also evident in this representation.

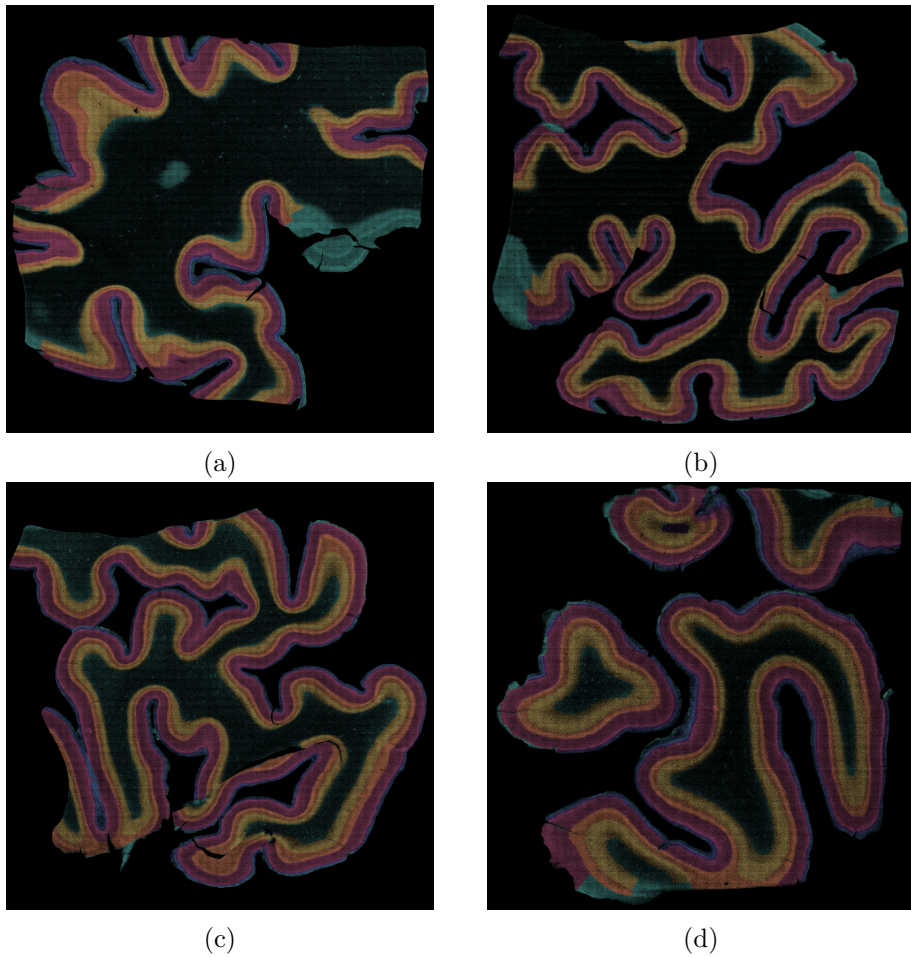


Figure 5.13: Cortical Layer Segmentation in the Broca Area: (a), (b), (c) and (d) represent, respectively the distributions of detected objects in slices 6, 18, 30 and 42 marked by their cortical layer classification. Individual layers of the cortex are individuated using segmentation masks traced on our acquisitions by the Hof Lab at Ichan School of Medicine at Mount Sinai. Mount Sinai also produced manual stereological neuron count estimates on layers 3, 5 and 6, here marked in purple, orange and yellow.

Slice	Layer	Population ($\times 10^5$) (detected)	Population ($\times 10^5$) (stereology estimate)
6	1	3.69 ± 0.45	n/a
	2	2.63 ± 0.37	n/a
	3	17.96 ± 2.67	9.05 ± 3.38
	5	11.11 ± 1.52	5.73 ± 2.22
	6	11.13 ± 1.88	8.14 ± 3.05
	18	1	3.15 ± 0.40
2		3.55 ± 0.61	n/a
3		19.99 ± 3.26	15.18 ± 5.48
5		12.84 ± 1.85	10.72 ± 3.87
6		12.13 ± 1.76	14.01 ± 5.05
30		1	3.49 ± 0.46
	2	2.66 ± 0.43	n/a
	3	21.04 ± 2.71	20.89 ± 7.24
	5	12.49 ± 1.90	12.49 ± 4.50
	6	10.30 ± 1.50	8.50 ± 3.07
	42	1	2.41 ± 0.34
2		2.42 ± 0.35	n/a
3		12.40 ± 1.76	6.28 ± 2.35
5		7.48 ± 0.94	5.63 ± 2.11
6		7.36 ± 1.04	8.14 ± 2.94

Table 5.4: Neuronal Population per Cortical Layer inside the Broca Area: countings of detected somas for each cortical layer have been compared to stereology based estimations. Stereological population estimates have been conducted for layers 3, 5 and 6 of the cortex, layer 4 is not visible in the Broca Area Sample. With the exception of layers 3 in slices 6 and 42, and layer 5 in slice 6, the number of automatically detected neurons per layer is compatible with stereological estimates.

Manual stereological analysis of the four samples was conducted by the Hof Lab at Icahn School of Medicine at Mount Sinai (New York, USA), in the collaborative context of the Brain Initiative Cell Census Network (BICCN) project. Mount Sinai provided stereology counting of the neuronal somas inside a 3.6 μm resolution version of our dataset that we were able to provide for their analysis. We used the layer segmentation masks to perform of measurements on individual cortical layers.

5.4.2 Segmentation Model Performance and Manual Annotation Challenges

Numerical results appear to be dominated by a marked tendency to false positive detection, this is also reflected in a modest numerical value of the F_1 score. The F_1 metric, defined as the harmonic mean of precision and sensitivity

$$F_1 = \frac{2TP}{2TP + FP + FN} \quad (5.1)$$

is an indicator of the overall quality of a segmentation and takes values between 0 and 1, we would expect a good segmentation score to be placed in the 0.7 to 0.9 range for a typical application.

Figure 5.9 gives insight on a possible reason for this other than the obvious difficulty of the task in itself. Visual inspection of the activation maps reveals that low intensity objects, as well as marginal sections of neuronal somas and near-plane appearances, consistently trigger positive response from the network: while detection of these objects in their entire extension is surely desirable, human annotators operating on bi-dimensional sections with limited contrast windows of the 16bit color range would encounter very high difficulty in distinguishing between these objects and background. Ultimately it's possible that ground truth annotations are affected by operator-induced over-representation of the negative class and label uncertainty due to fuzzy appearance of the ex-

act borders of the somas in low illumination areas, as well as near-plane effects. The problem of operator-induced contrast window selection bias can possibly be tackled using an *active labeling* approach, where a segmentation model, trained on an isolated partition of the manually annotated data, can be used to signal to the operator the presence of possible low intensity objects that are masked by his current contrast visualization window.

The metrics in table 5.3 are calculated with a fixed classification threshold of 0.5. Both the Receiver Operating Characteristic Curve plotted in figure 5.10, often known as a ROC curve, and the Precision-Recall curve trace the model's binary classification capacity at different values of its discrimination threshold, in the first case the True Positive Rate is plotted at the variation of False Positive Rates, in the second one the model's Precision (defined the number of true positives over all positives) is plotted against its Recall (True Positive Rate). Neither of the two curves has an intuitive explanation, but comparison with an *indecisive model* baseline with no discriminatory capacity, plotted in dashed lines, is a visual tool to assess the model's capacity to distinguish between positive and negative classes: the higher the area under these curves, the better the model. Both curves show that the model is more than capable of distinguishing neuronal somas from background.

Imaging of large and heterogeneous tissue samples, even after perfecting clearing, staining an imaging protocols, inevitably produces variability in image quality. Structural inhomogeneity of the tissue can result in significant difference in illumination conditions and horizontal striping artifacts, and antibodies may penetrate inhomogeneously in different parts of the tissue causing significantly different signal intensity across the imaged area. Moreover, the finite thickness of the light

sheet contributes to blurring the margins of individual objects defined on multiple planes. LSFM images can be hard to read even for specialized microscopists and are undoubtedly difficult to annotate when the objects are expressed in largely different areas of a 16bit monochromatic brightness range. It's expected that, in annotating this kind of data, we would have significant inter-operator variability.

5.4.3 Architecture Selection

The parametric formulation of NEUROresUNet was a planned design feature to facilitate model selection so that the effects of architectural optimization could be observed by launching multiple training sessions with varying hyperparameters. However, CNN model training is an inherently stochastic process: stochastic gradient descent, batch selection and real-time data augmentation are all determined by the initialization states of many internal Random Number Generators (RNGs). As a consequence, assessing the validity of hyperparameter selections should require a large number of training runs and in-depth statistical analyses for each of the parameters we wish to optimize: given the large amount of time and computational resources associated to the training process, relying on this approach would be extremely unpractical. As an alternative strategy, we chose to simulate computational determinism of the training process by tracing each random number generator and fixing its initialization stage. Even if complete determinism of the training process can't always be reached [10, 12] due random behavior in GPU computation¹, fixing random seeds for model selection is generally regarded to as a *safe practice* [2]. By forcing approximate determinism we can avoid expensive statistical assessments and significantly reduce

¹At the time of writing, since version 2.8.0 TensorFlow includes an experimental mode for OP determinism: in our experience this feature isn't - by itself - sufficient for complete computational determinism.

needed computational resources by directly comparing runs corresponding to different hyperparameter values.

We decided to include in our NeuroSegmenter framework the possibility of centralizing seed selection for its various components and for any external library involved in model training and data generation. By using this feature we were able to select an optimal number of levels for our network, a number of convolutional filters per convolutional block, as well as preemptively evaluating the effects of altering model hyperparameters, such as the learning rate, thus minimizing the number of needed training runs.

5.4.4 Potential Limitations of CNN-Based Segmentation Approaches

The main limitation of CNN-based segmentation techniques, including the one presented in this work, is the inherent request for significant quantities of ground truth data annotations. This kind of data is generally hard to obtain as it needs to be produced by specialized experts with high familiarity with both the imaging modality and the specific subject. Moreover, the data distribution described by the microscopy acquisitions can vary significantly depending both on tissue and clearing characteristics and imaging apparatus configuration. The high cost involved in producing suitable datasets for model training can be prohibitive for small sized research facilities and independently producing large volumes of manual annotations is often a target out of reach for most. The first line of defense against excessive data annotation costs and model under-generalization is represented by data augmentation: at training time the dataset is transformed by application of the non-deterministic transforms listed in 5.1, this can effectively widen the data space domain over which the model is trained, providing predictive capabilities over a variety of imaging and sample conditions which is not

necessarily represented in the original dataset. Although we consider the validity of our data processing approach to be limited to transpositional inference over the imaging acquisitions in which we sampled our relatively small manually annotated dataset, we included a large variety of data augmentation transforms as part of a generalization effort to potentially different samples and imaging setups which, at the time of writing, is still under evaluation and characterization.

Other than data augmentation, there are a number of approaches that can be considered to mitigate and reduce manual annotation requirements for application on different data and setups different from the ones we calibrated our system on, that we didn't include in this work. The provided models can be fine tuned with small quantities of application-specific data - without the need to produce an entirely new dataset - using a variety of *transfer-learning* techniques [7, 17, 30]. The main idea behind these methods is that the vast majority of low-level features is shared between similar datasets with slightly different samples and imaging setups: the bulk of feature extracting stages in the network can be left identical to the ones we provide in our model, while a few specifically collected data points of the new domain can be used to train selectively the last few layers. In this way we can provide a further level of generalization to different sample and imaging condition without the need of creating another dataset ex-novo.

Other approaches include the use of generative data models to simulate artificial data points of the target space to be used in training [1, 3, 15, 27] or the use of pseudo-labels produced by pre-existing models on unlabeled data [6, 25, 31]. Most of these methods share the same intuition that low-level features in similar datasets share some level of closeness that allows them to be learned from either from inference by pre-trained models on unlabeled data or - even more cheaply - from entirely synthetic datapoints, instead of expensive domain-specific manual

annotations.

5.5 Conclusions

We have presented a functional pipeline for quantitative exploration of brain tissue that employs biological tissue transformation and clearing with the SHORT protocol, high resolution Light Sheet Fluorescence Microscopy imaging with a custom-built apparatus, whole-sample digital reconstruction using custom developed software (ZetaStitcher), deep learning-enabled semantic segmentation of the data using the an original CNN model (NEUROresUNet) running on an internally developed high level deep learning framework (NeuroSegmenter), followed by instance segmentation and data aggregation. Innovations on the software side of the pipeline are paralleled by a specifically built data management flow and modular computational infrastructure design with the aim of making the whole process arbitrarily scalable to even larger areas.

The diSPIM LSFM apparatus, in combination with sample slicing, tissue transformation and clearing, has enabled for imaging of $500\mu\text{m}$ thick slices of the human Broca Area sample and its digital reconstruction on a large scale. We have demonstrated the use of a custom-designed CNN network for neuronal soma semantic segmentation on whole tissue slices, moreover we have employed a fast processing pipeline for separation, measurement and indexing of individual objects from the segmentation map. These methods, coupled with our modular and scalable processing architecture allow us to obtain detailed and queryable maps of neuronal somas in large sections of the human Broca Area.

Obtaining interrogable and queryable maps from biological samples can be considered the end goal of the whole tissue clearing, imaging and data processing pipeline: in this sense, the mere fact that we're able

to perform computer queries on tens of millions of individual neurons and obtaining their position and physical dimensions testimonies to the validity of our methods.

5.5.1 Data and Software Availability

The fluorescence imaging data relative to Brodmann Areas 44 and 45, after compression and reslicing to a 3.6 μm isotropic resolution, was formatted in the Brain Imaging Data Structure (BIDS) data specification and uploaded to the the DANDI Archive as part of a collaborative multi-modal atlas of the human Broca Area in the context of the Human Brain Cell Census Network (BICCN). The *ZetaStitcher*, *SPIMLab*, *NeuroSegmenter* and *pyometiff* software packages are available on GitHub [4, 5, 18, 19].

Bibliography

- [1] Antreas Antoniou, Amos Storkey, and Harrison Edwards. *Data Augmentation Generative Adversarial Networks*. en. Nov. 2017. URL: <https://arxiv.org/abs/1711.04340v3> (visited on 03/20/2023).
- [2] Steven Bethard. *We need to talk about random seeds*. Oct. 2022. DOI: 10.48550/arXiv.2210.13393. URL: <http://arxiv.org/abs/2210.13393> (visited on 01/11/2023).
- [3] Francesco Calimeri et al. “Biomedical Data Augmentation Using Generative Adversarial Neural Networks”. en. In: *Artificial Neural Networks and Machine Learning – ICANN 2017*. Ed. by Alessandra Lintas et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 626–634. ISBN: 978-3-319-68612-7. DOI: 10.1007/978-3-319-68612-7_71.
- [4] Filippo Castelli. *neurosegmenter*. Dec. 2022. URL: <https://github.com/filippocastelli/neurosegmenter> (visited on 12/20/2022).
- [5] Filippo Castelli and Giacomo Mazzamuto. *pyometiff*. Jan. 2023. DOI: 10.5281/zenodo.7520378. URL: <https://doi.org/10.5281/zenodo.7520378>.
- [6] Filippo M. Castelli et al. “Semantic Segmentation of Neuronal Bodies in Fluorescence Microscopy Using a 2D+3D CNN Training Strategy with Sparsely Annotated Data”. en. In: *Machine Learn-*

- ing, Optimization, and Data Science*. Ed. by Giuseppe Nicosia et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020, pp. 95–99. ISBN: 978-3-030-64583-0. DOI: 10.1007/978-3-030-64583-0_10.
- [7] Veronika Cheplygina, Marleen de Bruijne, and Josien P. W. Pluim. “Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis”. en. In: *Medical Image Analysis* 54 (May 2019), pp. 280–296. ISSN: 1361-8415. DOI: 10.1016/j.media.2019.03.009. URL: <https://www.sciencedirect.com/science/article/pii/S1361841518307588> (visited on 09/06/2021).
- [8] Irene Costantini et al. *A multimodal imaging and analysis pipeline for creating a cellular census of the human cerebral cortex*. en. Oct. 2021. DOI: 10.1101/2021.10.20.464979. URL: <https://www.biorxiv.org/content/10.1101/2021.10.20.464979v1> (visited on 01/28/2023).
- [9] Irene Costantini et al. “A versatile clearing agent for multi-modal brain imaging”. en. In: *Scientific Reports* 5 (May 2015), p. 9808. ISSN: 2045-2322. DOI: 10.1038/srep09808. URL: <https://www.nature.com/articles/srep09808> (visited on 05/07/2019).
- [10] Duncan Riach. *Deep Learning Determinism*. May 2019. URL: https://www.youtube.com/watch?v=TB07_mUMt0U (visited on 01/11/2023).
- [11] Vladislav Gavryusev et al. “Swift light sheet volumetric charting of large human brain portions”. EN. In: *Biophotonics Congress: Biomedical Optics 2020 (Translational, Microscopy, OCT, OTS, BRAIN) (2020), paper JW5A.3*. Optical Society of America, Apr. 2020, JW5A.3. DOI: 10.1364/TRANSLATIONAL.2020.JW5A.3. URL: <https://www.osapublishing.org/abstract.cfm?uri=Translational-2020-JW5A.3> (visited on 09/05/2020).

- [12] Nick Harris. *A Workaround for Non-Determinism in TensorFlow*. en-US. URL: <https://www.twosigma.com/articles/a-workaround-for-non-determinism-in-tensorflow/> (visited on 01/11/2023).
- [13] Kaiming He et al. “Deep Residual Learning for Image Recognition”. In: *arXiv:1512.03385 [cs]* (Dec. 2015). URL: <http://arxiv.org/abs/1512.03385> (visited on 08/25/2019).
- [14] Kaiming He et al. “Identity Mappings in Deep Residual Networks”. In: *arXiv:1603.05027 [cs]* (Mar. 2016). URL: <http://arxiv.org/abs/1603.05027> (visited on 08/25/2019).
- [15] Reka Hollandi et al. “A deep learning framework for nucleus segmentation using image style transfer”. en. In: *bioRxiv* (Mar. 2019), p. 580605. DOI: 10.1101/580605. URL: <https://www.biorxiv.org/content/10.1101/580605v1> (visited on 09/08/2020).
- [16] Fabian Isensee et al. *batchgenerators - a python framework for data augmentation*. Jan. 2020. DOI: 10.5281/zenodo.3632567. URL: <https://zenodo.org/record/3632567> (visited on 12/23/2022).
- [17] Hee E. Kim et al. “Transfer learning for medical image classification: a literature review”. en. In: *BMC Medical Imaging* 22.1 (Apr. 2022), p. 69. ISSN: 1471-2342. DOI: 10.1186/s12880-022-00793-7. URL: <https://doi.org/10.1186/s12880-022-00793-7> (visited on 03/20/2023).
- [18] Giacomo Mazzamuto. *GitHub - lens-biophotonics/SPIMlab: SPIMlab is a data acquisition and control software for Light Sheet Microscopy*. URL: <https://github.com/lens-biophotonics/SPIMlab> (visited on 01/28/2023).
- [19] Giacomo Mazzamuto. *lens-biophotonics/ZetaStitcher*. Oct. 2022. URL: <https://github.com/lens-biophotonics/ZetaStitcher> (visited on 11/21/2022).

- [20] Dirk Merkel. “Docker: lightweight Linux containers for consistent development and deployment”. In: *Linux Journal* 2014.239 (Mar. 2014), 2:2. ISSN: 1075-3583.
- [21] Evan Murray et al. “Simple, scalable proteomic imaging for high-dimensional profiling of intact systems”. In: *Cell* 163.6 (Dec. 2015), pp. 1500–1514. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.11.025. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5275966/> (visited on 05/13/2019).
- [22] Nobuyuki Otsu. “A Threshold Selection Method from Gray-Level Histograms”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (Jan. 1979), pp. 62–66. ISSN: 2168-2909. DOI: 10.1109/TSMC.1979.4310076.
- [23] Luca Pesce et al. “3D molecular phenotyping of cleared human brain tissues with light-sheet fluorescence microscopy”. en. In: *Communications Biology* 5.1 (May 2022), pp. 1–12. ISSN: 2399-3642. DOI: 10.1038/s42003-022-03390-0. URL: <https://www.nature.com/articles/s42003-022-03390-0> (visited on 01/11/2023).
- [24] Luca Pesce et al. “Fast volumetric mapping of human brain slices”. In: *Neurophotonics*. Vol. 11360. SPIE, Apr. 2020, pp. 43–48. DOI: 10.1117/12.2555644. URL: <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11360/113600A/Fast-volumetric-mapping-of-human-brain-slices/10.1117/12.2555644.full> (visited on 12/15/2022).
- [25] Hieu Pham et al. “Meta Pseudo Labels”. en. In: 2021, pp. 11557–11568. URL: https://openaccess.thecvf.com/content/CVPR2021/html/Pham_Meta_Pseudo_Labels_CVPR_2021_paper.html (visited on 03/20/2023).

- [26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv:1505.04597 [cs]* (May 2015). URL: <http://arxiv.org/abs/1505.04597> (visited on 05/13/2019).
- [27] Veit Sandfort et al. “Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks”. en. In: *Scientific Reports* 9.1 (Nov. 2019), p. 16884. ISSN: 2045-2322. DOI: 10.1038/s41598-019-52737-x. URL: <https://www.nature.com/articles/s41598-019-52737-x> (visited on 03/20/2023).
- [28] Douglas Thain, Todd Tannenbaum, and Miron Livny. “Distributed computing in practice: the Condor experience.” In: *Concurrency - Practice and Experience* 17.2-4 (2005), pp. 323–356.
- [29] Donghao Zhang et al. “Automated 3D Soma Segmentation with Morphological Surface Evolution for Neuron Reconstruction”. en. In: *Neuroinformatics* 16.2 (Apr. 2018), pp. 153–166. ISSN: 1559-0089. DOI: 10.1007/s12021-017-9353-x. URL: <https://doi.org/10.1007/s12021-017-9353-x> (visited on 09/09/2020).
- [30] Fuzhen Zhuang et al. “A Comprehensive Survey on Transfer Learning”. In: *Proceedings of the IEEE* 109.1 (Jan. 2021), pp. 43–76. ISSN: 1558-2256. DOI: 10.1109/JPROC.2020.3004555.
- [31] Yuliang Zou et al. *PseudoSeg: Designing Pseudo Labels for Semantic Segmentation*. en. Oct. 2020. URL: <https://arxiv.org/abs/2010.09713v2> (visited on 03/20/2023).

Chapter 6

Conclusions

Quantitative investigation of extended areas of brain tissue is an inherently complex challenge that needs the orchestrated cooperation of many independent competences. Choices in biological treatment of the samples are intimately correlated to microscopy imaging needs, elaborated data flows must be designed around computational limitations and the automated processing approaches are selected to satisfy expected computational performance targets while achieving satisfactory predictive performance.

In this work we tried to offer a strategy to tackle the problem of mapping the neuronal structures in fluorescence microscopy of human brain tissue to a large and currently unparalleled scale. We designed our processes in the perspective of offering a scalable methodology that could be applied to even larger acquisition targets by means of modular scaling of the involved imaging, storage and computational resources.

We presented two instances of the same end-to-end pipeline for tissue preparation, imaging, sample reconstruction and automated cell segmentation, conjugated in two imaging technologies (Two Photon Fluorescence Microscopy and Light-sheet Fluorescence Microscopy) operat-

ing with different working principles, as well as two different approaches to automated histological analysis.

The two microscopy techniques both cover wide-ranging and overlapping application targets, each bringing its own set of requirements and advantages, ultimately answering different imaging needs: Two Photon Confocal Microscopy combines very high optical sectioning and low visual artifacts with a standard, imaging geometry but its scanning operating principle limits its application spectrum to modestly sized samples. Light-sheet Fluorescence Microscopy, on the other hand, is characterized by geometrically higher volumetric throughput with comparable optical sectioning, but the possibility of implementing multiple views and multi-spectral imaging (in the Di-CLSM variant apparatus we introduced) comes with the implicit need of significant data managing infrastructure to handle massive data flows. Moreover, it suffers from technique-specific artifacts, such as horizontal striping, that generally have a higher visual impact than the ones encountered in TPFM. Future perspectives of scaling the methodology to complete brain areas, and eventually entire organs, pose a definite preference towards the LSM approach for its volumetric reach, making TPFM a better choice for reduced scale applications.

Both variations of the presented pipeline are characterized by the presence of an intermediate sample reconstruction step using the ZetaStitcher software, but differ in their automated cell segmentation steps: in the first case we relied on a lightweight texture classification model to make optimal use of limited ground truth data annotations, in the other case we proposed a fully customizable and tunable architecture that could be scaled to match both ground truth availability and image complexity constraints.

The high volumetric throughput of LSM makes it necessary to also define a modular and scalable computational infrastructure for data

analysis, which, in our design, takes full advantage of modular process containerization using Docker and the HTCondor cluster computation scheduler which makes it ready for deploy on HPC supercomputing facilities. The architecture proposed in this work can easily be adapted to large scale LSFM imaging by addition of computational nodes, up to supercomputers and computational clusters like CINECA.

We then explored the issue of sharing the collected data with the neuroscience community at large, which rendered necessary the definition of an extension to a data sharing format (the BIDS specification [4]) which has since been positively accepted and integrated [1], as well as the introduction of a custom library - *pyometiff* - for interaction with the OME-TIFF format [2] which not only complemented the aforementioned specification extension proposal, but is currently used in other elaboration pipelines in the neuroscience imaging area[7].

The proposition value of this work lies mainly in the definition of an original pipeline structure for automated cell detection and characterization starting from raw samples, which can be replicated as it is or adapted in its parts to specific biological, imaging and computational constraints. We used this pipeline to create histological censuses of brain samples cataloging and characterizing individual neurons in the imaged areas. The end goal of this work is to propose a pipeline that could ultimately be scaled to entire brains and could be used to create extended histological censuses. The general methodological framework established in this thesis will serve - and already does - as a general scaffold for our experiments in large scale imaging and mapping aimed at reaching the volumetric scales of entire human central nervous systems.

6.1 Future Perspectives

The general methodological framework proposed can be expanded beyond its current implementation: here we propose three possible targets that are currently being explored, namely the possibility of integrating additional color channels to the CNN segmentation step, the possibility of working with multiple views of the same sample and, lastly, wide scale application of this method to larger targets in order to create neuronal cell atlases of entire brains.

6.1.1 Multi-channel and Multi-target

As for now, automated cell detection, in the second variant of the pipeline for soma detection and segmentation, makes use of a single imaging channel expressing anti-NeuN signal, but fluorescence multi-channel views of the same tissue can easily be integrated in our segmentation model: in our case, the chosen signal sufficiently correlated with the visual features of interest, but the proposed CNN model can be adapted to work on multi-spectral data, either to improve single-class predictive capabilities or to perform multi-class segmentation with many types of objects. These applications would require additional ground truth annotations to compensate for increased model complexity but can easily extend the application range of the proposed analysis pipeline to those cases where a single channel is not sufficient to fully characterize the objects of interest in the entire sample. A potential application for this case would be the automated characterization of excitatory and inhibitory neuron populations, the distinction of which generally requires multi-channel co-detection.

6.1.2 Multi-View

Many LSFM setups offer the possibility of simultaneous acquisition of multiple views from the same sample: this property can be used to compensate for volumetric resolution anisotropy by combining the two independently acquired anisotropic stacks into a single isotropic volume [5, 6, 8]. Operating on higher resolution stacks is an interesting road that would increase segmentation performances but trade-offs with the significant computational overhead and increased data sizes need to be carefully reviewed, as predictive accuracy advantages can easily be outweighed by increased complexity. A possible, more ambitious road for large-scale multi-view analysis of brain samples would ideally involve specifically designed 3D CNN models operating simultaneously on the two anisotropic stacks. Such models would implicitly learn to perform multi-view deconvolution before segmentation: this approach is still mostly unpaved, architectural research in this direction could directly unlock untapped potential in the diSPIM technology.

6.1.3 Multi-Modal Cell Atlases

Fluorescence Microscopy of human brain tissue involves physical sectioning of the analyzed samples for uniform clearing, staining and light penetration. The processes needed for obtaining fluorescence imaging inevitably causes nontrivial geometrical distortions of the tissue that would prevent easy reconstruction of different acquisitions. To cover the gap between single-slice analysis and alignment of whole areas to standardize brain atlases, a mesoscopic reference is needed to tie the LSFM-detected neurons to a global reference. We presented LSFM imaging of the Broca Area that was acquired as part of a collaborative effort [3] towards the construction of kind of multi-modal atlas. The pipeline we created was tested on the samples described during this work but

wide-scale application is still unexplored. We're currently working on applying our methodological solution to a larger number of samples, and primarily on the rest of the Broca Area specimen we've presented. Modularity of our processing approach enables us to look at HPC super-computing solutions as a future development of this work for scaling our analyses and drastically reducing time needed for obtaining neuronal maps.

Bibliography

- [1] Marie-Hélène Bourget et al. “Microscopy-BIDS: An Extension to the Brain Imaging Data Structure for Microscopy Data”. In: *Frontiers in Neuroscience* 16 (Apr. 2022), p. 871228. ISSN: 1662-4548. DOI: 10.3389/fnins.2022.871228. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9063519/> (visited on 05/18/2022).
- [2] Filippo Castelli and Giacomo Mazzamuto. *pyometiff*. Jan. 2023. DOI: 10.5281/zenodo.7520378. URL: <https://doi.org/10.5281/zenodo.7520378>.
- [3] Irene Costantini et al. *A multimodal imaging and analysis pipeline for creating a cellular census of the human cerebral cortex*. en. Oct. 2021. DOI: 10.1101/2021.10.20.464979. URL: <https://www.biorxiv.org/content/10.1101/2021.10.20.464979v1> (visited on 01/28/2023).
- [4] Krzysztof J. Gorgolewski et al. “The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments”. en. In: *Scientific Data* 3.1 (June 2016), p. 160044. ISSN: 2052-4463. DOI: 10.1038/sdata.2016.44. URL: <https://www.nature.com/articles/sdata201644> (visited on 01/10/2023).
- [5] Min Guo et al. “Rapid image deconvolution and multiview fusion for optical microscopy”. en. In: *Nature Biotechnology* 38.11 (Nov. 2020), pp. 1337–1346. ISSN: 1546-1696. DOI: 10.1038/s41587-

- 020-0560-x. URL: <https://www.nature.com/articles/s41587-020-0560-x> (visited on 01/17/2023).
- [6] Bihe Hu, Guang Li, and J. Quincy Brown. “Enhanced resolution 3D digital cytology and pathology with dual-view inverted selective plane illumination microscopy”. EN. In: *Biomedical Optics Express* 10.8 (Aug. 2019), pp. 3833–3846. ISSN: 2156-7085. DOI: 10.1364/BOE.10.003833. URL: <https://opg.optica.org/boe/abstract.cfm?uri=boe-10-8-3833> (visited on 01/17/2023).
- [7] Min Jang et al. *Spatial transcriptomics for profiling the tropism of viral vectors in tissues*. en. Jan. 2023. DOI: 10.21203/rs.3.rs-1486912/v1. URL: <https://www.researchsquare.com> (visited on 01/24/2023).
- [8] Yicong Wu et al. “Simultaneous multiview capture and fusion improves spatial resolution in wide-field and light-sheet microscopy”. EN. In: *Optica* 3.8 (Aug. 2016), pp. 897–910. ISSN: 2334-2536. DOI: 10.1364/OPTICA.3.000897. URL: <https://opg.optica.org/optica/abstract.cfm?uri=optica-3-8-897> (visited on 01/17/2023).