POLITECNICO DI MILANO

Department Of Electronics, Information and Bioengineering

Master of Science in Biomedical Engineering

Cell-type specific connectivity analysis in the mouse somatosensory cortex

Supervisor: Prof. Alessandra PEDROCCHI

Co-supervisor: Francesco Jamal SHEIBAN

Alessandra Maria TRAPANI

Candidate:

Giulia RUFFO

Identification number: 905195

Academic Year 2021-2022

The brain: if you feed it, it works.
If you let it go and retire it, it weakens.
Its plasticity is formidable.
For this we must continue to think.

- Rita Levi Montalcini

Contents

1 Introduction **1**
1.1 The nervous system . . . 2
1.1.1 Neuron . . . 2
1.2 Computational Neuroscience . . . 3
1.2.1 Brief Introduction . . . 3
1.2.2 Single neuron models . . . 4
1.2.3 Synaptic Plasticity . . . 6
1.2.4 Spiking neurons models . . . 8
1.2.5 Neural networks models . . . 13

2 State of the Art **15**
2.1 Machine Learning for neuron classification . . . 15
2.2 3-D volume reconstruction . . . 18
2.3 UNI-EM . . . 21
2.4 Flood-Filling Networks . . . 23
2.5 FLoRIN . . . 27
2.6 Cell-type structure and connectivity map . . . 29

3 Materials and Methods **33**
3.1 Experimental Data . . . 33
3.1.1 MICrONS . . . 34
3.2 Data Processing . . . 36

4 Discussion **39**

5 Greetings **41**

Elenco delle figure **43**

Elenco delle tabelle **47**

Bibliography **49**

Abstract

In the last century research, technological advancement and new scientific research have increased and improved research on the brain and nervous system. This is also thanks to the progress made in the field of molecular biology, electrophysiology and computational neuroscience, enclosed in a single field called neuroscience.

Neuroscience represents the scientific study of the nervous system and investigates its functioning, development, maintenance and anatomy. The goal of neuroscience is to understand not only how the nervous system works in health conditions, but also, when it does not work properly.

This thesis will focus in particular on analyzing neurons, the type of connections they make and how they do it.

Neurons

Neurons are the fundamental units of the nervous system. They are responsible for carrying information throughout the human body using electrical and chemical signals in order to help the coordination all of the necessary functions of life. There are three fundamental parts that compose these cells:

- the **soma**, or cellular body, in which is possible to find the nucleus and and various cytoplasmic organelles. From it, the dendrites and the axons originate;
- the **dendrites** carrying information from other neurons to the soma. They are the "input" part of the cell;
- the **axons** carrying information from the soma to other cells. They represent the "output" part of the cell. It propagates information to other neurons by forming synapses with the dendrites of other neurons.

Computational neuroscience

Computational neuroscience is the field of study in which mathematical tools and theories are used to investigate brain function. It can also incorporate diverse approaches from electrical engineering, computer science and physics in order to understand how the nervous system processes information. In this field of study are included all the methods that are being used to represent the nervous system and its functioning.
When it comes to modeling individual elements of the brain, e.g. neurons, there is a problem related to the balance between the computational cost and the availability of acquired data. The mathematical models empolyed for this scope are called **single neuron models** and can be either be matematical abstraction of the neurons (**point neuron models**) or complex detailed models, ecompassing both morphological aspects (*compartment models*) and electrophysiology (*H-H models*).
The **H-H model** is a mathematical model that describes the variations of membrane conductance, by reproducing the dynamic characteristics of ionic channels (and thus, the overall spiking behaviors) using an electrical equivalent circuit. **Compartmental models** use multiple parameters to describe different HH models one for each cell's

compartment, so to explain the different electrophysiological properties that different part of the same neuron exhibits.

Point neuron models describe the firing properties of neurons in a less complex way. This category includes the *integrated-and-fire*, *Izhikevic*, *Adaptive Exponential Leaky Integrate-and-Fire*, *Generalized LIF* and *Extended-Generalized LIF* models. Thanks to a lighter computational cost, point neuron models can be connected in **spiking neural networks, SNNs**, mathematical models that mimic the information processing of real neurons, i.e. biological behaviour and biochemical variations within neurons when there is an action potential.

This thesis focused on analyzing the different experimental tools used to reconstruct the neural connectivity of the mice cerebral cortex. Collecting such information about could eventually guide the design of more bio-realistic SNNs.

Methods

There are several methods for the acquirement of images of the brain. The best technique, however, to investigate the detailed structure of tissues, cells, organelles and macromolecular complexes cellular components in detail, is **electron microscopy**.

Over the years, several projects have focused on the creation of a software tool that simultaneously allowed to annotate and distinguish, within EM images, neurons and their characteristics (excitatory, inhibitory, etc.). Thus, the focus of this thesis was initially directed on the study conducted by Kashturi et al. in which the authors tried to analyze and describe automated technologies to probe the structure of neural tissue, in order to create a 3-D reconstruction of a sub-volume of mouse neocortex. They acquired high-resolution images (3nm/pixel), making sure that the boundaries of the membrane were easy to reconstruct and then, using the same samples, the imaged the sections at lower resolutions in order to acquire images of larger tissue volumes, and made the data publicly available.

The first attempt in analyzing Kasthuri et al.'s data was a program called *VAST*, created by the authors themselves, which allows users to highlight and distinguish the

various cellular components in the input images, organize the results in an annotation framework and export the results for the 3-D analysis. Once created, the ground-truth can be used in the image processing tool *RhoANA* to train a machine learning model to segment and classify all the cellular (axons, dendrites and glia) and sub-cellular components (synapses, synaptic vesicles, spines, spine apparati, post-synaptic densities and mitochondria). The software has five stages of processing: membrane classification, 2-D segmentation, 3-D segmentation, block matching and global remap. However, since the approach is exper-based and time consuming, it couldn't be used in the present thesis.

Another program that has been taken into account is *UNI-EM*. This environment, oriented to biological researchers that lack programming skills, implements several 2-D and 3-D CNNs on the widely used Tensorflow framework, as well as the proofreading software Dojo and a series of 2-D/3-D filters for classic image processing. The data processing pipeline in this program also consist of five steps: pre-processing, training, inference, post-processing and, finally, visualization and proofreading. An example of how the program works is given in the paper itself [1]. In this last one they used a particular type of CNN, called *flood-filling network*, which shows high accuracy in neuron segmentation but requires a pre-processing step to create the ground-truth. Flood-Filling Networks (FFNs) are a class of neural networks designed for instance segmentation of complex and large shapes, particularly in volume EM datasets of brain tissue, in which a recurrent 3-D convolutional network directly produces individual segments from a raw image. Januszewski et al. demonstrated a novel segmentation approach using the FFNs. It has two channels as input: one with the raw image intensities and the second indicating the position of the object mask in the form of a probability map, created by the network itself. It consists of a deep stack of 3-D convolutional modules with Rectified Linear Unit (ReLU) non-linearities. The input mask in the second channel is incomplete in most inference iterations of the network, and the FFN is trained to extend it within its field of view (FoV). The spatial size of the network FoV is set to $33 \times 33 \times 17$ voxels. All convolutions use $3{\times}3{\times}3$ filters. Inside, the convolutional module, is composed of two convolutional layers, with skip connections between them.

Althuogh machine learning models can be useful as a faster approach, it can have many cons due to the lack of data or the proper neural network to use as a fully automated method, which hasn't been created yet. A new machine learning-free open source method was created by Shahbazi et al., in 2018, called *Flexible Learning-free Reconstrucion of Imaged Neural volumes (FLo-RIN)* and doesn't require a training step. It's divided in three stages: segmentation, identification and reconstruction. Reconstructions created by FLoRIN can be used as a starting point for deciding how to study neural volumes: experts can gain a high-level view and choose regions to study at higher resolutions. Thus, the reconstructions done using FLoRIN don't represent a complete analysis, and require a further processing step with the time-consuming and expert-based methods mentioned above.

The data made available by Kasthuri et al. were insufficient and, in some cases, incomplete for the thesis work to be carried out. For this reason, a new data set provided by the Allen Institute [2] was taken into account. Together with a consortium of laboratories, they created an innovative program called *MICrONS*, which maps the function and the way cortical circuits connect. The Cortical MM^3 database was taken into account. It spans a 1.4mm x 0.87mm x 0.84mm volume of cortex in a P87 mouse and it was imaged using two-photon microscopy, microCT, and serial electron microscopy, and then reconstructed using a combination of AI and human proofreading. The anatomical data contains more than an estimated 200,000 cells, and 120,000 neurons. The database is divided into different dataframes for different usage, but for this project the focus was only onto two dataframes: "Cell Types", in which it is possible to find a subset of nucleus in a 100μm column manually classified as well as the complete segmented subset; "Synaptic Connectivity" with which it is possible to query many aspects of the data, including synapses. To process the data *Google Colab* has been used and it hosts a Jupyter notebook without any setup requirement, avoiding the need of specific GPUs hardware. All the MICrONS Binder dataframes are collected in an hosted server called minnie65. In order to access it, it was necessary to register to the CAVEclient framework. As stated earlier, two are the set of data considered among

the others, and they have been used to create a dataset with all the data necessary for the probability map reconstruction. First, it has been considered the query table with the data of a cortical column in the *"Cell-type"* dataframe. From this, it was possible to highlight each neuron's current state (excitatory/inhibitory) and identified by an ID. To retrieve the position of these neurons in the whole (4 x 4 x 40)nm volume, a new table was considered. The position has been expressed in the three dimensions *(x,y,z)*. The last step was the concatenation of these two dataframes with the table containing the automated synapse detection. Nucleus are recognizable through their location in the synapse (pre- or post-synaptic cleft), which allowed also the distinction for the layers.

Results and discussion

Only after the complete dataset was created, it was clear that all the data from Cortical MM^3 only comprised a portion of the whole cortex (specifically, only layer L4, L5, L6). For future works, it would then be necessary to find a new dataset that includes information about all the layers. After a full 6-layers set would be obtained, it would be possible to use the data extracted from the chosen dataset to generate a set of connectivity rules to be employed when reproducing and simulating realistic brain networks models (such as the cortical column model realized in T.C.Potjans and M. Diessman [3]), improving their biological plausibility.

Sommario

Nel secolo scorso , il progresso tecnologico e la nuova ricerca scientifica hanno aumentato e migliorato la ricerca sul cervello e sistema nervoso. Questo grazie anche ai progressi compiuti nel campo della biologia molecolare, elettrofisiologia e neuroscienze computazionali, facente tutti di un unico campo chiamato neuroscienze.
La neuroscienza rappresenta lo studio scientifico del sistema nervoso e ne indaga il suo funzionamento, sviluppo, manutenzione e anatomia. L'obiettivo delle neuroscienze è capire non solo come funziona il sistema nervoso in condizioni di salute, ma anche quando non funziona correttamente.
Questa tesi si concentrerà in particolare sull'analisi dei neuroni, il tipo di connessioni che fanno e come le fanno.

Il neurone

I neuroni sono le unità fondamentali del sistema nervoso. Sono responsabili del trasporto di informazioni in tutto il corpo umano utilizzando segnali elettrici e chimici per aiutare il coordinamento di tutte le funzioni necessarie della vita. Ci sono tre parti fondamentali che compongono queste cellule:

- il **soma**, o corpo cellulare, in cui è possibile trovare il nucleo e vari organelli citoplasmatici. Da esso provengono i dendriti e gli assoni;
- i textbfdendriti che trasportano informazioni da altri neuroni al soma. Sono la parte "input" della cellula;
- gli textbfassoni che trasportano informazioni dal soma ad altre cellule. Rappresentano la parte "output" della cellula nervosa. Propagano le informazioni ad altri neuroni formando le sinapsi con i dendriti di altri neuroni.

Neuroscienze Computazionali

La neuroscienza computazionale è il campo di studio in cui gli strumenti e le teorie matematiche sono usate per studiare la funzione del cervello. Può incorporare diversi approcci da ingegneria elettrica, informatica e fisica per capire come il sistema nervoso elabora le informazioni. In questo campo di studio sono inclusi tutti i metodi che vengono utilizzati per rappresentare il sistema nervoso e il suo funzionamento. Quando si tratta di modellare singoli elementi del cervello, ad es. i neuroni, c'è un problema legato all'equilibrio tra il costo computazionale e la disponibilità dei dati acquisiti. I modelli matematici impiegati per questo scopo sono chiamati textbfsingle neuron models e possono essere o una forma matematica dei neuroni astratta(**point neuron models**) o modelli dettagliati complessi, comprendenti entrambi gli aspetti morfologici (*compartment models*) ed elettrofisiologici (*modelli H-H*).

L'**H-H** è un modello matematico che descrive le variazioni della conduttanza della membrana, riproducendo le caratteristiche dinamiche dei canali ionici, utilizzando un circuito elettrico equivalente. I **modelli compartimentali** utilizzano più parametri

per descrivere diversi modelli HH uno per ogni compartimento cellulare, in modo da spiegare le proprietà elettrofisiologiche che si manifestano in diverse parti dello stesso neurone.

I **Modelli di neuroni puntiformi** descrivono le proprietà di sparo dei neuroni in modo meno complesso. Questa categoria include i modelli *integrated-and-fire*, *Izhikevic*, *Adaptive Exponential Leaky Integrate-and-Fire*, *Generalized LIF* e *Extended-Generalized LIF*. Grazie ad un costo computazionale più basso, i modelli di neuroni puntiformi possono essere collegati alle **spiking neural networks, SNNs**, modelli matematici che imitano l'elaborazione delle informazioni dei neuroni reali, ovvero il comportamento biologico e le variazioni biochimiche all'interno dei neuroni quando c'è un potenziale di azione.

Questa tesi si concentra sull'analisi dei diversi strumenti sperimentali utilizzati per ricostruire la connettività neurale della corteccia cerebrale dei topi. La raccolta di tali informazioni potrebbe eventualmente guidare la progettazione di SNN più realistiche.

Metodi

Ci sono diversi metodi per l'acquisizione di immagini del cervello. Tuttavia, la migliore tecnica per studiare dettagliatamente la struttura dettagliata dei tessuti, delle cellule, degli organelli e delle componenti cellulari dei complessi macromolecolari, è la **microscopia elettronica**.

Nel corso degli anni, diversi progetti si sono concentrati sulla creazione di uno strumento software che contemporaneamente permettesse di annotare e distinguere, all'interno delle immagini EM, i neuroni e le loro caratteristiche (eccitatori, inibitori, ecc.). Per questo motivo, il focus di questa tesi è stato inizialmente diretto sullo studio condotto da Kashturi et al. in cui gli autori hanno cercato di analizzare e descrivere tecnologie automatizzate per sondare la struttura del tessuto neurale, al fine di creare una ricostruzione 3-D di un sub-volume della neocorteccia del topo. Hanno acquisito im-

magini ad alta risoluzione (3nm/pixel), facendo in modo che i confini della membrana fossero facili da ricostruire e poi, usando gli stessi campioni, hanno acquisito le immagini delle sezioni a risoluzioni più basse ottenendo così immagini su grandi porzioni di tessuto; il tutto reso disponibili pubblicamente. Il primo tentativo di analizzare i dati di Kasthuri et al. è stato un programma chiamato VAST, creato dagli autori stessi, che permette agli utenti di evidenziare e distinguere le varie componenti cellulari nelle immagini di input, organizzare i risultati in un quadro di annotazione ed esportare i risultati per l'analisi 3-D. Una volta creata, la ground-truth, questa può essere utilizzata nello strumento di elaborazione delle immagini *RhoANA* per addestrare un modello di apprendimento automatico a segmentare e classificare tutte le componenti cellulari (assoni, dendriti e glia) e sub-cellulari (sinapsi, vescicole sinaptiche, spine, apparati vertebrali, densità post-sinaptiche e mitocondri). Il software ha cinque fasi di elaborazione: classificazione della membrana, segmentazione 2-D, segmentazione 3-D, block matching e rimappaggio globale. Tuttavia, poiché l'approccio si basa sul grado di esperienza dell'utente, e in più richiede tempo, non potrebbe essere utilizzato nella presente tesi.

Un altro programma che è stato preso in considerazione è *UNI-EM*. Questo ambiente è orientato ai ricercatori biologici che non hanno competenze di programmazione. Implementa diversi CNN 2-D e 3-D sul framework Tensorflow, così come il software Dojo e una serie di filtri 2-D/3-D per l'elaborazione delle immagini. La pipeline di elaborazione dati in questo programma consiste in cinque passaggi: pre-elaborazione, formazione, inferenza, post-elaborazione e, infine, visualizzazione e proofreading. Un esempio di come funziona il programma è dato nel documento stesso [1]. In quest'ultimo hanno usato un particolare tipo di CNN, chiamato *flood-filling network*, che mostra un'elevata precisione nella segmentazione dei neuroni ma richiede un passo di pre-elaborazione per creare la ground-truth. Le Flood-Filling Networks (FFNs) sono una classe di reti neurali progettate per esempio la segmentazione di forme complesse e di grandi dimensioni, in particolare nei set di dati EM di volume di tessuto cerebrale, in cui una rete convoluzionale 3-D ricorrente, produce direttamente singoli segmenti da un'immagine grezza.

Januszewski et al. hanno dimostrato un nuovo approccio di segmentazione utilizzando le FFN. Ha due canali come input: uno con le intensità dell'immagine grezza e il secondo che indica la posizione della maschera dell'oggetto sotto forma di una mappa di probabilità, creata dalla rete stessa. Consiste in una pila profonda di moduli convoluzionali 3-D delle unità lineari rettificate (Relu). La maschera di ingresso nel secondo canale è incompleta nella maggior parte delle iterazioni di inferenza della rete, e la FFN è addestrata ad estenderla nel suo campo visivo (FoV). La dimensione spaziale della rete FoV è impostata a 33 x 33 x 17 voxel. Tutte le convoluzioni utilizzano filtri 3 x 3 x 3. All'interno, il modulo convoluzionale, è composto da due strati convoluzionali.

Nonostante i modelli di apprendimento automatico possono essere utili come approccio più veloce,essi possono avere molti svantaggi a causa della mancanza di dati o la decisioine sulla rete neurale corretta da utilizzare come metodo completamente automatizzato, che, ad oggi, non è stato ancora creato. Un nuovo metodo open source senza apprendimento automatico è stato creato da Shahbazi et al., nel 2018, chiamato *Flexible Learning-free Reconstrucion of Imaged Neural volumes (Flo-RIN)* e non richiede lo step di training. È diviso in tre fasi: segmentazione, identificazione e ricostruzione. Le ricostruzioni create da Florin possono essere utilizzate come punto di partenza per decidere come studiare i volumi neurali: gli esperti possono ottenere una vista ad alto livello e scegliere le regioni da studiare a risoluzioni più elevate. Pertanto, le ricostruzioni fatte con Florin non rappresentano un'analisi completa e richiedono un'ulteriore fase di elaborazione tramite metodi che richiedono tempo e esperti come menzionato precedentemente.

I dati messi a disposizione da Kasthuri et al. erano insufficienti e, in alcuni casi, incompleti per lo svolgimento del lavoro di tesi. Per questo motivo, è stato preso in considerazione un nuovo set di dati fornito dall'Allen Institute [2] . Insieme ad un consorzio di laboratori, essi hanno creato un programma innovativo chiamato *MICrON*, che mappa la funzione e il modo in cui i circuiti corticali si connettono. Per questo motivo, è stato preso in considerazione il database Cortical MM^3. Si estende su un 1.4mm x 0.87mm x 0.84mm volume di corteccia in una cavia P87 e le immagini sono state acquisite utilizzando microscopia a due fotoni, microCT e microscopia elettronica seriale, e poi ricostruiti utilizzando una combinazione di AI e proofreading. I dati

anatomici contengono più di 200.000 cellule e 120.000 neuroni. Il database è diviso in diversi dataframe per i diversi utilizzi, ma per questo progetto l'attenzione è stata focalizzata solo su due dataframe: "Cell Types", in cui è possibile trovare un sottoinsieme di nuclei in una colonna corticale 100μm classificato manualmente, così come l'intera parte di sezione in analisi; "Synaptic Connectivity" nel quale è possibile trovare molti aspetti dei neuroni, in particolare le sinapsi. Per elaborare i dati è stato utilizzato *Google Colab* che ospita un notebook Jupyter senza alcun requisito di installazione, evitando così la necessità di hardware GPU specifiche. Tutti i dati di MICrONS Binder sono raccolti in un server esterno chiamato minnie65. Per accedervi, era necessario registrarsi al framework CAVEclient. Come affermato in precedenza, due sono gli insiemi di dati considerati e sono stati utilizzati per creare un dataset finale con tutti i dati necessari per la ricostruzione della mappa di probabilità. In primo luogo, è stata considerata la tabella con i dati di una colonna corticale nel dataframe *"Cell-type"*. Da qui è stato possibile evidenziare lo stato attuale di ogni neurone (eccitatorio/inibitorio) e identificarlo con ID. Per recuperare la posizione di questi neuroni nell'intero volume (4 x 4 x 40)nm, è stata considerata una nuova tabella e la loro posizione è stata espressa nelle tre dimensioni *(x,y,z)*. L'ultimo passo è stata la concatenazione di questi due dataframe con la tabella contenente il rilevamento automatico delle sinapsi. I nuclei sono riconoscibili attraverso la loro posizione nella sinapsi (pre- o post-sinaptica), che ha permesso anche la distinzione nei vari layer.

Risultati e discussione

Solo dopo la creazione del dataset completo, era chiaro che tutti i dati di Cortical MM^3 comprendevano solo una parte dell'intera corteccia (in particolare, solo i layer L4, L5, L6). Per lavori futuri, sarebbe quindi necessario trovare un nuovo set di dati che includa informazioni su tutti i layer. Dopo averlo ottenuto, sarebbe possibile utilizzare i dati estratti dal set di dati scelto per generare un insieme di regole di connettività da utilizzare durante la riproduzione e la simulazione di modelli realistici di reti cerebrali (come il modello di colonna corticale realizzato in T.C.Potjans e M. Diessman [3]), migliorando la loro plausibilità biologica.

Capitolo 1

Introduction

The brain is definitely the most important organ of the nervous system and certainly the most complex organ of our body. The first studies on the brain belong to the philosophy of thought and date back to the ancient medicine of Egypt and Greece. But it is only from the second half of the twentieth century, when modern neuroscience was born, that brain research became an autonomous discipline focusing on the study of the nervous system and how this organ can really control any aspect of the body (e.g.,movement, sensations, emotions) and how its damages can compromise the abilities of a living being. Moreover, it was only at the end of the nineteenth century, after the invention of the microscope, that Ramón y Cajal showed that the nervous system was not formed by a continuous reticulum but by different neural cells (the neurons), using the method of silver staining developed by Golgi (published in 1885). [4] Despite the rapid advancement of neural interfacing technologies in the last century, brain studies have not progressed as fast. This chapter will also introduce the concept of computational neuroscience.

1.1 The nervous system

The nervous system is the major controlling, regulatory, and communicating system in the body. It is the center of all mental activity including thought, learning, and memory. Together with the endocrine system, the nervous system is responsible for regulating and maintaining homeostasis. Through its receptors, the nervous system keeps us in touch with our environment, both external and internal.
Like other systems in the body, the nervous system is composed of organs, principally the brain, spinal cord, nerves, and ganglia. These, in turn, consist of various tissues, including nerve, blood, and connective tissue. Together these carry out the complex activities of the nervous system.

1.1.1 Neuron

Neurons are the fundamental units of the brain and nervous system, the cells responsible for receiving sensory input from the external world, for sending motor commands to our muscles, and for transforming and relaying the electrical signals at every step in between. The neuron has a cell body, or *soma*, in which the nucleus and other cytoplasmic organelles reside (see Figure 1.1). These structures are fundamental for cellular metabolism and play an important role in the production of neurotransmitters useful for synaptic transmission. From the cellular body branch out prolongations, the *neutrites*, which in turn differ in dendrites and axons.

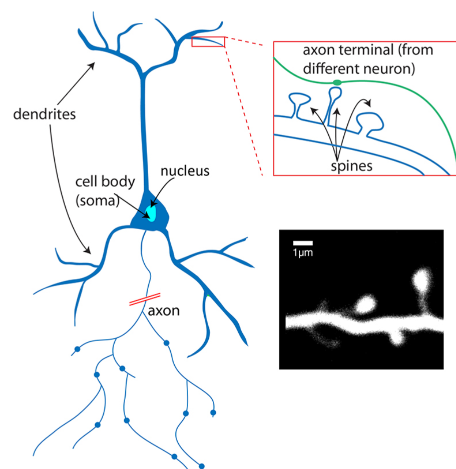


Figura 1.1: *Structure of the neuron*

Dendrites are branches of the soma that receive the nerve signal from afferent (or pre-synptic) neurons and transmit it to the cellular body. They are often covered with spread small protusions called dendritic spines that represent the post-synaptic sites of the cell.
The **axon**, instead, is an extension of the soma that starts from the body of the nerve cell. It has a smooth and regular cylindrical shape and in the final part you can find a branch to which the synaptic endings are attached. These ones generates synapses and transmit the nervous signal from one neuron to the next.

1.2 Computational Neuroscience

1.2.1 Brief Introduction

Computational neuroscience is a branch of neuroscience that studies, through mathematical models and theoretical analysis, the nervous system at different scales, from single cells to vast neural networks comprising the entirety of the body. Thomas Trappenberg defines it as:

> *"Computational neuroscience is the theoretical study of the brain used to uncover the principles and mechanisms that guide the development, organisation, information-processing and mental abilities of the nervous system."*
> [5]

Over the years, the desire to better understand how the brain works has led neuroscience researchers to develop this new field using mathematical, statistical and theoretical models that could best represent biology, physiology and in general the mechanisms of various areas of the brain. This approach, therefore, is different from other research fields that try to mimic intelligent behavior through schematic and abstract models, such as machine learning, artificial intelligence, etc.

1.2.2 Single neuron models

HH model

As stated earlier, neurons communicate between each other through action potentials, also called *spikes*. The first to explain what happens at microscopic level, i.e. the behaviour of a single neuron when an action potential occurs, were Hodgkin and Huxley in 1952 [6].

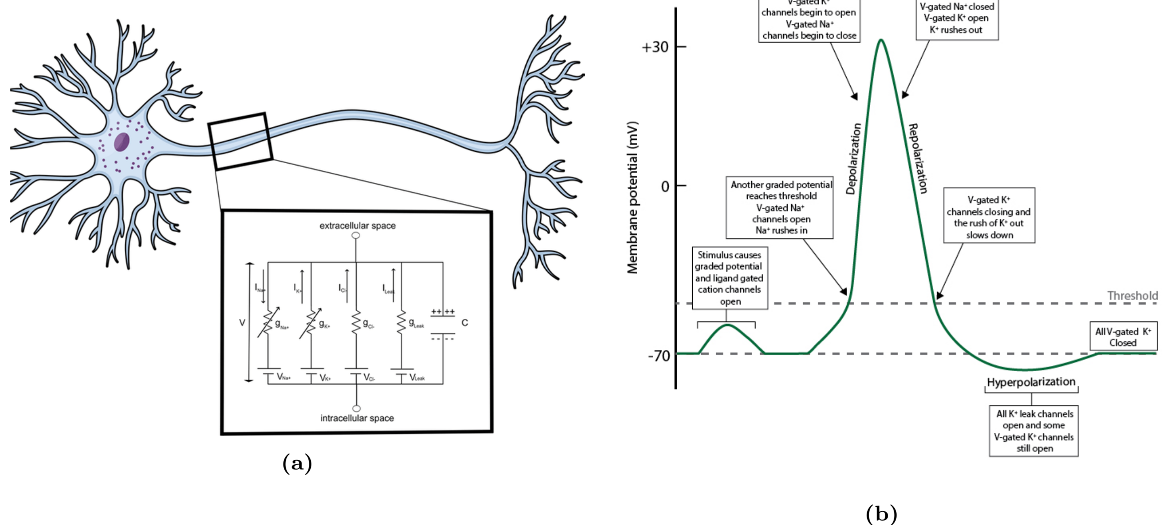


Figura 1.2: *(a) Representation of the H-H model circuit of the single neuron [7]. It consist of: a membrane capacitor; the conductances of potassium, sodium and chlorine. The membrane currents are mutually independent but depends on the membrane potential and the value of the membrane capacitor. (b) Action potential of a neural cell membrane. [created by BYU-I student, Kaylynn Loyd 2013].*

In the HH model the generation of an action potential is described according to the variation of the NA^+ and K^+ concentrations between the inside of the membrane and the extracellular space surrounding it. To do this, the model uses different parameters to evaluate the activation of the membrane ion channels: n - associated with potassium - m and h - associated with sodium. As shown in Figure 1.2b, the variation of m increases the conductance of Na^+ with consequent opening of the associated ion channels. This allows the sodium ions to enter the membrane, which depolarizes until a balance is reached between the inside of the membrane and the outside. At this point, the Na^+ channels close and, in the meantime, n grows, opening the potassium ion channels and allowing them to efflux. The reduction of the Na+ influence and

the increase of the K^+ outflow allow the membrane to repolarize, whose potential will return to rest values.

Compartmental model

Compartmental models provide a better description of the spike propagation in the axon, as they account for how the characteristics of the membrane change along the whole structure of the neuron. In this model, multiple cell compartments (i.e., spatial segments) are described and every single compartment has its conductance properties. Only two channels are considered in the HH model which, although they are the main factors for the generation of the action potential, are not the only ionic channels present. In these models, instead, they are considered and contribute to the modulation of the concentrations between the outside and the inside of the cell and in addition make the firing pattern dependent on previous events [8]. For a single compartment we will have: where, in each compartment, the membrane voltage is calculated as:

$$c_m \frac{dV(t)}{dt} = -g_{leak}[V(t) - V_{leak}] - g_{Na}(V,t)[V(t) - V_{Na}^{rev}] - g_K(V,t)[V(t) - V_K^{rev}] + I_{ext}$$

$$\frac{dV}{dt} = -\frac{1}{C_m} * \left\{ \sum [g_i * (V - V_i)] + i_{inj} \right\}$$

The HH model consists of a fourth order differential equation (the dimensions are V,h,m,n) which is, therefore, very complicated to solve and without analytical solution but it is possible to apply numerical methods to obtain a result in specific cases.

1.2.3 Synaptic Plasticity

The changing and the shaping of connections in our brain is known as *synaptic plasticity*, or rather, a change in the efficacy of synaptic transmission induced by a certain pattern of neuronal activity [9]. One of the first people to think about the plasticity in our brain was *Donald Hebb*[10]. When an action potential reaches the end of a pre-synaptic cell it causes the release of the neurotransmitters. These cross the synaptic cleft and bind to receptors on the post-synaptic cell. The degree of voltage change in the post-synaptic neuron and the way this voltage change is what determine the strengthening or weakening of the synapse. There is a critical window of time for synaptic plasticity with the peak time for changes to synaptic strength being 20 ms before and after an action potential.

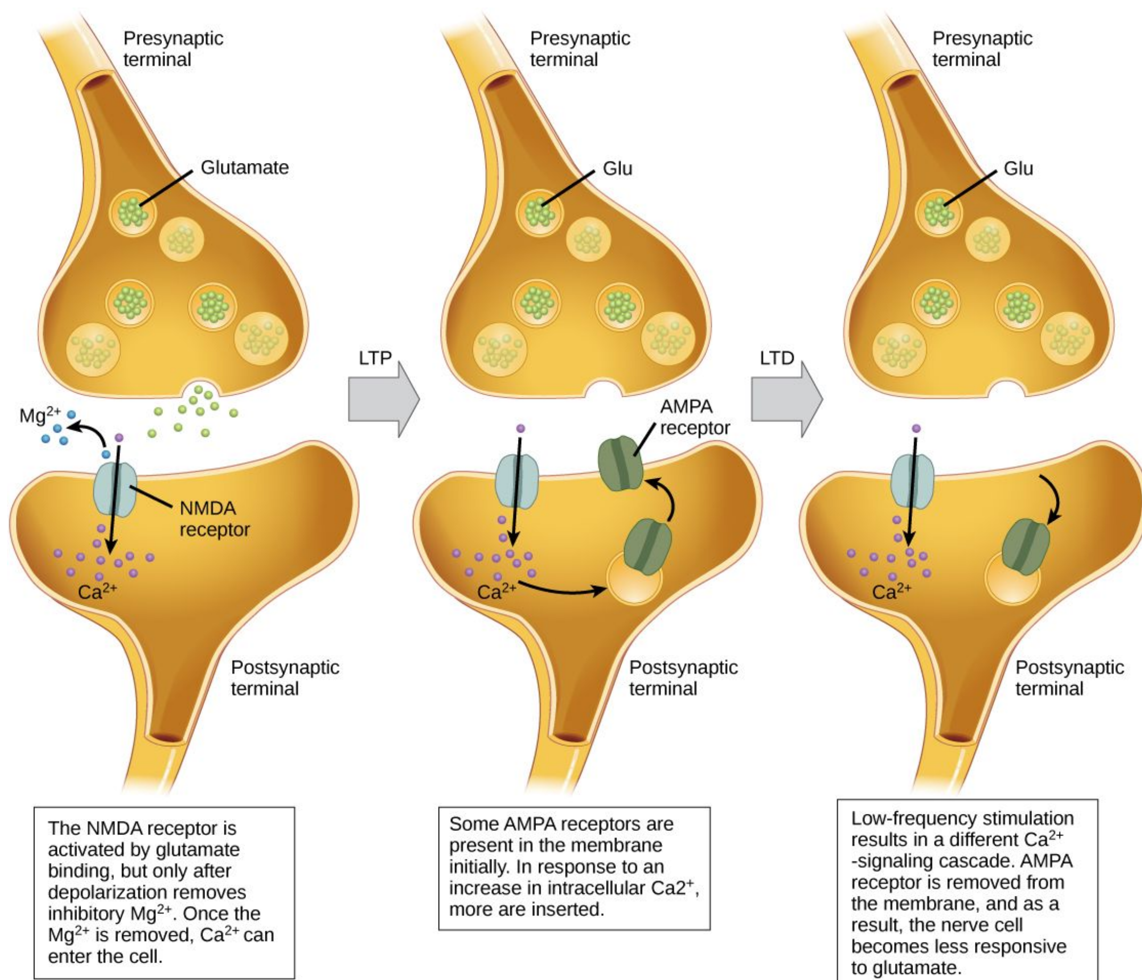


Figura 1.3: *Synaptic Plasticity. How does LTP and LTD occur [11]*

This is known as **Spike Timing-Dependent Plasticity** or **STDP**. The best way to understand the plasticity of our brain is to explain what happens in the hippocampus through the *in-vitro setup* which make possible to observe the modifications between pre- and post-synaptic neurons.

The glutamate (Glu) is fundamental for the synaptic plasticity as it is used as a neurotransmitter. There are two important Glu receptors: *AMPAR*, which are permeable to potassium and sodium and its inward flux which depolarizes the cell; *NMDAR*, "blocked" by magnesium and negative voltages and therefore do not significantly contribute to post-synaptic depolarization of the cell. However once the cell is depolarized, the magnesium is displaced and ions then flow through the NMDA receptors, allowing also calcium to flow through. If the pre-synaptic neuron fired first, it becomes depolarized and releases Glu. The glutamate binds to AMPA receptors at the post-synaptic neuron causing it to depolarize.As the cell becomes depolarized NMDA receptors become unblocked. The glutamate binds to them and this causes a large calcium influx. This process activates calcium dependent kinases which alterate the recycling of AMPAR and induce what is called *Long-Term Potentiation, LTP*. If the post-synaptic neuron fires first it becomes depolarized. As it is repolarizing, the pre-synaptic neuron fires and releases glutamate. When this reaches the post-synaptic cell, it finds it repolarizing and at a lower voltage, that means there are fewer NMDAR available to bind to. This leads to a more moderate calcium influx. This process activates protein phosphatases and induce what is called *Long-Term Depression, LTD.*

1.2.4 Spiking neurons models

In this section, multiple simplified neuron models will be introduced. Such models have been developed to allow a less computationally-intensive description of the firing properties of neurons, to be employed in large scale simulations of neural networks.

Integrate-and-fire

The most common model to describe spiking neurons is the integrate-and-fire. It's a phenomenological model, which uses simple electronic circuits to describe the mechanisms leading to the generation of an action potential. The model, shown in the Figure 1.4, is actually the **Leaky Integrate-and-Fire, (LIF)** and describes the development of the sub-threshold membrane potential depending on the firing instant, $t^{(k)}$, which is essential for the calculation of the action potential.

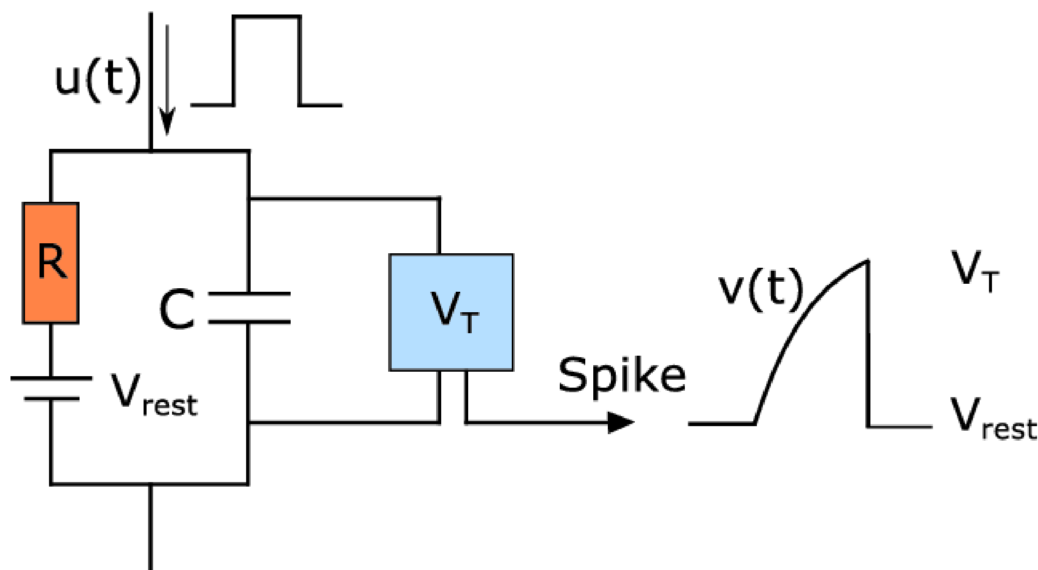


Figura 1.4: *Scheme of the Leaky Integrate-and-Fire (LIF) model. The current injected into the system changes the membrane potential $V_m(t)$. When this exceeds a predetermined firing threshold, then an action potential is generated and the circuit closes, then the membrane potential returns to the initial rest value [12]*

The equations describing the model are:

$$C * dV_m(t)/dt = 1/R_m * [V_m(t) - V_{rest}] + I_{ext}(t) \tag{1.1}$$

which describes the sub-threshold behaviour through a passive circuit; while the

$$\tau_m * \frac{d\nu(t)}{dt} = -\nu(t) + RI(t) \tag{1.2}$$

describes the effects on the membrane potential over time, where τ_m is the membrane time constant. This last equation allows to predict the evolution of the membrane potential knowing the value of the current over time, I(t) [13].

Izhikevic model

In 2003, a new model was proposed by Eugene Izhikevich, i.e. **Izhikevich model** [14], to reduce the complexity introduced by the HH model from four differential equations in the original model to two in the updated model, while retaining as much realism as possible.

Through this model it is possible to obtain different firing patterns, similarly to what happens biologically [15]. This model has been successfully used to represent cortical and thalamic neuron dynamics [8].

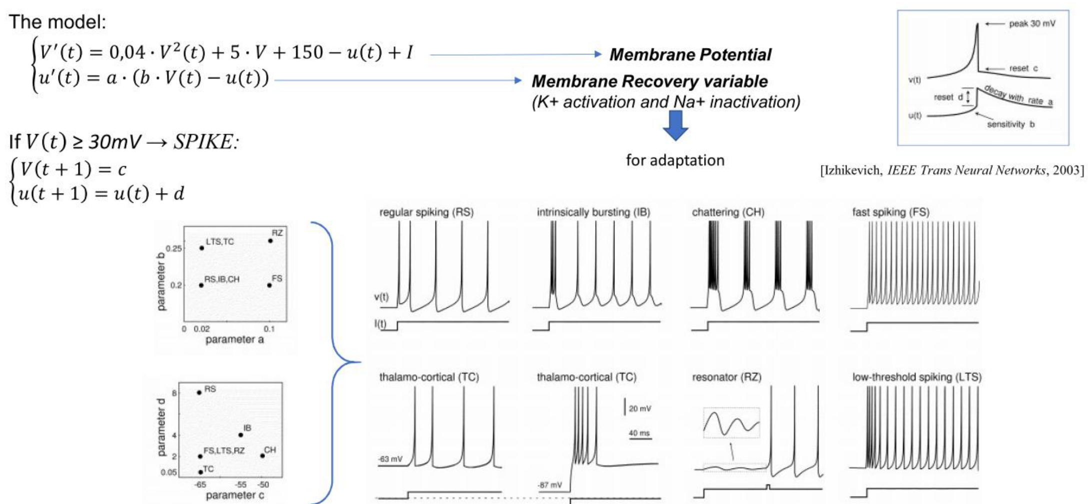


Figura 1.5: *Dynamic of Izhikevich model of the neuron [5, 8].*

Adaptive Exponential Leaky Integrate-and-Fire model (non linear)

This model consist of a system of two differential equations (Figure 1.6). There are two main variables : V_m, the membrane potential and an adaptive current coupled with V_m. Both modulates adaptation or bursting, depending on the value of the coupling constant.

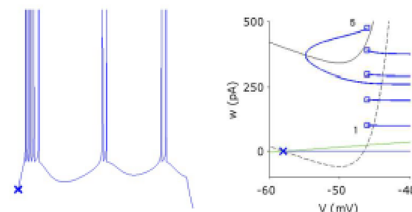


Figura 1.6: *Dynamic of Adaptive Exponential LIF model [8].*

The main properties are:

1. multiple electroresponsive properties based on parameter values;
2. replacement of the strict voltage threshold by a more realistic smooth spike initiation zone (given by the exponential term (*Brette and Gerstner, 2005*);
3. sub-threshold resonances or adaptation as in the Izhikevich model.

Generalized LIF (GLIF)

Since this model is a variation of LIF model, it is still composed by two differential equations but in this case there are three state variables: the membrane potential V(t), the spike-triggered current I(t) and a spike-triggered threshold V_{th}.
The equations, which describes this model, are:

$$\begin{cases} C_m * V'(t) = -g_L \cdot (V(t) - E_L) + \sum_j I_j(t) + I_e \\ I'(t) = -k_j \cdot I_j(t) \\ V'_{th} = a \cdot (V(t) - E_L) - b \cdot (V_{th}(t) - V_\infty) \end{cases} \tag{1.3}$$

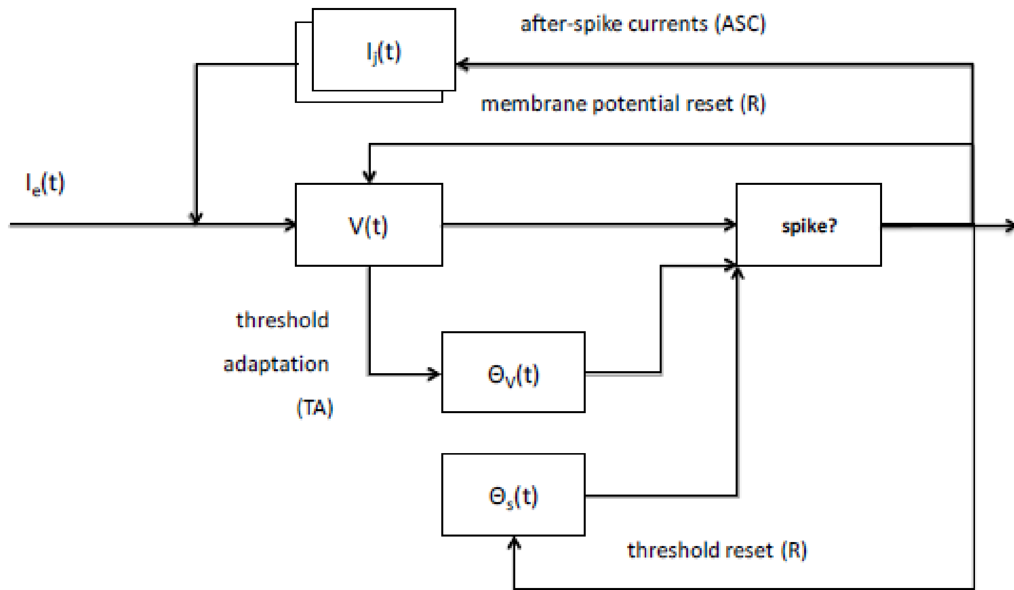


Figura 1.7: *Block diagram of Generalized LIF model [Mihalas and Niebur,Neural Comput.,2009].*

Spike take places when V(t)>V_{th} and then updated according to :

$$\begin{cases} V(t+1) = V_r \\ I_j(t+1) = R_j \cdot I_j(t) + A_j \\ V_{th}(t+1) = max(V_\infty, V_{th}(t)) \end{cases} \tag{1.4}$$

The membrane acts as a low-pass filter κ(t) on the input current I(t) to produce the modelled potential V(t). The exponential non-linearity (escape-rate) transforms this voltage into an instantaneous firing intensity λ(t), according to which spikes are generated. Each time a spike is emitted, both a current η(t) and a movement of the firing threshold γ(t) are triggered. GLIF models also include: spike-triggered currents I_j with different dynamics and reset rules depending on fast or slow subcellular mechanisms; spike-triggered firing threshold V_{th} with own dynamics; escape-rate non-linear rule for spike emission *(Mihalaş and Niebur, 2009; Pozzorini et al., 2015; Teeter et al., 2018)*. Since the coefficients of the differential equations form a triangular matrix, the set of equations can be sequentially solved. Internal currents are exponentially decaying and do not depend on other variables, the evolution of the membrane voltage depends on the internal currents, and the evolution of the threshold depends on the membrane

voltage. The GLIF model accurately predicts the occurrence of individual spikes with millisecond precision. In order to prove this, the response of a L5 pyramidal neuron to a fluctuating input current was recorded intracellularly, repeating the protocol nine times to assess the reliability of the response. The GLIF model was able to accurately predict both the subthreshold and the spiking response of the cell, therefore it has been proven capable to reproduce cortical neurons spiking patterns with high accuracy (*for more details, see Pozzorini et al., 2015 [16]*).

Extended-generalized LIF neuron model (E-GLIF)

This new model, proposed by *Geminiani et al.* in 2018 [17], is a compromise between model complexity, biological plausibility and computational efficiency. It consists of three state variables: the membrane potential and two intrinsic currents (I_{adap} and I_{dep}), associable to lots of biological mechanisms. Like the GLIF model, this one can be optimized without metaheuristic methods but just by using optimization algorithms [16, 18]. The model is defined as follows:

$$\begin{cases} \frac{dV_m}{dt} = \frac{1}{C_m} \cdot (\frac{C_m}{\tau_m} \cdot (V_m(t) - E_L) I_{adapt}(t) + I_{dep}(t) + I_e + I_{stim}) \\ \frac{dI_{adap}(t)}{dt} = k_{adapt} \cdot (V_m(t) - E_L) - k_2 \cdot I_{adapt}(t) \\ \frac{dI_{dep}(t)}{dt} = -k_1 \cdot I_{dep}(t) \end{cases} \tag{1.5}$$

where :

- I_{stim} = external stimulation current;
- C_m = membrane capacitance;
- τ_m = membrane time constant;
- E_L = resting potential;
- I_e = endogenous current
- k_{adap},k_2 = adaptation constants;
- k_1 = I_{dep} decay rate.

Spike is generated when : $\lambda(t) = \lambda_0 e^{\frac{Vm(t)-V_{th}}{\tau V}}$ where V_{th} is the threshold potential, while λ_0,τV are the escape rate parameters. The update is given by:

$$\begin{cases} V_m(t^+_{spk}) = V_r) \\ I_{adap}(t^+_{spk}) = I_{adap}(t_{spk}) + A_2 \\ I_{dep}(t^+_{spk}) = A_1 \end{cases} \tag{1.6}$$

where: t^+_{spk} is the time immediately following the spike time t_{spk}; V_r is the reset potential; A_1,A_2 are the model currents update constants.

1.2.5 Neural networks models

At the macroscopic level the objective is therefore to represent the behaviour of a series of neurons that create a vast neural network capable of learning the various tasks that allow everyone to make movements, perceive sensations, recognize images, sounds, tastes, etc. Since the neuronal circuit, and so also its functioning, is very complicated, over time scientists have tried to reproduce the behaviour of neurons through mathematical and statistical models. In particular, *artificial neural networks (ANNs)* have been used, which have undergone improvements over time to better represent, at a physiological level, the learning process in the brain. The first generation of ANNs was proposed in 1943 by McCulloch and Pitts [19], and consists of a series of binary threshold units, also called MCP neurons. These will be able to generate an action potential if the sum of the input signal weights exceeds the threshold value. Depending on the value assumed by the weights, the individual neurons can have an excitatory (positive value), inhibitory (negative value) or ineffective (null value) contribution which means that the input signal has no effect on the neuron. But this model, in reality, does not really explain the behaviour of neurons during the transmission of information. For this reason, in the second generation, neuron activation functions have been introduced, which give a different output value depending on the type of function and whether the action potential exceeds, or not, the threshold. Depending on the type of activation, we can find linear (for example Heaviside) and non-linear (for example logsig and

tanh) functions. With this new generation also layers have been introduced, i.e. the model architecture is modified by rearranging the neurons in several layers, where each layer processes the output of the neurons of the previous layer. This type of model is used in Feed-Forward Neural Network (FFNN), because the output is calculated in one direction from input to output, or the Recurrent Neural Network (RNN). Biologically speaking, neurons, after generating a spike, enter a period of "*refractoriness*". This consists of a time interval (about 10ms) and can be of two types: absolute or relative. In the absolute refractoriness period the neuron is in a resting state in which it will not be able to generate any spike, regardless of the amplitude of the next incoming stimulus. The period of relative refractoriness, instead, follows the previous one and during this time it is possible to excite the cell again, as long as the second incoming stimulus is wider than the first one. The first two generations of neural networks do not take these periods into account. They encode the output of the neuron by means of a binary output - 0 or 1 - which allows, however, to have a good representation of the real functioning of neurons but not in a complete way, since the calculation of the output signal is done through iterations and not continuously over time [20].

Spiking Neural Networks, SNNs have been created, in the field of computational neuroscience, as an attempt to reproduce the "communication methods" of neurons, i.e. biological behaviour and biochemical variations within neurons when there is an action potential. In particular, SNNs are a variation of Artificial Neural Networks (ANNs), where the neuron is represented as a computational unit that receives multiple inputs giving a single binary output signal. SNNs therefore represents the third generation of ANNs and, compared to previous generations, represent in a quasi-realistic way what happens during the transmission of information. The substantial difference lies in the use of a series of impulses instead of individual firing frequencies in certain time intervals, as happens with the two previous generations. This allows SNNs to encode spatio-temporal information at the level of communication and computation, just as neurons do in reality. The advantages of using this model are mainly: the transmission of information through very weak signals, since the rate encoding is very robust to noise; validation of the models from experimental data and unsupervised learning that works with the same learning rules as neuronal cells.

Capitolo 2

State of the Art

The brain is an extremely complex organ and although research on its functioning began more than a century ago, there is still some doubt about how some mechanisms and processes work, especially in certain situations or diseases. Cortical circuits are part of this category. In fact, it is not yet clear how many classes of cortical neurons exist. Neuronal classification remains a challenging topic since it's unclear how to designate a neuronal cell class and what are the best features to define them by (DeFelipe et al., 2013).

2.1 Machine Learning for neuron classification

Numerous studies have been carried out on the different methods which can be used to classify neurons efficiently and qualitatively according to their different characteristics: e.g. morphological, physiological, molecular. The work carried out by Vasques et al.[21] aims to create a list of all known and discovered methods up to now, starting from the extraction of morphological characteristics up to the neuronal classification, analyzing and comparing the accuracy of different machine learning methods one by one. **Data extraction and preprocessing** The data were acquired from NeuroMorpho.org, classified into distinct layers and/or m-types of the somatosensory cortex in rats. A tool, called L-Measure, was used to extract forty-three morphological features for each neuron. L-Measure (LM) is a freely available software tool for the quantitative

characterization of neuronal morphology. LM computes a large number of neuroanatomical parameters from 3D digital reconstruction files starting from and combining a set of core metrics [22].
Before data could be used for the various methods of classification of neurons, a preprocessing step was necessary to eliminate the possible missing values. These were presented in the form of white spaces, NaNs or other placeholders. These values were then replaced by the mean value of the processed features for a given class. The algorithms were implemented using different normalization methods, depending on the algorithm used. **Supervised Learning** Different algorithms were used and compared, where the training set represents the measurements on the features of neurons along with the name of the type of neuron associated with them indicating the class of observation.

- Naive Bayes;
- k-Nearest Neighbors;
- Radius Nearest Neighbors;
- Nearest Centroid Classifier (NCC);
- Linear Discriminant Analysis (LDA);
- Support Vector Machines (SVM);
- Stochastic Gradient Descent (SGD);
- Decision Tree (DT);
- Random forests classifier;
- Extremely randomized trees;
- Neural Networks (NN);
- Classification and Regression Tree;
- CHi-squared Automatic Interaction Detector (CHAID);

- Exhaustive CHAID;
- C5.0.

The algorithms follow some steps:

1. Normalize each of the neuron feature values;
2. Instantiate the estimator;
3. Fit the model according to the given training data and parameters;
4. Assign to all neurons the class determined by it's exemplar;
5. Compute the classification accuracy.

Unsupervised Learning The unsupervised learning is quite different from the supervised one. The training set has no class label is not defined because unknown, thus the goal is to establish the existence of classes/clusters with data provided. Again, different algorithms were evaluated :

- K-Means;
- Mini Batch K-Means;
- The K-Means algorithm used on PCA-reduced data;
- Ward (with and without connectivity constraints);
- Mean Shift.

The step followed by the algorithm were :

1. Normalize each of the neuron feature values using the l2 norm [-1,1];
2. Instantiate the estimator;
3. Fit the model according to the given training data and parameters;
4. Assign to all neurons the corresponding cluster number;

5. Algorithm Assessment.

In this case, the authors also used the *Affinity propagation algorithm* [23] which creates clusters by sending messages between pairs of samples until convergence.
From this paper it is clear that there is still a lot of work to do in terms of neuron classification. An improvement in this field can be made by increasing the number of morphological, electrophysiological and molecular data, in order for researchers to develop better classifiers. In this sense, neuroinformatics has an important role by working on how to acquire multi.modal data and develop new methods and tools to enhance data analysis. One example of scientific research project is the Human Brain Project (HBP). Among all of the others objectives, HBP aim to make easier for scientists to organize and access data, e.g. neuron morphologies.

2.2 3-D volume reconstruction

As stated earlier, the brain and his functioning is still not fully discovered. In particular, the cellular organisation of the mammalian brain is more complicated than that of any other known biological tissue [24].
In their research, Kasthuri at al., tried to analyze and describe automated technologies to probe the structure of neural tissue, in order to create a 3-D reconstruction of a sub-volume of mouse neocortex in which are visible also cellular object and many sub-cellular components. The authors understood that, for the reconstruction of synaptic connectivity, they should analyze as many connections as possible by observing not only the synapses created by each axon, but also when axons and dendrites approach and do not establish connections.They acquired high-resolution images (3nm/pixel), making sure that the boundaries of the membrane were easy to reconstruct and then, using the same samples, the imaged the sections at lower resolutions in order to acquire images of larger tissue volumes. So, after the acquisition of the images with EM technology, using a particular stain called ROTO, they created a multi-scale digital volume in such a way as to make easier the recognition of the tissues surrounding the area of interest to be reconstructed. To do this, they first have imaged all the sections at a low resolution (2 um/pixels). Then they have imaged a sub-volume (a radial strip of

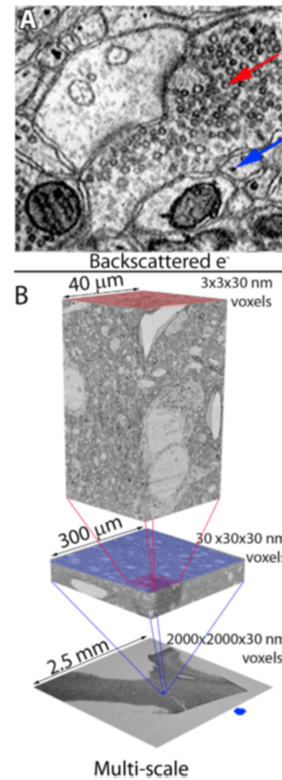


Figura 2.1: *EM Imaging brain sections.*

(a) *(A) A section of somatosensory neocortex imaged in a scanning EM. The red arrow shows synaptic vescicles. The blue one, a strongly labeled membranous tube found in unmyelinated axons.*
(B) The strategy for placing high-resolution images in a larger anatomical context [24].

celebral cortex from the pia to the white matter, 500 um wide and 1 mm long. 29 um/pixel). Finally they have imaged a box, a voxel 40x40x50 nm^3 that transected the apical dendritic bundle of a cortical mini-column at high resolution (3 nm/pixel). As stated also previously, there are currently no fully automated image segmentation methods. For the construction of a "semi-automated" method there is a need for a manual segmentation of the images that will be used in the training phase of the model. For this reason they developed a computer-assisted manual space-filling segmentation and annotation program called **VAST**. This program allows the user:

- to work with EM images online, overcoming the problem of saving large amounts of data in local memory;
- to "color" the images at multiple scales of resolution;
- to organize the results in a flexible annotation framework;
- to export results for 3-D visualization and analysis.

RhoANA It's an image processing tool developed to automatically segment large volumes of EM data. It was introduced for the first time by Kaynig et al. in 2015, and was born from the need to segment and classify, through a model of machine learning, all the cellular components (axons, dendrites and glia) and sub-cellular (synapses, synaptic vesicles, spines, spine apparati, post-synaptic densities and mitochondria) in volumes of brain tissue [24]. In 2018 a new updated version was created, called RhoanaNet, by Knowles-Barley S. et al.,[25]. The new RhoanaNet pipeline consists of five main stages (Figure 2.3):

1. *Membrane classification*, where cell membranes are identified in the 2-D images producing a membrane probability map;

2. *2-D segmentation*, based on the membrane probability for each image. These are then grouped across sections in order to create 3-D neuron reconstructions;

3. *3-D segmentation*. For this, 3-D blocks are cropped from the full volume and each block is processed individually;

4. *Block matching*, where blocks are matched pairwise with neighboring blocks, and overlapping objects are joined;

5. *Global remap*, matched objects are joined globally to produce a consistent segmentation for the full volume.

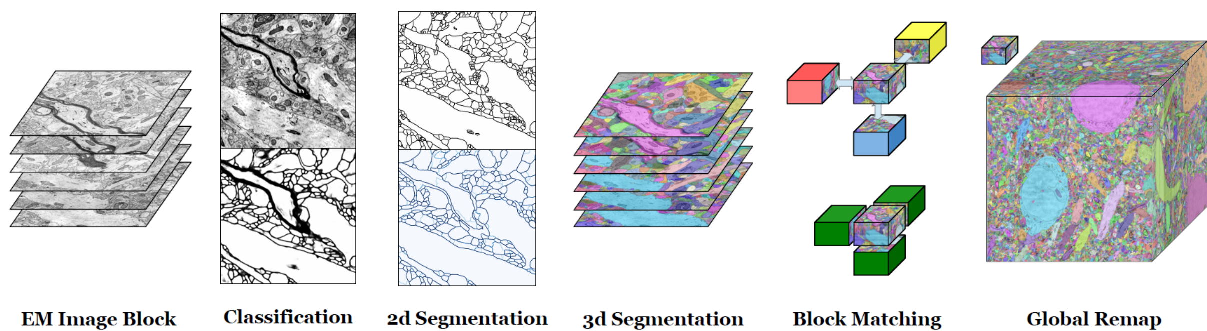


Figura 2.3: *The RhoANA segmentation pipeline [25].*

The authors found that, despite the rapid growth of fully automated methods, there is still a need to do several steps, both before and after using the tool. First, a manual segmentation of the images acquired through EM as it could be met with errors of

misclassification or tracing the contours of neurons, which will then affect the entire work. And then a proofreading step to make sure that the work done by the model really corresponds to what is in the volume under examination (neurons, synaptic connections, etc.). This last step can be done through a program, also available online, called *Dojo*, which allows you to better edit and render the final images. Currently the general development of new image segmentation and reconstruction techniques for Connectomics is hindered by the lack of available benchmark data sets. For this reason, different programs has been evaluated and tested to analyze the images from this paper.

2.3 UNI-EM

As previously introduced, advanced connectomics laboratories have developed their own software pipelines to employ CNN-based segmentation, including *RhoANA*, *Eyewire*, and the *FFN* segmentation pipeline. The main objective of these is large-scale 3-D reconstructions that are conducted by teams including computer experts for setup and maintenance. UNI-EM is a UNIfied Environment for CNN-based automated segmentation of EM images for researchers with limited programming skills. It implements several 2-D and 3-D CNNs on the widely used Tensorflow framework, as well as the proofreading software Dojo and a series of 2-D/3-D filters for classic image processing [1]. To test the program, the authors did few example workflows on the mouse somatosensory cortex. One of these, is focused on neuron segmentation using a particular type of CNN, called **Flood-Filling Networks** (FFNs).These type of NNs show high accuracy in neuron segmentation but require a pre-processing step for the 3-D ground truth, which is possible to create using *Dojo* or *VAST*. The program follows five steps:

1. *Pre-processing.* Stacks of target EM images and ground truth images are converted into FFN-specialized style files ($\sim$ 1h computation time);

2. *Training.* FFNs are trained with the pre-processed EM-image files ($\sim$ 2 weeks computation time);

3. *Inference.* The trained FFNs are applied to a stack of test EM images for the inference of 3D segmentation ($\sim$ 1 h computation time);

4. *Post-processing.* The output segmentation files are converted into a PNG file stack (∼ 10 min computation time);

5. *Proofreading and visualization.* The converted PNG files and EM images are imported into Dojo for proofreading as well as the 3-D annotator for visualization.

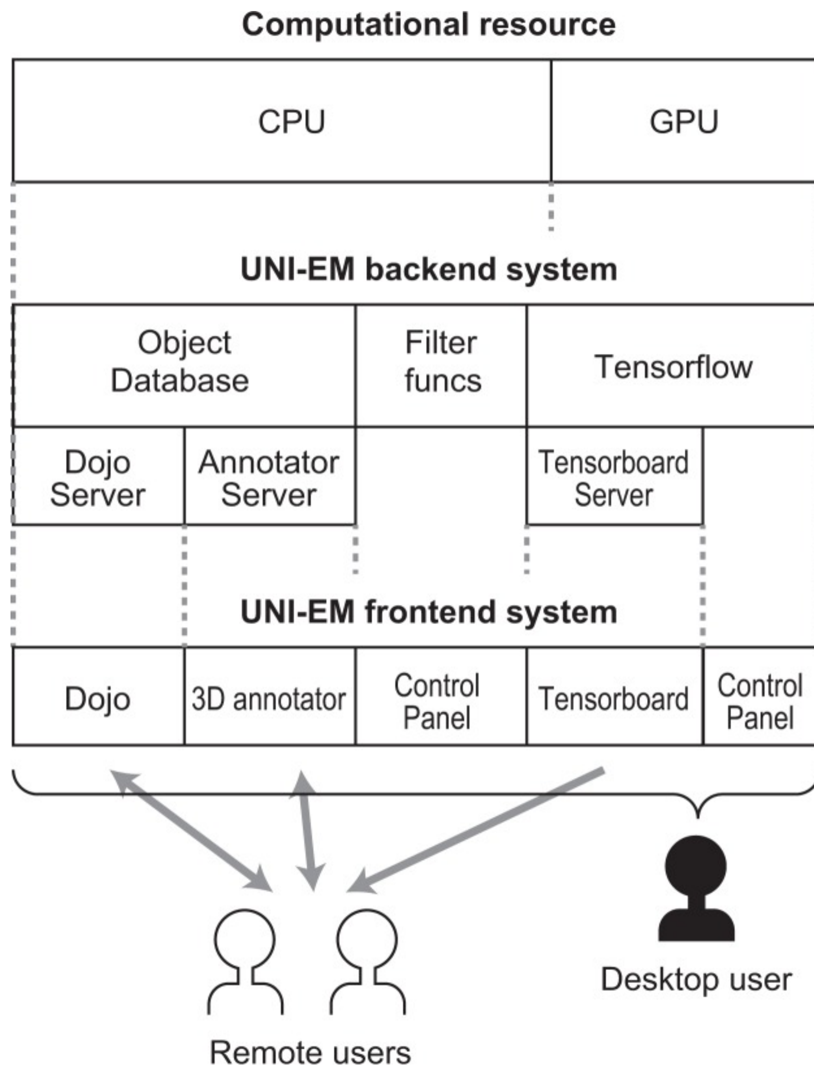


Figura 2.4: *Architecture of UNI-EM [1]*

2.4 Flood-Filling Networks

Flood-Filling Networks are a class of neural networks designed for instance segmentation of complex and large shapes, particularly in volume EM datasets of brain tissue, in which a recurrent 3-D convolutional network directly produces individual segments from a raw image. As can has been seen in previous research, finding an efficient neural network is critical to achieving accurate segmentation results. In this sense, the incorporation of supervised learning into boundary detection has been crucial. From a machine learning point of view, however, a drawback of this pipelined approach is the fundamental disconnect between the learned part of the pipeline (pixel-wise boundary prediction) from the subsequent steps (connected components, watershed, etc) that produce the actual segments. For this reason, Januszewski et al. introduced this new segmentation approach. The primary contributions of their work are [26]:

- A convolutional network architecture that introduces the notion of an "object mask channel" to both specify the target object and provide an explicit memory of segmentation state across recurrent iterations of the network;

- A recurrent procedure for iterating the network inference dynamics over multiple overlapping fields of view in order to segment arbitrarily large objects that are initialized from a single seed pixel.

- An experimental evaluation on a challenging connectomic dataset, demonstrating how a single flood-filling network can outperform a highly optimized and state-of-the-art pipeline consisting of three successive algorithmic steps: boundary detection by 3d convolutional neural networks, affinity graph watershed segmentation, and pairwise object agglomeration.

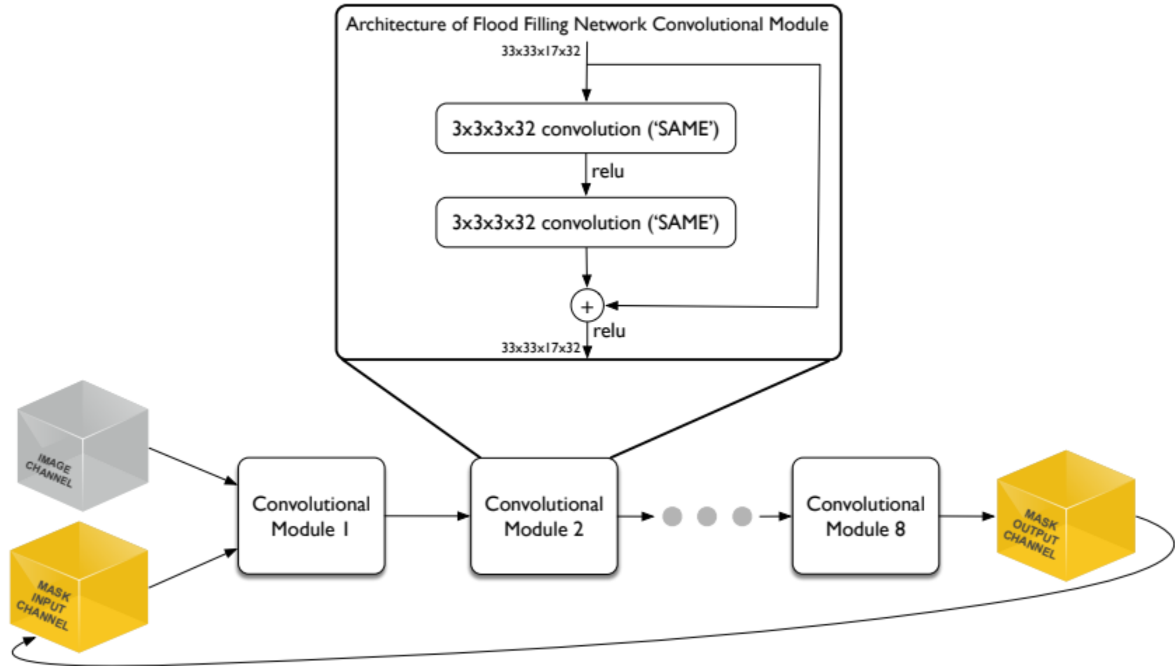
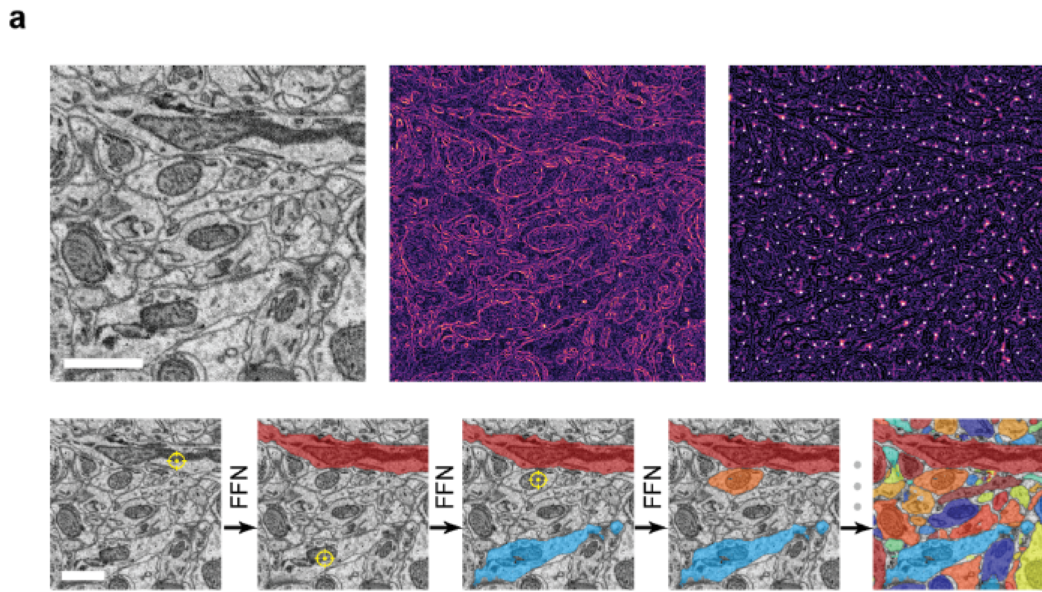
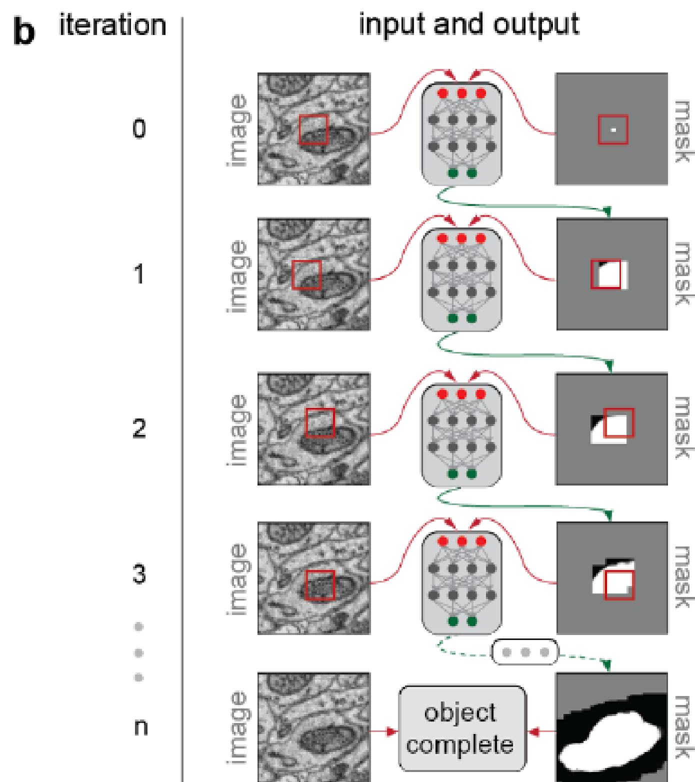


Figura 2.5: *Structure of the FFNs [26].*

Architecture. Given a 3-D subvolume of data as input, the FFN produces an object mask probability map, which is going to be the same size as the input volume. As can be seen in Figure 2.5, the network has actually two channels as input: one providing the raw image intensities (grayscale map); the second one provides the local state of the object mask in the form of a probability map. The network itself consists of a deep stack of 3-D convolutional modules with Rectified Linear Unit (ReLU) non-linearities. Since the dimension of the input and the output are the same, all convolutions use the SAME mode. The SAME mode for a convolutional layer with the kernel size k_x x k_y $\times$ k_z is implemented by padding the input with $[k_x/2]$, $[k_y/2]$, $[k_z/2]$ zeros on each side, respectively in the x, y, and z dimensions. The input mask in the second channel is incomplete in most inference iterations of the network, and the FFN is trained to extend it within its field of view (FoV). The spatial size of the network FoV is set to 33 $\times$ 33 $\times$ 17 voxels, corresponding roughly to a cube in physical space due to the anisotropy of the underlying image data. All convolutions use 3$\times$3$\times$3 filters. Inside, the convolutional module, is composed of two convolutional layers, with skip connections between them.



(a) *Segmentation of a subvolume with an FFN.*



(b) *The flood-filling inference process for a single object.*

Figura 2.6: *Flood-Filling Network's process.*

Workflow. First step is training. FFNs are trained to extend an input object mask into the full extent of the object within the network's field of view. A special case is the first iteration of the network, in which a single pixel is encoded in the center of the input object mask to specify the particular object the network should be extending. A training example consists of an object fragment contained within a $49 \times 49 \times 25$ subvolume, with the object of interest overlapping the center voxel of the subvolume. The objective of the network is to predict the voxel mask of the object, with a value of 0.95 indicating that a given voxel belongs to the object of interest, and a value of 0.05 that it does not. The values 0.05 and 0.95 are chosen as soft-target equivalents of 0 and 1, with the 0.05 softness parameter ensuring that the internal activations of the network are not pushed towards $\pm\infty$. In the first step of training, the FoV of the network is set to the center of the $49{\times}49{\times}25$ subvolume of the current training example, and the center voxel of the input object mask channel is set to the target value of 0.95. The remaining voxels are set to 0.05. The network is then run in forward mode, resulting in a partial predicted mask covering the current FoV. This mask is saved within a "canvas" covering the whole training subvolume and the FoV of the network is moved to a new location. Every FFN has an associated $\vec{\Delta}$ vector determining the step size to take in every direction when moving the FoV, as well as a threshold value t_{move} which needs to be reached or exceeded in the current object mask at the voxel that is to become the new center of the FoV. For every such plane the object mask voxel with the highest value v is found, and if $v \geq t_{move}$ the location of that voxel is added to a list of new positions for the FFN. Once all 6 planes are examined, the list is sorted in descending order of activation of the corresponding mask voxel and appended to a queue of new positions. Inference proceeds by pulling new positions from the queue and moving the FFN to them. After every such move, the search for new locations relative to the current position of the network is repeated. Inference ends when the queue is empty. To prevent the network from getting stuck in a loop, a set of already visited locations V is maintained.

2.5 FLoRIN

Machine learning algorithms show some constraints due to the lack of data or the best type of neural network to choose that detects and identifies microstructures as a semi-automated method. In their paper, Shahbazi et al., in 2018, introduced a new metod called **Flexible Learning-free Reconstrucion of Imaged Neural volumes** (FLoRIN), which exploits structure-specific contextual clues and requires no training [27]. FLoRIN is an open-source software package (https://github.com/CVRL/florin-scirep), which is divided into three stages, each of which use a series of learning-free image processing methods in either 2D or 3D:

Stage 1. ***Segmentation.***

With a set of raw images of a neural volume as an input, the program will, optionally, adjust the grayscale histogram of the images and then threshold the result. Thresholding is carried out using the N-Dimensional Neighborhood Thresholding (NDNT) algorithm, an extension of the method described by Bradley and Roth [28], which is able to focus in on small regions, making it robust to grayscale shifts and distant noise that would otherwise confound global thresholding methods. The result of the Segmentation stage is a binarized version of the raw images that labels potential microstructures of interest.

Stage 2. ***Identification.***

Starting from the binary output of the previous stage, the software performs morphological operations before grouping connected components. In this stage it is also possible to remove all the noise and artifacts that comes within the segmentation step. The remaining connected components are then found and filtered into groups based on their geometric and grayscale properties. Information about the connected components are saved to a database for further review.

Stage 3. *Reconstruction.*

After the identification, FLoRIN saves each class of microstructures into an individual volume. These ones can be output in a number of standard file types for compatibility with various post-processing and rendering software. Along with the 3D reconstruction, FLoRIN outputs a Statistical Report that contains spatial, morphological, and grayscale information about each segmented microstructure.
Reconstructions created by FLoRIN can be used as a starting point for deciding how to study neural volumes: experts can gain a high-level view and choose regions to study at higher resolutions. It can be used to statistically study reconstructed volumes, including cell count, distribution of cell sizes, etc. Learning-free methods are able to function across modalities because they are unbiased with respect to specific inputs, unlike supervised methods which are inherently biased by the training data.

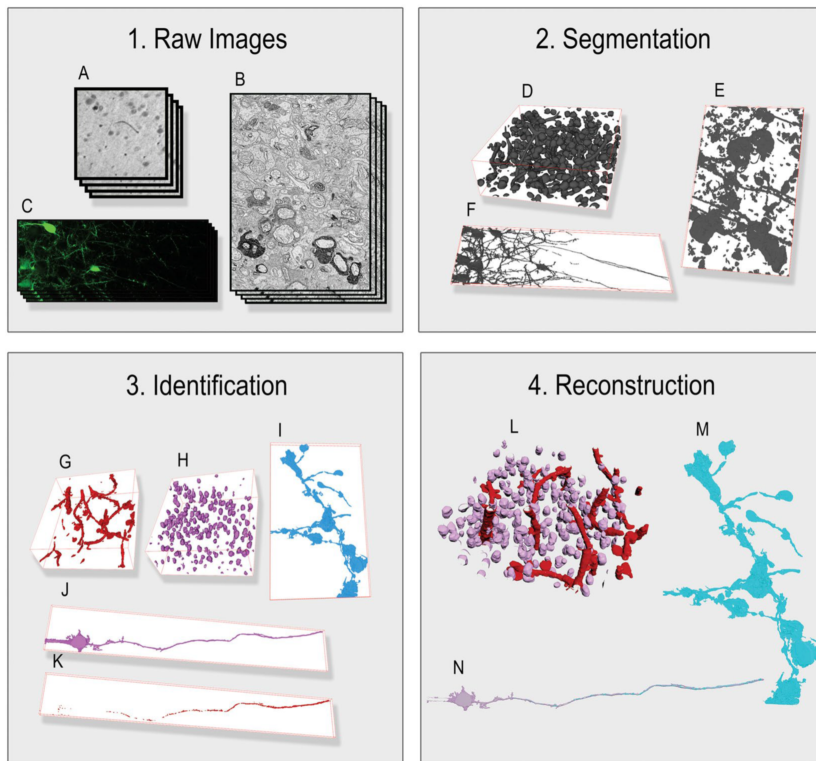


Figura 2.7: *FLoRIN Framework [27]*

2.6 Cell-type structure and connectivity map

The local cortical network is considered to be one of the main pillars of brain functioning. Cortical function depends on the type of synaptic connectivity that neurons establish, as well as the synaptic function of each of them. It is not yet known how these connections occur during cellular development. There are several factors and mechanisms that, during neurogenesis, differentiate neurons. These include the morphogenesis of pre- and post-synaptic processes and activity dependent mechanisms and intercellular molecular determinants such as transmembrane and secreted molecules, many of which have also been implicated in neurodevelopmental disorders [29].

Experimentally, methodological advances in the last decade provided new comprehensive data sets on cell-type specific connectivity structure and activity. At the same time, modelling has been established as a tool to relate network structure to activity dynamics. In this sense, connectivity maps have been created, using the Peters' rule. Peters originally proposed that the synaptic connectivity between cell types reflects the average spatial overlap between the presynaptic axon of one cell type and the different postsynaptic targets in a volume of cortex. If Peters' rule explains the connectivity patterns of the neocortex, mechanisms governing the development of each cell type's characteristic axonal and dendritic morphologies, including their vertical and horizontal distribution and their density, would establish the predictable patterns of intracortical synaptic connectivity among cortical cell types by regulating the average axodendritic overlap for different cell-type combinations.

In his work, Potjans et al. proposed a new way to employ the connectivity maps. It uses a full-scale spiking network modeling to relate the connectivity structure of the cortical microcircuit to its activity. As can be seen in Figure 2.8, the network model presents four out of five layers of cortex (from L2/3 to L6), each consisting of two populations of excitatory and inhibitory neurons, where the "connection probability" represents the probability that a neuron in the pre-synaptic population forms at least one synapse with a neuron in the post-synaptic population.

For the anatomical map, it has been considered data based on reconstructed neurons and a modified version of Peters' rule based on the layer-specific distributions of den-

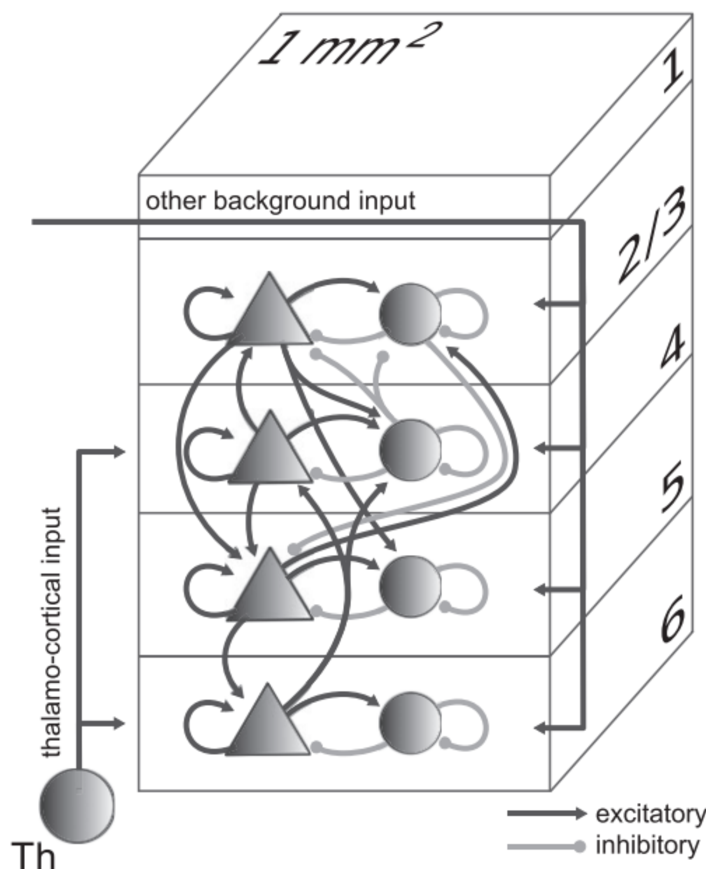


Figura 2.8: *Potjans-Diesmann model definition [3]*

drites. These provides the relative number of synapses participating in a connection as well as the total number of synapses. the product of these measures gives the absolute number of synapses K for any connection. The corresponding connection probabilities C_a are calculated base on the assumption that synapses are randomly distributed, allowing multiple contacts between any two neurons:

$$C_a = 1 - (1 - \frac{1}{N^{pre}N^{post}})^K \tag{2.1}$$

but it is possible to use the first-order Taylor series approximation:

$$C_a = \frac{K}{N^{pre}N^{post}} \tag{2.2}$$

The derived connection probabilities are inversely proportional to the considered surface area πr^2. This can be easily understood watching the eq.(2.2), where the number

of neurons N and synapses K increase almost linearly with the surface area. The physiological hit rate estimates provide the physiological map (p). Therefore, it is possible to combine multiple independently measured hit rates for the same connection by a weighted sum:

$$C_p = \frac{\sum_i R_i Q_i}{\sum_j Q_j} \tag{2.3}$$

where R_i and Q_i are the hit rate and the number of tested pairs in the i-th experiment, respectively, with probabilities of the L2/3i to L5e and of the L4i to L2/3i connections set to 0.2. In their experiment, the authors classified all connections into 2 main groups: Recurrent intralayer connections and connections between different layers.

Capitolo 3

Materials and Methods

In the last decade there has been a rapid expansion in the field of micro-connectomics, which aims to do a 3-D reconstruction of neural networks from stacks of 2-D EM images. Neuroscientists have already successfully reconstructed large-scale neural circuits from different species, such as mice and fruit flies [1]. For this, convolutional neural networks (CNNs) have been largely used because enable the automated image segmentation. Several laboratories and teams developed their own software pipelines. However, due to the amount of data collected and the histological knowledge required to do the pre-processing, the complexity of these software makes their use difficult even for computer experts. Currently, the aim is to find a way to make the 3-D reconstruction fully automated without the need of pre-processing the EM images to create the ground truth data and also to create affordable software for all, even the most inexperienced.

3.1 Experimental Data

Initially, the research carried out by Kasthuri et al. was taken into account. As mentioned in the previously in the section 2.2, the authors have tried to create an innovative method to investigate the structure of neural tissue at the nanometer level through semi-automated technologies. They investigate aspects of the connectivity of excitatory axons and found interesting patterns that wouldn't have been visible with lower resolutions methods. The images, their segmentation and annotated databases

linked to the images, as well as the software created, have been made available online (it is possible to find the description here: http://openconnecto.me/Kasthurietal2014/). However, even if the data were available, it was found that they were not complete and therefore inadequate for the project. For this reason, research was carried out for additional, more comprehensive databases that were available as an open source.

3.1.1 MICrONS

The Machine Intelligence from Cortical Networks (MICrONS) it's an ambitious program to map the function and connectivity of cortical circuits, using high throughput imaging technologies, with the goal of providing insights into the computational principles that underlie cortical function in order to advance the next generation of machine learning algorithms. The data, collected from the visual cortex of a mice, are available on the website (https://www.microns-explorer.org). This website serves as a data portal to release connectivity and functional imaging data collected by a consortium of laboratories led by groups at the Allen Institute for Brain Science, Princeton University, and Baylor College of Medicine, with support from a broad array of teams, coordinated and funded by the IARPA MICrONS program. These data include large scale electron microscopy based reconstructions of cortical circuitry from mouse visual cortex, with corresponding functional imaging data from those same neurons [2].
The *Cortical* MM^3 database was taken into account. It spans a $1.4mmx.87mmx.84mm$ volume of cortex in a P87 mouse and it was imaged using two-photon microscopy, microCT, and serial electron microscopy, and then reconstructed using a combination of AI and human proofreading. With an estimation of 75.000 pyramidal neurons, the volume contains also the single cell responses to a variety of visual stimuli. Then, the same portion of the brain was imaged at high resolution using EM and processed with a machine learning reconstruction pipeline to generate a large scale anatomical connectome. The anatomical data contains more than an estimated 200,000 cells, and 120,000 neurons. Automated synapse detection measured more than 523 million synapses. The database is divided into different dataframes for different usage.
In this project, I focused onto two specific sets of data:

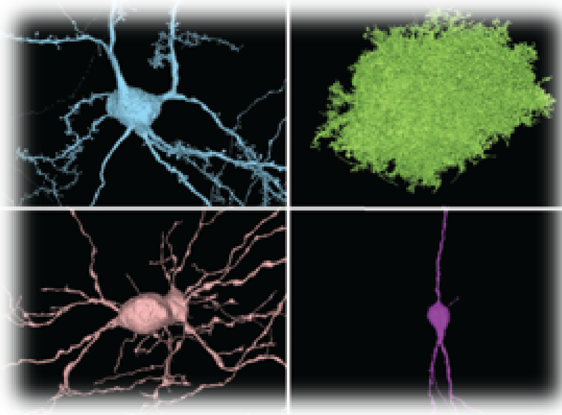


Figura 3.1: *Cell Types dataframe [2]*

- **Cell Types**. In this dataframe it possible to explore the morphology and connectivity of excitatory , inhibitory and non-neuronal cell types in the volume. It consists of a subset of nucleus (n=2204) detected in a 100 μm column, which were manually classified to distinguish non-neuronal, excitatory and inhibitory cells. The nucleus segmentation detected 171.818 connected components of nucleus objects with 82.000 neurons for a total of 99.6% neurons with a 96.9% of precision [2];

- **Synaptic Connectivity**. With this dataframe it is possible to query many aspects of the data as annotation tables, including synapses. At the moment only larger proofread volume synapses are usable.

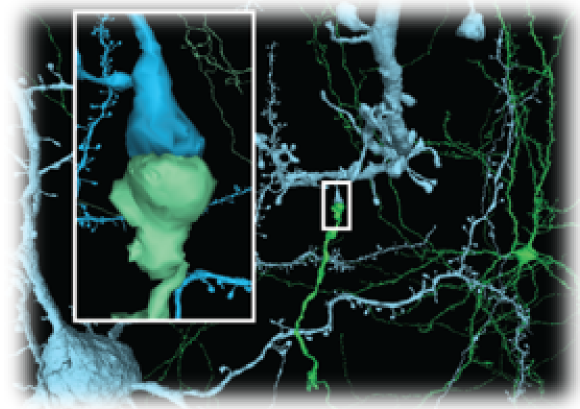


Figura 3.2: *Synaptic Connectivity dataframe [2]*

For both of these dataframes it was necessary to register as a user in the *CAVE-client* infrastructure (the repository can be found here: https://github.com/seung-lab/CAVEclient).

3.2 Data Processing

To process the data **Google Colab** has been used. It allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs [30]. After the registration to the CAVEclient framework, it was necessary to extract all the data from each MICrONS dataframe:

1. Started by querying the table that contains the data of a cortical column in *Cell Types*. Once this was done, a new table was created (Table 3.3);

ID	cell type
864691134945836023	excitatory
864691135294515893	excitatory
864691135974582895	inhibitory
864691135644554735	excitatory
864691136883828334	inhibitory
...	...

Tabella 3.1: *Example of the data extracted from the cortical column*

2. A new table containing the data of all the neurons in the segmented subset created using a trained multi-layer perceptron classifier to distinguish the type of neuron together with the location in the (4 x 4 x 40) nm voxels;

ID	cell type	position
864691136740606812	excitatory	[282608, 103808, 20318]
864691135366988025	excitatory	[110208, 153664, 23546]
864691135181741826	inhibitory	[166512, 174176, 24523]
864691135337690598	excitatory	[275616, 135120, 24873]
864691136883828334	inhibitory	[216064, 166800, 15025]
...	...	...

Tabella 3.2: *Example of the data extracted from the subset*

3. The next step was the concatenation of these two dataframes with the table containing the automated synapse detection performed by Nick Turner from the Seung Lab [31]. Here the nucleus were distinguished according to their location (pre- or post-synaptic cleft) including the corresponding layer.

pre-synaptic ID	**post-synaptic ID**	**pre-syn. type**	**post-syn. type**	**pre-syn. position**	**post-syn. position**	**layer (1-6)**
864691134945836023	864691134947465724	E	I	[282724, 261022, 22820]	[282732, 260974, 22834]	4
864691135294515893	864691135952035875	E	E	[286810, 238068, 21802]	[286892, 237968, 21804]	6
864691135974582895	864691135974582895	I	E	[298378, 243810, 25925]	[298388, 243778, 25927]	5
864691135644554735	864691136536564642	E	I	[309408, 244972, 19151]	[309456, 245044, 19136]	4
864691136883828334	864691136908297582	I	I	[275832, 250790, 22228]	[275894, 250834, 22218]	6
...						

Tabella 3.3: *Example of the final database*

Capitolo 4

Discussion

In the first phase of the thesis, we explored the state of the art related to the processes for the 3-D reconstruction of a volume of a mouse somatosensory cortex. The objective of this thesis is to design a method for the analysis of data and the reconstruction of a probability map of connection of neurons, which is available to all and at the same time simple to use. To do this, several methods already present in the field of scientific research have been examined, but they have shown some problems, such as the need to carry out a pre-processing step to manually segment images and create ground-truth. This, in particular, is an obstacle for all those who have no knowledge of histology, thus failing to distinguish the various cellular components present in EM images. Many of the studies conducted in this field, In fact, stated that nowadays it is not yet possible to create a fully automated method of machine learning just because it hasn't been created yet a neural network able to segment and annotate images accurately; The margin of error is still too high. Although manual segmentation is rough and not entirely accurate, it can be improved by using third-party programs, such as VAST or Dojo. This type of programs are very useful because they allow you to distinguish the various cellular components by coloring them, but especially because they are available online, thus avoiding the problem of local storage and the use of GPUs which are not appropriate for this type of analysis.

Initially, using the images analyzed by Kasthuri et al. and available online, several tests were carried out with the various programs, of which we discussed in the chapter 2. Unfortunately, the available hardware used in this project was not adequate for the type of analysis that these programs have to perform. In some cases, as also pointed out by the authors of RhoANA, image analysis and model traning require months of work (24/7), if not more, to be completed. For this reason I tried to modify the workflow to be followed in such a way as to "lighten" the computational aspect of the method. In addition, in some cases, such as that of FFNs, there were numerous problems related to the user interface and the saving of data in H5 format. It was therefore decided to tack on the data provided by the Allen Institute that were already complete with that part of manual segmentation required for the training of machine learning models, and in addition have numerous dataframes for the verification of the steps taken (e.g. proofreading/annotation). The only missing part, which was then carried out in this thesis, is the position of the neurons taken into analysis in the layers of belonging. Only after doing this, it was noticed that the data in the Cortical MM^3 database do not include the neurons of all six layers but only the last three, as a previous work had already been done on the first ones by the same consortium of laboratories.

As a future development, we recommend that you capture or use data that is complete and includes annotation and proofreading databases to verify that the connectivity map is actually correct.

Capitolo 5

Greetings

First of all, I would like to thank all the people who made this project possible: Professor Alessandra Laura Giulia Pedrocchi who strongly believed in this project, and the PhD candidates Francesco Jamal Sheiban, Alessandra Maria Trapani and Benedetta Gambosi who helped me during the whole period of the thesis. Thanks to all the other graduating students of the laboratory with who I shared ideas and knowledge every friday in the last year.

A heartfelt thanks to our parents who have always supported and encouraged me, not only during this thesis project but throughout my university career and in life. Finally a special hug to all friends and colleagues who have supported and bore with me all these years.

Acronyms

CAVEclient Connectome Annotation Versioning Engine client

CNN Convolutional Neural Network

EM Electron Microscopic images

FFNs Flood-Filling Networks

FLoRIN Flexible Learning-free Reconstruction of Imaged Neural volumes

H-H Hodgkin-Huxley

LIF Leaky Integrate-and-Fire

LM L-Measure

NDNT N-Dimensional Neighborhood Thresholding

NN Neural Network

ReLU Rectified Linear Unit

UNI-EM UNIfied Environment

Elenco delle figure

1.1 Structure of the neuron . . . 2
1.2 (a) Representation of the H-H model circuit of the single neuron [7]. It consist of: a membrane capacitor; the conductances of potassium, sodium and chlorine. The membrane currents are mutually independent but depends on the membrane potential and the value of the membrane capacitor. (b) Action potential of a neural cell membrane. [*created by BYU-I student, Kaylynn Loyd 2013*]. . . . 4
1.3 Synaptic Plasticity. How does LTP and LTD occur [11] . . . 6
1.4 Scheme of the Leaky Integrate-and-Fire (LIF) model. The current injected into the system changes the membrane potential V_m(t). When this exceeds a predetermined firing threshold, then an action potential is generated and the circuit closes, then the membrane potential returns to the initial rest value [12] . . . 8
1.5 Dynamic of Izhikevich model of the neuron [5, 8]. . . . 9
1.6 Dynamic of Adaptive Exponential LIF model [8]. . . . 10
1.7 Block diagram of Generalized LIF model [Mihalas and Niebur,*Neural Comput.*,2009]. . 11

2.1 EM Imaging brain sections. . . . 19
2.3 The RhoANA segmentation pipeline [25]. . . . 20
2.4 Architecture of UNI-EM [1] . . . 22
2.5 Structure of the FFNs [26]. . . . 24
2.6 Flood-Filling Network's process. . . . 25
2.7 FLoRIN Framework [27] . . . 28
2.8 Potjans-Diesmann model definition [3] . . . 30

3.1 Cell Types dataframe [2] . . . 35
3.2 Synaptic Connectivity dataframe [2] . . . 35

Elenco delle tabelle

3.1 Example of the data extracted from the cortical column 36
3.2 Example of the data extracted from the subset 36
3.3 Example of the final database 37

Bibliografia

[1] H. Urakubo, T. Bullmann, Y. Kubota, S. Oba, and S. Ishii, “Uni-em: an environment for deep neural network-based automated segmentation of neuronal electron microscopic images,” *Scientific reports*, vol. 9, no. 1, pp. 1–9, 2019.

[2] M. C. et al., “Functional connectomics spanning multiple areas of mouse visual cortex.” https://www.microns-explorer.org. bioRxiv 2021.07.28.454025; doi: https://doi.org/10.1101/2021.07.28.454025.

[3] T. C. Potjans and M. Diesmann, “The cell-type specific cortical microcircuit: relating structure and activity in a full-scale spiking network model,” *Cerebral cortex*, vol. 24, no. 3, pp. 785–806, 2014.

[4] E. Kandel, “Essentials of neural science and behavior,” *Appleton&Lange Norwalk, CT*, vol. 6, no. 23, pp. 425–451, 1995.

[5] T. Trappenberg, *Fundamentals of computational neuroscience*. OUP Oxford, 2009.

[6] A. L. Hodgkin and A. F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *The Journal of physiology*, vol. 117, no. 4, p. 500, 1952.

[7] J. Isakovic, I. Dobbs-Dixon, D. Chaudhury, and D. Mitrecić, “Modeling of inhomogeneous electromagnetic fields in the nervous system: a novel paradigm in understanding cell interactions, disease etiology and therapy,” *Scientific Reports*, vol. 8, 12 2018.

[8] A. Pedrocchi, “Computational neuroscience - modelling and simulation.” [PPT slides]. Retrived from:, 2019-2020.

[9] T. Tabata and M. Kano, *Synaptic Plasticity in the Cerebellum*, pp. 63–86. 01 2009.

[10] D. O. Hebb, *The organization of behavior: A neuropsychological theory.* Psychology Press, 2005.

[11] O. O. CNX, “How neurons communicate,” 2014.

[12] A. Nandi, H. Schättler, J. Ritt, and S. Ching, “Fundamental limits of forced asynchronous spiking with integrate and fire dynamics,” *The Journal of Mathematical Neuroscience*, vol. 7, 12 2017.

[13] L. M. P. Ravazzini, *Principi di Bioelettricità e Bioelettromagnetismo.* Pàtron editore, 2011.

[14] E. M. Izhikevich, “Simple model of spiking neurons,” *IEEE Transactions on neural networks*, vol. 14, no. 6, pp. 1569–1572, 2003.

[15] S. Donachy, “Spiking neural networks: neuron models, plasticity, and graph applications,” 2015.

[16] C. Pozzorini, S. Mensi, O. Hagens, R. Naud, C. Koch, and W. Gerstner, “Automated high-throughput characterization of single neurons by means of simplified spiking models,” *PLoS Comput Biol*, vol. 11, no. 6, p. e1004275, 2015.

[17] A. Geminiani, C. Casellato, F. Locatelli, F. Prestori, A. Pedrocchi, and E. D’Angelo, “Complex dynamics in simplified neuronal models: Reproducing golgi cell electroresponsiveness,” *Frontiers in Neuroinformatics*, vol. 12, p. 88, 12 2018.

[18] C. Teeter, R. Iyer, V. Menon, N. Gouwens, D. Feng, J. Berg, A. Szafer, N. Cain, H. Zeng, M. Hawrylycz, *et al.*, “Generalized leaky integrate-and-fire models classify multiple neuron types,” *Nature communications*, vol. 9, no. 1, pp. 1–15, 2018.

[19] W. Maass, “Networks of spiking neurons: the third generation of neural network models,” *Neural networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[20] J. Vreeken, “Spiking neural networks, an introduction,” 2003.

[21] X. Vasques, L. Vanel, G. Villette, and L. Cif, “Morphological neuron classification using machine learning,” *Frontiers in neuroanatomy*, vol. 10, p. 102, 2016.

[22] R. Scorcioni, S. Polavaram, and G. A. Ascoli, “L-measure: a web-accessible tool for the analysis, comparison and search of digital reconstructions of neuronal morphologies,” *Nature protocols*, vol. 3, no. 5, pp. 866–876, 2008.

[23] D. Dueck and B. J. Frey, “Non-metric affinity propagation for unsupervised image categorization,” in *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, IEEE, 2007.

[24] N. Kasthuri, K. J. Hayworth, D. R. Berger, R. L. Schalek, J. A. Conchello, S. Knowles-Barley, D. Lee, A. Vázquez-Reina, V. Kaynig, T. R. Jones, *et al.*, “Saturated reconstruction of a volume of neocortex,” *Cell*, vol. 162, no. 3, pp. 648–661, 2015.

[25] S. Knowles-Barley, V. Kaynig, T. Jones, A. Wilson, J. Morgan, D. Lee, D. Berger, N. Kasthuri, J. Lichtman, and H. Pfister, “Rhoananet pipeline: Dense automatic neural annotation,” 11 2016.

[26] M. Januszewski, J. Maitin-Shepard, P. Li, J. Kornfeld, W. Denk, and V. Jain, “Flood-filling networks,” *arXiv preprint arXiv:1611.00421*, 2016.

[27] E. Shahbazi, J. Kinnison, R. Vescovi, M. Du, R. Hill, M. Joesch, M. Takeno, H. Zeng, N. Costa, J. Grutzendler, N. Kasthuri, and W. Scheirer, “Flexible learning-free segmentation and reconstruction of neural volumes,” *Scientific Reports*, vol. 8, 09 2018.

[28] D. Bradley and G. Roth, “Adaptive thresholding using the integral image,” *J. Graphics Tools*, vol. 12, pp. 13–21, 01 2007.

[29] A. Y. Gutman-Wei and S. P. Brown, “Mechanisms underlying target selectivity for cell types and subcellular domains in developing neocortical circuits,” *Frontiers in neural circuits*, p. 98, 2021.

[30] “Google colaboratory (google colab).” https://research.google.com/colaboratory/faq.html.

[31] “Seung lab - computational neuroscience@princeton.” https://seunglab.org.