THE UNIVERSITY OF CHICAGO

POSTNATAL DEVELOPMENT OF SYNAPSES IN MOUSE PRIMARY VISUAL CORTEX REVEALED BY A CONNECTOMICS TOOLBOX

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE BIOLOGICAL SCIENCES
AND THE PRITZKER SCHOOL OF MEDICINE
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

COMMITTEE ON COMPUTATIONAL NEUROSCIENCE

BY
HANYU LI

CHICAGO, ILLINOIS
DECEMBER 2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I want to thank my advisor Dr. Narayanan 'Bobby' Kasthuri for his guidance and support over the last few years. Without his passion and enthusiasm, this thesis would not have been possible. I'd like to thank my thesis committee Jason MacLean, John Maunsell, Nicolas Brunel, and Viren Jain, for their great suggestions and insights along the way. I would like to thank my colleagues at Kasthuri lab and close collaborators: Gregg Wildenberg, Vandana Sampathkumar, Anastasia Sorokina, Griffin Badalamente, Dawn Paukner, Haotian Gan, Shuichi Shigeno, Joe Austin, and Wenbiao Gan for their help with my projects on various aspects. I would like to thank my collaborators at Argonne Leadership Computing Facility Tom Uram, Nicola Ferrier, Venkatram Vishwanath, William Scullin, and many others who introduced me to HPC and coached me in programming on supercomputers. I would then like to thank my mentors in connectomics at Google team: Peter H. Li, Michal Januszewski, and Viren Jain, for hosting me on an unforgettable internship. They inspired me to think big and taught me how to address scientific problems with engineering at a massive scale. I would like to thank my parents for their love and support over the years. They ignited my interest in science at an early age, which eventually prompted me on this quest. Finally, I would like to thank my wife Can Dong, for her love and companion over this incredible journey, and for the joy and adventures we shared together. This thesis would not have been possible without her by my side.

ABSTRACT

Understanding how a premature circuit adapts to environmental stimuli and matures could greatly improve our understanding of learning rules in a neural system. The development of synapses in the mammalian visual cortex during the critical period plays a significant role in the proper functioning of the visual system. Despite extensive studies with electrophysiology and light microscopy, there has not been a detailed investigation with connectomics.

The rapid development of electron microscopy and computing technology in the last few years gave rise to ever-increasing data size and made previously costly connectomics reconstruction feasible. However, tremendous computational challenges remain in the field. Better computing resource accessibility and a higher level of automation performance are required for the field to expand into mainstream neuroscience. Also, skepticism in connectomics remains on whether/what biological insights it can offer at the current scale(100 μm cubes). This thesis discusses three projects on different yet interconnected aspects of connectomics, addressing these concerns.

The first project describes a software pipeline for end-to-end EM connectomics reconstruction, integrating multiple advanced automation tools into High-performance computing(HPC) environment, utilizing the previously untapped power of supercomputers at national labs for connectomics for the first time.

The second project discusses a novel computational framework for neuronal subcompartment morphology classification and how it can perform neurite type classification at a massive scale with state-of-the-art accuracy. It is also demonstrated to detect and correct merge errors from segmentation, addressing a significant bottleneck in improving EM reconstruction accuracy.

A comparative study is conducted in the third project to reconstruct mouse primary visual cortex layer 4 samples at age P14, the beginning of the critical period, and P105, in adulthood. Automatic saturated segmentation and synapse prediction are run with HPC,

and focused proofreading of dendritic synapses is conducted. It is found that in the adult visual cortex, synapses increase substantially in both density and size, challenging a pruning centric hypothesis of circuit development. Also, a sharp increase of perisomatic shaft synapse density is observed, supporting the significance of modulatory inhibitory inputs, and the lack thereof, in determining the maturity of neurons in the visual cortex. Also, a significant increase in mitochondria coverage and its correlation with synapse density is reported, suggesting a vital role of mitochondria in the maturation of circuits.

CHAPTER 1

INTRODUCTION

"How does intelligence work?" is a fundamental question that has been pondered by some of the brightest minds of our civilization throughout human history. Despite a mechanistic understanding of intelligence, human beings have been exceptionally adept at using it. From our humble origins as hunter-gatherers threatened by predators to the society today capable of harnessing nuclear energy and sending probes to Mars, the brain, the substrate upon which humankind achieved all the marvels and wonders, remains essentially a mystery.

The primary technical limitation has been our ability to observe delicate structures of neural tissue. Much like how the telescope revolutionized astronomy and physics, the invention of the microscope was perhaps the most monumental technical advancement for our understanding of the brain. Modern neuroscience started when Ramón y Cajal performed Golgi staining in neural tissue and, for the first time, delineated individual neuronal morphology (Cajal 1888), with the compound optics of a microscope, demonstrating clear evidence for neuron doctrine, which claims that the brain is made up of separate cells that interact with each other.

1.1 The emergence of Connectomics with Automated Serial Electron Microscopy

Since the time of Cajal, neuroscience breakthroughs have been frequently associated with advancements in new techniques and technologies. Aside from the mechanistic understanding of neural systems through electrophysiology (Neher and Sakmann 1976) like the Hodgkin Huxley model (Hodgkin and Huxley 1952). There is an insatiable demand for higher resolution neural data that captures more neurons at the same time.

Tremendous progress has been made in light microscopy technology over the last few

decades. Milestones include confocal microscopy (Minsky 1988), fluorescent labeling (Chalfie et al. 1994), calcium imaging (Grynkiewicz, Poenie, and Tsien 1985; Grienberger and Konnerth 2012), two-photon microscopy (Helmchen and Denk 2005), and super-resolution imaging (Betzig et al. 2006), which led to a plethora of scientific discoveries and were lauded with a series of Nobel prizes(Chemistry 2008, Chemistry 2014). However, the resolution of optical microscopy remains physically limited by the wavelength of visible light at the theoretical upper limit of 200 nm. Super-resolution microscopy could breach the limit by using another active laser source like a flashlight for fluorescent molecules (Ke et al. 2016); another novel technique, expansion microscopy (F. Chen, Tillberg, and Boyden 2015), could enlarge the sample into optical range without disrupting cellular morphology. However, even with these advancements, it remains a huge challenge to acquire large-scale neural imagery at the synapse level with light microscopy. Part of the problem with optical microscopy is that even with nanoscale resolution, there remains the problem of staining/labeling. Most optical studies sample less than 1% of all neurons and rarely label organelles, etc.

On the other hand, electron microscopy has the unique advantage of easily reaching a nanometer-level resolution at the cost of not being able to image in-vivo. Compared to light microscopy, EM is considerably slower in acquisition speed, as electron beams need to be focused and sequentially traverse a field of view. Early applications of EM (Palade 1954) focused on studying intra-cellular organelles within snapshot samples, often in a limited number of slices. Over time, studies with manually collected series of thin slices were conducted. A principled advantage of EM-based approaches is that osmium, the heavy metal used as the principal contrast agent in most EM approaches, stains the membranes and organelles of every cell in every species so far tested (Palay et al. 1962).

The concept of "connectomics" originated from the idea that by mapping the entire connectivity matrix of a nervous system, a holistic view of the network topology can be obtained, from which fundamental principles of neural wiring can be deduced (Sporns, Tononi, and

Kötter 2005). The connectivity can be on different scales. For example, at macro-scale, the "human connectome project" (Van Essen et al. 2013) studies the correlation and connectivity between brain regions with fMRI. At mesoscale, multiple sparse labeling techniques like anterograde or retrograde tracing, the "Brainbow" (Livet et al. 2007), tissue clearing (Kim, Chung, and Deisseroth 2013; Shen et al. 2020), offer connectivity between subsets of neurons or track long-distance projections. However, achieving a micro-scale connectome requires the ability to find every synapse and separate individual neurons, which can still only be achieved with electron microscopy. The first widely recognized micro-connectome was from C.elegans in the 80s. (White et al. 1986), consisting of 302 neurons and all their connections. This pioneering work took tremendous manual effort over more than a decade with extremely limited digitization. The computers back then were not powerful enough for any sophisticated image processing. Therefore the connectome was essentially painted by humans slice by slice. The daunting technical challenge and prohibitive time cost drove neuroscientists toward more realistic and efficient techniques. Over the next two decades, light microscopy soared past EM and became the mainstream in neural imaging.

A major technological breakthrough that brought EM back into the mainstream was the automation of data collection, which drastically reduced the amount of human labor required in section slicing and picking, along with continued research over EM technology, the scale at which EM can be used for scientific inquiry is dramatically increased. (Briggman and Denk 2006; Briggman and Bock 2012)

Two paradigms were developed around the same time. The first one is "block-face" EM, where the sample is ablated with either a diamond knife or a focused ion beam and is interleaved with EM scanning (Denk, Horstmann, and Harris 2004; Heymann et al. 2006). The other approach is "serial sectioning" EM, where the sample is first cut and collected with an automatic device called "Automatic Tape Collecting Ultra-Microtome" (ATUM) (Hayworth et al. 2006; Kasthuri et al. 2007) on tape, which is further cut into segments and mounted

on silicon wafers. The wafers are then scanned with a scanning electron microscope(SEM). A slightly different approach uses Transmission Electron Microscope (Harris et al. 2006; Yin et al. 2020), which is incompatible with wafers and tapes since electron beams cannot penetrate them. Instead, sections are manually collected and mounted on copper grids. The compromise in collection speed is compensated by the superior data collection speed and much lower cost of TEM over SEM.

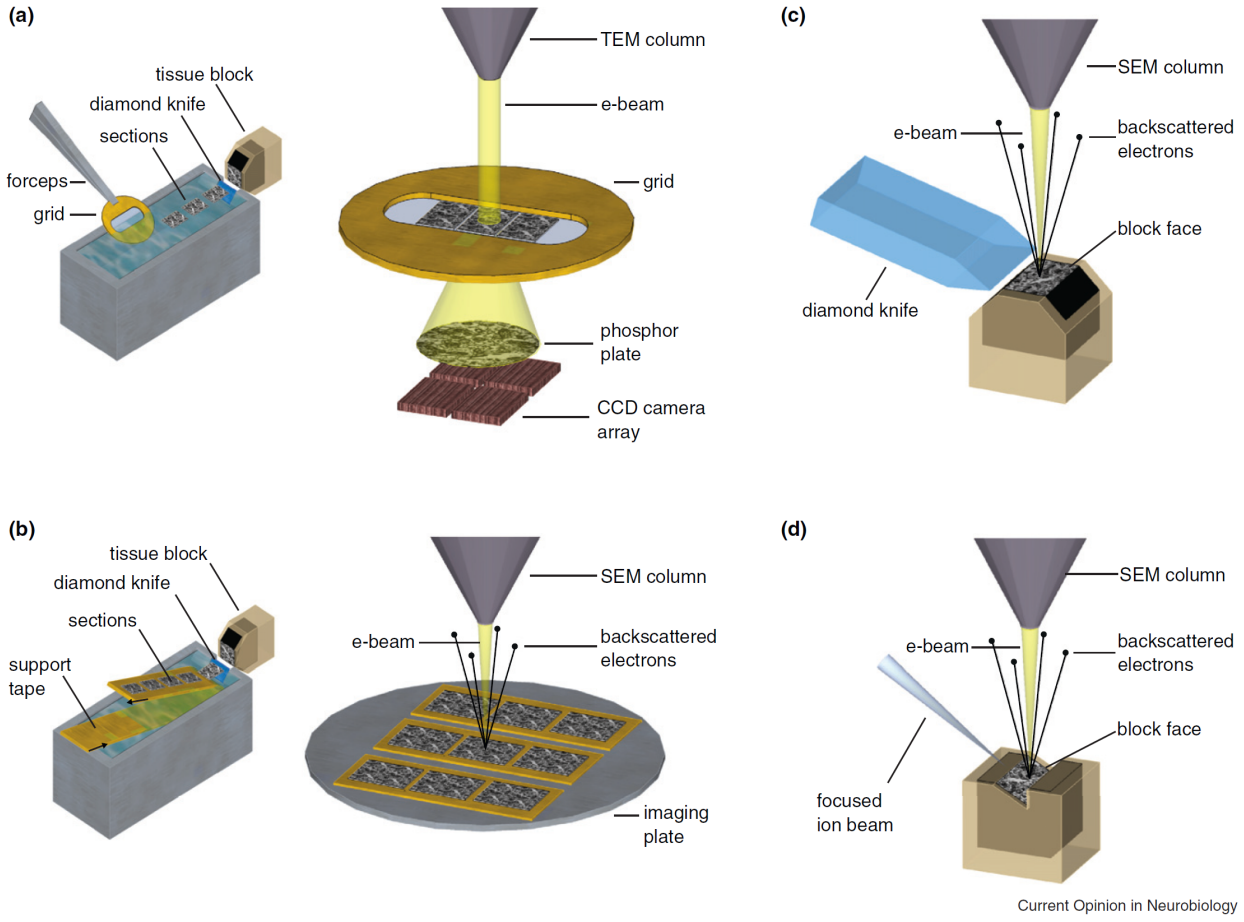


Figure 1.1: EM schematics, reprint from (Briggman and Bock 2012, with permission from publisher) (a). Serial section transmission electron microscopy (ssTEM) (b). Automated tape-collecting ultramicrotome scanning electron microscopy (ATUM-SEM) (c). Serial block-face scanning electron microscopy (SBEM) (d). Focused ion beam milling scanning electron microscopy (FIB-SEM)

The advantage of block-face EM is that it preserves the original coordinates of sections, making 3D slice alignment significantly easier. Also, it allows higher z resolution (20 nm)

since it does not require as much cutting precision as the serial sectioning approach.

On the other hand, the serial sectioning paradigm completely throws away the original spatial coordinate and requires significant effort realigning. Also, the cutting and automatic tape collection machinery can be error-prone over long cutting sessions at 30 - 40 nm thickness; tiny vibrations from the device, improper cutting speed, and non-uniform sample density could interrupt the continuity of sectioning. With current technology, at most several thousand sections can be collected without error. The major advantage over block-face is that the sections are preserved permanently and can be easily revisited and reacquired in case of bad acquisition, region selection, or for multi-scale imaging. Also, by separating cutting and scanning, the scanning process can be scaled horizontally, like distributing wafers to multiple SEMs or using Multibeam to scan up to 91 tiles simultaneously. (Eberle and Zeidler 2018; Shapson-Coe et al. 2021)

1.2 Advancements in Computational Tools

With automated data acquisition comes inevitably the challenge of data processing. To start with, a connectomics dataset can typically reach several hundred Gigabytes (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015) up to a Petabyte (Shapson-Coe et al. 2021) in recent studies. The computational tools originally designed for light microscopy quickly fell short (ImageJ, Amira, etc.). The computational infrastructure has not been ready to deal with large-scale 3D datasets until quite recently.

Alignment The first computational challenge in connectomics is image alignment, which is the process of registering a series of 2D images into a smooth 3D stack by matching features between neighboring image pairs and deforming images. The most frequently used algorithm is the Scale-invariant feature transform (SIFT) algorithm (Lowe 2004), which takes two similar images and detects local features that best match with each other and deform

the images linearly so that they can be stitched or registered. For smaller stacks, affine transform with SIFT alignment was quite effective already (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015).

However, with increasing sample volumes, alignment quickly turned non-trivial. The affine alignment(which allows only linear operations like translation, rotation, and shear mapping) would only work when the original image stack has superb uniformity and acquisition quality, which is increasingly difficult to guarantee with larger image sizes. The problem gets trickier when considering the sample defects that frequently occur during cutting(folds, tears, broken holes) or wafer preparation(carbon dust during coating, air bubbles, tape irregularities). The variety of artifact types often means there is no good way to provide a clean automated solution to cover all corner cases.

To address the limitation of linear methods, a crucial concept, "elastic alignment," was introduced and discussed in great detail in Saalfeld et al. 2012, in which the authors proposed to model each image as an elastic grid of nodes that interact with each other within and across slices. The alignment process then becomes finding an optimal solution to minimize the overall discrepancy within the elastic mesh. The trade-off is that the computational cost can be hundreds of times higher than a linear alignment. The idea is powerful and robust and is still applied broadly in the field a decade later.

The algorithm was first developed and included within TrakEM2 by Cardona et al. 2012, the first attempt to bundle various tools specifically designed for connectomics into one framework and provide a unified interface to interact with large image stacks beyond the RAM limit of typical workstations.

For larger image stacks, AlignTK by Arthur W Wetzel, Hood, and Dittrich 2013 developed on top of the idea of elastic alignment and offered more powerful nonlinear alignment. Several improvements upon TrakEM2 include the introduction of masks, allowing patches with defects to be masked out, finer scale patch registration, allowing better pixel-wise accu-

racy and native compatibility with high-performance computing facility. It was successfully applied in the zebrafish reconstruction project (Hildebrand et al. 2017).

A different tool written in Julia language (Macrina and Lh n.d.) follows similar design principles but uses cloud-based computing infrastructure. It was applied to successfully reconstruct a $250 \times 140 \times 90\ \mu m^3$ volume in mouse V1 described in Turner et al. 2020.

An internal elastic alignment engine by the Google team was developed on the same idea (Michał Januszewski, Jörgen Kornfeld, et al. 2018) and was applied in an even larger sample size (Shapson-Coe et al. 2021) with the help of massively parallel computing infrastructure.

More recently, novel alignment algorithms based on deep learning have been explored. In E. Mitchell et al. 2019, a dense vector field for the transformation between neighboring pairs of images is calculated with a trained convolutional network instead of with a predetermined algorithm like SIFT or elastic alignment. Although to what extent can it substitute previous tools remains to be seen.

Segmentation The biggest computational challenge in the connectomics pipeline is generally agreed to be segmentation (Jain, Seung, and Turaga 2010; Abbott et al. 2020). "Image segmentation" in its original form is concerned with identifying and separating objects in 2D images. Some non-machine learning methods include the watershed algorithm, which detects borders, calculates each pixel's distance to the nearest border, expands segments from each local minimum, and eventually groups each pixel to its nearest basin. The limitation of the watershed algorithm is that it requires the borders between objects to be well defined and trivially detected, which is rarely the case in natural images.

The groundbreaking work of LeCun et al. 1998 in using a convolutional neural network(CNN) to perform handwritten digit classification laid the foundation of modern deep-learning-based computer vision. The computers back then, however, were not powerful enough to perform more complicated real-life image classifications. For the majority of the 2000s, CNN was not the mainstream technique for computer vision.

Things started to change with the "ImageNet" project (Deng et al. 2009), which created a huge database with millions of labeled images and has been a golden standard for image classification benchmark ever since. Its unprecedented size of training data effectively fueled the meteoric rise of deep learning in computer vision. In 2012, AlexNet(Krizhevsky, Sutskever, and Hinton 2012) took the field by surprise and showed that deep learning could outperform traditional computer vision methods by a large margin as training data scales. Following AlexNet, neural network-based automatic image classification has seen significant improvements over the next few years. In 2014, VGG19 was proposed as a 19 layer deep convolutional network that achieved state-of-the-art performance (Simonyan and Zisserman 2014). The introduction of the inception module in (Szegedy et al. 2016) and residual connection in ResNet (He, X. Zhang, et al. 2016) significantly increased the depth of trainable networks to as deep as 101 convolutional layers(ResNet-101) and vastly improved image classification performance. The idea was also expanded to semantic segmentation by introducing a Region of Interest module to find individual objects and their contours. (He, Gkioxari, et al. 2017; Howard et al. 2017) Automatic systems based on these network architectures have already been widely applied in autonomous vehicles, industrial robots, smart cameras, and image-rich social network platforms.

In the context of medical imaging, segmentation also plays a crucial role. For example, there is significant medical application value in detecting tumors out of healthy tissue, find regions with abnormality with MRI, or separating different anatomical substructures from XRay scans. CNN was quickly adopted in the medical imaging field as well. By far, the most influential neural network architecture in this domain has been UNet (Ronneberger, Fischer, and Brox 2015), which inherited ideas from the fully convolutional network(FCN) (Long, Shelhamer, and Darrell 2015) and built a hierarchical encoder-decoder network with skip connection in between layers at the same downsample level. Its power lies in its ability to encode images into features at different resolution levels and recombine multi-resolution

features back into a pixel-level accurate mask prediction. Since its conception, UNet has effectively become the de-facto standard in medical image segmentation, and numerous modifications and variants were proposed on top of it (K. Lee et al. 2017; Heinrich et al. 2018).

In connectomics, segmentation of neurites is a necessary step for any automated analysis. The problem is complicated because the dataset is 3D instead of 2D, and neurites are densely packed with each other. At an early stage of connectomics, convolutional neural networks had been proposed to predict cell membranes against intracellular space and other organelles (Jain, Murray, et al. 2007; Ciresan et al. 2012). The predicted membrane will be the boundary between neurons; from there, a watershed could be used to produce an over-segmentation, upon which pieces are agglomerated into complete neurites. This paradigm dominated connectomics segmentation for a while (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015; K. Lee et al. 2017; Funke et al. 2018). Overall, UNet has been used to predict cellular membranes with great success, as shown in the leaderboard of CREMI challenge (Cremi.org n.d.), in which the top-ranking methods uniformly used some variant of UNet for membrane detection. On top of that, the current state-of-the-art with UNet + agglomeration paradigm (K. Lee et al. 2017) also uses UNet to predict the affinity between voxels across z dimension for better agglomeration.

While the prediction of cell membranes is already extremely accurate, minor errors like a blurred piece of the membrane could lead to merge errors that would be hard to fix in the downstream process. The agglomeration step is also error-prone with thin neurites like axons and dendritic spine necks.

In order to address the limitation of boundary prediction + agglomeration paradigm. A novel paradigm flood-filling network was proposed by Michał Januszewski, Jörgen Kornfeld, et al. 2018, in which the authors bypassed the boundary prediction step in the established segmentation paradigm and used a recurrent convolutional neural network to predict the object mask one instance at a time. Flood-filling network(FFN) achieved state-of-the-art

performance and was later used to reconstruct several increasingly larger samples: zebra finch area x: $96 \times 98 \times 114\ \mu m^3$, full Drosophila brain: $995 \times 537 \times 283\ \mu m^3$ and more recently a piece of human cortex tissue reaching roughly $1 mm^3$.

Synapse Prediction Another major computational challenge is to predict synapses automatically. At the early stage of connectomics, synapses were manually annotated. (Helmstaedter, Briggman, Turaga, et al. 2013; Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015) However, synapses have distinct morphology under EM. Each synapse has a "T-bar" shaped dark region at the contact site between axon and dendrite in the drosophila brain, while in the mammalian cortex, each synapse is associated with a vesicle cloud and a significantly darker post-synaptic density. The uniformity of these features suggests a trained approach and is a perfect use case of machine learning.

Convolutional neural networks were quickly adopted and used for this task (Dorkenwald, Schubert, et al. 2017), in which vesicle clouds, synaptic junctions(post-synaptic densities) are predicted as separate tissue classes. This idea was later explored in Heinrich et al. 2018, and a larger variant of UNet was used to detect Drosophila synaptic clefts. In a recent study, on top of predicting the post-synaptic site mask, UNet was also used to predict a 3D vector field of most likely synapse direction, demonstrating the versatility of the approach. These UNet-based methods were subsequently used in biological studies (Turner et al. 2020; Jorgen Kornfeld et al. 2020; Shapson-Coe et al. 2021).

Neurite Classification Outputs from the segmentation pipeline are often still fragmented and contain errors. By classifying neurite fragments into axons, dendrites, glia, and other useful sub-types, cross-class agglomeration errors can be detected and automatically corrected. This approach has been proposed as an alternative strategy to push the segmentation algorithm's accuracy even higher (Schubert et al. 2019). Also, having a subcompartment classification would help automated other downstream analyses like cell type classification

or synapse error checking.

Chapter 3 discusses our effort in building a state-of-the-art pipeline for this task.

1.3 High-performance computing in neuroscience

High-performance computing(HPC), more commonly called supercomputers, has been a crucial part of scientific advancements in the last few decades. Apart from computer science, It is intensively used to drive research in physics, chemistry, climate, material science, engineering, and biological sciences.

While the use of HPC had been common for structural molecular biology and bioinformatics, it had not gained much attention in neuroscience until recently (Bouchard et al. 2016). The biggest reason is that experimental neuroscience based on electrophysiology and light microscopy did not require the level of computing power HPC offers. Even with in-vivo calcium imaging, which generates live video data that could reach hundreds of Gigabytes and is orders of magnitude more data-intensive than electrophysiology, the computation required for the raw video is not that intensive. Typically the video recording is pre-processed into highly sparsified signal traces for each neuron at the very early stage of the pipeline.

Things started to change with connectomics. In the previous sections, multiple computational tools developed during 2012-2020 were discussed, and very few of them(perhaps except AlighTK) were designed to be HPC-native. The key issue is that the core of a connectomics pipeline, the segmentation engine, relies on an efficient deep learning infrastructure, which is not something the HPC field had been optimizing for. Traditionally the focus of supercomputers had been CPU intensive workloads like the simulation of particles or materials at the atomic level, but the rise of deep learning along with the growing prominence of General-Purpose Graphics Processing Units(GPU) in recent years started to disrupt that paradigm and pushed the HPC community to rethink the design of future supercomputers. Before the next-gen supercomputer is ready, the void is filled by cloud computing, like Amazon EC2

and Google Cloud, which are currently driving the largest connectomics projects like (Turner et al. 2020; Shapson-Coe et al. 2021).

Acknowledging the surging demand for GPU in the scientific and engineering fields, the next-gen supercomputers Summit(2018, Vazhkudai et al. 2018) and Aurora(2021, R. Stevens et al. 2019) were designed to be GPU-centric. Summit is equipped with 27648 Nvidia Tesla V100 GPUs and Aurora will be equipped with custom-designed GPUs with a whopping 1 exaFLOPs of peak performance. With their vastly improved performance in AI, it will be interesting to see how HPC can accelerate connectomics in this decade.

Chapter 2 discusses our effort in setting up a connectomics pipeline geared for next-gen supercomputer clusters.

1.4 The critical period and the development of visual cortex

With technological advancements, the growing ability of EM connectomics in revealing ultrastructural details has raised interest in applying it to some of the most intriguing open questions in neuroscience, for example, the development of synapses in the central nervous system (Schmidt et al. 2017; Gour et al. 2021). The development of the mammalian brain, particularly the human brain, is notably slower and could have a more profound influence on adult behavior. The idea is that by collecting high-resolution large-scale reconstructions across developmental stages, we could gain valuable insights into how neuronal and synaptic morphologies change and mature, which is a crucial piece of the puzzle of intelligence.

The major difficulty is that collecting a EM dataset is still costly and time consuming, so the age and brain region need to be carefully chosen if a study involves more than one sample. Ideally, the different samples should have dramatic structural differences previously established, or hypothesized, by electrophysiology or light microscopy. The mouse primary visual cortex during critical period fits the criteria and is therefore a good target for connectomics.

The critical period is a developmental stage during which the sensory cortex at an imma-

ture state goes through dramatic change driven largely by sensory stimuli that optimizes the circuitry permanently (S.-Y. Choi 2018). Beyond the critical period, the same level of environmental influence would no longer be able to induce large-scale rewiring in the circuitry despite a certain level of plasticity. Conversely, lack of proper exposure to environmental stimuli could severely disrupt cortex development and impair certain behaviors. The critical period was first hypothesized from observation in human language acquisition, which is anecdotally known to be significantly easier at an early age compared to later in adulthood.

David Hubel and Torsten Wiesel extensively studied the critical period in a series of groundbreaking experiments in cat and macaque visual cortices. (David H Hubel and Wiesel 1970) They discovered that ocular dominance, a phenomenon that information from two eyes relayed from Lateral Geniculate Nucleus(LGN) terminate in alternation columns in the binocular region in visual cortex layer IVC (David H Hubel and Wiesel 1962), relies on visual experience during the critical period to develop properly. In their experiments, a monkey with one eyelid sutured for the first 6 months after birth had a severely disproportional ocular dominance column pattern, where inputs from the closed eye were greatly weakened (David Hunter Hubel et al. 1977). A similar phenomenon has also been discovered in mice (Hensch and Fagiolini 2005; Hensch 2005).

The mechanisms for the critical period have been extensively studied. A crucial player is believed to be the shift in excitatory-inhibitory balance and the maturation of GABAergic neurons (Chattopadhyaya et al. 2004). In the mouse primary visual cortex, the functional maturation of GABAergic inhibition proceeds during a protracted postnatal period from the time of eye-opening to early adulthood (Z. J. Huang et al. 1999; Morales, S.-Y. Choi, and Kirkwood 2002; Chattopadhyaya et al. 2004). Apart from inhibitory synapse maturation, myelination is believed to play a crucial role in consolidating axonal targeting and ending the critical period.

One hypothesis for synapse development during the critical period is pruning, which as-

sumes the visual cortex before eye-opening has exuberant weak synapses and, over time gets trimmed according to visual signals. This hypothesis finds its root from the experiments at neuromuscular junctions, in which several axon branches coinnervate the same muscle fiber but competition among them results in the strengthening of one and elimination of others (Narayanan Kasthuri and Jeff W Lichtman 2003). Also, in the developing human cortex, synapse density was found to decrease (Huttenlocher 1984). Recent in-vivo imaging studies have shown monocular deprivation induces dendritic spine elimination in the developing mouse visual cortex (Zhou, Lai, and Gan 2017). While these findings support synapse pruning as an important aspect of visual cortex rewiring by visual experience during development at the light microscopic level, the extent of synapse pruning covering the critical period has not been investigated at the EM level and it will be the focus of my study in Chapter 4.

CHAPTER 2

A FULL CONNECTOMICS PIPELINE ON HIGH-PERFORMANCE COMPUTING FACILITY

This chapter is a full reprint of **"Toward an Automated HPC Pipeline for Processing Large Scale Electron Microscopy Data"** (Vescovi, H. Li, et al. 2020), in which I was a co-first author. The work is included with permission from all authors.

2.1 Abstract

We present a fully modular and scalable software pipeline for processing electron microscope (EM) images of brain slices into 3D visualization of individual neurons and demonstrate an end-to-end segmentation of a large EM volume using a supercomputer. Our pipeline scales multiple packages used by the EM community with minimal changes to the original source codes. We tested each step of the pipeline individually, on a workstation, a cluster, and a supercomputer. Furthermore, we can compose workflows from these operations using a Balsam database that can be triggered during the data acquisition or with the use of different front ends and control the granularity of the pipeline execution. We describe the implementation of our pipeline and modifications required to integrate and scale up existing codes. The modular nature of our environment enables diverse research groups to contribute to the pipeline without disrupting the workflow, *i.e.* new individual codes can be easily integrated for each step on the pipeline.

2.2 Introduction

Microscopy images are a significant source of insight and raw information for neuroscience. Modern techniques in electron microscopy (EM) allow scientists the ability to image at such high resolution that every single synaptic connection can be distinguished (Briggman and

Denk 2006; Helmstaedter, Briggman, and Denk 2008; Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015). Furthermore, acquisition automation has enabled us to acquire large volumes of microscopy data spanning several resolutions with minimal human involvement in the acquisition.

The popularization of these automated imaging systems has made acquiring large amounts of data the standard operation for many laboratories (Rubin 2006) (Sunkin et al. 2012), and although most of them are able to physically handle the amount of data, there is an increasing need for streamlining the pipeline. This necessity arises because of the growing acquisition speed of microscopes, leading to an exponential growth in data throughput(Keller, Zeidler, and Kemen 2014). While different techniques are emerging to solve each step of the upstream process, they still have their own independent development communities(O'Toole et al. 2018; Cardona et al. 2012; Arthur W Wetzel, Hood, and Dittrich 2013) and there are very few laboratories with the capability of carrying out the entire process by themselves. Individually these algorithms contribute to the study of neuroscience image data; it is, however, non-trivial to chain these modules together and deploy them in one coherent environment for end-to-end connectomics projects.

The continuous use of electron microscopes can produce single datasets that reach multiple petabytes of data, which cannot be processed on local workstations or small clusters and therefore require High Performance Computing (HPC) facilities. We propose to deploy and chain these different processing libraries into a single *microscope to HPC* workflow and provide a way for the user to interact with the data and its processing in real time. Our package is implemented in Python and is called HAPPYNeurons (HPC Automated Pipeline for Processing Yotta Neurons) (Vescovi and H. Li 2020).

An alternative solution is the use of cloud-based services (e.g. AWS, Google Cloud etc.), but the data size and computation time make the cost infeasible for most laboratories. On the other hand, HPC facilities at national labs have vast storage and computing capabilities

that have not been tapped in the field of connectomics due to accessibility or differences from established technology stacks. Our pipeline aims to address this limitation by incorporating state-of-the-art open source connectomics tools into an HPC environment and building an end-to-end pipeline for EM segmentation, while increasing accessibility to users and labs.

On a technical level, we propose to use MPI (Gropp, Thakur, and Lusk 1999) as a parallelization layer for each step in order to keep the internal mechanics of the original software mostly intact, while achieving compatibility with most HPC infrastructure. This allows for rapid deployment of new tools on the pipeline. It is not the goal of our work to improve the sample preparation or acquisition, but instead to enable the user to take advantage of large scale computing facilities in order to process the data. This processing is currently done in several parts that will be described in greater detail in the next chapter. Our contributions to the field of neuroscience and HPC are: 1) deployment on HPC of EM tools necessary to go from raw images to final reconstruction; 2) wrapping the tools in an operation database that can be used to create custom pipelines, 3) deployment of an computational environment that permits the user to interact with, annotate, and visualize the data without the need to transfer outside of the HPC facility.

2.3 Related work

The field of *connectomics* has been blooming with different communities trying to understand the underlying connectivity map of neural tissue. In the particular case of electron microscopy there have been parallel contributions into sample preparation, data acquisition and the diverse steps on the complex data processing involved. The sheer complexity of the problem means every step of it can still be improved and ongoing efforts on different aspects of the problem can be seen from various labs and groups in the community (C. M. Schneider-Mizell et al. 2020; Motta et al. 2019; Jorgen Kornfeld et al. 2020).

For alignment, computer vision methods (O'Toole et al. 2018; Cardona et al. 2012; Arthur

W Wetzel, Hood, and Dittrich 2013) are still heavily used to assemble the raw data into 3D volumes, but are slowly being surpassed by machine learning-driven methods (E. Mitchell et al. 2019). Segmentation has long been the rate-limiting step and still requires a lot of human annotation. Originally it could take weeks to trace a single neuron through a stack of images. With the aid of deep learning algorithms, segmentation can be vastly accelerated. Efforts on neural networks such as U-Net (J. Wu, W. M. Silversmith, and Seung 2019) and Flood-Filling Network (FFN) (Michał Januszewski, Jörgen Kornfeld, et al. 2018) have proven successful for the task of automatic segmentation of neurons.

Human-intensive data annotation is crucial to establish datasets for machine learning approaches. The web-based package webKnossos (Boergens et al. 2017) enables laboratories to deploy an intuitive interface to annotate datasets, without requiring annotators to transfer up to petascale datasets between their institution and the hosting site. We leverage webKnossos in our work to make the increasingly large connectomics datasets available for annotation by distant annotators.

Upon completion of reconstruction, datasets are meshed for visualization and made available via Neuroglancer (Maitin-Shepard n.d.), a program developed by Google that visualizes flat, black-and-white electron images, related labels, and reconstructions as a colourful 3D forest of neurons.

2.4 Computational Pipeline

Electron microscope image processing follows a certain number of pre-defined steps from raw data to a final scientific result. Given the data throughput of modern microscopes, executing all those steps within the same facility as the microscope is a challenge. Figure 2.1 describes the connection between the electron microscope lab at Argonne National Laboratory and the Argonne Leadership Computing Facility (ALCF), showing the services involved in our pipeline environment. After the microscope finishes acquiring an image (or set of images), it

triggers an action that is stored on an external database server. This database of actions then controls the transfer and processing of data through the storage and computing resources. On the front end side the user can either manipulate the data or the actions to make the pipeline unique to each sample.

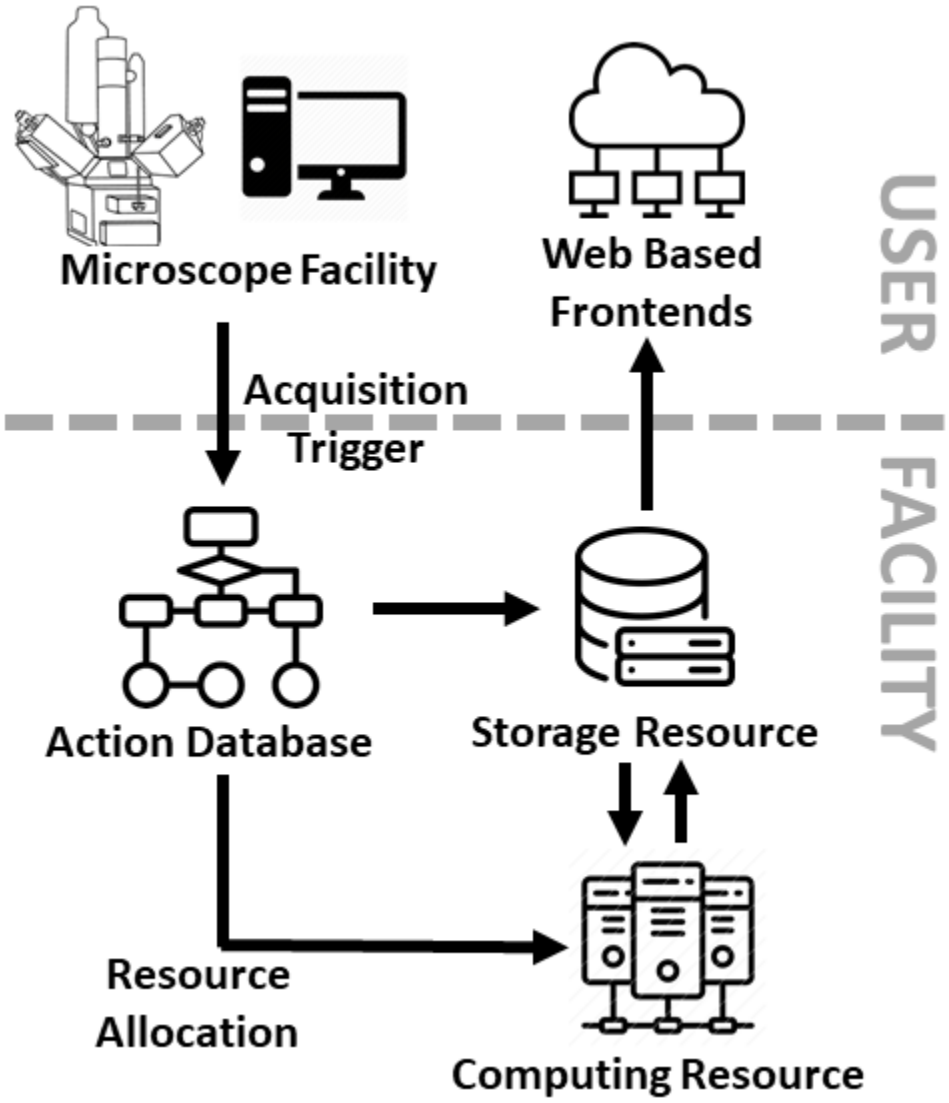


Figure 2.1: Model of the flow of data between the electron microscope facility (top side) and the HPC facility (lower side). The microscope acquisition populates the action database which controls the storage and computing resources. The user can visualize and manipulate the data and actions through web-based front ends.

For simplicity we will describe the basic steps involved in processing electron microscope

images individually. Even though these steps are described in a sequential fashion, in practice the processing is conducted through an iterative process involving automated computational steps and human-intensive guidance.

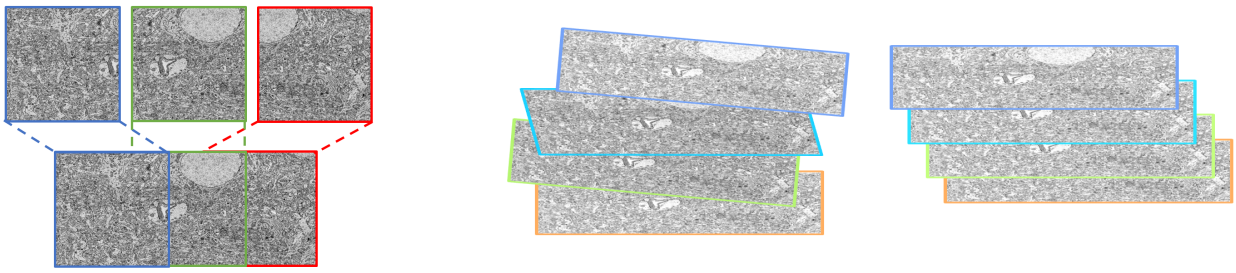
2.4.1 Processing methods for EM data

In order to make the the interaction with the data as easy as possible for the user, we encoded basic operations that can eventually be described as sequential pipelines. Since human validation is necessary at multiple steps, the user can choose where and when to interact with the pipeline. These operations are described below:

Montage is the process of positioning and merging overlapping image tiles into a single larger image. Given sufficient metadata about the arrangement of the tiles, this step can be executed largely without user intervention. We implemented a headless macro for TrakEM2 (Cardona et al. 2012) and developed a Python wrapper for MPI parallelization of this procedure. The mutual independence of image sections makes it possible to trigger the montage operation during acquisition, once the full set of images for a tile have been acquired, and process it on the fly.

Alignment is the process of ensuring that neighboring images in the stack are aligned according to their contained features, and is a crucial step for serial electron microscopy. In our pipeline, we use AlignTK (Arthur W Wetzel, Hood, and Dittrich 2013) to perform elastic alignment on the montaged image stack. We implemented wrappers for AlignTK's core functionality to better adapt to parallel deployment on HPC, along with a set of utility tools for image preprocessing, including contrast normalization, scaling, and artifact thresholding. Figure 2.2 shows examples of montage and alignment images.

Segmentation is the process to assign unique IDs to individual neurite objects in a 3D volume and is one of the key challenges in connectomics analysis. The current state of the art, FFN, has achieved great success in accuracy and scale (Michał Januszewski, Jörgen Kornfeld,



(a) Montage Example

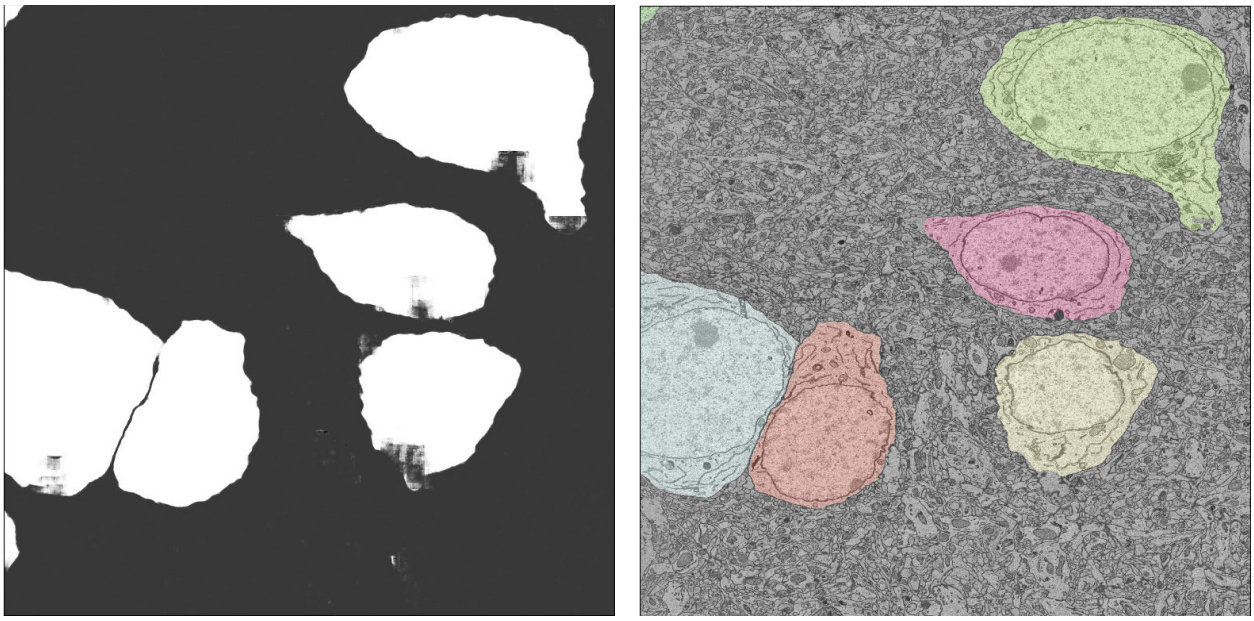
(b) Alignment example

Figure 2.2: Example of the montage process (left) and alignment process (right).

et al. 2018; Zheng, Lauritzen, et al. 2018; Jorgen Kornfeld et al. 2020) over the last few years. Although it was originally designed for distributed computing platforms and despite preliminary efforts on distributed training(Dong et al. 2019), it has not previously been deployed on HPC infrastructure for large scale segmentation. In this pipeline, we made modifications to the open source release of Google's FFN. First, we added MPI-based parallelization for execution at large-scale HPC facilities. Second, we added support for reading precomputed volume(W. Silversmith n.d.[a]) data as input, in addition to *HDF5* which reduces repetitive data usage and seamlessly integrates with the visualization engine Neuroglancer. Third, we implemented a reconciliation step that merges overlapping subvolume inference results into a final segmentation in precomputed format.

Mask Prediction: In practice, a prerequisite for FFN is identifying tissue masks that would disrupt segmentation. These masks are used to omit imaging artifacts or large objects that can make the final segmentation less accurate, like cell-bodies and blood vessels. For cell-bodies and blood vessels, we implemented and ran a classic 2D U-Net(Ronneberger, Fischer, and Brox 2015). We created manual annotations on every 100 images at 4x resolution and used these to train a U-net model, which was used for patch-wise inference over the full volume. Manual seeds were placed at the center of each cell-body, and a 3D watershed algorithm was run to provide initial segmentation of cell bodies and blood vessels. Figure 2.3 shows the process of creating the cell mask from the U-Net probabilities (on the left) to

the actual cell body masks (on the right).



(a) U-Net cell body probabilities

(b) U-Net Watershed Overlay

Figure 2.3: Result of the U-Net segmentation of large body sizes. These can be subsequently used to mask-out already known objects in the final segmentation.

Mesh Generation produces a mesh-based representation to support 3D visualization of the segmented objects. Currently this step is achieved using the Python library Igneous(W. Silversmith n.d.[b]).

Skeletonization creates a point graph for every object and can also be processed by using the TEASAR (Sato et al. 2000) implementation inside Igneous.

Manual Annotation: We used WebKnossos (Boergens et al. 2017) to provide manual volumetric annotations as a training/validation set for FFN. This step is human-intensive, and is typically approached iteratively, with the biologist annotating an initial sample partially, rerunning training and inference with FFN, and making further corrections. To accommodate data format requirements of different packages and to streamline this process, we implemented utilities for easy transformation of data formats between WebKnossos cube, which is used by WebKnossos, stacks of tiff images, and HDF5 (Folk et al. 2011), which are traditional data formats, and Precomputed (W. Silversmith n.d.[a]), which is used by

CloudVolume and Neuroglancer.

User inspection of intermediate results between pipeline stages is currently essential. To this end, we have developed a number of facilities to enable interaction with the datasets that reside on ALCF systems. In the case of connectomics datasets, the intermediate results can be very large, typically involving many images in the range of hundreds of megapixels each; the ability to view these results quickly, in-place on ALCF systems, is critical. For each intermediate dataset, we have developed code to produce downsampled versions of select output data, and a Jupyter notebook template which can be copied into the target run directory to view the downsampled data. The JupyterHub deployment at ALCF has direct access to the Theta and Cooley filesystems, creating a highly usable environment for viewing, which can be customized with additional Python-driven analyses.

While segmentation results can be viewed using Jupyter notebooks as above, these results are more typically visualized in 3D using the Neuroglancer application. We have deployed Neuroglancer at ALCF to support viewing segmentation results, which we demonstrate later in the text. Whereas Jupyter notebooks access Theta-resident data directly, Neuroglancer retrieves data using web protocols; to support this, we transfer segmentation results to the 3PB Petrel community storage system(Allcock et al. 2019) at ALCF, and expose the data to Neuroglancer using standard web protocols.

2.4.2 Workflow Management

We have designed a pipeline that encompasses the individual software packages described above, allowing them to be executed independently or assembled into more automated workflows. These software packages are leveraged in an iterative fashion, varying parameters to achieve desired accuracy on each particular dataset, and software development is ongoing. To account for this scenario, our pipeline is modular, supports multiple interfaces, and aims to enable the entire data life cycle from raw images to final results. Figure 2.4 shows the

electron microscope operations described above with the I/O present on our package.

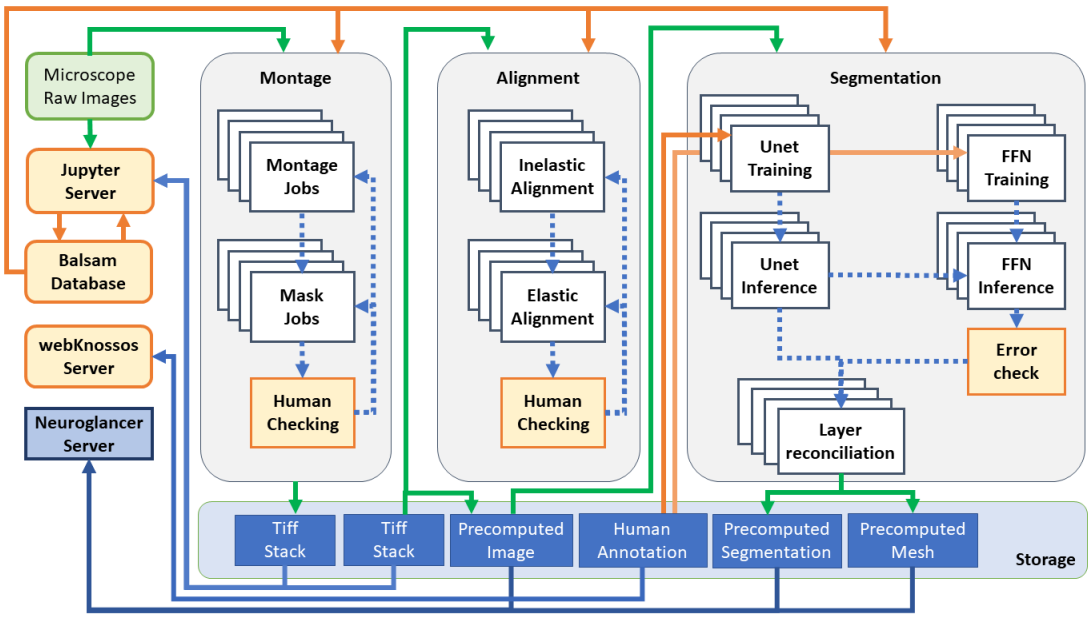


Figure 2.4: Schematic of the pipeline. The green box represents the data acquisition. Orange boxes and arrows represent human interactions in the pipeline. The white boxes represent HPC submitted jobs. Blue boxes represent the data storage and visualization server. Green arrows represent I/O from the computing resources.

Given the complex nature of the acquired data, we acknowledge that human intervention is required during the process; our goal is to facilitate this while minimizing the iteration time and making interaction with the data easier. This is important given that any imperfection or artifact in the sample and data acquisition can cause the steps of the pipeline to fail and require human intervention.

The pipeline is implemented as a Python library, with an API that exposes the individual applications to be run, together with specifications of input data and configuration, producing a collection of jobs that will be offloaded to our HPC facilities. This functionality is generalized such that choice of execution machine can be made at the time of execution rather than being tightly integrated into the job definitions, which permit users to encapsulate the jobs on their own schedulers depending on the computational infrastructure. On our HPC resources, we achieve further flexibility by relying on the Balsam workflow toolkit

(Salim et al. 2019). Using Balsam's Python programming interface, we populate a database with the desired pipeline actions and, using the specifics of the target computing resource, define a collection of jobs to be run. Balsam manages the execution of these jobs on the target computing resource, optimizing for concurrency and throughput, handling errors, and providing monitoring and reporting details as the pipeline jobs are executed. This level of control frees the user from laborious management of the compute jobs and enables Balsam to systematically manage execution and data management ensuring that the compute resources are used efficiently.

Pipeline users interact with Balsam via two interfaces: a Python API to define steps in the workflow and a command line interface for allocating resources and launching applications at the appropriate scale. The mapping of parallel tasks to MPI ranks varies across applications: rank/section for montage with TraekEM2, rank/section-pair for alignment with AlignTK, and rank/subvolume for segmentation using FFN. This interface is mirrored in the Python/Balsam interface. Input data and configuration details are provided via Python calls, and passed to Balsam, which handles defining jobs in the underlying job database.

Once the job database has been populated , one can use the Balsam command-line interface to submit jobs to the Theta queues for execution and to monitor the jobs as they run.

2.5 Results

In our experiments we used two HPC resources: Theta, an 11.69 PFLOPs supercomputer and Cooley, a GPU cluster with Nvidia K80s, both at the ALCF. Theta is composed of 4392 compute nodes, each with a 64-core, 1.3-GHz Intel Xeon Phi 7230 processor, 192GB DDR4 RAM and 16GB high-bandwidth MCDRAM. When needed, we also used a workstation with dual Xeon E5 2630v4, 256GB memory and two Titan X Pascal GPUs. Our usage of FFN relied on TensorFlow 1.14.

The microscope used on our experiments was the Zeiss SEM Gemini 300 (Carl Zeiss and Sigma 2011), which can provide images at up to 6nm of spatial resolution. The acquisition automation is done by the Atlas software from Zeiss.

The sample tissue was dissected from a 14 day old, post-natal mouse brain in primary visual cortex layer 4, and was prepared according to the protocol described by Hua *et al.* (Hua, Laserstein, and Helmstaedter 2015). The tissue was then cut into 40 nm slices on ATUM(R. Schalek et al. 2011) and scanned with the microscope at 6 nm resolution and 3.5 µs pixel dwell time. For each of the 1312 slices, two 10833 x 14000 pixel tiles were scanned in sequence with 5% overlap.

2.5.1 Workflow Validation

As an initial validation step, two stages of the pipeline–montage and alignment–were run on a subset of the data, using the Balsam execution backend. For montage, 128 sections of data were selected, with a corresponding Balsam job describing the input and configuration for TrakEM2. These jobs ran on 32 Theta nodes, with Balsam managing the distribution of work to compute nodes as they became available. We demonstrate this approach in the current case because this flexible approach to computing will become essential in the context of larger image stacks in the future. A similar approach was taken to run image alignment on this 128-image stack. For alignment, the database was populated with jobs to run on 16 nodes of the Cooley visualization cluster, with jobs distributed to compute nodes as they became available.

To highlight user interactivity with the pipeline, we provide a Jupyter notebook example where, given a raw dataset, the pipeline stages described above are submitted to a Balsam database with standard (or user-provided) configurations for execution on Theta. This approach permits the user to run collections of jobs multiple times, such as to perform parameter sweeps. We created Balsam jobs for the TrakEM2 montage step with multiple

configurations, varying the maximum and minimum octave used in the search for the correct overlap between two images, which also affects the run time, and calculated the error rate of those configurations on a given dataset (as shown on Table 2.1). An initial analysis of the resulting images was conducted to identify montage failures, using the image size as a proxy. The accumulated error was calculated by the number of images that were corrected by changing the parameters. The final accumulated error shows the fraction of images that could not achieve a correct montage with any of the tested parameter sets; these, therefore, must be corrected through direct user intervention. We continue to develop metrics for identifying montage errors, in the interest of further automating this process.

TrakEM2 - Min	TrakEM2 - Max	RunTime	Error Rate	Accumulated Error
400	2000	100min	35%	35%
400	3000	260min	15%	10%
400	3500	450min	9%	6%
1000	3500	520min	6%	1%

Table 2.1: Execution times of TrakEM2 headless montage macro on a test dataset (6x2 tiles of 15000x15000 pixels and 1128 slices divided into 8 different folders that represent the acquisition sessions). Each line shows the results of 8 Balsam jobs with 32 nodes, 4 ranks per node and the values for the TrakEM2 parameters for the minimum and maximum octaves used by the montage macro script.

Lastly, we simulated online processing of images from the electron microscope by triggering a transfer of images on a schedule that approximates typical operation. For the dataset described in this work, each section was imaged as two separate tiles, each 8-bit tile having image dimensions 10833x14000 and occupying 151MB. The imaging time for each tile is on the order of 10 seconds, so a full section is imaged every 20 seconds. In this simulation, we transferred a full section from the microscope-connected machine to Theta every 20 seconds and added a montage job to the Balsam database, continuously, over a period of three hours. In the current paradigm, at this rate, a wafer of 200 sections would be imaged in about one hour, producing 30GB; this equates to a daily rate of 720GB. It is clear from this experi-

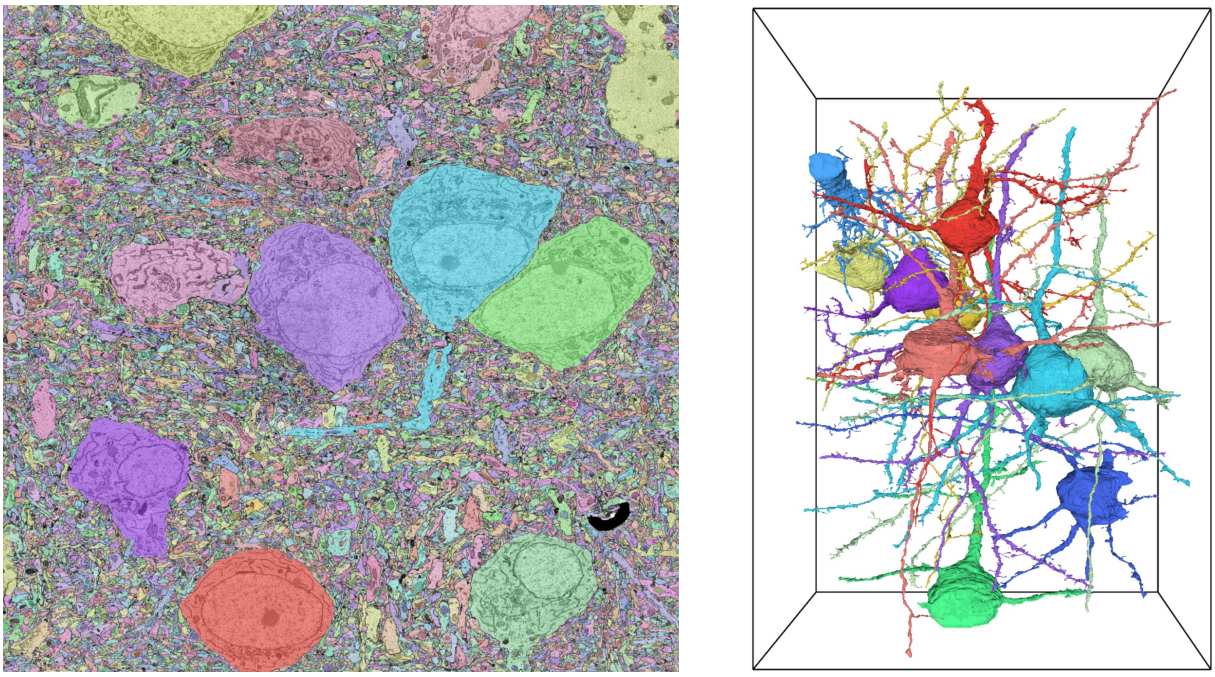


Figure 2.5: Final visualization using Neuroglancer. On the left, an example of a raw image (black and white) overlapped with the inference labels(colors). On the right, 3D rendering of the same cells.

ment that Theta is able to keep pace with the incoming jobs at this rate (each TrakEM2 job was run on a single Theta node, using 64 cores per node, and 2 threads per core, as we determined this to be the optimal configuration, with runtime averaging 440 seconds). To achieve this, we began with an initial allocation of Theta nodes, with the Balsam executor configured to grow and shrink the pool of nodes as needed, corresponding with the flow and ebb of incoming jobs. This demonstration is, in itself, not a compelling demonstration of the full extent of the current capability; it does, however, show that we have the technology in place to trigger image transfer and job injection when a section has been imaged, which will become a necessity in the future, where we anticipate transfer of images from multiple microscopes simultaneously to process on ALCF supercomputers. We are currently undertaking a scaling study to examine bounds on throughput in this scenario, which will be the focus of a future publication.

2.5.2 Complete Pipeline

We demonstrate the pipeline being executed from raw tiles to the final reconstruction on a 90 x 125 x 52 μm volume of neural tissue. Each part of the pipeline was executed as a standalone call from a bash script using our Python wrappers or, where appropriate, by calling applications directly.

After montage and alignment, the data size was 15000x20800x1312 voxels at 6x6x40 nm^3 resolution, in 8 bit grayscale, with a total size of 324 GB. Segmentation was carried out at 2x lower resolution to reduce merge errors and to increase speed. To perform training, we acquired an FFN model trained on the Kasthuri11 (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015) dataset from the authors of (Michał Januszewski, Jörgen Kornfeld, et al. 2018) as an initial checkpoint. Using manual annotations of a 256x256x128 voxel volume from our own dataset, we incrementally trained the base model until accuracy saturated at 0.91; this training was run on a separate workstation with dual Titan X GPUs for 12 hours, as a transfer learning job that didn't require Theta-scale computing. Before proceeding to inference, we first performed cell-body and vessel masking with U-Net/watershed on the workstation at 4x downsampled resolution, and used that as an initial segmentation. We then split a total volume of 6700x9900x1312 voxels (at 12x12x40 nm^3 resolution) into 3618 512x512x128 cubes with 32x32x16 overlap in each dimension. Inference jobs with the trained model were run on 32 nodes of the Cooley cluster each with 2 NVIDIA K80 GPUs, with one MPI rank per GPU, for a total of 72 hours; afterwards the subvolumes were reconciliated (recombined into a full volume) on a workstation for final visualization and error checking. Figure 2.5 shows the visualization of the reconstructed data in Neuroglancer. Our pipeline have been tested on datasets as large as 1Tb and it can be scaled to larger volumes as long as the resource allocation allows it. The limitations come from data sharing between different resources and the final user. Another limitation comes from the increase in the error rate of individual algorithms in larger volumes.

2.6 Closing Remarks

In summary, HAPPYNeurons provides a software pipeline to integrate electron microscopes with HPC facilities. We demonstrate an end-to-end connectomics reconstruction pipeline using HPC resources. This is achieved by wrapping multiple libraries as a coherent set of operations and providing the ability to chain them together using Balsam to enable more optimized scheduling on supercomputers. Due to the modular design of the workflow, the wrapped applications can be combined according to the needs of the current application and dataset, and new modules can be added with ease. HPC facilities can be used in a seamless manner, enabling the processing of large scale data without the monetary burdens of cloud computing. This integration paves the way for using supercomputers for connectomics reconstruction, in preparation for the deluge of data anticipated from faster next-generation microscopes, and to enable exascale computers to process it. This modular design also allows for different processes of the pipeline to target different accelerators on the HPC resources (i.e. dedicated GPU's). We are currently studying the execution of our pipeline at larger scale on Argonne supercomputers; these results will appear in a future publication.

Acknowledgements

We thank all members of the Kasthuri Laboratory at University of Chicago for providing the insights into Electron Microscopy and biology. We thank Dr. Shuichi Shigeno for data used in the preliminary tests. We also thank Michal Januszewski for sharing a pre-trained FFN model and insights into FFN's hyper-parameters. This research is funded in part by, and used resources of, the Argonne Leadership Computing Facility which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357 and funded in part by NIH: 5R01MH110932-04.

CHAPTER 3

AUTOMATIC CLASSIFICATION OF CONNECTOMICS RECONSTRUCTION AND APPLICATION IN ERROR CORRECTION

This chapter is a full reprint of **"Neuronal Subcompartment Classification and Merge Error Correction"** (H. Li et al. 2020), in which I was the first author. The work is included with permission from all authors.

3.1 Abstract

Recent advances in 3d electron microscopy are yielding ever larger reconstructions of brain tissue, encompassing thousands of individual neurons interconnected by millions of synapses. Interpreting reconstructions at this scale demands advances in the automated analysis of neuronal morphologies, for example by identifying morphological and functional subcompartmentswithin neurons. We present a method that for the first time uses full 3d input (voxels) to automatically classify reconstructed neuron fragments as axon, dendrite, or somal subcompartments. Based on 3d convolutional neural networks, this method achieves a mean f1-score of 0.972, exceeding the previous state of the art of 0.955. The resulting predictions can support multiple analysis and proofreading applications. In particular, we leverage finely localized subcompartment predictions for automated detection and correction of merge errors in the volume reconstruction, successfully detecting 90.6 % of inter-class merge errors with a false positive rate of only 2.7 %.

3.2 Introduction

Recent advances in 3d electron microscopy (EM) have enabled synaptic-resolution volumetric imaging of brain tissue at unprecedented scale (Zheng, Lauritzen, et al. 2018; Dorkenwald,

Turner, et al. 2019; Xu et al. 2020). Semi-automated reconstructions of these volumes yield thousands of neurons and neuronal fragments, interconnected by millions of synapses (Dorkenwald, Schubert, et al. 2017; Michał Januszewski, Jörgen Kornfeld, et al. 2018; P. H. Li et al. 2019; Buhmann et al. 2019). Together, reconstructed neurons and synapses within each dataset describe a "connectome": a connectivity graph whose structure is anticipated to underlie the computational function of the tissue (Dasgupta, C. F. Stevens, and Navlakha 2017; Jorgen Kornfeld et al. 2020). Interpreting neural connectivity at this scale is a significant undertaking. One means to enhance interpretability is to use ultrastructural and morphological details of neuronal fragments to distinguish their functional subcompartments. For example, the classical description of "neuronal polarity", i.e. the flow of information within vertebrate neurons, from dendritic subcompartments, into the soma, and out through the axon, remains central to understanding connectivity (Swanson and Jeff W Lichtman 2016). Although trained human reviewers can classify many neuronal fragments with respect to subcompartment, the growing scale of connectomic reconstructions demands automated methods. A recent approach to this problem was based on training random forest classifiers on manually defined features extracted from neurite segments and separately detected organelles such as mitochondria or synapses (Dorkenwald, Schubert, et al. 2017; Motta et al. 2019). A later extension improved accuracy by classifying 2d projections of neurites and their organelles with convolutional neural networks (CNNs), a technique called Cellular Morphology Networks (CMNs) (Schubert et al. 2019) . However, an approach based on a full 3d representation of neuron fragments, which retains the maximum morphological and ultrastructural information, has not been previously demonstrated. Another application for subcompartment predictions is in proofreading, e.g. to correct errors in automated reconstructions. Prior works proposed to detect merge errors through identification of morphologically unlikely cross-shaped fragments (Motta et al. 2019; Meirovitch et al. 2016) or used a 3d CNN trained specifically to detect merge (Rolnick et al. 2017) or split errors (Haehn et al. 2018). Strong bi-

ological priors dictate that vertebrate neurons have only one major axonal branch extending from the soma, and that dendritic and axonal subcompartments do not typically intermingle within a neurite (Swanson and Jeff W Lichtman 2016). Prior work used these cues to tune agglomeration via sparse subcompartment predictions in a multicut setting, which optimizes over an explicit edge-weighted supervoxel graph (Krasowski et al. 2017; Pape et al. 2019). Alternatively, violation of biological priors in subcompartment predictions can be used to detect post-agglomeration reconstruction errors, which can then be flagged for efficient human proof-reading workflows (Hubbard et al. 2020), or fully automated error correction. In the following we (1) present a system for neuronal subcompartment classifica-tion based on 3d convolutional neural networks, (2) demonstrate finely localized subcompartment predictions whose accuracy exceeds state-of-the-art, and (3) show how these predictions can be used for high-fidelity detection and correction of agglomeration errors in an automated segmentation.

3.3 Materials and Methods

3.3.1 Datasets

We used an automated Flood-Filling Network (FFN) segmentation of a $114 \times 98 \times 96$ μm volume of zebra finch Area X brain tissue acquired with serial blockface EM at a voxel resolution of $9 \times 9 \times 20$ nm (Michał Januszewski, Jörgen Kornfeld, et al. 2018). Base FFN supervoxels (SVs) were agglomerated (Fig. 1a) via FFN resegmentation, with additional post-processing applied to the agglomeration graph to reduce merge and split errors (Jorgen Kornfeld et al. 2020). We also used precomputed organelle probability maps for synaptic junctions and vesicle clouds (Dorkenwald, Schubert, et al. 2017) in some experiments. The agglomerated segmentation was skeletonized via TEASAR (Sato et al. 2000), and the resulting skeletons were sparsified to a mean inter-node spacing of 300 nm and eroded so that terminal nodes were at least 100 nm from the segment boundary. A subset of the objects in

the volume were manually classified by human experts as axon, dendrite, or soma, of which 27 objects were used for training (32.8k axon, 8.4k dendrite, 7.5k soma nodes), 6 objects for validation (2.6k, 1.0k, 2.0k), and 28 objects for evaluation (29.3k, 38.2k, 32.7k). Datasets are available from the CMN authors on request (Schubert et al. 2019).

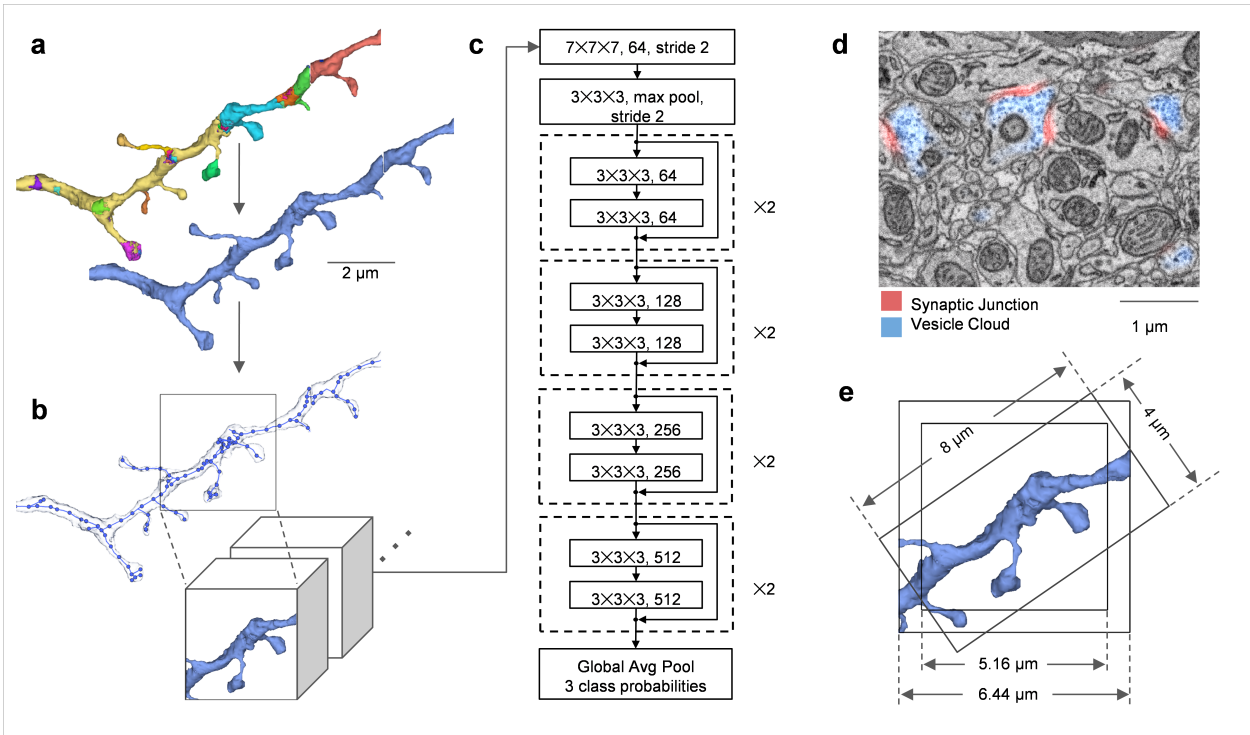


Figure 3.1: **Neural subcompartment classification with 3d CNN.** (a) The segmentation consists of base SVs (top, different colors) that were agglomerated into more complete neuron segments (bottom, solid) (Michał Januszewski, Jörgen Kornfeld, et al. 2018). (b) Input FOVs are centered at node positions from automated skeletonization of the segmentation mask. (c) The classifier architecture is a 3d extension of a ResNet-18 CNN, and outputs probabilities for axon, dendrite, and soma subcompartment classes. (d) For some experiments, we provided additional input channels, e.g. the contrast normalized (Zuiderveld 1994) EM image, or precomputed organelle probability maps (Dorkenwald, Schubert, et al. 2017). (e) Illustration of the two primary FOV sizes used in our experiments, approximately 5.16 or 6.44 µm on a side. For comparison, we also illustrate the neurite-aligned 4x4x8 µm FOV employed by the previous CMN approach (Schubert et al. 2019)

3.3.2 Classification of neural subcompartment with 3d CNNs

Classifier input fields of view (FOVs) were centered at neuron skeleton node locations, with the segment mask extracted from the neuron's agglomerated segmentation (Fig. 3.1b). However, multiple axonal and dendritic processes from the same neuron sometimes pass close to each other, even if their connection point is far outside the FOV. Therefore, it was beneficial to remove segment mask components in the FOV that were not connected with the component at the center. Disconnected component removal was done at full $9 \times 9 \times 20$ nm resolution, prior to downsampling the block to the network input resolution.

Classifier architectures were derived from the ResNet-18 CNN model (He, X. Zhang, et al. 2016), with convolution and pooling layers extended to 3d (Fig. 3.1c). Neuronal morphology was provided to classifiers as a 3d binary segment mask. When additional input channels (EM image, organelle masks; Fig. 3.1d) were provided, the segment mask was applied to the other channels instead of being provided separately, with areas outside the mask set to zero. Input data was provided at $36 \times 36 \times 40$ nm resolution in blocks of 129 or 161 voxels on a side, for a total field of view of $4.64 \times 4.64 \times 5.16$ or $5.80 \times 5.80 \times 6.44$ μm respectively (Fig. 3.1e). Network output comprised probabilities for axon, dendrite, and soma subcompartment classes.

For training, the input locations were class balanced by resampling skeleton nodes for underrepresented classes multiple times per epoch. We also applied random 3d rotations to the input as a training data augmentation. Networks were trained via stochastic gradient descent, with learning rate 0.003 and batch size 64 for 1.5M steps. For the best performing network, with two input channels and 6.44 μm field of view, the total number of trainable parameters was 33.2M.

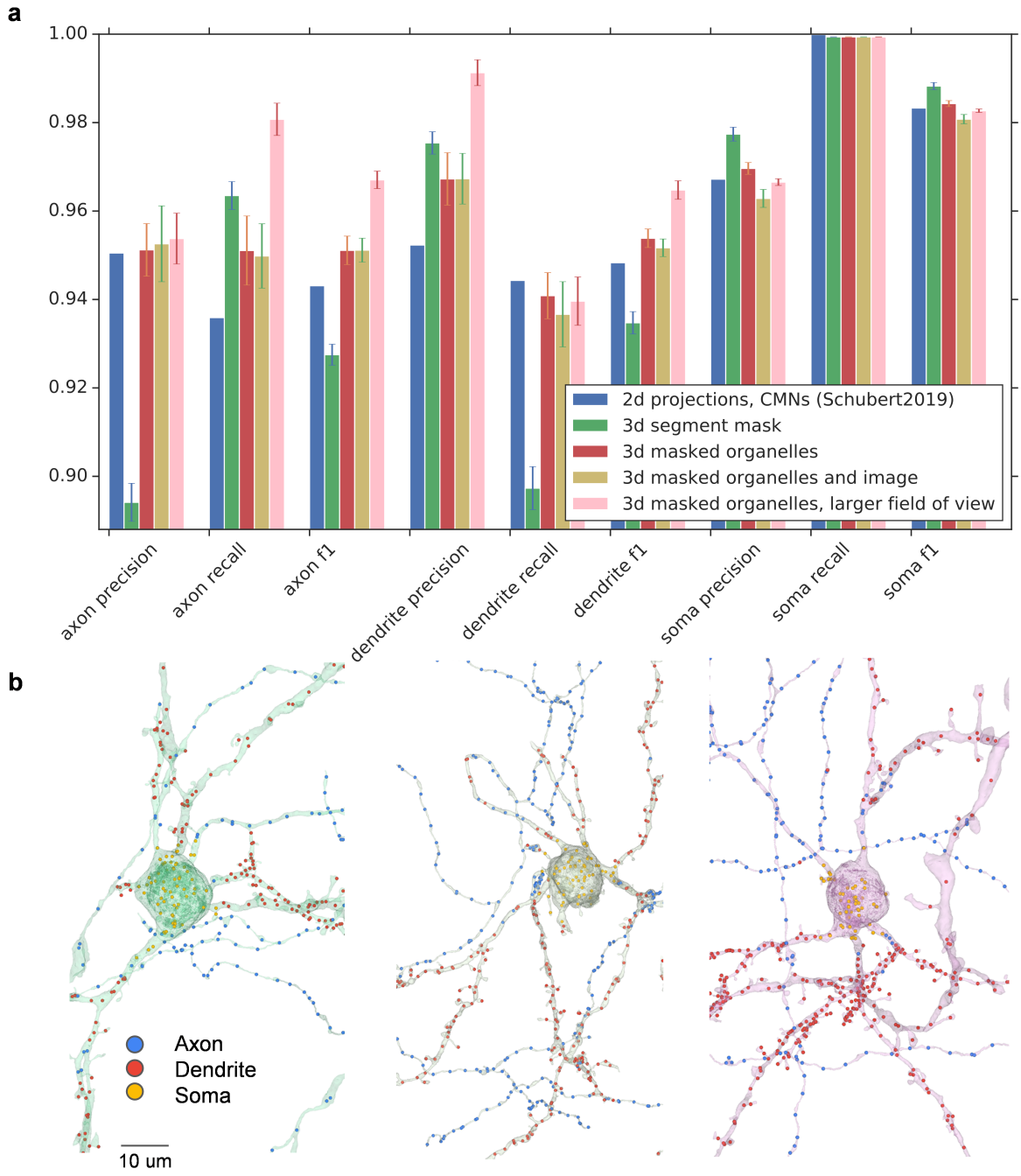


Figure 3.2: **Subcompartment classification results.** (a) Node classification performance on axon, dendrite, and soma labeled examples. (b) Skeleton node classifications of three automated neuron reconstructions outside the train, validation, and evaluation sets.

3.3.3 Automated detection and correction of merge errors

We applied top-performing subcompartment predictions (Fig. 2) to further improve neuron reconstruction quality by detecting merge errors between different classes. FFN reconstruction biases base SVs (Fig. 1a) to have very few merge errors via an oversegmentation consensus procedure (Michał Januszewski, Jörgen Kornfeld, et al. 2018), so we focused on errors in SV agglomeration. Once agglomeration errors are identified and localized, they can be fixed efficiently by simply removing the bad agglomeration graph edges, either under human review (Hubbard et al. 2020) or automatically. We used subcompartment predictions to identify all somas and branches, and then to detect and correct two classes of agglomeration errors: axon/dendrite branch merge errors, and soma/neurite merge errors. As ground truth, we manually identified 132 agglomerated neurons that contained merge errors and annotated their bad agglomeration graph edges. This yielded 473 branches, among which there were 83 branch merge errors and 56 soma merge errors. Together these represent a significant fraction of all merge errors identified through an exhaustive screening of the reconstruction.

3.3.4 Branch merge error correction by graph cut consistency score

Branch merge errors involve a mis-agglomerated axon and dendrite (Fig. 3.3a). Intuitively, branches that contain a merge error tend to have lower overall node prediction consistency (defined as weighted mean probability of dominant class type). Removing a bad agglomeration edge should improve the node consistency of the two resulting subgraphs.

The input to our system is the skeleton of the agglomerated neuron with node class predictions, and the neuron's agglomeration graph, where each skeleton node contains information about the base SV it belongs to. The workflow is as follows:

Step 1: Identify soma. If the segment has > 200 soma classified nodes, find the SV with the most soma nodes.

Step 2: Separate branches from soma. After removing the primary soma SV, each

remaining subgraph of the agglomeration graph is considered a branch if it contains > 100 nodes.

Step 3: Compute node weights (optional). Automated skeletonization sometimes over-clusters nodes within thicker objects. Densely clustered nodes can optionally be down-weighted by 1 / (node count in 500 nm radius - 2) to discount the nodes in excess of the three expected within each 500 nm of clean path length. Furthermore, neurite nodes proximal to the soma (within 5-10 µm) tend to have inconsistent class predictions. These nodes can also be optionally down-weighted by 0.01, which effectively ignores them except in cases where soma proximal nodes are the only nodes on a branch.

Step 4: Group predictions. Node predictions are aggregated by base SV, to compute weighted mean class probabilities P_{SV} and node count w_{SV} for each SV (Fig. 3.3b).

Step 5: Compute cut scores. Any cycles are first removed, then edges are traversed from leaf nodes in. At each edge, the branch is conceptually divided into subgraphs Gleave and Gremain, and the "cut consistency score", a measure of how many nodes belong to their respective majority classes post- versus pre-cut, is computed (Fig. 3.3b):

$$\frac{max \sum_{SV}^{G_{leave}} P_{SV} w_{SV} + max \sum_{SV}^{G_{remain}} P_{SV} w_{SV}}{max \sum_{SV}^{G_{branch}} P_{SV} w_{SV}} \tag{3.1}$$

Step 6: Detection. The highest predicted cut score is thresholded to determine if the branch contains a merge error, with constraints that Gleave and Gremain must have different majority class types and their weighted sizes must be > 50.

Step 7: Correction (optional). The suggested agglomeration edge is removed, and majority vote pooling is performed within subcomponents. Branch pooled node prediction accuracy is compared pre- and post-cut (Fig. 3.3e).

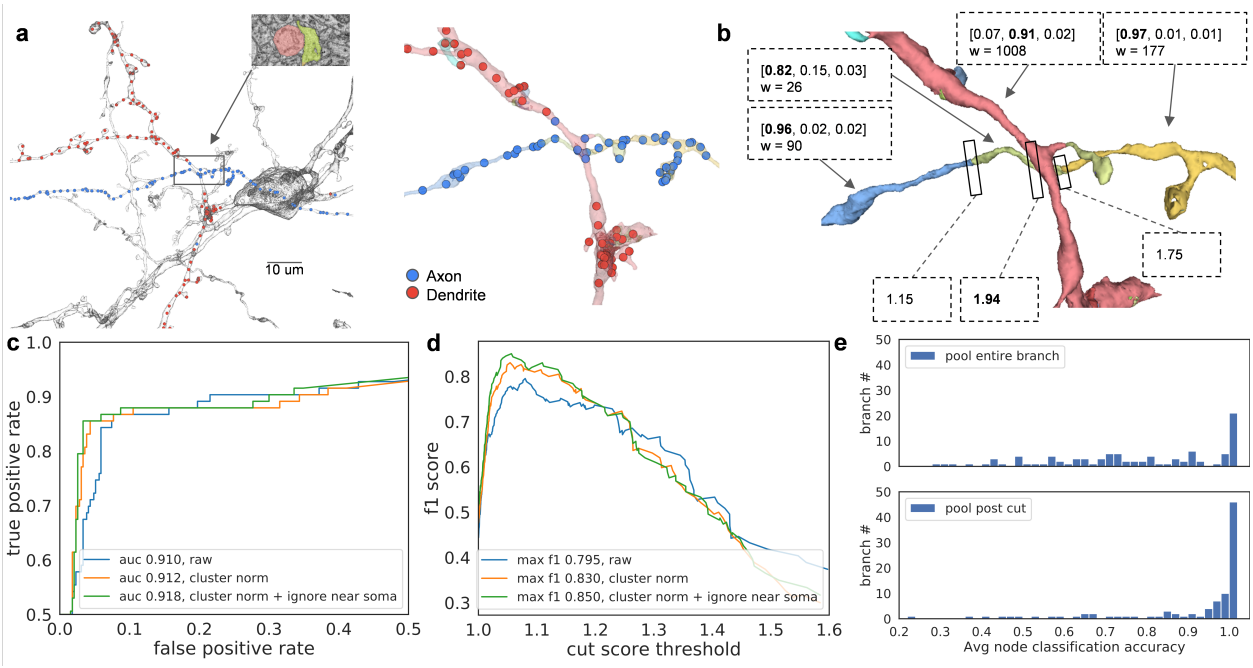


Figure 3.3: **Correction of branch merge errors via subcompartment prediction.** (a) Left, view of an agglomerated segment centered on a branch merge error, with the node predictions for the branch overlaid. Inset shows the EM image and overlaid base SVs for the merge. Right, zoomed in view of the merge error, with base SVs in different colors. (b) Node predictions are aggregated to get class probabilities [axon, dendrite, soma] and weight (node count) per SV. Three SVs are predicted axon, one dendrite, reflecting the merge error. Candidate neuron cuts are annotated with their consistency improvement scores (Eq. 1). (c) ROC plot showing detection performance as the cut score threshold is varied. Separate curves show three variants with different node reweighting (to address node clustering or nodes close to the soma). (d) The f1 of merge detection versus cut score threshold. (e) Branch-wise majority vote pooled class accuracy distribution before (top) and after (bottom) applying suggested cuts.

3.3.5 Soma merge error correction by trajectory of primary neurite

The second error mode involves a neurite fragment that is mis-agglomerated with soma (Fig. 3.4). We observed that these errors can be fixed with a simple heuristic based on branch trajectory relative to the soma surface. The pipeline is as follows:

Steps 1-2: See branch merge detection pipeline above.

Step 3: Distance to soma. For each axon or dendrite node, compute the distance d_s to the nearest soma node.

Step 4: Distance to branch root. For each branch, the root is the node with minimum d_s. Compute the distance d_r from each branch node to the root (Fig. 3.4b). **Step 5:** Fit slope. For each branch, compute a linear fit to d_s versus d_r (Fig. 3.4c) for nodes within a tunable distance to soma. The slope of the fit is then thresholded to determine if a branch is a soma merge error (Fig. 3.4c-e).

3.4 Results

3.4.1 Subcompartment classification performance of 3d CNNs

We compared the performance of our 3d CNN classifiers to previous state-of-the-art results from CMNs (Schubert et al. 2019), in terms of class-wise precision, recall, and f1 metrics (Fig. 3.2a). For each trained 3d CNN, we saved parameter checkpoints throughout the training period and screened them on a small manually labeled validation set. For most models, performance on the validation set approached or exceeded 0.99 on all metrics (not shown), but validation performance was useful for tracking convergence, confirming there was no overfitting to the training set, and for avoiding checkpoints where training temporarily became unstable. We then applied the ten checkpoints with highest validation accuracy to the larger evaluation set to compute the mean and standard deviation for each metric.

Compared with CMNs (Fig. 3.2a, blue), a network analyzing voxel representation of 3d segment shape alone was competitive (green). Adding vesicle cloud and synaptic junction organelle probability map channels allowed the 3d CNN to exceed state of the art (red). Interestingly, further adding the full EM image channel had negligible impact (yellow). Expanding the field of view for the masked organelles network from $5.16\mu m$ to $6.44\mu m$ yielded the best performing system tested (pink). We also tested expanding the FOV further, increasing the input resolution, increasing the CNN depth, and providing different input channel configurations; see supplemental Table S1. Of the top ten checkpoints from the best per-

forming model, the median overall node accuracy on the evaluation set was 97.1%, and mean f1 across classes was 0.972. We then used this median checkpoint to predict node classes for reconstructed neurons and fragments throughout the entire volume. Predicted skeletons demonstrate good class consistency within soma and neurites, with some ambiguity at the interface between branch and soma (Fig. 2b). Based on the predictions, the volume contains 3.25 m total axon path length, and 0.79 m dendrite path length, a ratio of 4:1 that is similar to the 5:1 ratio previously reported (Schubert et al. 2019). However, the total path length here significantly exceeds that previously reported, probably due to differences in skeleton sparsity, so the absolute lengths here should be considered an upper bound.

3.4.2 Agglomeration merge error detection and correction

We fed subcompartment predictions back to detect and correct two classes of reconstruction merge errors that occur during SV agglomeration: axon/dendrite branch merges, and soma/neurite merges.

The branch merge error correction system is based on analyzing the predicted subcompartment class consistency of agglomerated segments, with and without candidate cuts applied (Fig. 3.3a-b). We first considered branch merge error detection performance, and plotted the receiver operating characteristic (ROC) curve by varying the cut score threshold (Fig. 3.3c-d). In areas with many small SVs, several nearby cut candidates can have equivalent impact, so predicted cuts that fell within four agglomeration graph edges of ground truth cuts were considered correctly detected. The best detection performance was at 1.05 cut score threshold, with f1 of 0.850 (see also Table S2).

Merge error detections can be used to flag the location for human review. We also calculated the node prediction accuracy improvement after directly applying the suggested cut. For the 96 branches with either a predicted merge or ground truth merge, we manually determined their nodewise ground truth class as axon or dendrite, then performed majority

vote predicted class pooling before and after applying predicted cuts. Comparing class pooled accuracy pre- and post-cut (Fig 3.3e), the mean node prediction accuracy improves from 0.804 to 0.886.

We addressed the second category of agglomeration merge errors, between somas and nearby neurites, by analyzing the trajectory of the branch relative to the somal surface (Fig. 3.4a-c). We found the best performance is achieved by sampling skeleton nodes within the initial 10 µm from the soma, yielding an f1 of 0.923 at a slope of 0.78. (Fig. 3.4d-e; see also Table S3).

Combined, branch and soma merge analyses detected 90.6% of merge errors, with a false positive rate of only 2.7%.

3.5 Conclusions & Discussion

To make volume EM datasets of brain tissue easy to analyze at scale, it is crucial to reduce the data they contain to more compact and semantically meaningful representations. Segmentation and synapse detection provide an important first step in this process. Here we presented a system that can provide further information about the biological identity of neurites by predicting subcompartment types, and feed back to the preceding reconstruction stage through automated correction of agglomeration errors.

We expect this approach to be useful for brain circuit analyses, and to be applicable to diverse datasets. We also anticipate that the approach could be extended to finer grained subcompartment classification. For example, the subcompartment localization of a postsynaptic site on e.g. a dendritic spine, dendritic shaft, soma, or axon initial segment is linked to both the synapse's functional impact as well as the identity of its presynaptic partner ((PING et al. 2008; Contreras, D. J. Hines, and R. M. Hines 2019). Another related application is in the identification of neuronal subtypes, whose shared structural and functional properties can enhance connectome interpretability by organizing thousands of individual

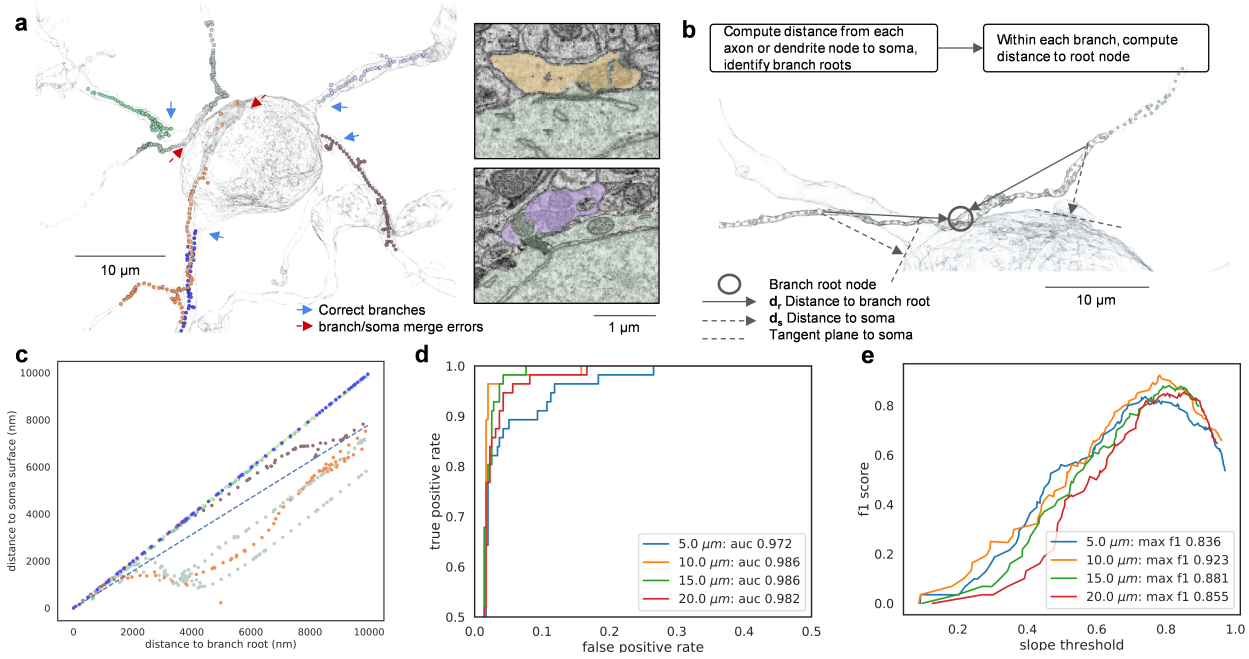


Figure 3.4: **Correction of soma merge errors.** (a) Example of a soma with multiple neurite branches, each in a different color. Two of the branches were erroneously agglomerated to the soma. (b) For each node along a branch, the distance to the nearest soma node is computed. The distance to the branch root (defined as the branch node closest to the soma) is also computed. (c) The soma distance versus branch root distance for the nodes comprising branches from (a), with matching color-coding. The dashed line of slope 0.78 separates the trajectories of correct branches that run primarily radially out from the soma, from the soma merge error branches that run primarily tangential. (d) ROC plot showing performance of merge error detection as slope threshold is varied. Separate curves show results with nodes at different distances from the soma included in the analysis. (e) The f1 of merge error detection versus slope threshold.

neurons into a reduced complement of conceptual roles (Jiang et al. 2015; Gouwens et al. 2020; Grünert and Martin 2020).

The primary advantages of our system are its simplicity, and its ability to capture complete local information about a neurite, resulting in a new state of the art. A fundamental limitation is that processing efficiency drops with increasing field of view as the neurite of interest fills a progressively smaller fraction of the voxels that need to be processed. This limitation could be mitigated by using an alternative representation of sparse 3d data (Kipf and Welling 2016; Qi et al. 2017; Riegler, Osman Ulusoy, and Geiger 2017; Mescheder et al.

2019; B. Graham, Engelcke, and Van Der Maaten 2018).

3.6 Acknowledgments

We thank Philipp Schubert and Jörgen Kornfeld for sharing detailed CMN results and training data, as well as the EM volume. We thank Jeremy Maitin-Shepard and the anonymous reviewers for comments on the manuscript.

3.7 Supplemental Material

3.7.1 Node classification ablations and extensions

Table S1 presents performance metrics on the evaluation set for 15 node classification experiments, as well as the performance of the previous state-of-the-art CMNs (Schubert et al. 2019). Bold rows indicate data already plotted in the main text (Fig. 2).

Rows 5-7 and 9 focus on changing the input channels. Rows 5 and 6 show the relative performance of using only synaptic junction or vesicle cloud probability maps alone, rather than both together (row 3); the vesicle clouds perform better, but neither is sufficient on its own. Rows 7 and 9 show that adding mitochondria probability maps does not improve performance, and may cause some degradation.

Row 10 shows that further expanding the field of view (FOV) to 193 voxels on a side degrades performance relative to the 161 voxel model (row 8). As FOV increases, the neuron segment mask is increasingly sparse, i.e. the input is increasingly empty. This may cause the model to train less smoothly, consistent with the larger checkpoint variance.

Rows 11 and 12 show that increasing input resolution to 18x18x20 nm degrades performance relative to the 36x36x40 nm models (e.g. row 3). The number of voxels in the input block for the higher resolution models remained the same, so the resulting reduction in FOV in terms of microns likely explains the degradation. Interestingly, adding the full EM image

does improve the performance of these higher resolution models, in contrast to the lower resolution models where adding the image has little effect (rows 3 and 4). This is consistent with our observation that as human viewers we find a significant amount of detail is lost in the EM image when downsampling from 18x18x20 to 36x36x40 nm.

Rows 13-16 repeat several of the experiments using a deeper ResNet-50 network architecture. Overall, the results are similar to ResNet-18 performance. The best performing ResNet-18 model performs somewhat better than its ResNet-50 equivalent (rows 8 and 16) and is significantly less computationally intensive, so we favored ResNet-18 in experiments and production.

	FOV (vox)	Res (nm)	Extra Inputs	Model Depth	Axon			Dendrite			Soma		
					prec	recall	f1	prec	recall	f1	prec	recall	f1
1	CMN				0.951	0.936	0.943	0.952	0.944	0.948	0.967	1	0.983
2	129	40		18	0.894 ± 0.004	0.964 ± 0.003	0.928 ± 0.002	0.975 ± 0.003	0.897 ± 0.005	0.935 ± 0.003	0.977 ± 0.002	0.999 ± 0.000	0.988 ± 0.001
3	129	40	sj, vc	18	0.951 ± 0.006	0.951 ± 0.008	0.951 ± 0.003	0.967 ± 0.006	0.941 ± 0.005	0.954 ± 0.002	0.970 ± 0.001	0.999 ± 0.000	0.984 ± 0.001
4	129	40	im, sj, vc	18	0.953 ± 0.009	0.950 ± 0.007	0.951 ± 0.003	0.967 ± 0.006	0.937 ± 0.007	0.952 ± 0.002	0.963 ± 0.002	0.999 ± 0.000	0.981 ± 0.001
5	129	40	sj	18	0.892 ± 0.006	0.956 ± 0.006	0.923 ± 0.004	0.970 ± 0.005	0.892 ± 0.005	0.929 ± 0.003	0.972 ± 0.001	0.999 ± 0.000	0.986 ± 0.001
6	129	40	vc	18	0.917 ± 0.006	0.953 ± 0.008	0.934 ± 0.003	0.968 ± 0.006	0.912 ± 0.006	0.939 ± 0.003	0.970 ± 0.001	0.999 ± 0.000	0.984 ± 0.001
7	129	40	sj, vc, mito	18	0.950 ± 0.007	0.959 ± 0.011	0.954 ± 0.004	0.974 ± 0.008	0.940 ± 0.006	0.957 ± 0.003	0.971 ± 0.001	0.999 ± 0.000	0.985 ± 0.001
8	161	40	sj, vc	18	0.954 ± 0.006	0.981 ± 0.004	0.967 ± 0.002	0.991 ± 0.003	0.940 ± 0.005	0.965 ± 0.002	0.967 ± 0.001	0.999 ± 0.000	0.983 ± 0.000
9	161	40	sj, vc, mito	18	0.934 ± 0.007	0.969 ± 0.009	0.951 ± 0.003	0.981 ± 0.007	0.924 ± 0.006	0.952 ± 0.002	0.967 ± 0.001	0.999 ± 0.000	0.983 ± 0.000
10	193	40	sj, vc	18	0.940 ± 0.014	0.970 ± 0.020	0.955 ± 0.007	0.983 ± 0.016	0.923 ± 0.013	0.952 ± 0.005	0.962 ± 0.001	0.999 ± 0.000	0.980 ± 0.000
11	129	20	sj, vc	18	0.902 ± 0.011	0.937 ± 0.009	0.919 ± 0.004	0.954 ± 0.007	0.919 ± 0.011	0.936 ± 0.004	0.992 ± 0.002	0.999 ± 0.000	0.995 ± 0.001
12	129	20	im, sj, vc	18	0.938 ± 0.010	0.935 ± 0.012	0.936 ± 0.003	0.954 ± 0.008	0.932 ± 0.009	0.943 ± 0.003	0.971 ± 0.002	0.999 ± 0.000	0.985 ± 0.001
13	129	40		50	0.898 ± 0.006	0.960 ± 0.004	0.928 ± 0.002	0.973 ± 0.003	0.900 ± 0.006	0.935 ± 0.002	0.976 ± 0.001	0.999 ± 0.000	0.987 ± 0.001
14	129	40	im	50	0.951 ± 0.009	0.959 ± 0.006	0.955 ± 0.002	0.974 ± 0.004	0.937 ± 0.009	0.955 ± 0.003	0.966 ± 0.002	0.999 ± 0.000	0.982 ± 0.001
15	129	40	sj, vc	50	0.950 ± 0.005	0.958 ± 0.006	0.954 ± 0.002	0.972 ± 0.005	0.941 ± 0.005	0.956 ± 0.001	0.971 ± 0.001	0.999 ± 0.000	0.985 ± 0.000
16	161	40	sj, vc	50	0.951 ± 0.002	0.975 ± 0.003	0.963 ± 0.001	0.986 ± 0.003	0.938 ± 0.002	0.961 ± 0.001	0.967 ± 0.001	0.999 ± 0.000	0.983 ± 0.000

Table 3.1: Node classification ablations and extensions vox: voxels on a side; Res: roughly isotropic resolution; prec: precision; im: EM image; sj: synaptic junctions; vc: vesicle clouds; mito: mitochondria

3.7.2 Detailed branch merge detection metrics

Table S2 shows the detailed performance metrics for the best performing (in terms of f1 metric) cut score threshold settings of the branch merge detection pipeline, corresponding to data plotted in the main text (Fig. **??**c-d).

Node reweighting	Precision	Recall	F1	AUC
no reweighting	0.753	0.843	0.795	0.91
clusters	0.807	0.855	0.83	0.912
clusters, near soma	0.845	0.855	**0.85**	0.918

Table 3.2: Branch merge detection performance (best f1)

3.7.3 Detailed soma merge detection metrics

Table S3 shows the detailed performance metrics for the best performing (in terms of f1 metric) slope threshold settings of the soma merge detection pipeline, corresponding to data plotted in the main text (Fig. 3.4d-e). After excluding predicted soma merge error branches, we were able to identify a single correct axonal branch 82.4% of the time.

Sampling range	Precision	Recall	F1	AUC
$5\mu m$	0.852	0.821	0.839	0.972
$10\mu m$	**0.885**	**0.964**	**0.923**	**0.986**
$15\mu m$	0.839	0.929	0.881	0.986
$20\mu m$	0.779	0.946	0.855	0.982

Table 3.3: Soma merge detection performance (best f1)

CHAPTER 4

COMPARATIVE STUDY OF SYNAPSE DEVELOPMENT FROM P14 TO P105

This chapter has not been published in a peer-reviewed journal by the submission time of this thesis. I would be the first author of the manuscript that we plan to submit in the near future, encompassing the content of this project. Co-authors include Gregg Wildenberg, Tom Uram, Nicola Ferrier, Narayanan Kasthuri.

4.1 Abstract

The mammalian visual cortex goes through massive rewiring throughout the critical period. Understanding how synapses develop from an immature state into fully functioning neural circuits is of great interest. Despite advancements in connectomics in recent years, saturated reconstruction of more than one developmental stage in the visual cortex has not been demonstrated. In this study, we acquired and reconstructed tissue samples from mouse primary visual cortex layer 4 at age P14 and P105 with serial electron microscopy, then performed automatic segmentation with Flood-filling network and synapse prediction with UNet. We found that synapses increase in both size and density from eye-opening to early adulthood over the critical period, challenging a pruning centric view of circuit development. Also, a sharp increase of somatic and perisomatic shaft synapse density is observed, supporting the significance of modulatory inhibitory inputs in regulating the visual cortex development. Furthermore, a significant increase in mitochondria coverage and its correlation with synapse density is reported, suggesting the importance of mitochondria in dendrite and synaptic maturation.

4.2 Introduction

A fundamental concept of mammalian brain development is that connections between neurons are exuberant in the newborn and, over postnatal life, neurons "prune" back synapses on some target neurons while strengthening connections on others (Neniskyte and Gross 2017). Classic examples of pruning are the development of ocular dominance columns in the cat and primate primary visual cortex, where cortical Layer 4 neurons receive connections from thalamic neurons both representing left and right eye inputs at birth, but those inputs gradually segregate such that 'columns' of the cortex are responsive primarily to one eye in the adult (David H Hubel and Wiesel 1962; David H Hubel and Wiesel 1968). It was observed with sensory deprivation, thalamocortical synapses are pruned and axon arbors retract (Antonini and Stryker 1993; Antonini, Fagiolini, and Stryker 1999). It was also reported in primates, approximately 5000 synapses are lost per second between 2.7 and 5 years old (Bourgeois and Rakic 1993). Similar pruning processes have been documented for retinal ganglion cells inputs onto thalamic neurons in the LGN (Guido 2008) and in the cerebellum, where multiple climbing fibers innervate an individual Purkinje cell at birth, but in adulthood, only one climbing fiber remains (Hashimoto and Kano 2013). Finally, in the peripheral nervous system, motor neurons prune inputs onto individual muscle fibers (Narayanan Kasthuri and Jeff W Lichtman 2003), and pre-ganglionic axons prune inputs on ganglion neurons in retina development (D'Orazi, Suzuki, and Wong 2014). Also, in a more recent work, it was found in mouse V1 layer 2/3, oligodendrocyte precursor cells prune axons (Buchanan et al. 2021). These experiments suggest that pruning could be a universal phenomenon.

However, there remain large gaps in the pruning story. Since much of the data was collected using proxies for synapses, e.g., trans-synaptic labeling using radioactive amino acids (Crowley and Katz 2000), or focused on spines (Narayanan Kasthuri and Jeff W Lichtman 2004; Moyer and Zuo 2018). There has been concern that technical limitations potentially

may have overemphasized pruning or that pruning varies substantially across animals or even within animals (Horton and Hocking 1996). A gap relevant to this proposal is that previous documentation of pruning, particularly in the cortex, has centered almost exclusively on higher-order mammals, e.g., cats, nonhuman primates, and humans (Huttenlocher 1984). There is little documentation of synaptic pruning in the mouse cortex, where a better understanding of the process would allow investigation with the wide variety of tools, genetic and otherwise, available in that species. A few examples are fixed time point fluorescence data, where the limited resolution can not clearly visualize synapses and could lead to misidentification of synapses (Grutzendler, Narayanan Kasthuri, and Gan 2002). A recent study using in-vivo time-lapse imaging has revealed that monocular deprivation induces dendritic spine elimination in the developing mouse visual cortex (Zhou, Lai, and Gan 2017), but the difficulties with time-lapse imaging in pups prevent the investigation of synaptic structural dynamics over a wider range in the living visual cortex. Thus, there remains a gap in comprehensively detailing the amount of pruning against synaptogenesis.

Large volume EM connectomics has revealed lots of insights into adult mouse circuits. Notable examples include (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015; Motta et al. 2019; C. M. Schneider-Mizell et al. 2020), but "developmental connectomics", a sub-field using connectomics tools to study the development, is rather new. Due to the time and resource cost in collecting an EM dataset, few studies go into the temporal dimension and attempt reconstructing multiple samples until recently. While EM reconstruction has been performed in a study of C.elegans development (Witvliet et al. 2021) and studies of axonal development in the entorhinal cortex and the somatosensory cortex Schmidt et al. 2017; Gour et al. 2021, it has not been used to investigate synapse changes on dendrites during visual cortex development.

We report an analysis of dendritic synapse development during the critical period with a semi-automatic connectomics pipeline on supercomputers.

4.3 Results

4.3.1 Reconstruction of mouse V1 layer 4 samples at two different ages

We acquired and performed saturated reconstruction over a $120 \times 80 \times 52$ μm block from Layer 4 visual cortex of a P14 male mouse and a $120 \times 130 \times 36$ μm block from the same region of a P105 male mouse with an automatic tape-collecting ultramicrotome scanning electron microscopy (ATUM-SEM) pipeline (Fig. 4.1), sampling the beginning of the critical period and early adulthood. This is the first attempt in studying the development of the visual cortex beyond the critical period with a connectomics toolbox at such scale as far as we know.

Connectomics studies in the last few years typically either focus on automatic reconstruction of one sample as large as technology allows (Hildebrand et al. 2017; P. H. Li et al. 2019; Motta et al. 2019; Turner et al. 2020; Shapson-Coe et al. 2021) or study several ages of a region with skeleton tracing based reconstruction and manual synapse annotation (Schmidt et al. 2017; Gour et al. 2021).

We differ from the standard practices and aim to reconstruct moderately sized samples at two important time points in the developmental history of a specific cortical region while also providing saturated automated segmentation and synapse prediction with state-of-the-art machine learning tools, allowing for a wide variety of probes into the morphological and topological differences emerging over the course of development.

Previous studies focused on the target preference of axons (Schmidt et al. 2017; Gour et al. 2021) and relied heavily on manual skeleton annotation. In this study, we focused our attention on dendrites and study how synapses evolve through the critical period.

In order to perform high throughput connectomics reconstruction. We built an end-to-end pipeline for EM data processing integrated with high-performance computing(HPC) facilities, combining several state-of-the-art methods from the field that were previously

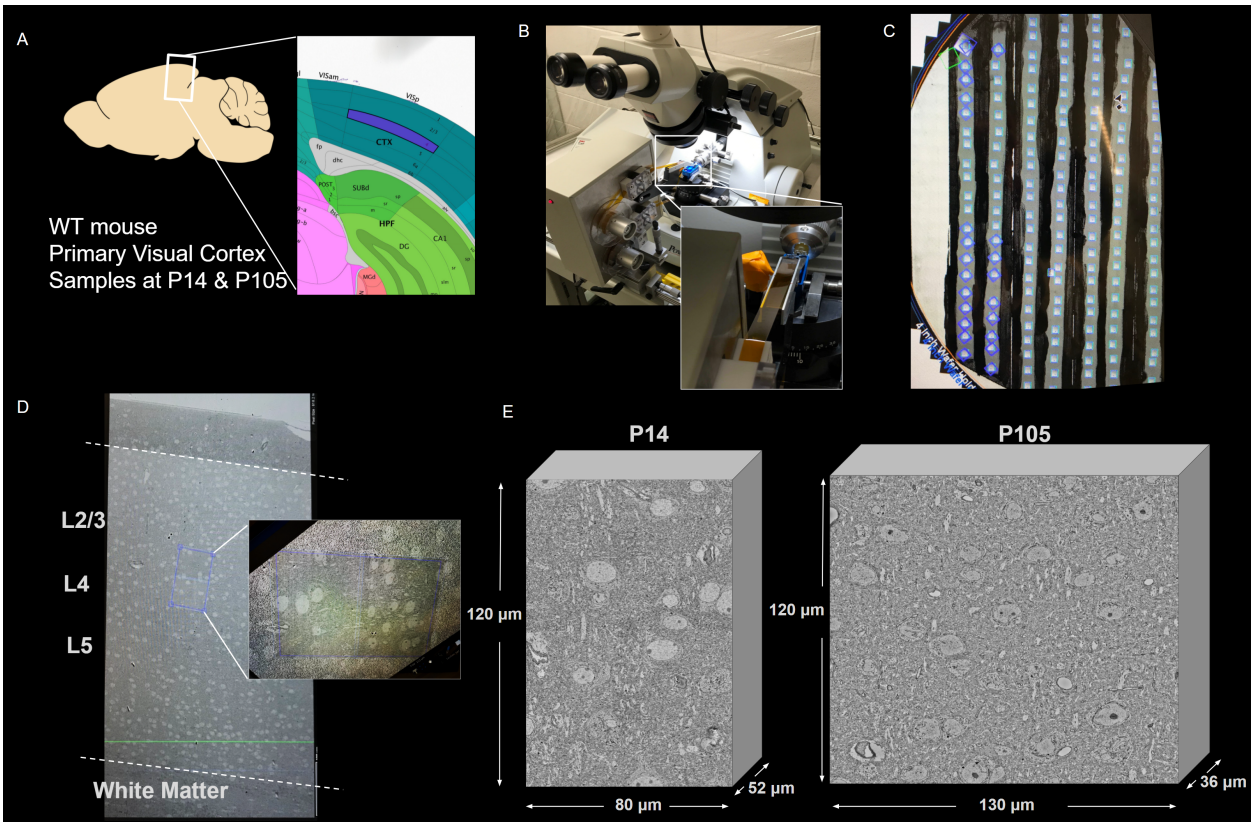


Figure 4.1: Acquisition of P14 and P105 V1 datasets: (A). Samples are dissected from the V1 region, referencing Alan Atlas (Brain Science 2007; Lein et al. 2007). (B). ATUM(RMC) collecting sections. (C). Sections mounted on wafer and field-of-views are selected with ATLAS(Zeiss) software. (D). A focused view of all layers in a slice of P14 sample, inset showing the actual region acquired with 6nm resolution. (E). Full dimensions of the two datasets collected. (left) P14 (right) P105

designed for different platforms. The integration with the HPC aspect is discussed in chapter 2 (Vescovi, H. Li, et al. 2020) in more detail, while the specific methods and steps for this experiment are discussed in more detail in the materials and methods section.

Briefly, after data collection with ATUM-SEM, the raw image tiles are montaged with TrakEM2 and aligned with AlignTK. We then used 2D UNet for soma, blood vessel, and myelin mask prediction, creating an exclusion mask and initial segmentation containing just soma and blood vessels. We then used one of the state-of-the-art methods, Flood-filling network, for large-scale segmentation (Fig. 4.2). Finally, we used 3D UNet for various intracellular organelle classifications and performed synapse and mitochondria detection (Fig.

4.3), providing a comprehensive reconstruction of the two volumes.

Overall, the automated synapse detection system returned 174207 synapses in P14, and 377445 in P105, albeit with primarily recall errors that require manual proofreading. We thoroughly proofread 88 dendrites from the P14 dataset originating from 41 cell bodies and 107 dendrites from the P105 dataset from 50 cell bodies, overall proofreading 6809 synapses from P14, 13509 from P105.

4.3.2 Significant increase in synapse density

Synaptogenesis and synape pruning are important aspects of synaptic circuit development. The large-scale structural change during the critical period is of particular interest. Most previous studies relied on time-lapse light microscopy, focused on individual branches, and had trouble tracking all synapses with limited resolution. With a connectomics pipeline, we could investigate the shift in synapse distribution and morphology at an unprecedented scale.

It is known that during the critical period, dendritic spines are highly motile and undergo tremendous reorganization. Numerous transient spines are formed and eliminated (Majewska and Sur 2003), but the balance between synaptogenesis and pruning and the shift in synapse density cannot be easily determined without EM reconstruction.

Previous work in mouse neuromuscular junctions found that "At birth, each neuromuscular junction is co-innervated by approximately ten highly intermingled axons (versus one in adults)" (Tapia et al. 2012). excessive axonal arbors die out in the first few postnatal days through an activity-dependent process for motor system to mature. Whether the same pruning principle holds in the central nervous system is unclear given its complexity.

In a recent study (Buchanan et al. 2021) in P36 mouse primary visual cortex layer 2/3, oligodendrocyte precursor cells were found to engulf terminal axon branches and included numerous phagolysosomes, suggesting their substantial contribution to the refinement of

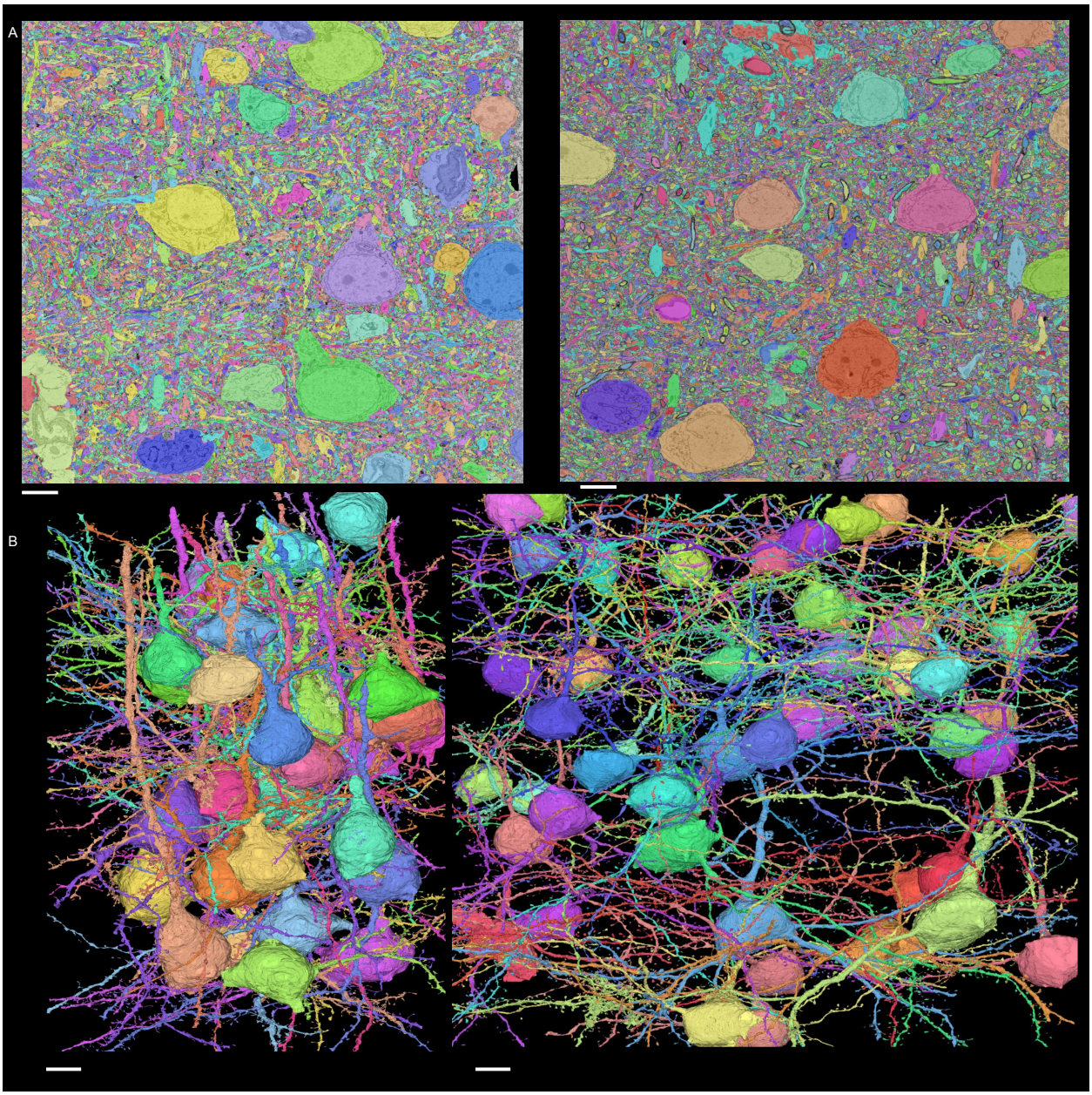


Figure 4.2: Demonstration of 3D Segmentation: (A). Electron microscopy images overlaid with segmentation from flood-filling network at $12 \times 12 \times 40$ nm resolution. (B). 3D rendering of proofread reconstructions(the proofreading process is further discussed in methods). all images oriented such that apical dendrites extend upward, all scale bars 10 μm

neuronal circuits during the critical period by pruning excess axons.

In another recent connectomics study in the somatosensory cortex by Gour et al. 2021. The authors examined axons in the primary somatosensory cortex(S1) from age P7 to P28.

They observed an increase in axonal synapse density and an increase in preference towards soma or apical dendrite from P9 to P14, suggesting an increase in synapse density but did not report the pattern on dendrites. Also, the studies focused primarily on inhibitory axons and did not show the development of spine synapses, which are primarily excitatory.

In this study, we try to address the question of synapse density change during the critical period with a focus on dendrites instead of axons and ask what the overall synapse distribution patterns differ and whether pruning is the dominant trend in cortex development.

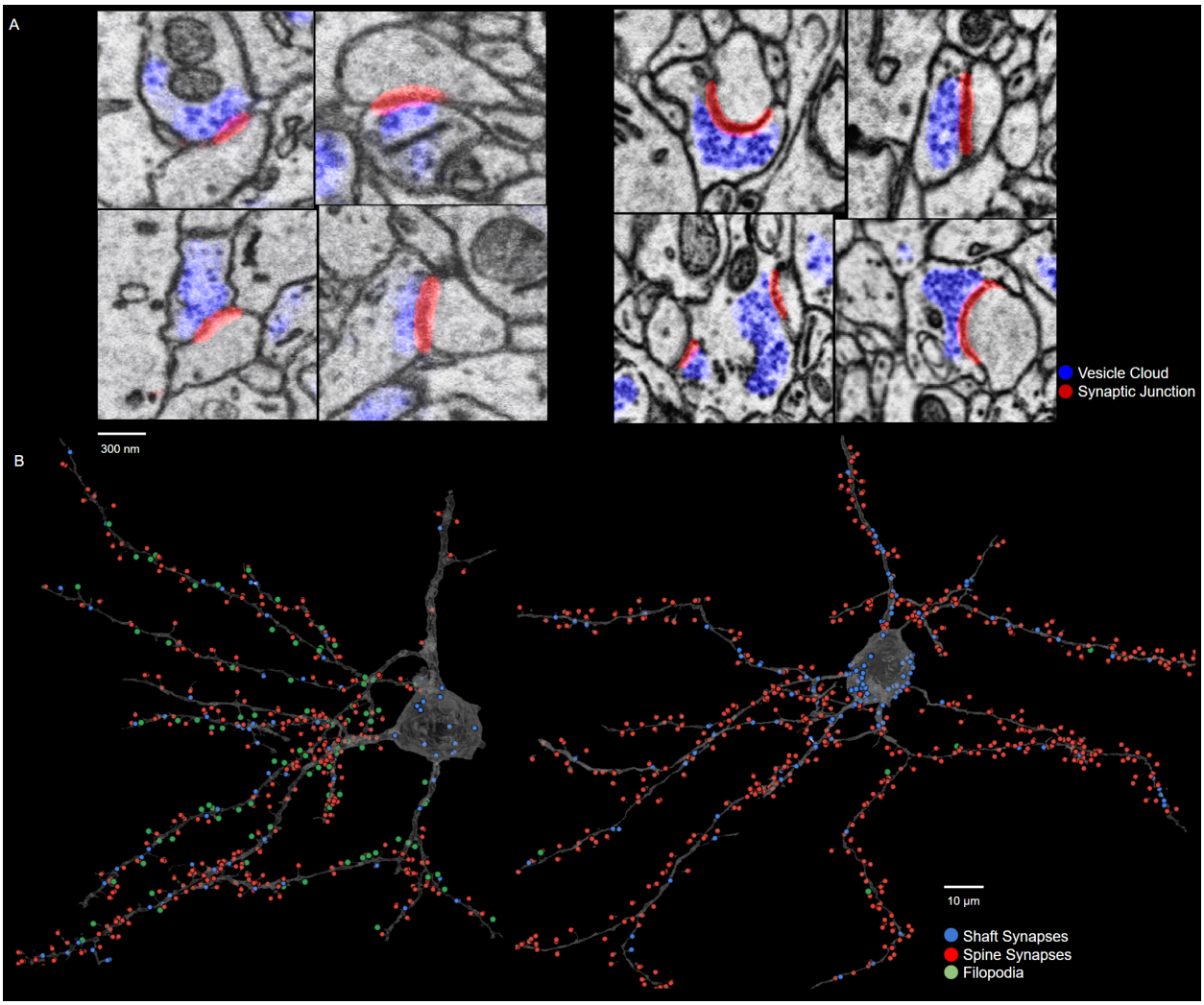


Figure 4.3: Demonstration of Synapses (left column: P14, right column: P105): (A). Electron microscopy images overlaid with automatically predicted vesicle cloud and synaptic junctions. (B). Detected synapses(manually proofread ones are also included).

In our datasets, we first observed a significant increase in synapse density in the P105 sample vs. the P14 sample, contrary to the hypothesis that synapses are predominantly pruned in the central nervous system, as implied by studies at neural muscular junctions. Among the 88 dendrites we thoroughly proofread from P14 data, we observed a mean synapse density of 0.43 ± 0.11 $synapse/\mu m$ and for 107 dendrites in P105, 0.69 ± 0.16 $synapse/\mu m$, amounting to 60.4% increase (MU Test pvalue $6.17e-26$ (Fig. 4.4left). Overall, the cable length fully annotated in P14 data reaches 15.4 mm and in P105 reaches 19.5 mm.

On top of that, we also manually annotated filopodia along these dendrites and found significantly higher density in P14 data, while in P105, filopodia are extremely rare: 0.075 ± 0.038 in P14 and 0.005 ± 0.008 in P105 (MU Test pvalue $4.43e-32$) (Fig. 4.3B, Fig. 4.4 middle). We used the distinct morphological criteria of filopodia as long protrusions from dendrite that do not form a synapse.

We also proofread synapses onto soma and found 18.31 ± 12.15 in P14 and 42.85 ± 28.80 in P105, marking a 134% increase. (Mann-Whitney U test pvalue $2.65e-8$), fitting what was described in (Gour et al. 2021).

Combining the observations, we could establish a picture of the overall trend of synaptic development. The synapse density increases dramatically through the critical period, and filopodia are ubiquitous at the beginning of eye-opening but largely disappear in the adult visual cortex.

The connectomics data provide new evidence suggesting synaptogenesis from filopodia as the dominant trend in postnatal cortex development in the critical period and significantly outweighs pruning.

4.3.3 Increase in synapse size

A major advantage of connectomics is its ability to resolve ultra-structural neural morphology. With a combination of automatic mask prediction and manual correction, we were able

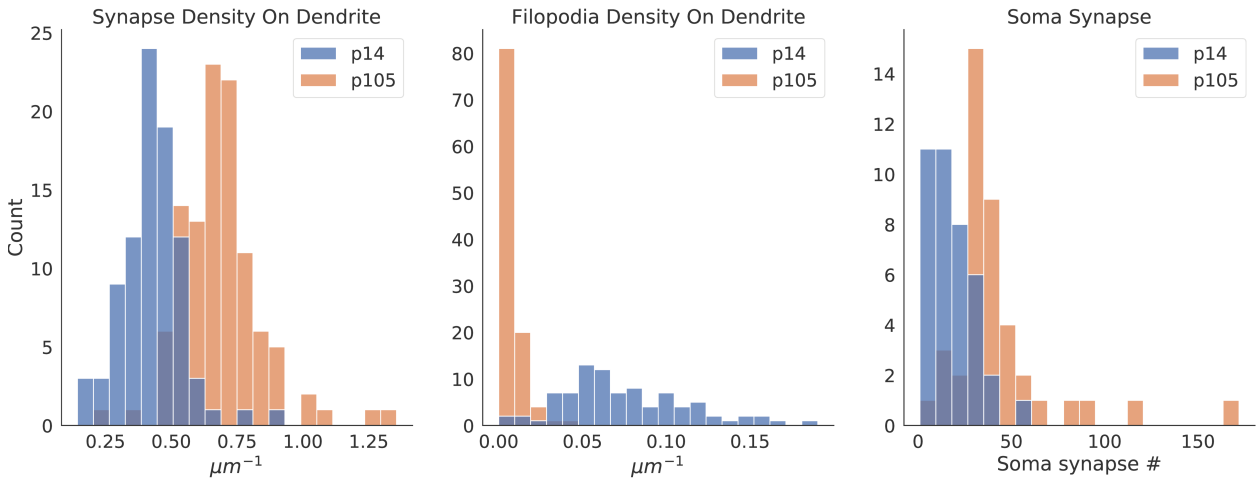


Figure 4.4: Synapse Density Statistics: (left). Synapse density along dendrites, data point number: p14: 88, p105: 107 (middle). Filopodia density along dendrites, data point number: p14: 88, p105: 107 (right). Soma synapse count, data point number: p14: 39, p105: 41

to quantitatively measure each synapse for its vesicle size, synaptic junction size, and spine head size(more details on how to detect spine head in the methods section).

We compared the distribution of these metrics at P14 and P105 and observed a significant increase in all three aspects: 18.1% in spine size, 66.5% in synaptic junction size, 14.8% in vesicle cloud size (all p ¡ 1e-5). Suggesting an overall maturation of synapses. All three metrics follow a roughly log-normal distribution, as previously seen in (Jorgen Kornfeld et al. 2020; Ofer et al. 2021). Overall, the increase in synapse size on these metrics suggest stronger and more mature synaptic connections, which is expected from the increased neural activity after eye opening.

4.3.4 Shaft vs. spine synapses

The next question we ask is whether the increase in synapse density is uniform. It has been discovered from previous studies with light microscopy that maturation of inhibitory synapses plays a major role in determining the beginning and end of the critical period, and excitatory/inhibitory ratio shift could play a major role in the stabilization of circuits.

We first classified synapses into spine versus shaft using its distance to dendritic branch as

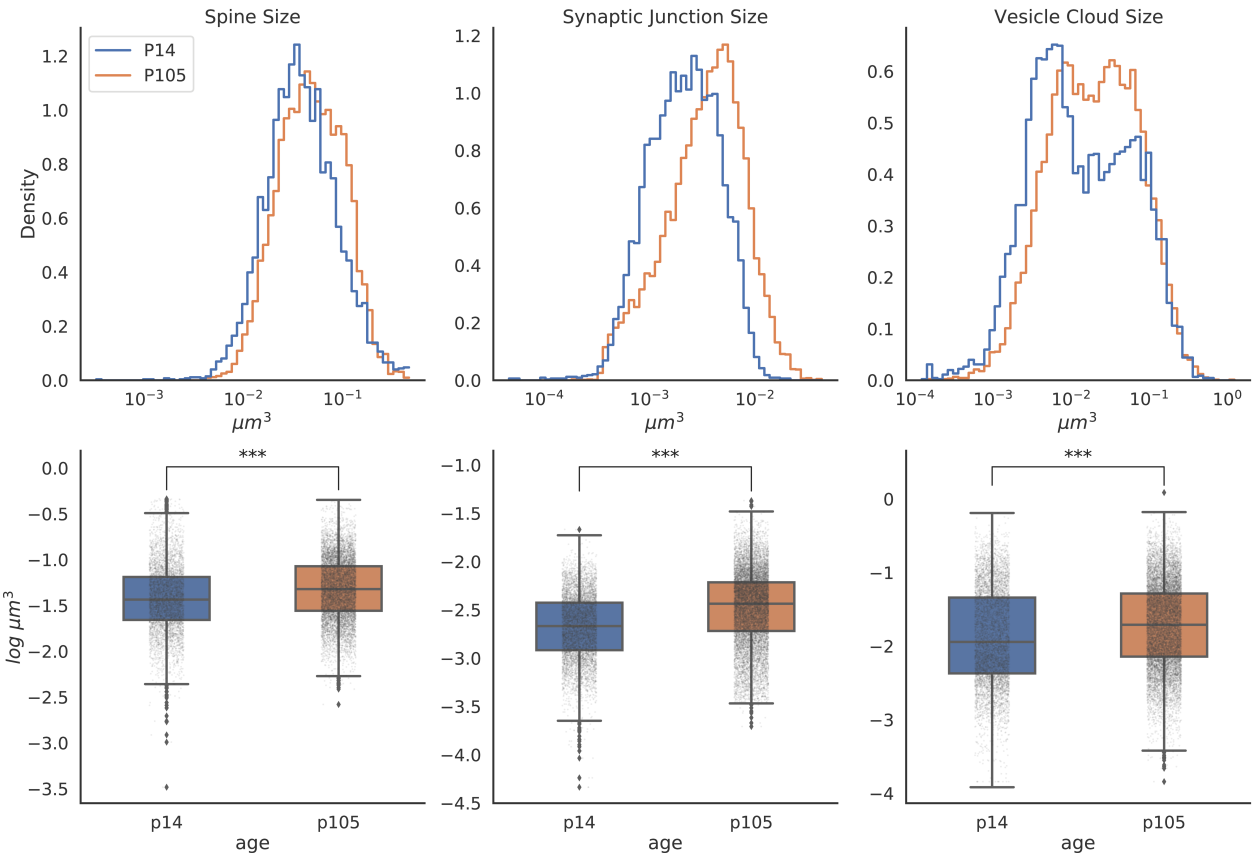


Figure 4.5: Synapse Size Statistics: (upper row). Dendritic Spine, Synaptic junctions, vesicle clouds show log-normal distribution, and in all three categories, P105 has larger synapses. number of observations: P14: spine: 5204, synaptic junctions, 6732, vesicle cloud: 6780, P105: spine: 9776, synaptic junction, 13478, vesicle cloud, 13500 (lower row). The same statistics are shown in box plots. All three categories have $p < 0.0001$.

heuristics (Fig. 4.17). All soma synapses are considered shaft synapses. For each synapse, we find its nearest skeleton node and find its path distance to soma. We then plotted synapse distribution against path distance to the soma (Fig. 4.7A) and show that P105 has a significantly higher shaft synapse ratio in perisomatic dendrites. (examples in Fig. 4.7B). Excluding somatic synapses, the shaft-to-spine synapse ratio drop from 2:1 to 1:4 in P105, about 30μm away from soma, while P14 drops from 1:1 to 1:4. This significant shift provides evidence for the role of PV(+) inhibitory neurons, this significant shift is consistent with the role of PV(+) neurons during the critical period of visual cortex development.

4.3.5 Evidence for critical period synapse development pattern

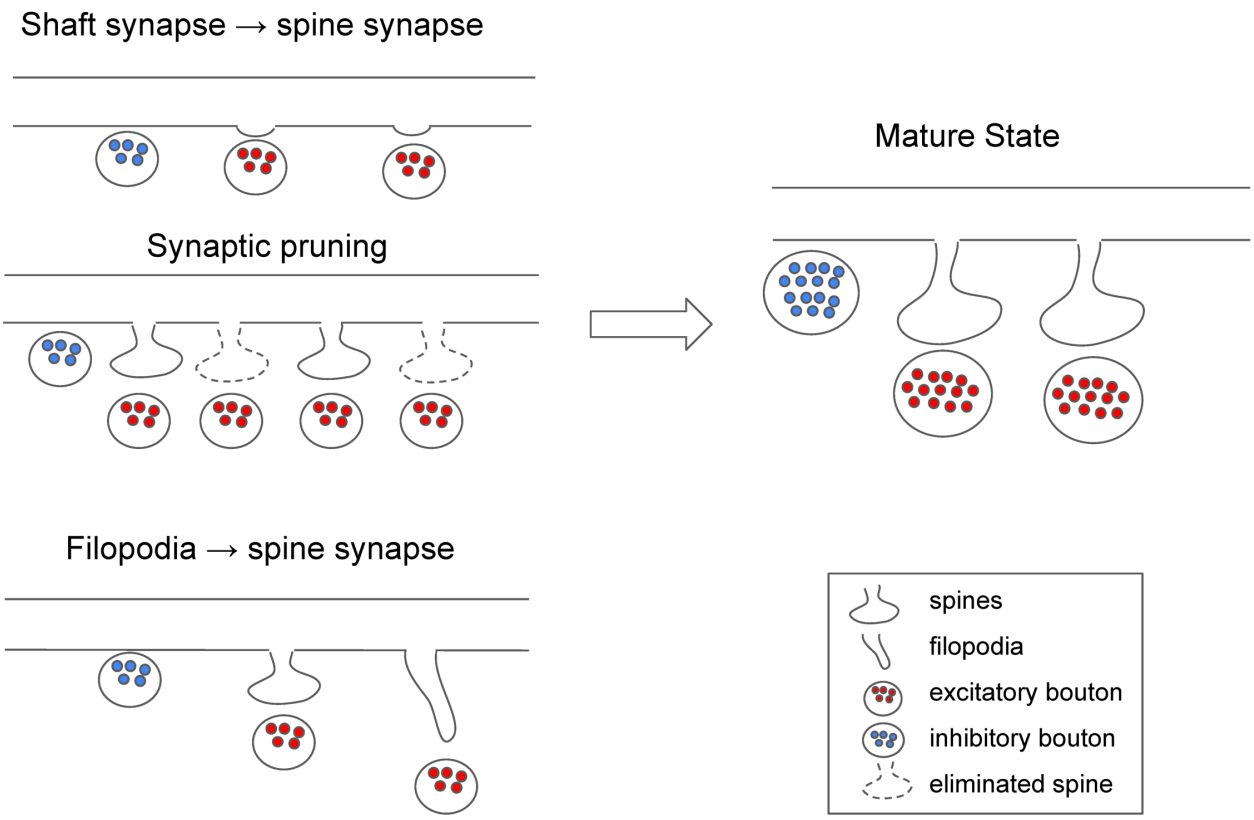


Figure 4.6: Different Synapse Rearrangement Models:

Several potential mechanisms of post-natal synapse rearrangement have been raised and discussed (Yuste and Bonhoeffer 2004):

shaft to spine The Miller/Peters model hypothesizes that axons first form shaft synapses at early postnatal ages and induce spine growth to form spine synapses (Miller and Peters 1981) (Fig. 4.6, first model). This has been challenged by the lack of direct observation of the process and that dendritic spines could spawn spontaneously without pre-synaptic contact. If the model is true, we expect a higher rate of shaft (or stubby) synapses in P14 vs. P105. However, we did not find that in our data; on the contrary, the shaft synapse rate increased from P14 to P105, suggesting this is not the likely model in critical period development of the visual cortex.

pruning Pruning has been suggested in Purkinje cell and neuromuscular junctions as the primary mechanism for circuit maturation. A recent study in the P36 mouse visual cortex showed that oligodendrocytes were involved in pruning. For the same mechanism to hold in the central nervous system, we would expect to observe a significantly higher number of weak synapses in P14 compared to P105, which is not the case.

Synaptogenesis with filopodia The filopodia model is, however, strongly supported by our data. P14 filopodia density is overwhelmingly higher than P105, consistent with previous studies with light microscopy (Konur and Yuste 2004). The decrease in filopodia density and an increase in spine synapse density suggest large-scale filopodia maturation into spines.

The increase in the soma and perisomatic shaft synapses from P14 to P105 is also noted. To further understand the mechanism of this increase, we demonstrate a manually agglomerated axon that spans half the volume and makes synapses preferentially onto somatic and peri-somatic regions.

4.3.6 Mitochondria size development and correlation with synapse density

The role of mitochondria in the process of development is also of great interest. Previous studies suggest mitochondria play a crucial role in sustaining long-term plasticity (Divakaruni et al. 2018; Rangaraju, Lauterbach, and Schuman 2019). Previously in Turner et al. 2020, it was shown for the first time with connectomics tools that in basal dendrites of mouse visual cortex layer 2/3 at age P36, there is a positive correlation between synapse density and mitochondria's coverage ratio.

With our reconstructions, we could study how mitochondria develop throughout the critical period and cross-check with the results from layer 2/3 in the previous study (Turner et al. 2020). Our automatic mitochondria segmentation with UNet + connected component returned 398278 instances in P14 and 533019 in P105. (only those $1000 < \textit{voxel count} <$

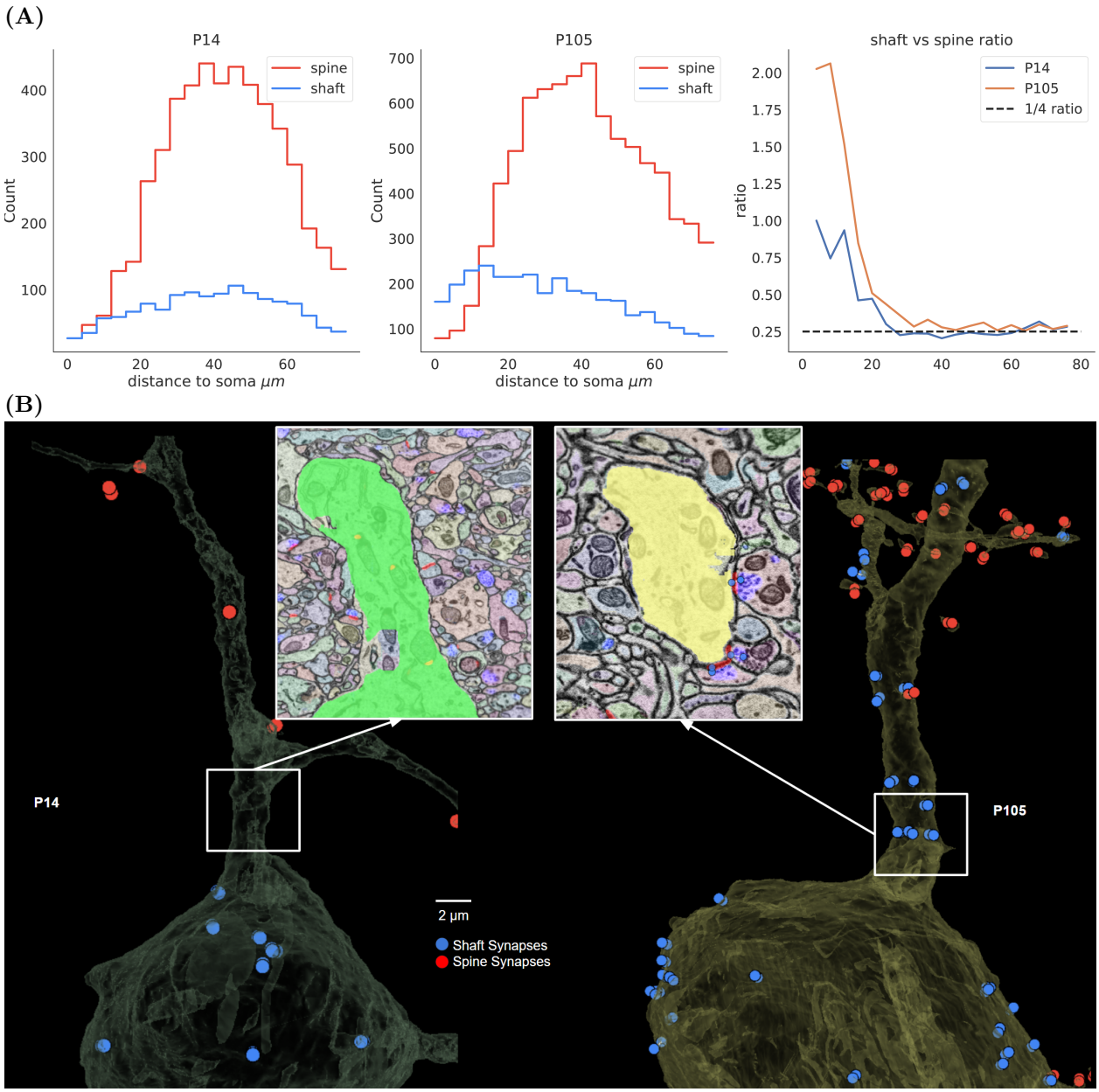


Figure 4.7: Shaft vs Spine Synapses. (A). (left) Histogram of the total number of synapses at different distances to soma in p14, $4\mu m$ per bin, all dendrites grouped; (middle) Same histogram for P105; (right) Shaft to synapse ratio. P105 shows a significant increase in perisomatic shaft synapse versus P14, while further away from soma, both ages approach 1/4 shaft to spine synapse ratio. (B). Examples of apical dendrites in P14(left), P105(right) are demonstrated, where P105 shows significant enrichment in perisomatic shaft synapses.

50000 are considered valid). Although it was not possible to proofread the prediction over full volume, we expect the mask prediction to be highly accurate while many split errors

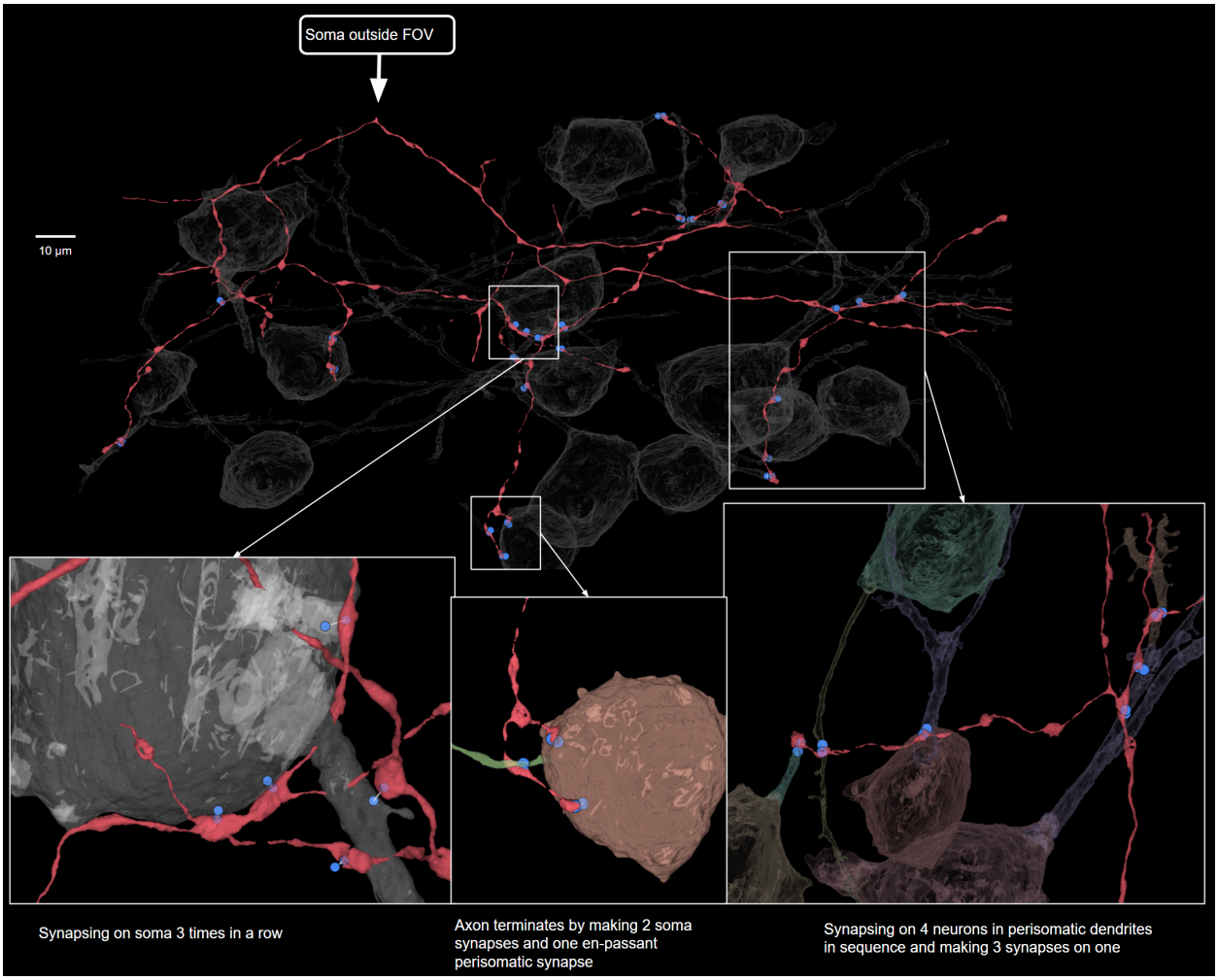


Figure 4.8: Inhibitory Axon Example (main). An inhibitory axon is manually agglomerated. It peruses the volume and makes 24 synapses at somatic or peri-somatic sites with 14 cells. The somata of this axon are outside the field of view, and its location is deduced from its branching pattern. (inset left). The axon makes 3 synapses onto one soma (inset middle). The axon terminates by making 2 synapses onto soma and one en-passant synapse on dendrites (inset right). The axon makes peri-somatic synapses with 4 separate cells in sequence and making 3 consecutive en-passant synapses with one dendrite.

remain within the instance segmentation due to thin necks of longer mitochondria. We observed a significant 78.4% increase in mitochondria size in P105 than in P14 ($p \approx 0$, Fig. 4.10(A) left), primarily in length, suggesting maturation and genesis of mitochondria within dendrites, which can be seen in a $2048 \times 2048 \times 512$ voxel subvolume in (Fig. 4.9B).

We then compared mitochondria density along dendrites. For each annotated dendrite,

we define "mitochondria density" as the total volume of all the mitochondria it contains divided by its total cable length. We observed a 66.7% higher mitochondria density in P105, as shown in Fig. 4.10(A) right $p \approx 1e-20$. An example comparison in Fig. A also demonstrates the overall higher coverage of mitochondria within dendrite (Fig. 4.9A).

We also ask whether the change in mitochondria coverage is correlated with synapse density. We use "mitochondria coverage", similar to the definition in Turner et al. 2020, as the percentage of skeleton nodes which has mitochondria nearby within a small radius. We then compared the correlation between mitochondria coverage against synapse density. We found that in P105, the correlation is considerably stronger, with Pearson's r = 0.495, p = 5.8e-8 , while in P14, r = 0.125, p = 0.246 (Fig. 4.10(B) left). We also broke down into Apical and Basal and observed that the correlation is much stronger in basal in both cases(Fig. 4.10(B) middle & right). This suggests that the correlation between synapse and mitochondria is not yet strongly correlated at the beginning of the critical period, while gradually strengthened during the course of development.

Significant increase in aspinous dendrite mitochondria coverage We also observed a special case of mitochondria distribution in aspinous dendrites, whose synapses are primarily on shafts. Most notably, in P105 aspinous dendrite, the mitochondria could almost cover the entire dendritic branch(92%), as shown in (Fig. 4.11lower), while a P14 aspinous dendrite only covers 64%. In terms of mitochondria density, the P14 aspinous dendrite is 44.5% higher than average, and the P105 one is 135% higher than average, suggesting that in P105, the mitochondria density disparity in aspinous neuron is even more significant. Although aspinous dendrites are too rare within the annotated set to be statistically significant(2 in P14 and 5 in P105), the few cases we observed uniformly showed a strong pattern.

The dramatic increase in mitochondria density in aspinous neurons could partially be explained by the higher synapse density, which requires more Ca2+ and has a higher metabolic load. For the two examples shown, both have higher synapse density than average(2.56 times

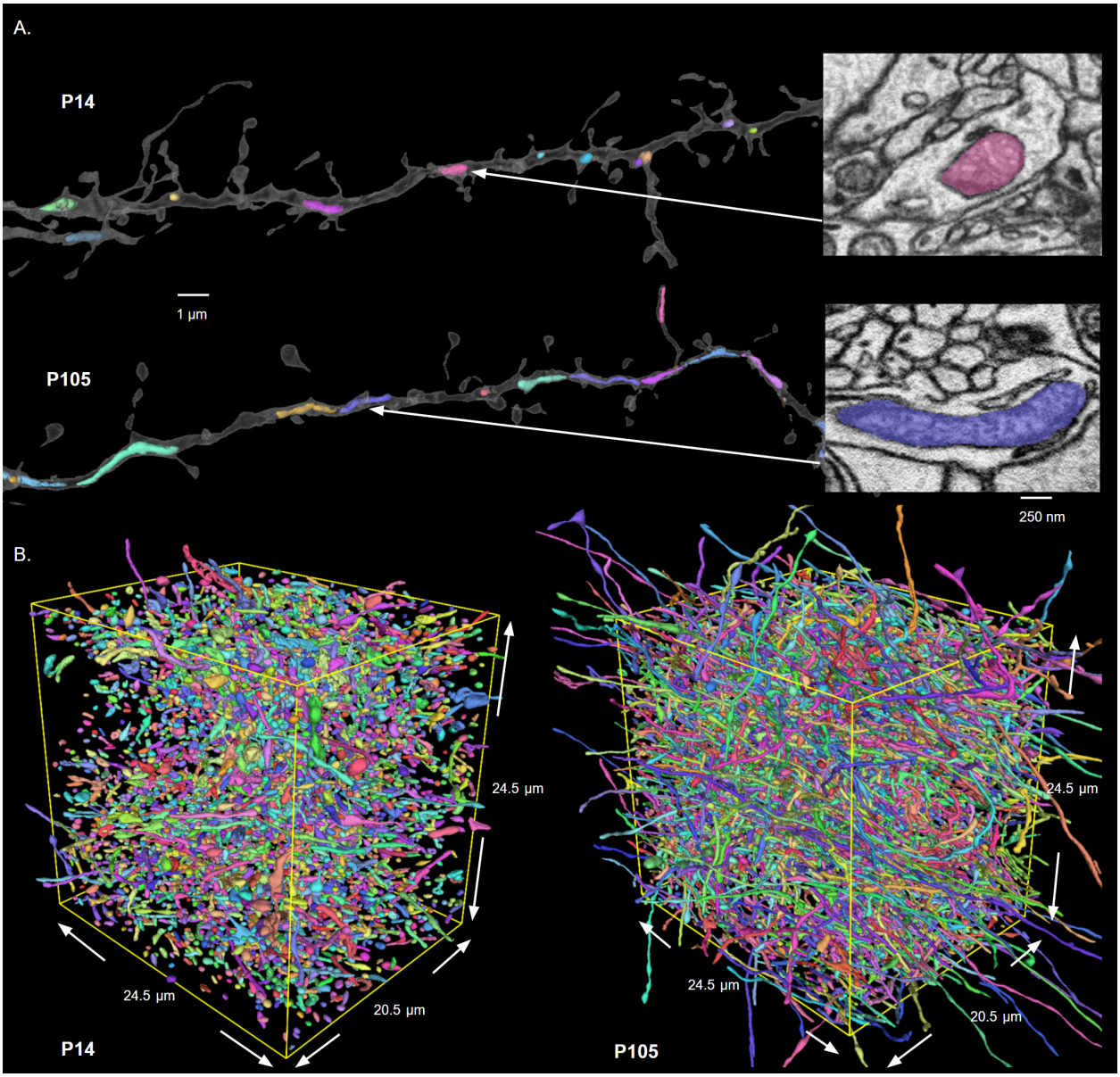


Figure 4.9: Mitochondria Reconstruction: A. Examples of dendrites in P14 and P105 overlayed with mitochondria, P105 has larger synapses. B. Examples of subvolumes filled with mitochondria within FOV.

in P14, 1.57 times in P105).

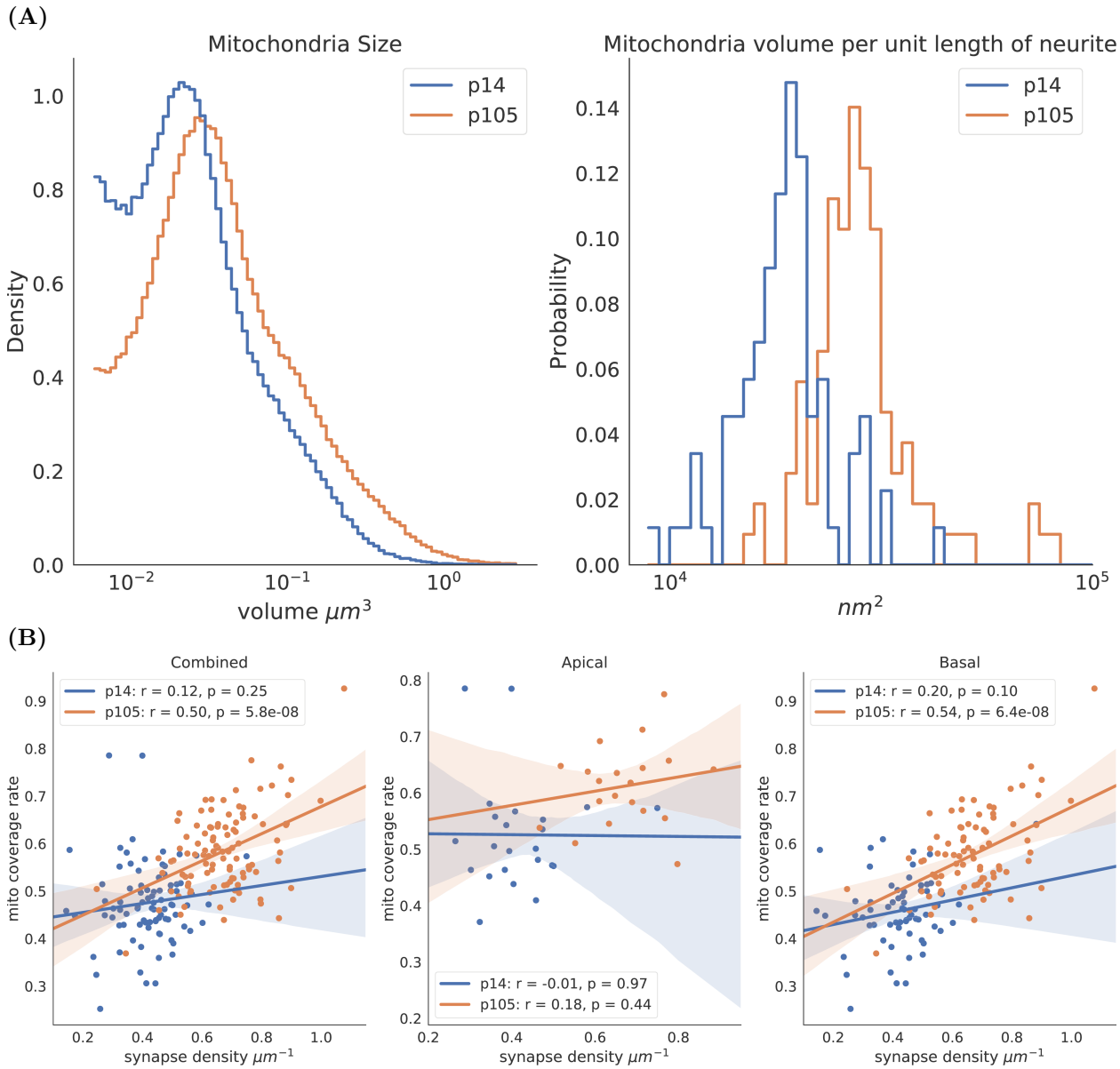


Figure 4.10: Mitochondria Statistics. (A). (left) Histogram of all mitochondria sizes (right) Histogram of mitochondria density within dendrites (B). (left) Correlation between synapse density and mitochondria coverage rate in all dendrites. (middle) Same as (left) but with apical dendrites only. (right). Same as (left) but with basal dendrites only.

4.4 Discussion

Achievements We presented the first juvenile(P14) connectomics sample of mouse V1 at the beginning of the critical period with saturated segmentation and synapse prediction,

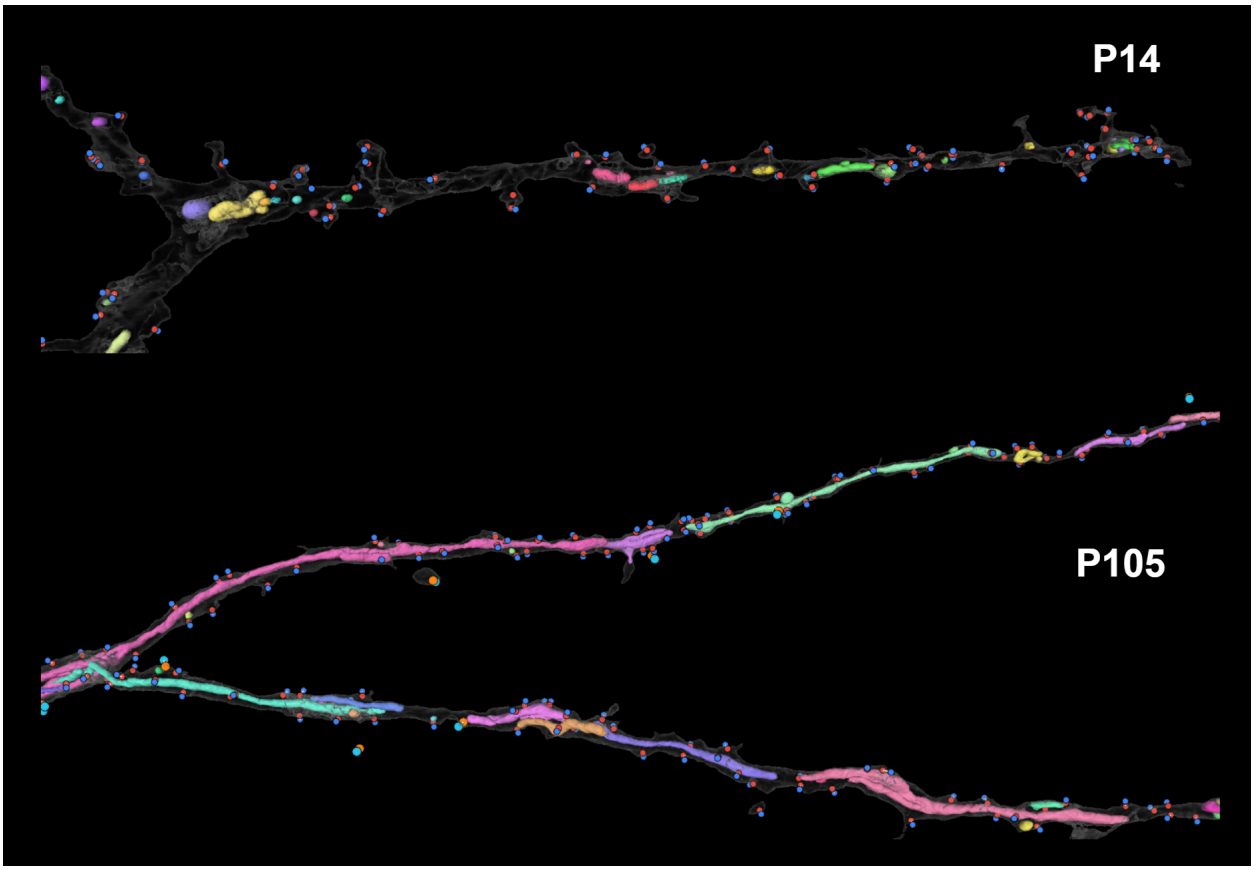


Figure 4.11: Aspinous dendrite mitochondria: Example aspinous dendrites from P14 and P105, showing higher coverage of mitochondria, especially in P105.

providing rich information about the neural anatomical substrate of a developing circuit. We believe this would be beneficial for the field in complementing functional studies with light microscopy. Comparing the juvenile versus adult sample allows developmental models to be tested at unprecedented scale and precision.

We used a connectomics toolbox to determine various ultrastructural changes of synapses and mitochondria throughout the critical period. From P14 to P105, we observed a widespread increase in synapse density, synapse size, perisomatic shaft synapse ratio, and mitochondria coverage/synapse density correlation. The overwhelming synapse addition and maturation from P14 to P105 and a sharp drop in filopodia density suggest that filopodia growth-based synaptogenesis overwhelms pruning as the dominant pattern in critical period synapse de-

velopment.

The structural change in the critical period is crucial in optimizing the visual circuity to adapt to the environment. Our connectomics reconstruction would be of great value for the field in studying the organizational principle and could potentially be used to inspire future machine learning architectures.

Limitations and future directions In this study, we focused on dendrites proximal to soma in two age points. We are primarily limited by data acquisition time cost, automated system accuracy, and manual proofreading resources. In the future, we would like to address these limitations and have more time points across critical periods, and reconstruct larger volumes, increasing the temporal and spatial scale of the study.

One obvious direction is to scan larger samples. Apart from investing more scanning time and use more powerful microscopes like Multibeam SEM, there are several technical options to improve efficiency in data acquisition. We are exploring potential paths like using aluminum tape with better conductance or swapping out carbon with platinum nano-particles in wafer coating. These technical improvements aim to increase the electron signal to allow the use of an in-lens detector, which is orders of magnitude faster than the backscatter detector we are currently using. Having a stronger signal also means we could decrease pixel dwell time or scan at a lower resolution to achieve similar quality.

Aside from increasing scanning size, we could also use APEX staining to include long-range information that otherwise could not be captured within the volume. For example, in a recent study, APEX labeling was used to mark thalamocortical axons (Sampathkumar et al. 2021; Martell et al. 2017), which would be extremely useful for identifying the sensory cortex that receives input from the thalamus, like V1 layer 4 we are studying. However, it is still challenging to perform APEX staining in younger animals.

On the computational front, the machine learning systems would naturally benefit from more training data. From our experience, we will need much higher accuracy in segmentation

to drastically reduce proofreading efforts, which is essential for more complex connectomics analysis. The datasets we present in this paper are already impossible for human annotators to review thoroughly in a reasonable amount of time. Beyond the scope of this paper, the state-of-the-art petascale segmentation of the human cortex sample (Shapson-Coe et al. 2021) would require astronomical manual effort to proofread fully. Therefore it is an ongoing effort to push segmentation accuracy as high as possible, meanwhile adding automatic error detection mechanisms (H. Li et al. 2020). The amount of manual proofreading scales inversely with the accuracy of automated systems. With better segmentation, more axons can be proofread and included to study the topological changes at the network level.

4.5 Methods

4.5.1 Sample Preparation and Data Acquisition

Animals

Two wild-type male mice (C57BL/6) at age P14 and P105 were used in this study. The perfusion procedures were followed according to animal regulations at the University of Chicago's Animal Resources Center (ARC) and approved IACUC protocol.

Perfusion and Sample Staining

The brains of the two mice were prepared in almost the same manner as previously described in (Hua, Laserstein, and Helmstaedter 2015; Wildenberg et al. 2020). Briefly, the anesthetized animal was first transcardially perfused with 0.1 M Sodium Cacodylate (cacodylate) buffer, pH 7.4 (Electron microscopy sciences, EMS), followed by a fixative containing 2% paraformaldehyde (EMS), 2.5% glutaraldehyde (EMS) in 0.1 M Sodium Cacodylate (cacodylate) buffer, pH 7.4 (EMS). We used 10 ml of cacodylate buffer followed by 20 ml of fixative. The brains were removed and placed in fixative, for P14, 6 hours at 4^{o}C, for P105, at least 24 hours at 4^{o}C. For each sample, a $300\mu m$ vibratome section encompassing V1 was removed and put into a fixative for 24 hours at 4^{o}C. Mouse V1 was identified using the Allen Brain Institute reference atlas (atlas.brain-map.org), and a $1\ mm \times 2\ mm$ piece of tissue was excised from the coronal section encompassing all cortical layers of V1. V1 brain slices were then rinsed in cacodylate buffer at room temperature and stained sequentially with 2% osmium tetroxide (EMS) in cacodylate buffer, 2.5% potassium ferrocyanide (Sigma-Aldrich), thiocarbohydrazide, unbuffered 2% osmium tetroxide, 1% uranyl acetate, and 0.66% Aspartic acid buffered Lead (II) Nitrate with extensive rinses between each step except potassium ferrocyanide. The tissues were then dehydrated in ethanol and propylene oxide and infiltrated with 812 Epon resin (EMS, Mixture: 49% Embed 812, 28% DDSA,

21% NMA, and 2.0% DMP 30). The resin-infiltrated tissue was cured at 60^{o}C for 3 days.

Slicing and Wafer preparation

Using a commercial ultramicrotome (Powertome, RMC), the cured blocks were further trimmed to $0.8 \times 1.5\ mm^2$ rectangles. With the 1.5 mm orientation covering cortical layers. Equipped with an Ultra 35 diamond knife(Diatome), 1500 and 3000 40nm thick sections were collected from each block on polyamide tape (Kapton) using an automated tape collecting device (ATUM, RMC) before the cutting process becomes unstable. The collected tapes are then cut into strips and assembled on 4-inch silicon wafers as previously described in (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015). 300 sections are mounted on each wafer. To enhance the EM signal, the wafers are coated with carbon.

SEM data acquisition

The serial sections were acquired semi-automatically using backscattered electron detection with a Gemini 300 scanning electron microscope (Carl Zeiss) equipped with ATLAS software for automated wafer imaging. In the first round, a low-resolution (100 nm), large-coverage ($400 \times 400\ \mu m^2$ patch is scanned for each section focusing on layer 4. In the second round, a focused region-of-interest(ROI) is manually selected from the low-resolution scan using a fixed set of cell bodies as fiduciary points. Once all regions on a wafer are determined, an automatic acquisition session starts and iterates over each ROI at 6 nm resolution at 1-microsecond pixel dwell time. For the P14 sample, two 16000×14000 pixel tiles over 1312 sections were collected; for p105 samples, four 15000×15000 pixel tiles over 913 sections were collected. In both acquisition sessions, there is a 15% overlap between neighboring tiles. Each acquisition session took 300 hours of imaging time. After the acquisition, tiles are manually inspected, and blurry or off-region sections are reacquired as needed.

4.5.2 Data Analysis

The data analysis pipeline includes various open-source software packages and a set of tools we developed to perform 3D reconstruction of neural tissue.

Processing connectomics data has been a challenging and actively researched problem over the last few years for a few reasons:

1. The data size is significantly larger than the RAM size of a typical workstation. Apart from posing considerable logistic challenges in data storage and management, it also requires non-trivial software engineering efforts to process in a reasonable time frame.

2. Different tools specialize in different pipeline steps, have different data format requirements, and are designed with different infrastructures in mind.

3. The state-of-the-art methods nowadays ubiquitously use computer vision algorithms based on deep learning as the core engine, requiring a tremendous amount of manual labels as training data. The automatic predictions and outputs often need to be further inspected by human annotators to eliminate remaining errors before biological statistics can be performed, further increasing the amount of human effort needed.

Our data processing pipeline is built on high-performance computing infrastructure and emphasizes scalability, with Message Passing Interface(MPI) (Gropp, Thakur, and Lusk 1999) as the main parallelization paradigm. More details are explained in Chapter 2, and in this section, the focus will be on the actual execution of each step for this particular study.

Montage

The raw image collected from EM is in the form of tile grids. Each region-of-interest consists of 2 or 4 separate scans that need to be montaged into a single image. We use parallel macros based on the montage tool of Fiji/TrakEM2 (Schindelin et al. 2012; Cardona et al. 2012) with SIFT algorithm (Lowe 2004).

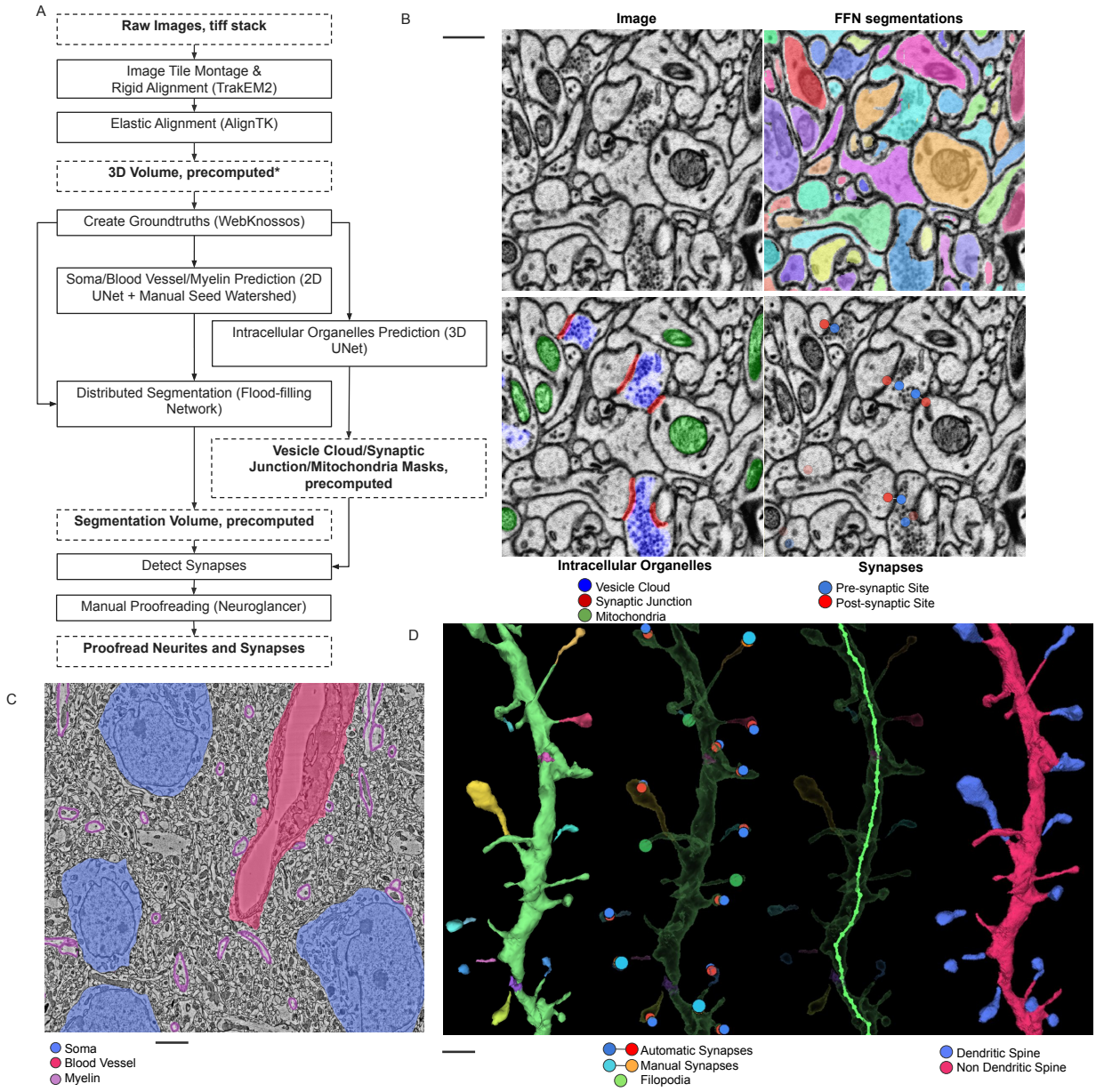


Figure 4.12: Data Analysis Pipeline: (A). Detailed workflow of connectomics reconstruction (B). (From upper-left to lower-right) Raw image, FFN segmentation, Organelle predictions of vesicle cloud, synaptic junction, and mitochondria, predicted synapses(C). Mask prediction of Soma, Blood Vessel, and Myeline(in P105 dataset). (D). (From left to right) Segmentation fragments manually agglomerated; Synapse Annotations on this piece of dendrite; Skeletonization of the main branch; Spine separation with the watershed algorithm.

Alignment

The montaged sections first go through a round of rigid alignment with Fiji/TrakEM2, forming a coarsely aligned stack. The images are then cut in x/y dimensions to eliminate

empty backgrounds. Severe artifacts like large chunks of carbon dust, cutting artifacts left by a diamond knife are marked manually and replaced by 0 value in Fiji.

The elastic alignment is performed with AlignTK on a supercomputer cluster (Arthur W Wetzel, Hood, and Dittrich 2013; Vescovi, H. Li, et al. 2020). The idea was to model the image stack as a grid of nodes connected by springs that elastically interact with neighboring nodes within the same section and across the z dimension. Specifically, a warping map between each pair of adjacent sections is first calculated to determine the non-linear deformation needed to register the two. Then all the pairwise registration maps are taken into a global relaxation process to determine the optimal absolute positioning of each node to match the images into one coherent stack while retaining the original image geometries. The deformation map is then applied to the original image stack and created an aligned tiff stack.

Segmentation

The segmentation pipeline is centered around one of the state-of-the-art methods Flood-Filling Network(FFN) (Michał Januszewski, Jörgen Kornfeld, et al. 2018). A forked version of FFN based on the open-source repo (github.com/Hanyu-Li/ffn) is built with additional functionalities like multi-GPU training with Horovod (Sergeev and Balso 2018), distributed training based on MPI, and integration with cloud-based data formats used in Neuroglancer (Maitin-Shepard n.d.) and Cloudvolume (W. Silversmith n.d.[a]). Additionally, we developed a UNet engine, "EM_mask" (H. Li 2020a), which performs tissue classification for EM datasets, and an open-source suite of tools to work with EM data packaged in (H. Li 2020b).

Preprocessing The aligned image stack is first converted to "precomputed" format, a data format designed and popularized by Google (Michał Januszewski, Jörgen Kornfeld, et al. 2018) and further supported by Seung Lab (W. Silversmith n.d.[a]) to address large

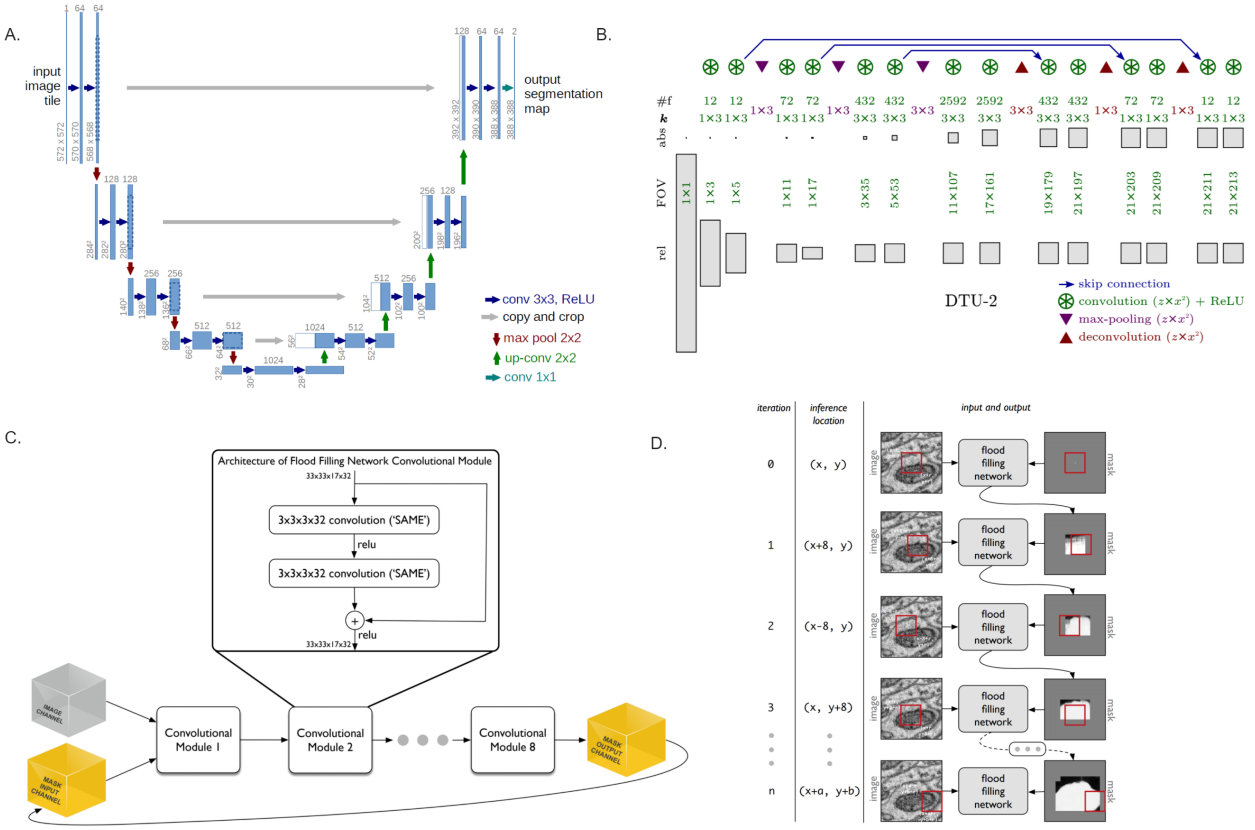


Figure 4.13: Neural Network Architectures: A. Classic 2D UNet (Ronneberger, Fischer, and Brox 2015), used for soma, blood vessel, and myelin mask prediction B. Distance-transform UNet (DTU) (Heinrich et al. 2018), used for synaptic junction, vesicle cloud, and mitochondria mask prediction C. Flood-filling network module (Michał Januszewski, Maitin-Shepard, et al. 2016). D. Flood-filling network mechanism (Michał Januszewski, Jörgen Kornfeld, et al. 2018)

3D volumetric data challenges. Essentially, the tiff stack is converted to a multi-resolution 3D volume chunked into mini files that can be served through HTTP on-demand, allowing remote visualization with Neuroglancer within a web browser and efficient access of arbitrary chunks. A CLAHE("contrast limited adaptive histogram equalization", Zuiderveld 1994) filtered version is also created in order to strengthen the contrast between the cell membrane and intracellular space and improve the quality of FFN segmentation downstream.

Create Tissue Masks The first step was to produce masks of tissue types that should be excluded in the FFN segmentation process. In particular, soma and blood vessel tend to

create merge errors with their vastly larger size than neurites.

For the purpose of soma detection, the original resolution $6 \times 6 \times 40\ nm^3$ would be too high. Instead, the 4x downsampled volume at $24 \times 24 \times 40\ nm^3$ resolution is used. To generate training data, 10 sections uniformly sampled from within the stack are chosen, and their soma and blood vessels are thoroughly annotated(each instance of soma is assigned with a unique id paintbrush) with WebKnossos (Boergens et al. 2017), constituting 1% of the complete dataset. Borders between neighboring soma are assigned a larger weight (10.0, versus 1.0 baseline) to penalize the neural network on errors predict borders. The instance annotations are binarized to be used as labels. The training data are then packed into one h5 file with "image", "label" and "weight" datasets with equal dimensions.

The backbone is a 2D UNet with the architecture described in the (Ronneberger, Fischer, and Brox 2015) paper (Fig. 4.13A), with Adam optimizer(learning rate 0.0001, beta1 0.9, beta2 0.999, epsilon 1e-08), and random rotation augmentation.

Prior to training, random coordinates are generated with a balanced sampling strategy described in Michał Januszewski, Jörgen Kornfeld, et al. 2018 where the fraction of foreground pixel within the field-of-view is uniformly distributed between 0.0 and 1.0.

At training time, 512×512 pixel patches of image and label centered around the random coordinates are fed into the network and trained for 6-12 hours until saturation.

At inference time, the complete dataset is broken down into overlapping patches of 512×512 and distributed to multiple GPUs and the output logits are reassembled into a single "precomputed" volume. The output volume is then Gaussian-blurred across z dimension to smooth out irregular boundaries predicted by the 2D network. Blood Vessel mask is created in a similar fashion. Specifically for P105 data, myelination constitutes a significant portion of the volume, which is also predicted with the same method(myelination in P14 data is extremely rare).

Remaining imaging artifacts with brightness and contrast that deviate from those of

neurites are manually detected and painted with Webknossos.

The background mask is generated by thresholding the greyscale(0-255) image with a near 0 value and is then labeled into connected components, and regions larger than 8000 pixels are considered background.

Eventually, we merge all these masks into one categorically coded volume, in which 0: valid neurites, 1: soma, 2: blood vessel, 3: artifacts, 4: background, 5: myelin, so that in the next stage, only valid neurites are considered.

Generate initial segmentation As mentioned above, soma and blood vessels, if not preprocessed, can be the major contributor to merge error in instance segmentation. It is, therefore, beneficial to pre-segment those objects at a lower resolution before performing FFN.

For soma, the center of each soma is manually annotated in Neuroglancer. Moreover, a seeded watershed is performed on the 16x downsampled soma mask at $96 \times 96 \times 80\ nm^3$ resolution volume to generate instance segmentation of soma. The watersheded result is scaled up to $24 \times 24\ nm^3$ resolution and cropped by mask volume.

For blood vessels, the instance segmentation is simply generated by connected components since different blood vessel branches are separated by large distance.

The instance segmentation of soma and blood vessel are then merged and scaled up to $6 \times 6 \times 40\ nm^3$ resolution, creating an initial segmentation. (Fig. 4.14)

Instance segmentation Segmentation is performed with Flood Filling Network. For each dataset, we created $512 \times 512 \times 128$ (33.55M) voxel annotation at $12 \times 12 \times 40\ nm^3$ resolution with Webknossos. The network configuration is the same as used in Michał Januszewski, Jörgen Kornfeld, et al. 2018 (Fig. 4.13C, D). Specifically, the FOV size is 33,33,17, with 8 residual modules, 8-pixel step size in-plane, and 4-pixel step size in the z dimension.

For training, we acquired a pre-trained checkpoint that has been extensively trained

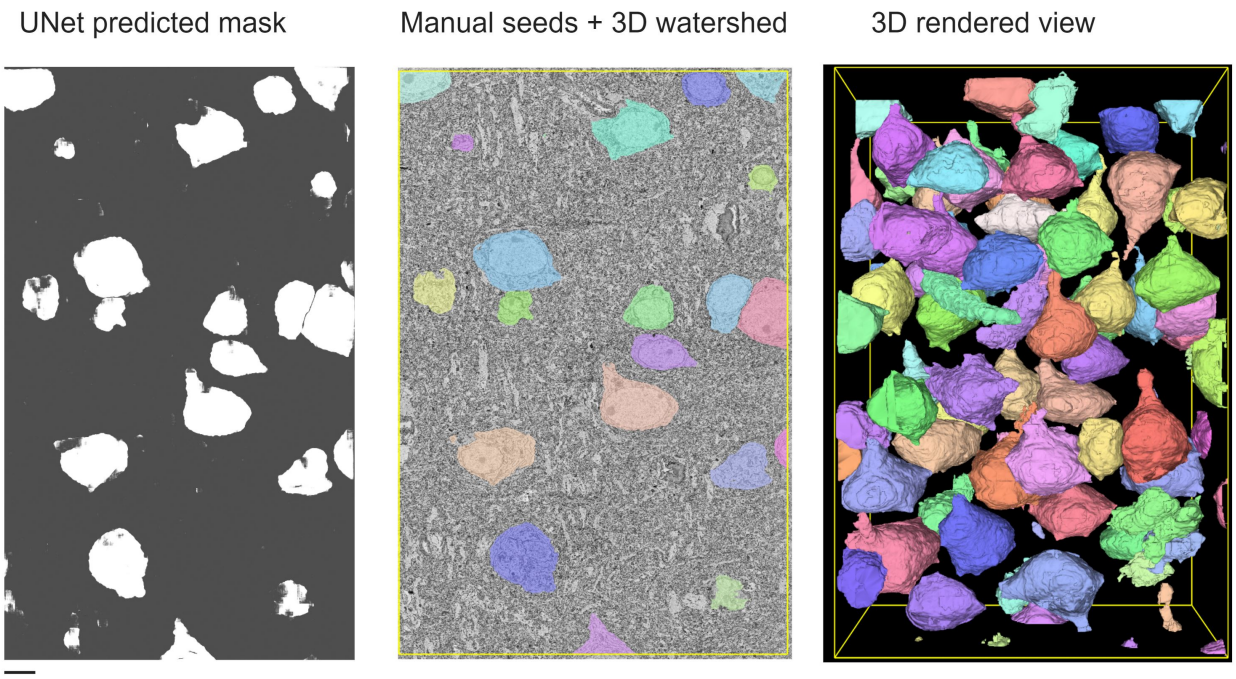


Figure 4.14: Generating Soma Instance Segmentation, (left) UNet predicted logits, (middle) Instance segmentation overlayed on image, (right). 3D rendering of soma. scale bar 10 μm

on the K11 dataset(adult mouse brain S1 region, Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015) from the authors of Michał Januszewski, Jörgen Kornfeld, et al. 2018. We then continued training on our own annotations separately for P14 and P105 datasets on a GPU cluster with the following configurations: optimizer: SGD, learning rate: 0.001. The training is sped up with multi-GPU acceleration through Horovod (Sergeev and Balso 2018). We train for ¿20 hours until saturation on training data(¿90% accuracy). In order to increase the amount of annotation efficiently, we resort to a bootstrapping technique. We run a round of inference with this network in another subvolume in a different region and generate a coarse segmentation. This segmentation has noticeably lower quality than that of training data and is uploaded back into Webknossos for further manual correction. This set is then used to continue training the checkpoint with the same configuration until saturation(¿94% accuracy).

For inference, the tissue mask generated in the previous step is provided as an exclusion

mask, and FFN is only run in regions with mask value = 0(valid neurite). Instance segmentation, including soma and blood vessel, is used to populate the initial canvas of FFN inference. For P14 data, the overall volume was split into 3618 $512 \times 512 \times 128$ voxel sub boxes with $32 \times 32 \times 16$ voxel overlap and distributed to 64 Nvidia K80 GPU workers and ran for a total of 72 hours. For P105 data, the volume was split into 2990 $512 \times 512 \times 256$ voxel boxes with $64 \times 64 \times 32$ overlap on a more powerful cluster with 64 A100 GPUs and ran for about 38 hours.

The output of distributed inference is a collection of independently segmented subvolumes. The next step is to reconcile segment IDs and assemble the subvolumes into one "precomputed" volume. Specifically, first, the segment IDs in each subvolume are offset to be globally unique(each subvolume is incremented by the maximum segment ID from the previous one). Second, for each pair of neighboring subvolumes, the overlapping region from both are cut and compared, from which overlapping segment IDs > 250 *voxels* are added into a global merge graph as an edge. Finally, each connected component of this graph is joined into one ID, and the subvolumes are remapped accordingly and written into one large "precomputed" volume.

Mesh and skeleton generation In order to render 3D objects in Neuroglancer, the object meshes are generated with marching cube algorithm using "igneous" (W. Silversmith n.d.[b]) at $24 \times 24 \times 40$ nm^3 resolution. The skeletonization is performed with the TEASER algorithm (Sato et al. 2000), implemented in "Kimimaro" (W. Silversmith and J. Alexander Bae 2020) and wrapped within "igneous".

Semi automatic correction of merge errors Unlike split errors, which can be joined easily by annotators or an automated algorithm post-segmentation, merge errors are by far the more problematic error type in EM segmentation. Common causes include 1. Cell membrane degradation/breakage over the process of tissue preservation and histology (Fig.

4.15); 2. Mis-alignment of neurites from alignment step; 3. Unaddressed imaging artifacts throwing off the segmentation algorithm.

Although a merge error hot spot is localized, its effect can propagate far beyond its local field-of-view and falsely label two or more objects with the same ID throughout the entire volume that's hard to disentangle. It is non-trivial to modify segment IDs over the entire volume, which requires significant data IO.

What further complicates the problem is the fact that merge errors are hard to detect, a merge error would usually only manifest itself after the segmentation is complete over the whole volume and a human annotator or an automatic system is able to review the 3D reconstruction (H. Li et al. 2020). Automated systems cannot achieve 100% accuracy and may have limitations in error type(for example, neurite class-based detection can only deal with axon-dendrite merge but not axon-axon or dendrite-dendrite merge). Either case, it is even more difficult to detect merge errors within FFN inference subvolumes, which typically only cover less than 10 um fragments of neurites, at which scale the morphological information would not allow meaningful detection based on automated systems (H. Li et al. 2020) and would be impractical for human annotators to scan and eliminate prior to full volume reconciliation.

We designed a pipeline to minimize manual intervention while allowing automatic error correction. Instead of relying on exhaustive inspection and paintbrush correction, we utilized the fact that a merge error is typically localized in one subvolume. In contrast, the merged objects in neighboring subvolumes are often correctly separated and can be detected retrospectively through inspecting the reconciled volume. Therefore as long as the merged object is split locally within the subvolume, the reconciled volume would be rid of the merge error.

We first use the heuristic that if two neighboring objects are segmented by FFN in one subvolume but ended up joined in the global reconciliation graph, it likely includes a merge

error. A set of candidates are generated and inspected in Neuroglancer. To aid manual annotation, a reconciled volume without segment ID remapping is assembled. (Fig. 4.15A, which creates the grid-looking segmentation).

Each candidate object is then proofread thoroughly in Neuroglancer, and line annotations are placed such that they cross the merge error "bridge"(Fig. 4.15A(3)). For each line annotation, a disk-shaped mask is created from its midpoint perpendicular to the line direction and used to cut the object in question into two parts(mask out the disk region and perform watershed from two endpoints). We focused our annotation effort on the most detrimental category of errors where a major neurite branch is merged with another or with glia. All splits are performed in batch mode after annotations are complete.

The processed subvolumes are reconciled again with the same pipeline, and merge errors decrease significantly (an example shown in Fig. 4.15C).

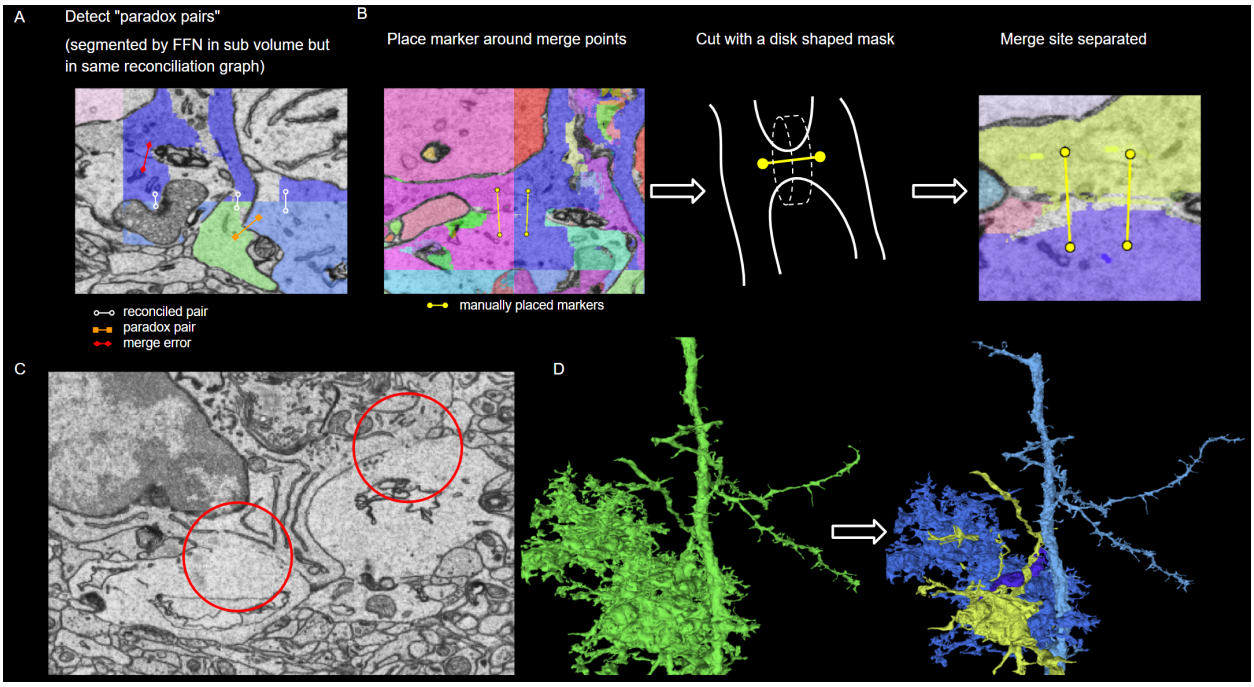


Figure 4.15: Merge Correction. A. Detecting merge error by finding paradox pairs in reconciliation graph. B. Workflow for merge error correction: 1. manually inspect and find merge error sites, place a line annotation marker at each such site. 2. Cut with a disk-shape mask. 3. Separated segments. C. Common cause of merge errors: cellular membrane break. D. A glia-dendrite merge error corrected.

Synapse prediction

Vesicle cloud and synaptic junction prediction In EM, each synapse is morphologically defined by a pre-synaptic vesicle cloud and a post-synaptic density/synaptic junction. For this task. Similar to tissue mask prediction, manual annotations are first generated with Webknossos. Two $1024 \times 1024 \times 128$ voxel subvolumes are traced for each dataset, in which each instance of vesicle cloud and synaptic junction is assigned a unique ID.

We then convert vesicle clouds(VC) and synaptic junctions(SJ) into two separate masks to train separate UNets. Architecture-wise, we use a variant of 3D UNet called "distance-transform UNet" previously described in (Heinrich et al. 2018) (Fig. 4.13B). The network is fully convolutional and accepts flexible input size. During training, $128 \times 128 \times 12$ voxel patches are randomly sampled around synapse sites as input. We use an Adam optimizer with a learning rate of 0.001, batch size 8. The training was performed on 2 to 4 GPUs over 12 hours until saturation(voxel accuracy over 95%).

The inference is set up in a similar fashion for soma and blood vessel mask prediction. We use a larger field of view $782 \times 782 \times 50$ since the GPU RAM can hold larger tensors in inference time without the need to hold gradients. Also, the batch size is lowered to 1. The inference is distributed to 8 A100 GPUs, and it takes about 1 hour to finish.

Infer synapses After finishing the prediction of vesicle cloud and synaptic junction mask, we utilize the fact that a synapse is defined by the juxtaposition of the two and use a heuristic to detect synapse instances.

We perform synapse detection in a chunkwise fashion distributed with MPI. For each chunk, we first identify all instances of vesicle cloud(VC) and synaptic junction(SJ) by connected components and only keep the VCs with sizes above 100 voxels and SJs with sizes above 25 voxels. Then for each VC instance, the largest 3 SJs bordering it are selected as candidate synapses. By overlaying and comparing with segmentation volume, the pre-

synaptic segment in which the VC resides and the post-synaptic partner it targets can both be determined.

We can then build a table of synapses, including crucial information like pre/post-synaptic segment IDs, 3D coordinates, vesicle cloud size, synaptic junction size, etc.

Correction of Proofreading

The accuracy of automatic processing does not yet reach the level where it can be readily used in statistical analysis. The remaining errors are manually reviewed and proofread with an annotation engine we developed with Neuroglancer's Python API, including two variants: an "agglomerator" and a "synapse proofreader". Since this study is focused on synapse distribution on dendrites, we direct our annotation resources to the dendrites originating from soma within the volume.

Fix split errors We manually find the initial segment of each neurite branch originating from soma and cycle through them in the "agglomerator". Its remaining segments are added in Neuroglancer and saved as a group of segment IDs for each initial segment. In practice, the majority of the time was spent on adding dendritic spines, which frequently drop off in the segmentation stage due to their thin spine necks at $12 \times 12\ nm$ resolution segmentation.

For each dataset, about 200 dendrites were thus proofread, and among them, 88 from P14 and 107 from P105 are selected as valid dendrites for the next stage. The ones that extend out of the volume due to bad orientation or contains merge errors are discarded. We estimate about 100 human working hours for this step.

Merge and sparsify skeletons For each fully agglomerated dendrite, the skeletons previously generated from its fragments using "Kimimaro" are merged into one and sparsified such that each edge is 250 nm long while all leaf nodes and branch nodes(degree not equal to 2) are preserved.

Quantify distance to soma For each annotated dendrite, we find its root by picking the minimum of an NDTree query between all its skeleton nodes and the mean coordinate of soma nodes. We then find the shortest path distance to root for each skeleton node as the path distance to soma. (Fig 4.16C)

Proofread synapses For each proofread dendrite, we query the synapse table for entries with post-synaptic ID contained in the dendrite's segment ID group and pre-populate the Neuroglancer-based "synapse proofreader" with these automatically predicted synapses.

A human annotator will then inspect each such dendrite and add synapses in the form of line annotations, starting from the pre-synaptic side, often within vesicle cloud, and ending in the post-synaptic partner. Also, false-positive predictions are removed. (Fig. 4.16A).

In a separate round, apart from synapse annotations, we manually find filopodia based on morphology and place point markers at the tip of each one. (Fig. 4.16A,B green annotations).

In practice, we found that the predominant error mode was recall error due to 1. missing synaptic junction, in cases where the post-synaptic density did not have a strong signal or when the synapse is oriented in the z-axis and got cut in the middle; 2. non or small vesicle cloud. We observed many cases where a weak synapse has only a tiny vesicle cloud associated with it. Overall, we estimate about 150 human working hours.

Unify manual and automatic synapses We then use the following heuristic to infer vesicle cloud and synaptic junction size for manual synapses. For cases where the SJ is missing, we infer SJ size by dilating the pre and post-synaptic neurite segments and take the intersection area as SJ. For missing VC, which usually only has a few vesicles and did not pass the threshold to get predicted as vesicle cloud, we chunk around the pre-synaptic position with $32 \times 32 \times 8$ voxel cube and mask with pre-synaptic segment, and then we use the dark voxels within this mask as a proxy of VC size.

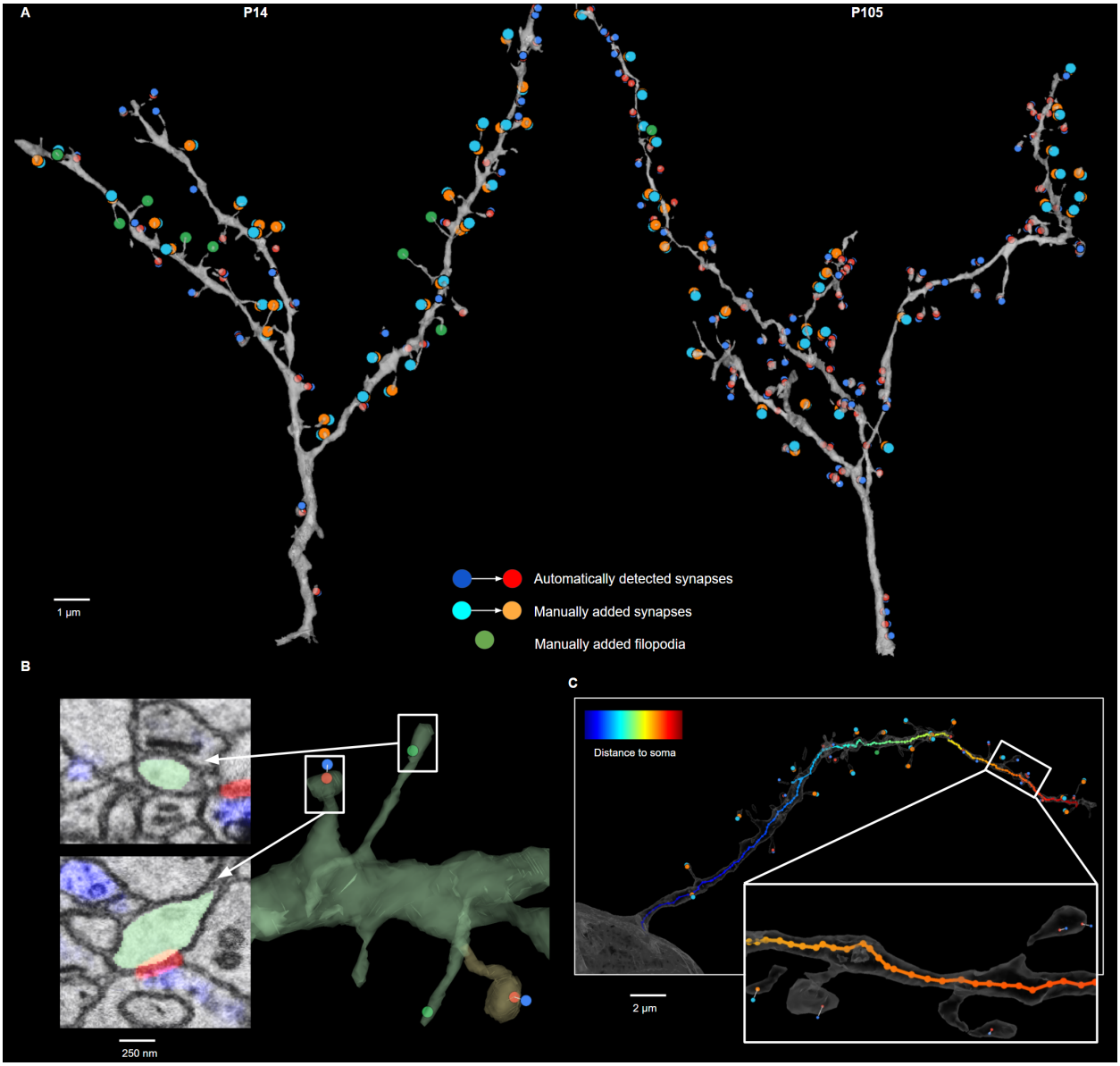


Figure 4.16: Manual Synapse Proofreading. A. Example dendrites from P14(left) and P105(right) displayed with automatic synapses, manual synapses, and filopodia. B. Example of spine synapse vs. filopodia, which uses thin protrusions without forming any synapse. C. Distance to soma is calculated for each skeleton node, and synapses are then mapped to their nearest skeleton node. (inset). Focused view of a portion of skeletonization.

Separate Spine vs. Shaft Synapse Since the skeletonization only covers the neurite branch(Fig. 4.12), we could calculate the distance between the post-synaptic position and the nearest skeleton node. Since we also have the radius at each skeleton node, the difference between the two is the distance between synapse and surface of neurite, which can determine

if the synapse is on the spine or shaft. We visually inspected the classification with these heuristics and found that apart from corner cases where the spine orients back towards the neurite branch, the heuristic is mostly correct and can be used to determine spine vs shaft synapses automatically.

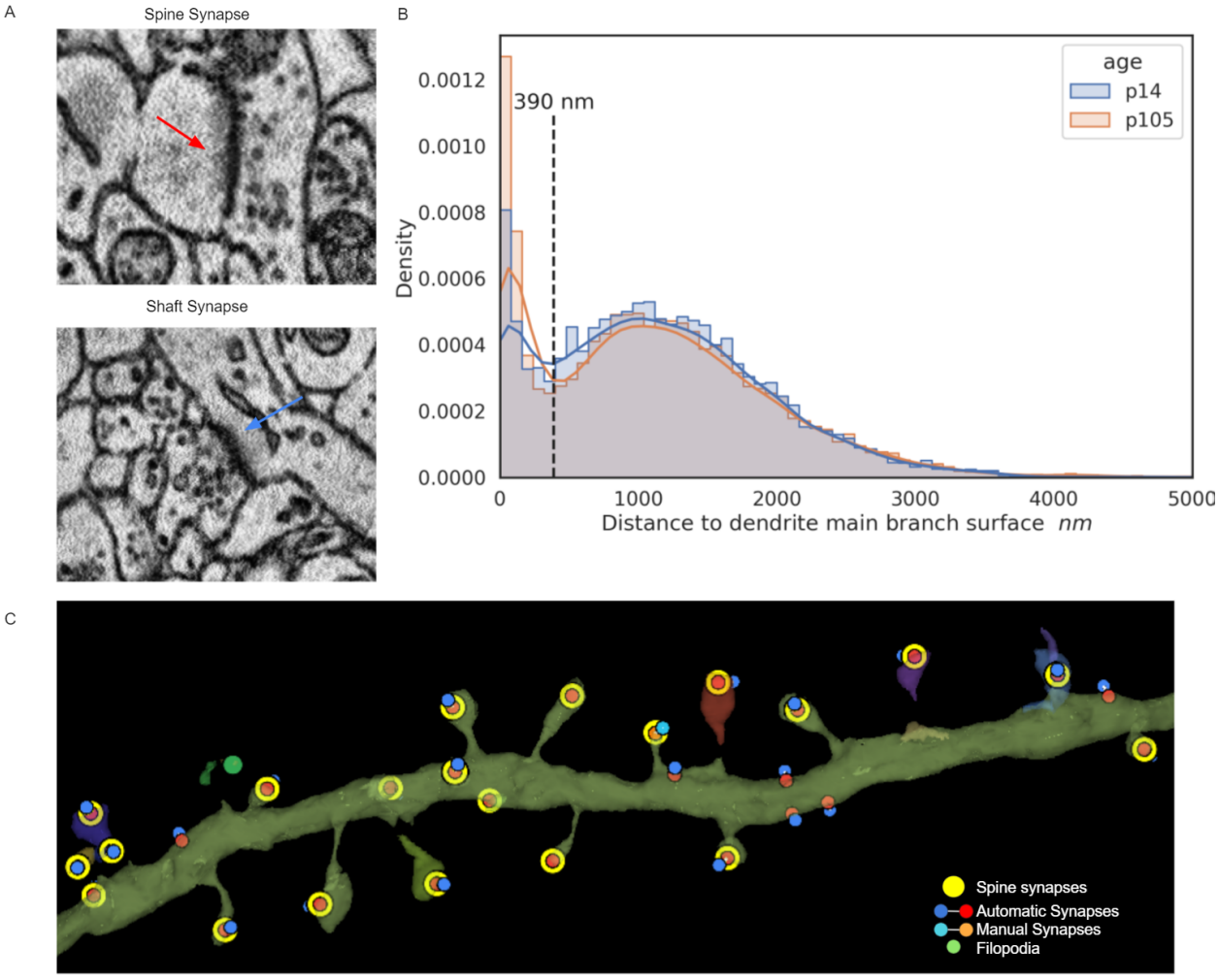


Figure 4.17: Spine vs. shaft synapse: (A). Spine synapse and shaft synapse under EM. (B). Histogram of synapse distance to dendrite surface. 390 nm is used to split spine vs. shaft synapse (C). A piece of dendrites with spine synapses marked in yellow with the previous criteria in (B).

Get spine size Spine size is obtained automatically from synapses proofread in the previous step.

We first create an NDTree with the spatial coordinates of skeleton nodes for each dendrite. Then for each synapse classified as spine synapse, we query the skeleton node NDTree with its post-synaptic coordinates(which is a good approximate of the dendritic spine head tip location) for its nearest partner.

For each synapse, from its post-synaptic coordinate, we query the nearest skeleton node on the dendritic branch through an NDTree query. Then, we cut a bounding box of segmentation for each pair of points and select the mask covering the corresponding dendrite and perform watershed from these two points. The part that started from the post-synaptic coordinate is used as spine size.

Mitochondria prediction Mitochondria are predicted similarly as synaptic junctions and vesicle clouds, with a 3D UNet. (Fig. 4.12B). Instance segmentation was performed with a chunkwise connected component labeling, followed by reconciliation described in the segmentation step.

CHAPTER 5

CONCLUSIONS AND DISCUSSIONS

5.1 Achievements

In this thesis, we present three separate but inter-connected projects that span the entire connectomics pipeline from computational infrastructure, novel method development to the application of connectomics to biological inquiry. Combined, this thesis summarizes a comprehensive effort in making connectomics more accessible, powerful, and biologically useful. The first two projects focused on the computational side of connectomics and these tools can be useful for the field. The third project takes a critical look at the role of pruning in the maturity of visual system and raises interesting questions for further studies.

1. On the computational infrastructure: A full connectomics pipeline was established on a high-performance computing(HPC) platform, enabling large-scale connectomics studies with supercomputers (Vescovi, H. Li, et al. 2020). The infrastructural effort formed the foundation of the analysis performed in chapter 4 and is driving multiple ongoing connectomics projects beyond the scope of this thesis (for example Wei et al. 2021). The complexity of connectomics data processing has been a major bottleneck that is preventing it from being used by mainstream neuroscience community. We believe the software framework will be a useful addition to the field, especially in providing smaller labs with less computing expertise access to computing resources at national labs. By establishing a connectomics pipeline on supercomputers, we expect to provide a centralized infrastructure for more labs that are interested in connectomics and accelerate the scientific output.

2. On novel method development: An automatic neural subcompartment classification system was developed with collaborators at Google (H. Li et al. 2020). The system reached

state-of-the-art performance in predicting axon, dendrite and soma compartments of neurons, and could be applied in merge error detection and correction. The system addresses two problems: first, a large-scale automatic segmentation requires automatic downstream analysis, for example, distinguishing the functional subcompartments help determine neuronal polarity; second, despite progress in segmentation algorithm performance, there remains significant amount of merge errors that are beyond what human annotators could proofread, by detecting cross-class merges, this dramatically reduce the workload for merge error correction. In the recent petascale human cortex study (Shapson-Coe et al. 2021), the system was successfully applied to further increase reconstruction accuracy.

3. On biological inquiry with connectomics toolbox: A comprehensive study of synapse development throughout the critical period in mouse visual cortex was performed. We combined state-of-the-art computational methods with serial electron microscopy to provide the first saturated reconstruction of multiple development stages in the mouse cortex. We demonstrated with the high-res reconstructions that synapses increase in density and size in primary visual cortex layer 4 over the course of critical period. This is surprising because there are clear evidences that support the existence of pruning in central nervous system, and the idea of a premature circuit eliminating weak connections to become mature makes logical sense. Our observations suggest that the overwhelming trend in critical period development is new synapse formation and maturation, contrary to what a pruning hypothesis suggests.

5.2 Outlook

EM Connectomics in recent years saw tremendous progress and is increasingly being recognized by mainstream neuroscience as a potential game-changer. In the early days, connectomics met with significant skepticism behind the hype and excitements. Critics of connectomics often cite the full C.elegans connectome as an example where a full static map of

connectivity is not enough to help us bridge the structure and behavior of a model system. With only 302 neurons, what appeared to be an easy network to simulate proved to be much harder than originally expected. Despite ongoing efforts to simulate the biophysical dynamics on top of the full connectome, there still is not a satisfying digital worm that mirrors the behavior of a real one (Sarma et al. 2018). Apart from the difficulty in reproducing environment and sensory signal simulation, the failure is often attributed to the complex biochemical modulation that could not be captured by the connectivity matrix and the fact that C.elegans transmits signals with largely analog synapses instead of spikes.

Ever increasing sample size With the advent of Block-face EM and ATUM EM, a $100 \times 100 \times 100$ μm level sample could be collected and reconstructed within a reasonable(still much longer) life cycle of a typical neuroscience project. Connectomics projects at this level focused on small samples of tissues like mouse retina (Helmstaedter, Briggman, and Denk 2008), mouse S1 (Narayanan Kasthuri, Kenneth Jeffrey Hayworth, et al. 2015; Gour et al. 2021), or zebra finch area x (Jorgen Kornfeld et al. 2020), which is still the mainstream in connectomics as of the time of this thesis. The project described in Chapter 4 also falls in this category. At this scale, one sample could contain up to hundreds of neurons, with enough field of view to classify cell types and contains rich information about synapses. However, it is often not enough to cover the entirety of a cortical neuron, which could span as much as $150\mu m$ away from the soma. The bigger problem is that most neurites within such volume do not come from cells with soma in view, often case there is no good way to determine where a neurite comes from and from which cell type. These limitations considerably weaken our ability to build a complete connectivity graph out of a connectomics sample.

Things started to change when the field approached millimeter level with the "full adult fly brain" (FAFB) reconstruction with serial section TEM and FFN, reaching $995 \times 537 \times 283$ μm in dimensions (P. H. Li et al. 2019). Apart from the sheer increase in size over previous experiments, the FAFB dataset was the first complete reconstruction of a species'

central nervous system after completing C.elegans connectome three decades ago. Although there are still quite some segmentation errors that prevent immediate access to the full drosophila connectome, the accuracy has reached an extremely high level to be ready for biological study with the help of some proofreading.

The prominence of drosophila as a model system in neuroscience immediately brought attention from neuroscientists outside connectomics. Within one year of the release of the FAFB dataset, a plethora of biological studies based on it ensued (Scheffer et al. 2020; Zheng, F. Li, et al. 2020; A. S. Bates et al. 2020; N. Otto et al. 2020; Marin et al. 2020), with highly diverse topics based on different subsets of neurons of interest. The complete map of the drosophila brain quickly proved to be an effective atlas to benchmark previous studies and perhaps, more importantly, raise questions previously unthought of.

In the more recent human cortex project (Shapson-Coe et al. 2021), the authors reconstructed a $3 \times 2 \times 0.16\ mm$ cube of human cortex tissue with multi-beam SEM, reaching 2.1 PB of raw data. Some previously undocumented objects are revealed within the volume, like a whorl of loosely coiled myelin or a swollen dendritic spine packed with intramembranous objects. While a thorough comparison between human and other mammalian brains remains to be seen, it offers a rare glimpse into the human brain at extremely high resolution that could help identify some key structural differences that made human intelligence extraordinary.

The most ambitious connectomics project in preparation today is to reconstruct an entire mouse brain. It is hard to overstate the difficulty of an undertaking at such a scale. The size difference between fruit fly and mouse brain is illustrated vividly by Abbott et al. 2020 as "6.5 Boeing 747 airliners vs. distance from Boston to Lisbon". The unprecedented scale also brings about seemingly insurmountable technical hurdles from sample preparation all the way to data analysis.

The full dataset of a mouse brain is roughly a centimeter cube estimated to be 1 Exabyte.

The storage and analysis of such data would challenge the computational infrastructure of even the top-tier supercomputers and tech giants and would require continuous development in technologies throughout the entire pipeline. Several actively researched areas include image compression (Minnen et al. 2021), which aims to use a CNN to denoise raw images and increase data compression rate without hurting the segmentation quality. There is also an ongoing effort to optimize and improve state-of-the-art segmentation algorithms like Flood-filling networks and UNet with affinity (K. Lee et al. 2017) and other segmentation methods are being explored. There is also an effort to use a generative adversarial network to adapt raw image texture in order to reduce training data requirements (Michał Januszewski and Jain 2019). It would also be interesting to see how advancements in the broader computer vision field can be used in the context of EM.

Even with the most optimistic predictions, it will take years to make this happen. A full mouse brain reconstruction, upon completion, will fundamentally change the landscape of neuroscience, perhaps in ways we could not foresee today.

Connectomics as an infrastructure for neuroscience Apart from the excitement around the high-resolution map of an entire nervous system. The resurgence of interest in connectomics comes from the fact that it can be considered an infrastructure for the entire neuroscience field, very much like how the human genome project served the genetics field, or more recently, how Alphafold redefined proteomics (Jumper et al. 2021).

Accessibility to connectomics data is a much larger issue than in genetics due to data size and the 3D nature. When datasets are several hundred Gigabytes, it is still possible to move data between labs the traditional way. With the ever-increasing data size, especially Petascale volumes we are starting to see, it is no longer an option. Fortunately, in the last few years, several key engineering achievements made connectomics data more accessible: Neuroglancer (Maitin-Shepard n.d.) elegantly solved the engineering challenge of visualizing large 3D volume over the internet with a design like a "Google Map for connectomics", serving

arbitrarily large 3D data as mini-blocks of files at different zoom level, allowing easy remote access with just a web browser. CloudVolume (W. Silversmith n.d.[a]) was designed along with Neuroglancer and provided tools for any lab to convert their 3D data to compatible formats, meanwhile providing convenient tools to interface with large image volumes and perform automation and analysis. With these tools, connectomics labs could host their own data on in-lab servers or clouds. For labs with less access to computing resources, the "open connectome project" was also built (Vogelstein et al. 2018) to host datasets generated this way.

Parallel to these efforts, the "Webknossos" project provides a similar framework that serves data remotely in chunks. It has less powerful visualization(in 3D rendering) but is equipped with a more powerful annotation engine, with the ability to perform paintbrush annotation, which is not available in Neuroglancer. It allows easy distribution of groundtruth annotation workload, which in foreseeable future, still constitutes a large portion of a connectomics project.

With open accessibility, neuroscientists in other subfields can easily crosscheck activity recordings or light microscopy results with the high-resolution structural map, previous hypotheses can be tested and new circuitry can be discovered at an unprecedented pace.

Rethinking simulation The hope of simulating the brain is one of the driving forces behind the inception of computers. Alan Turing, the founding father of computer science and artificial intelligence, openly discussed his desire to build a machine to "imitate a brain" (Alan M Turing 1951) and turned his interest toward understanding the biochemical basis of intelligence in his final years (Alan Mathison Turing 1990). The mathematical foundation he laid out is still the basis of modern-day computers, but at his time, the nascent computers were not able to run any simulation even remotely close to what a biological brain can do. Among his long list of legacies, he postulated there could be a potential link between AI engineering and human psychology and left the opened-ended question for future computer

scientists and neuroscientists.

The Hodgkin Huxley model (Hodgkin and Huxley 1952) marked the beginning of computational neuroscience. By capturing the dynamics of spike generation in precise mathematical form, it made large-scale simulation of networks of neurons theoretically feasible. Such simulations often use semi-randomized network topology with a set of constraints like Erdos-Renyi network (Erdos, Rényi, et al. 1960), and run Hodgkin Huxley model or simpler Leaky Integrate-and-fire model for each neuron to generate spikes.

While tremendous insights had been gained over the years with this idealized paradigm, previous attempts at simulating the brain with a large-scale network based on artificially wired neurons had not achieved much success. The idea of running a massive collection of neurons based on the Hodgkin-Huxley model culminated in the ill-fated Blue Brain Project (Markram 2006), which optimistically predicted in 2006 that computing technology would be ready to fully simulate the human brain in a petaFLOPS level supercomputer within a decade. 15 years later, although the state-of-the-art supercomputers are approaching exaFLOPS level thanks to the quantum leap in GPU technology, we are still far from simulating the human brain. Simulation of even drosophila brain is still challenging and require a high level of abstraction (Givon and Lazar 2016), let alone mouse or human. The significant technical challenges and huge disappointments at the blue brain project (Frégnac and Laurent 2014; Mainen, Häusser, and Pouget 2016) put a pause on simulation in mainstream neuroscience.

The problem with the simulation approach lies not only in the difficulty of running a huge number of parallel neuron simulations with a plausible level of biophysical details but also in the lack of actual connectivity. However, with advancements in connectomics, it is no longer unthinkable to acquire a massive connectivity graph of neurons, which could potentially reignite interest in simulation. As a bonus, the axonal and dendritic morphologies are captured as well, so a neuron can be modeled with as much detail as desired with a much

higher level of biological authenticity. Combined with activity recording, the simulation could be tuned or trained to reflect live circuitry at a larger scale.

Connectomics and Deep Learning Till now, connectomics has primarily reaped benefits from the rapid advancements of deep learning, but many in the field wonder if and to what degree connectomics can inspire research in deep learning and AI. The link is fairly straightforward. The biological neural circuits are highly efficient computing machines optimized over the course of eons of evolution. By studying how such a computing machine is organized, we might be able to gain valuable insights into the design principles of artificial intelligence, and most importantly, the learning rules in a biological neural network.

Also, it would not be the first time when neuroscience played a critical role in the development of AI.

1. The whole concept of "deep learning", or deep neural network, was inspired by how actual neurons compute. The "perceptron", a model neuron that gathers inputs and produces an output, was proposed in as early as the 50s (Rosenblatt 1957).

2. The idea of convolutional neural network traces its root to Hubel and Wiesel experiments in cat visual cortex, in which it was found that neurons in LGN have center-surround dot-shaped receptive fields, and simple cells in V1 combine inputs from LGN into bar-shaped receptive field selective to various orientations, complex cells are tuned to more involved patterns and neurons more downstream in the visual pathway are selective to particular patterns like face (Yamins and DiCarlo 2016).

3. Reinforcement learning, a subfield in deep learning which gained massive attention after AlphaGo historically defeated grandmaster human players, was inspired by how the dopaminergic reward mechanism works in biological neural circuitry (Mnih et al. 2015). Striking parallel between reinforcement learning and neuroscience was also demonstrated by researchers from DeepMind, who trained a recurrent neural network for spatial navigation and found some units exhibiting grid cell-like receptive field patterns within the network

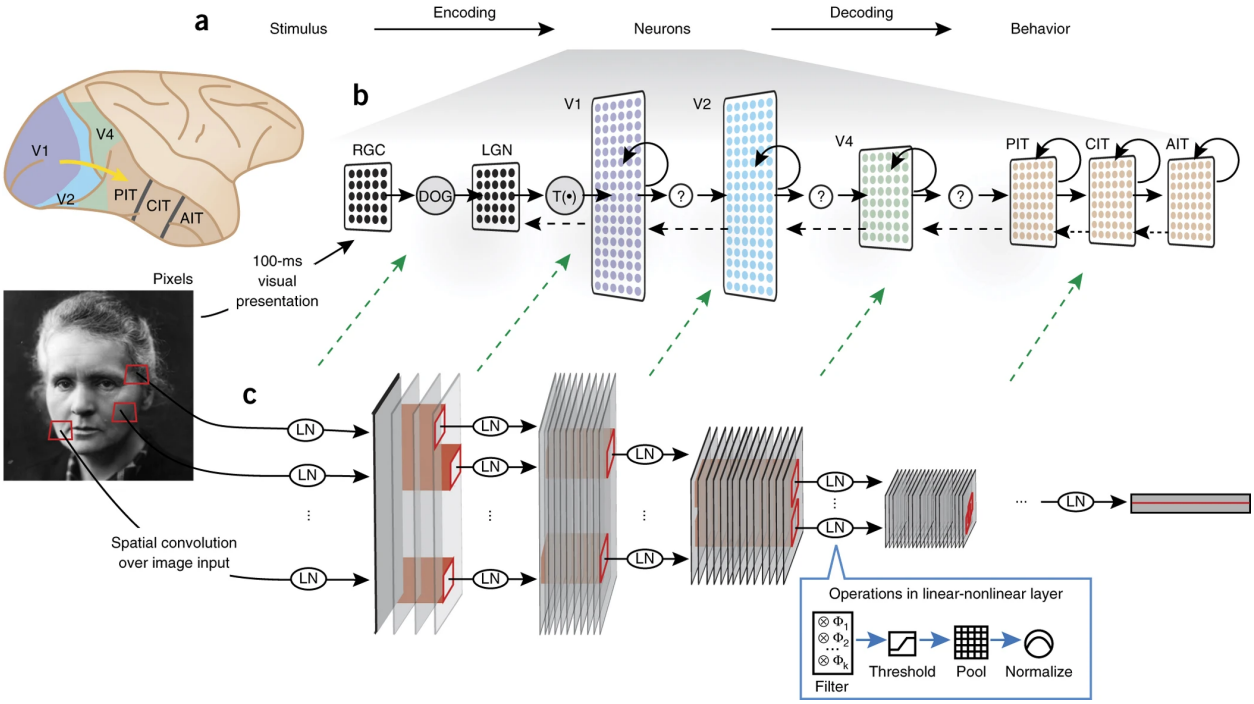


Figure 5.1: Biological parallel to CNN. The design of convolutional neural network draws inspiration from the flow of visual information through cortical regions. reprinted (Yamins and DiCarlo 2016) with permission from the publisher

spontaneously (Banino et al. 2018).

One of the mysteries in neuroscience is how a biological neural network trains itself. Artificial neural networks today overwhelmingly adopt a backpropagation algorithm to perform weight adjustments for each connection by calculating corresponding partial derivatives from the difference between the network output and training label. However, this mechanism requires each connection to receive a differentiated error signal from the final output, potentially numerous layers away. The exact equivalent of this error signal does not exist explicitly in the cortex, and it is therefore not clear how a synapse could decide if it should increase or decrease it is weight in a backpropagation fashion.

However, in recent years, it has been postulated that, although there is no exact equivalent of a backpropagation algorithm in the brain, the highly recurrent wiring and the inhibitory modulatory signals could form the substrate of error-based learning (T. P. Lillicrap et al.

2020). For example, the visual cortex has extensive recurrent connections both within and across sub-areas (Fig. 5.1), unlike a traditional CNN. It is therefore interesting to compare hypothesized novel learning networks against connectomics reconstructions in coming years.

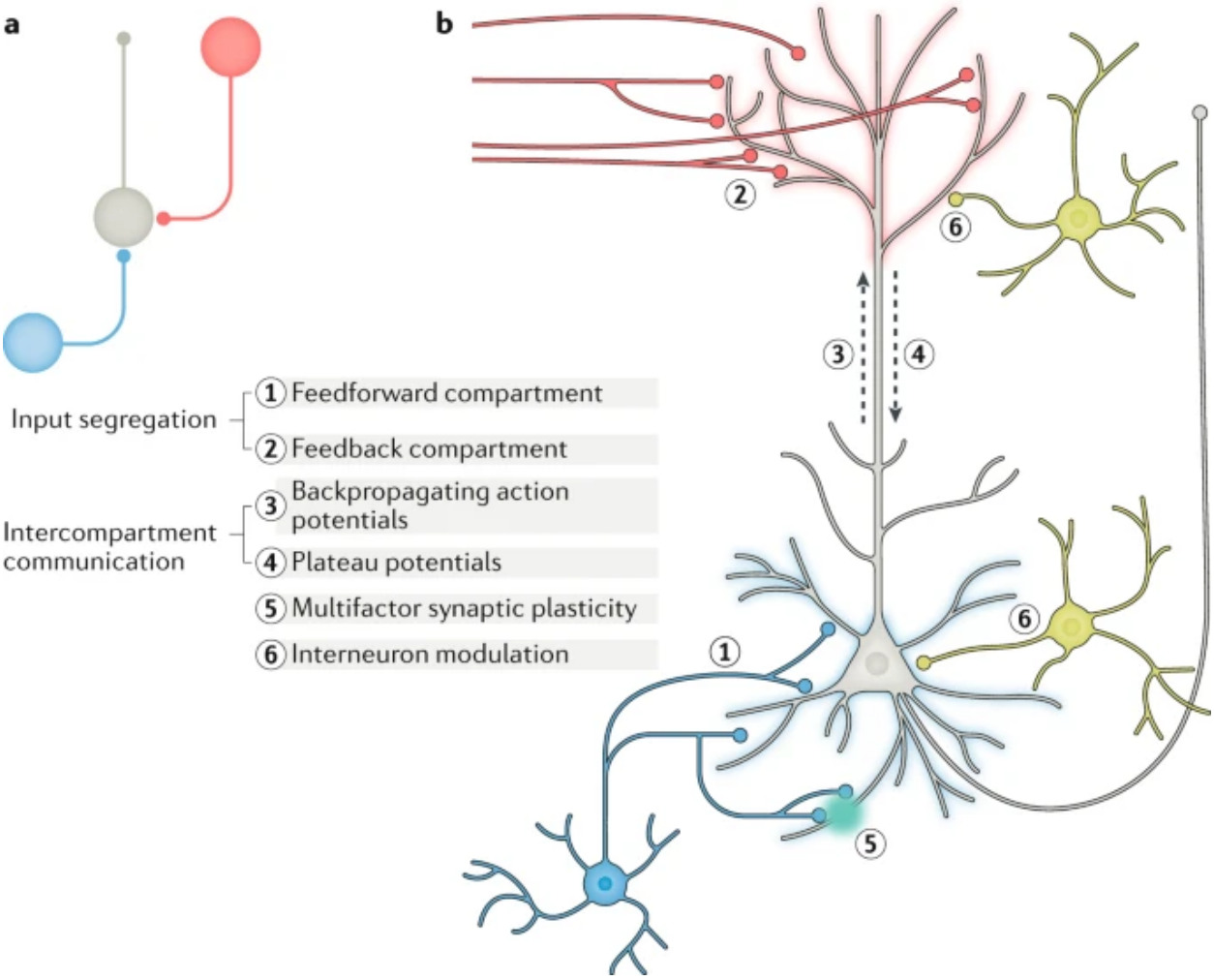


Figure 5.2: Biological basis of backpropagation a. Traditional model of a neuron(grey cell) with feedforward signal(blue, from lower-order cortical areas) and feedback signal(red, from higher-order cortical areas) b. A contemporary model of a cortical pyramidal neuron. In which inputs to a neuron are segregated into different compartments(apical, basal), and inhibitory neurons relaying provide a feedback signal that could serve as the training signal. reprinted (T. P. Lillicrap et al. 2020) with permission from the publisher

Functional and structural Correlation There are large collaborative efforts to bridge the gap between the function and structure of neural circuits. In the Machine Intelligence

from Cortical Networks (MICrONS) project (Turner et al. 2020), multiple labs collaborated on reconstructing a millimeter cube with fluorescent labeling and calcium imaging before the tissue is prepared for EM histology, providing both functional and structural data simultaneously for a subregion of reconstruction. There is still tremendous difficulty in putting together the resources and reconcile the huge resolution disparity between functional and structural data, but over time, with lower cost and better automation, this paradigm could one day become the standard practice in connectomics.

Why developmental connectomics The developmental connectomics paradigm we discussed in this thesis is another potential direction for future connectomics. EM Connectomics has a major flaw that frequently draws criticism from the anti-connectomics camp and raises concerns among even the most ardent advocates, which is its static nature. While light microscopy could be used to follow the day-to-day growth of a small number of neurites, connectomics captures a snapshot of a much large group of neurites. With advancing automation technology in the coming years, it is possible to envision a more efficient pipeline that could reconstruct many small samples across ages, compensating for the limitations inherent in connectomics.

In the context of this thesis, we are particularly interested in how an immature neural network with little, if not none, external stimulus becomes mature over the critical period by a concerted effect of genetic orchestration and activity-dependent synaptic plasticity. By providing multiple snapshots of the same region, we expand connectomics beyond a single time point and reveal a broader trend in development.

In this study, the overwhelming trend is that synapses increase in both quantity and strength, refuting a pruning centric view where a mature circuit comes from trimming exuberant connections. Possible explanations include the significant increase in excitatory input from LGN and heightened level of long term potentiation, but the exact mechanism behind the structural shift need to be further studied along with functional data. In future, it re-

mains to be seen to what extent the result is true for other cortical areas and how it differs in other model systems.

BIBLIOGRAPHY

Cajal, Santiago Ramón y (1888). *Estructura de los centros nerviosos de las aves*.

Neher, Erwin and Bert Sakmann (1976). "Single-channel currents recorded from membrane of denervated frog muscle fibres". In: *Nature* 260.5554, pp. 799–802.

Hodgkin, Alan L and Andrew F Huxley (1952). "A quantitative description of membrane current and its application to conduction and excitation in nerve". In: *The Journal of physiology* 117.4, pp. 500–544.

Minsky, Marvin (1988). "Memoir on inventing the confocal scanning microscope". In: *Scanning* 10.4, pp. 128–138.

Chalfie, Martin, Yuan Tu, Ghia Euskirchen, William W Ward, and Douglas C Prasher (1994). "Green fluorescent protein as a marker for gene expression". In: *Science* 263.5148, pp. 802–805.

Grynkiewicz, Grzegorz, Martin Poenie, and Roger Y Tsien (1985). "A new generation of Ca2+ indicators with greatly improved fluorescence properties." In: *Journal of biological chemistry* 260.6, pp. 3440–3450.

Grienberger, Christine and Arthur Konnerth (2012). "Imaging calcium in neurons". In: *Neuron* 73.5, pp. 862–885.

Helmchen, Fritjof and Winfried Denk (2005). "Deep tissue two-photon microscopy". In: *Nature methods* 2.12, pp. 932–940.

Betzig, Eric, George H Patterson, Rachid Sougrat, O Wolf Lindwasser, Scott Olenych, Juan S Bonifacino, Michael W Davidson, Jennifer Lippincott-Schwartz, and Harald F Hess (2006). "Imaging intracellular fluorescent proteins at nanometer resolution". In: *Science* 313.5793, pp. 1642–1645.

Ke, Meng-Tsen, Yasuhiro Nakai, Satoshi Fujimoto, Rie Takayama, Shuhei Yoshida, Tomoya S Kitajima, Makoto Sato, and Takeshi Imai (2016). "Super-resolution mapping of neuronal circuitry with an index-optimized clearing agent". In: *Cell reports* 14.11, pp. 2718–2732.

Chen, Fei, Paul W Tillberg, and Edward S Boyden (2015). "Expansion microscopy". In: *Science* 347.6221, pp. 543–548.

Palade, GE (1954). "Electron microscope observations of interneuronal and neuromuscular synapses". In: *Anat Rec* 118, pp. 335–336.

Palay, Sanford L, SM McGee-Russell, Spencer Gordon Jr, and Mary A Grillo (1962). "Fixation of neural tissues for electron microscopy by perfusion with solutions of osmium tetroxide". In: *The Journal of cell biology* 12.2, pp. 385–410.

Sporns, Olaf, Giulio Tononi, and Rolf Kötter (2005). "The human connectome: a structural description of the human brain". In: *PLoS computational biology* 1.4, e42.

Van Essen, David C, Stephen M Smith, Deanna M Barch, Timothy EJ Behrens, Essa Yacoub, Kamil Ugurbil, Wu-Minn HCP Consortium, et al. (2013). "The WU-Minn human connectome project: an overview". In: *Neuroimage* 80, pp. 62–79.

Livet, Jean, Tamily A Weissman, Hyuno Kang, Ryan W Draft, Ju Lu, Robyn A Bennis, Joshua R Sanes, and Jeff W Lichtman (2007). "Transgenic strategies for combinatorial expression of fluorescent proteins in the nervous system". In: *Nature* 450.7166, pp. 56–62.

Kim, Sung-Yon, Kwanghun Chung, and Karl Deisseroth (2013). "Light microscopy mapping of connections in the intact brain". In: *Trends in cognitive sciences* 17.12, pp. 596–599.

Shen, Fred Y, Margaret M Harrington, Logan A Walker, Hon Pong Jimmy Cheng, Edward S Boyden, and Dawen Cai (2020). "Light microscopy based approach for mapping connectivity with molecular specificity". In: *Nature communications* 11.1, pp. 1–12.

White, John G, Eileen Southgate, J Nichol Thomson, Sydney Brenner, et al. (1986). "The structure of the nervous system of the nematode Caenorhabditis elegans". In: *Philos Trans R Soc Lond B Biol Sci* 314.1165, pp. 1–340.

Briggman, Kevin L and Davi D Bock (2012). "Volume electron microscopy for neuronal circuit reconstruction". In: *Current opinion in neurobiology* 22.1, pp. 154–161.

Briggman, Kevin L and Winfried Denk (2006). "Towards neural circuit reconstruction with volume electron microscopy techniques". In: *Current opinion in neurobiology* 16.5, pp. 562–570.

Denk, Winfried, Heinz Horstmann, and Kristen M Harris (2004). "Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure". In: *PLoS biology* 2.11, e329.

Heymann, Jurgen AW, Mike Hayles, Ingo Gestmann, Lucille A Giannuzzi, Ben Lich, and Sriram Subramaniam (2006). "Site-specific 3D imaging of cells and tissues with a dual beam microscope". In: *Journal of structural biology* 155.1, pp. 63–73.

Hayworth, KJ, N Kasthuri, R Schalek, and JW Lichtman (2006). "Automating the collection of ultrathin serial sections for large volume TEM reconstructions". In: *Microscopy and Microanalysis* 12.S02, pp. 86–87.

Kasthuri, N, K Hayworth, J Lichtman, N Erdman, and CA Ackerley (2007). "New technique for ultra-thin serial brain section imaging using scanning electron microscopy". In: *Microscopy and Microanalysis* 13.S02, pp. 26–27.

Harris, Kristen M, Elizabeth Perry, Jennifer Bourne, Marcia Feinberg, Linnaea Ostroff, and Jamie Hurlburt (2006). "Uniform serial sectioning for transmission electron microscopy". In: *Journal of Neuroscience* 26.47, pp. 12101–12103.

Yin, Wenjing, Derrick Brittain, Jay Borseth, Marie E Scott, Derric Williams, Jedediah Perkins, Christopher S Own, Matthew Murfitt, Russel M Torres, Daniel Kapner, et al. (2020). "A petascale automated imaging pipeline for mapping neuronal circuits with high-throughput transmission electron microscopy". In: *Nature communications* 11.1, pp. 1–12.

Eberle, Anna Lena and Dirk Zeidler (2018). "Multi-beam scanning electron microscopy for high-throughput imaging in connectomics research". In: *Frontiers in neuroanatomy* 12, p. 112.

Shapson-Coe, Alexander, Michał Januszewski, Daniel R Berger, Art Pope, Yuelong Wu, Tim Blakely, Richard L Schalek, Peter Li, Shuohong Wang, Jeremy Maitin-Shepard, et al. (2021). "A connectomic study of a petascale fragment of human cerebral cortex". In: *bioRxiv*.

Kasthuri, Narayanan, Kenneth Jeffrey Hayworth, Daniel Raimund Berger, Richard Lee Schalek, José Angel Conchello, Seymour Knowles-Barley, Dongil Lee, Amelio Vázquez-Reina, Verena Kaynig, Thouis Raymond Jones, et al. (2015). "Saturated reconstruction of a volume of neocortex". In: *Cell* 162.3, pp. 648–661.

Lowe, David G (2004). "Distinctive image features from scale-invariant keypoints". In: *International journal of computer vision* 60.2, pp. 91–110.

Saalfeld, Stephan, Richard Fetter, Albert Cardona, and Pavel Tomancak (2012). "Elastic volume reconstruction from series of ultra-thin microscopy sections". In: *Nature methods* 9.7, pp. 717–720.

Cardona, Albert, Stephan Saalfeld, Johannes Schindelin, Ignacio Arganda-Carreras, Stephan Preibisch, Mark Longair, Pavel Tomancak, Volker Hartenstein, and Rodney J Douglas (2012). "TrakEM2 software for neural circuit reconstruction". In: *PloS one* 7.6, e38011.

Wetzel, Arthur W, Greg Hood, and Markus Dittrich (2013). "High-performance image registration: preparing for the acquisition of petascale connectomics image stacks". In:

Hildebrand, David Grant Colburn, Marcelo Cicconet, Russel Miguel Torres, Woohyuk Choi, Tran Minh Quan, Jungmin Moon, Arthur Willis Wetzel, Andrew Scott Champion, Brett Jesse Graham, Owen Randlett, et al. (2017). "Whole-brain serial-section electron microscopy in larval zebrafish". In: *Nature* 545.7654, pp. 345–349.

Macrina, Thomas and Dodam Lh (n.d.). *Alembic*. github.com/seung-lab/Alembic.

Turner, Nicholas L, Thomas Macrina, J Alexander Bae, Runzhe Yang, Alyssa M Wilson, Casey Schneider-Mizell, Kisuk Lee, Ran Lu, Jingpeng Wu, Agnes L Bodor, et al. (2020). "Multiscale and multimodal reconstruction of cortical structure and function". In: *bioRxiv*.

Januszewski, Michał, Jörgen Kornfeld, Peter H Li, Art Pope, Tim Blakely, Larry Lindsey, Jeremy Maitin-Shepard, Mike Tyka, Winfried Denk, and Viren Jain (2018). "High-precision automated reconstruction of neurons with flood-filling networks". In: *Nature methods* 15.8, pp. 605–610.

Mitchell, Eric, Stefan Keselj, Sergiy Popovych, Davit Buniatyan, and H Sebastian Seung (2019). "Siamese encoding and alignment by multiscale learning with self-supervision". In: *arXiv preprint arXiv:1904.02643*.

Jain, Viren, H Sebastian Seung, and Srinivas C Turaga (2010). "Machines that learn to segment images: a crucial technology for connectomics". In: *Current opinion in neurobiology* 20.5, pp. 653–666.

Abbott, Larry F, Davi D Bock, Edward M Callaway, Winfried Denk, Catherine Dulac, Adrienne L Fairhall, Ila Fiete, Kristen M Harris, Moritz Helmstaedter, Viren Jain, et al. (2020). "The mind of a mouse". In: *Cell* 182.6, pp. 1372–1376.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems* 25, pp. 1097–1105.

Simonyan, Karen and Andrew Zisserman (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna (2016). "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick (2017). "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969.

Howard, Andrew G, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam (2017). "Mobilenets: Efficient convolutional neural networks for mobile vision applications". In: *arXiv preprint arXiv:1704.04861*.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer, pp. 234–241.

Long, Jonathan, Evan Shelhamer, and Trevor Darrell (2015). "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Lee, Kisuk, Jonathan Zung, Peter Li, Viren Jain, and H Sebastian Seung (2017). "Superhuman accuracy on the SNEMI3D connectomics challenge". In: *arXiv preprint arXiv:1706.00120*.

Heinrich, Larissa, Jan Funke, Constantin Pape, Juan Nunez-Iglesias, and Stephan Saalfeld (2018). "Synaptic cleft segmentation in non-isotropic volume electron microscopy of the complete drosophila brain". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 317–325.

Jain, Viren, Joseph F Murray, Fabian Roth, Srinivas Turaga, Valentin Zhigulin, Kevin L Briggman, Moritz N Helmstaedter, Winfried Denk, and H Sebastian Seung (2007). "Supervised learning of image restoration with convolutional networks". In: *2007 IEEE 11th International Conference on Computer Vision*. IEEE, pp. 1–8.

Ciresan, Dan, Alessandro Giusti, Luca Gambardella, and Jürgen Schmidhuber (2012). "Deep neural networks segment neuronal membranes in electron microscopy images". In: *Advances in neural information processing systems* 25, pp. 2843–2851.

Funke, Jan, Fabian Tschopp, William Grisaitis, Arlo Sheridan, Chandan Singh, Stephan Saalfeld, and Srinivas C Turaga (2018). "Large scale image segmentation with structured loss based deep learning for connectome reconstruction". In: *IEEE transactions on pattern analysis and machine intelligence* 41.7, pp. 1669–1680.

Cremi.org (n.d.). *Cremi*. cremi.org/.

Helmstaedter, Moritz, Kevin L Briggman, Srinivas C Turaga, Viren Jain, H Sebastian Seung, and Winfried Denk (2013). "Connectomic reconstruction of the inner plexiform layer in the mouse retina". In: *Nature* 500.7461, pp. 168–174.

Dorkenwald, Sven, Philipp J Schubert, Marius F Killinger, Gregor Urban, Shawn Mikula, Fabian Svara, and Joergen Kornfeld (2017). "Automated synaptic connectivity inference for volume electron microscopy". In: *Nature methods* 14.4, pp. 435–442.

Kornfeld, Jorgen, Michał Januszewski, P Schubert, Viren Jain, Winfried Denk, and Michale S Fee (2020). "An anatomical substrate of credit assignment in reinforcement learning". In: *BioRxiv*.

Schubert, Philipp J, Sven Dorkenwald, Michał Januszewski, Viren Jain, and Joergen Kornfeld (2019). "Learning cellular morphology with neural networks". In: *Nature communications* 10.1, pp. 1–12.

Bouchard, Kristofer E, James B Aimone, Miyoung Chun, Thomas Dean, Michael Denker, Markus Diesmann, David D Donofrio, Loren M Frank, Narayanan Kasthuri, Chirstof Koch, et al. (2016). "High-performance computing in neuroscience for data-driven discovery, integration, and dissemination". In: *Neuron* 92.3, pp. 628–631.

Vazhkudai, Sudharshan S, Bronis R De Supinski, Arthur S Bland, Al Geist, James Sexton, Jim Kahle, Christopher J Zimmer, Scott Atchley, Sarp Oral, Don E Maxwell, et al. (2018). "The design, deployment, and evaluation of the CORAL pre-exascale systems". In: *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, pp. 661–672.

Stevens, Rick, Jini Ramprakash, Paul Messina, Michael Papka, and Katherine Riley (2019). *Aurora: Argonne's next-generation exascale supercomputer*. Tech. rep. ANL (Argonne National Laboratory (ANL), Argonne, IL (United States)).

Schmidt, Helene, Anjali Gour, Jakob Straehle, Kevin M Boergens, Michael Brecht, and Moritz Helmstaedter (2017). "Axonal synapse sorting in medial entorhinal cortex". In: *Nature* 549.7673, pp. 469–475.

Gour, Anjali, Kevin M Boergens, Natalie Heike, Yunfeng Hua, Philip Laserstein, Kun Song, and Moritz Helmstaedter (2021). "Postnatal connectomic development of inhibition in mouse barrel cortex". In: *Science* 371.6528.

Choi, Se-Young (2018). "Synaptic and circuit development of the primary sensory cortex". In: *Experimental & molecular medicine* 50.4, pp. 1–9.

Hubel, David H and Torsten N Wiesel (1970). "The period of susceptibility to the physiological effects of unilateral eye closure in kittens". In: *The Journal of physiology* 206.2, pp. 419–436.

— (1962). "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex". In: *The Journal of physiology* 160.1, pp. 106–154.

Hubel, David Hunter, Torsten Nils Wiesel, Simon LeVay, Horace Basil Barlow, and Raymond Michael Gaze (1977). "Plasticity of ocular dominance columns in monkey striate cortex". In: *Philosophical Transactions of the Royal Society of London. B, Biological Sciences* 278.961, pp. 377–409.

Hensch, Takao K and Michela Fagiolini (2005). "Excitatory–inhibitory balance and critical period plasticity in developing visual cortex". In: *Progress in brain research* 147, pp. 115–124.

Hensch, Takao K (2005). "Critical period plasticity in local cortical circuits". In: *Nature Reviews Neuroscience* 6.11, pp. 877–888.

Chattopadhyaya, Bidisha, Graziella Di Cristo, Hiroyuki Higashiyama, Graham W Knott, Sandra J Kuhlman, Egbert Welker, and Z Josh Huang (2004). "Experience and activity-dependent maturation of perisomatic GABAergic innervation in primary visual cortex during a postnatal critical period". In: *Journal of Neuroscience* 24.43, pp. 9598–9611.

Huang, Z Josh, Alfredo Kirkwood, Tommaso Pizzorusso, Vittorio Porciatti, Bernardo Morales, Mark F Bear, Lamberto Maffei, and Susumu Tonegawa (1999). "BDNF regulates the maturation of inhibition and the critical period of plasticity in mouse visual cortex". In: *Cell* 98.6, pp. 739–755.

Morales, Bernardo, Se-Young Choi, and Alfredo Kirkwood (2002). "Dark rearing alters the development of GABAergic transmission in visual cortex". In: *Journal of Neuroscience* 22.18, pp. 8084–8090.

Kasthuri, Narayanan and Jeff W Lichtman (2003). "The role of neuronal identity in synaptic competition". In: *Nature* 424.6947, pp. 426–430.

Huttenlocher, Peter R (1984). "Synapse elimination and plasticity in developing human cerebral cortex." In: *American journal of mental deficiency*.

Zhou, Yanmei, Baoling Lai, and Wen-Biao Gan (2017). "Monocular deprivation induces dendritic spine elimination in the developing mouse visual cortex". In: *Scientific reports* 7.1, pp. 1–11.

Vescovi, Rafael, Hanyu Li, Jeffery Kinnison, Murat Keçeli, Misha Salim, Narayanan Kasthuri, Thomas D Uram, and Nicola Ferrier (2020). "Toward an Automated HPC Pipeline for Processing Large Scale Electron Microscopy Data". In: *2020 IEEE/ACM 2nd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, pp. 16–22.

Helmstaedter, Moritz, Kevin L Briggman, and Winfried Denk (2008). "3D structural imaging of the brain with photons and electrons". In: *Current opinion in neurobiology* 18.6, pp. 633–641.

Rubin, Gerald M (2006). "Janelia Farm: an experiment in scientific culture". In: *Cell* 125.2, pp. 209–212.

Sunkin, Susan M, Lydia Ng, Chris Lau, Tim Dolbeare, Terri L Gilbert, Carol L Thompson, Michael Hawrylycz, and Chinh Dang (2012). "Allen Brain Atlas: an integrated spatio-temporal portal for exploring the central nervous system". In: *Nucleic acids research* 41.D1, pp. D996–D1008.

Keller, Anna Lena, Dirk Zeidler, and Thomas Kemen (2014). "High throughput data acquisition with a multi-beam SEM". In: *Scanning Microscopies 2014*. Vol. 9236. International Society for Optics and Photonics, 92360B.

O'Toole, Eileen, Peter van der Heide, J Richard McIntosh, and David Mastronarde (2018). "Large-scale electron tomography of cells using SerialEM and IMOD". In: *Cellular Imaging*. Springer, pp. 95–116.

Vescovi, Rafael and Hanyu Li (2020). *HappyNeurons*. github.com/ravescovi/HappyNeuron.

Gropp, William, Rajeev Thakur, and Ewing Lusk (1999). *Using MPI-2: Advanced features of the message passing interface*. MIT press.

Schneider-Mizell, Casey M, Agnes L Bodor, Forrest Collman, Derrick Brittain, Adam A Bleckert, Sven Dorkenwald, Nicholas L Turner, Thomas Macrina, Kisuk Lee, Ran Lu, et al. (2020). "Chandelier cell anatomy and function reveal a variably distributed but common signal". In: *bioRxiv*.

Motta, Alessandro, Manuel Berning, Kevin M Boergens, Benedikt Staffler, Marcel Beining, Sahil Loomba, Philipp Hennig, Heiko Wissler, and Moritz Helmstaedter (2019). "Dense connectomic reconstruction in layer 4 of the somatosensory cortex". In: *Science* 366.6469.

Wu, Jingpeng, William M Silversmith, and H Sebastian Seung (2019). "Chunkflow: Distributed Hybrid Cloud Processing of Large 3D Images by Convolutional Nets". In: *arXiv preprint arXiv:1904.10489*.

Boergens, Kevin M, Manuel Berning, Tom Bocklisch, Dominic Bräunlein, Florian Drawitsch, Johannes Frohnhofen, Tom Herold, Philipp Otto, Norman Rzepka, Thomas Werkmeister, et al. (2017). "webKnossos: efficient online 3D data annotation for connectomics". In: *nature methods* 14.7, pp. 691–694.

Maitin-Shepard, Jeremy (n.d.). *Neuroglancer*. github.com/google/neuroglancer.

Zheng, Zhihao, J Scott Lauritzen, Eric Perlman, Camenzind G Robinson, Matthew Nichols, Daniel Milkie, Omar Torrens, John Price, Corey B Fisher, Nadiya Sharifi, et al. (2018). "A complete electron microscopy volume of the brain of adult Drosophila melanogaster". In: *Cell* 174.3, pp. 730–743.

Dong, Wushi, Murat Keceli, Rafael Vescovi, Hanyu Li, Corey Adams, Elise Jennings, Samuel Flender, Thomas Uram, Venkatram Vishwanath, Nicola Ferrier, et al. (2019). "Scaling Distributed Training of Flood-Filling Networks on HPC Infrastructure for Brain Mapping". In: *2019 IEEE/ACM Third Workshop on Deep Learning on Supercomputers (DLS)*. IEEE, pp. 52–61.

Silversmith, William (n.d.[a]). *Cloudvolume*. github.com/seung-lab/cloud-volume.

— (n.d.[b]). *Igneous*. github.com/seung-lab/igneous.

Sato, Mie, Ingmar Bitter, Michael A Bender, Arie E Kaufman, and Masayuki Nakajima (2000). "TEASAR: Tree-structure extraction algorithm for accurate and robust skeletons". In: *Proceedings the Eighth Pacific Conference on Computer Graphics and Applications*. IEEE, pp. 281–449.

Folk, Mike, Gerd Heber, Quincey Koziol, Elena Pourmal, and Dana Robinson (2011). "An overview of the HDF5 technology suite and its applications". In: *Proceedings of the EDBT/ICDT 2011 Workshop on Array Databases*, pp. 36–47.

Allcock, William E, Benjamin S Allen, Rachana Ananthakrishnan, Ben Blaiszik, Kyle Chard, Ryan Chard, Ian Foster, Lukasz Lacinski, Michael E Papka, and Rick Wagner (2019). "Petrel: A Programmatically Accessible Research Data Service". In: *Proceedings of the*

Practice and Experience in Advanced Research Computing on Rise of the Machines (learning), pp. 1–7.

Salim, Michael A, Thomas D Uram, J Taylor Childers, Prasanna Balaprakash, Venkatram Vishwanath, and Michael E Papka (2019). "Balsam: Automated scheduling and execution of dynamic, data-intensive hpc workflows". In: *arXiv preprint arXiv:1909.08704*.

Carl Zeiss, SMT and VP Sigma (2011). *Detection principles based on GEMINI technology*.

Hua, Yunfeng, Philip Laserstein, and Moritz Helmstaedter (2015). "Large-volume en-bloc staining for electron microscopy-based connectomics". In: *Nature communications* 6.1, pp. 1–7.

Schalek, R, N Kasthuri, K Hayworth, D Berger, J Tapia, J Morgan, S Turaga, E Fagerholm, H Seung, and J Lichtman (2011). "Development of high-throughput, high-resolution 3D reconstruction of large-volume biological tissue using automated tape collection ultramicrotomy and scanning electron microscopy". In: *Microscopy and Microanalysis* 17.S2, pp. 966–967.

Li, Hanyu, Michał Januszewski, Viren Jain, and Peter H Li (2020). "Neuronal Subcompartment Classification and Merge Error Correction". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 88–98.

Dorkenwald, Sven, Nicholas L Turner, Thomas Macrina, Kisuk Lee, Ran Lu, Jingpeng Wu, Agnes L Bodor, Adam A Bleckert, Derrick Brittain, Nico Kemnitz, et al. (2019). "Binary and analog variation of synapses between cortical pyramidal neurons". In: *BioRxiv*.

Xu, C Shan, Michal Januszewski, Zhiyuan Lu, Shin-ya Takemura, Kenneth Hayworth, Gary Huang, Kazunori Shinomiya, Jeremy Maitin-Shepard, David Ackerman, Stuart Berg, et al. (2020). "A connectome of the adult drosophila central brain". In: *BioRxiv*.

Li, Peter H, Larry F Lindsey, Michał Januszewski, Mike Tyka, Jeremy Maitin-Shepard, Tim Blakely, and Viren Jain (2019). "Automated reconstruction of a serial-section EM Drosophila brain with flood-filling networks and local realignment". In: *Microscopy and Microanalysis* 25.S2, pp. 1364–1365.

Buhmann, Julia, Arlo Sheridan, Stephan Gerhard, Renate Krause, Tri Nguyen, Larissa Heinrich, Philipp Schlegel, Wei-Chung Allen Lee, Rachel Wilson, Stephan Saalfeld, et al. (2019). "Automatic detection of synaptic partners in a whole-brain Drosophila EM dataset". In: *bioRxiv*.

Dasgupta, Sanjoy, Charles F Stevens, and Saket Navlakha (2017). "A neural algorithm for a fundamental computing problem". In: *Science* 358.6364, pp. 793–796.

Swanson, Larry W and Jeff W Lichtman (2016). "From Cajal to connectome and beyond". In: *Annual Review of Neuroscience* 39, pp. 197–216.

Meirovitch, Yaron, Alexander Matveev, Hayk Saribekyan, David Budden, David Rolnick, Gergely Odor, Seymour Knowles-Barley, Thouis Raymond Jones, Hanspeter Pfister, Jeff William Lichtman, et al. (2016). "A multi-pass approach to large-scale connectomics". In: *arXiv preprint arXiv:1612.02120*.

Rolnick, David, Yaron Meirovitch, Toufiq Parag, Hanspeter Pfister, Viren Jain, Jeff W Lichtman, Edward S Boyden, and Nir Shavit (2017). "Morphological error detection in 3D segmentations". In: *arXiv preprint arXiv:1705.10882*.

Haehn, Daniel, Verena Kaynig, James Tompkin, Jeff W Lichtman, and Hanspeter Pfister (2018). "Guided proofreading of automatic segmentations for connectomics". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9319–9328.

Krasowski, NE, Thorsten Beier, GW Knott, Ullrich Köthe, Fred A Hamprecht, and Anna Kreshuk (2017). "Neuron segmentation with high-level biological priors". In: *IEEE transactions on medical imaging* 37.4, pp. 829–839.

Pape, Constantin, Alex Matskevych, Adrian Wolny, Julian Hennies, Giulia Mizzon, Marion Louveaux, Jacob Musser, Alexis Maizel, Detlev Arendt, and Anna Kreshuk (2019). "Leveraging domain knowledge to improve microscopy image segmentation with lifted multicuts". In: *Frontiers in Computer Science* 1, p. 6.

Hubbard, Philip M, Stuart Berg, Ting Zhao, Donald J Olbris, Lowell Umayam, Jeremy Maitin-Shepard, Michal Januszewski, William T Katz, Erika R Neace, and Stephen M Plaza (2020). "Accelerated EM connectome reconstruction using 3D visualization and segmentation graphs". In: *BioRxiv*.

Zuiderveld, Karel (1994). "Contrast limited adaptive histogram equalization". In: *Graphics gems*, pp. 474–485.

(PING, Petilla Interneuron Nomenclature Group et al. (2008). "Petilla terminology: nomenclature of features of GABAergic interneurons of the cerebral cortex". In: *Nature reviews. Neuroscience* 9.7, p. 557.

Contreras, April, Dustin J Hines, and Rochelle M Hines (2019). "Molecular specialization of GABAergic synapses on the soma and axon in cortical and hippocampal circuit function and dysfunction". In: *Frontiers in molecular neuroscience* 12, p. 154.

Jiang, Xiaolong, Shan Shen, Cathryn R Cadwell, Philipp Berens, Fabian Sinz, Alexander S Ecker, Saumil Patel, and Andreas S Tolias (2015). "Principles of connectivity among morphologically defined cell types in adult neocortex". In: *Science* 350.6264.

Gouwens, Nathan W, Staci A Sorensen, Fahimeh Baftizadeh, Agata Budzillo, Brian R Lee, Tim Jarsky, Lauren Alfiler, Katherine Baker, Eliza Barkan, Kyla Berry, et al. (2020). "Toward an integrated classification of cell types: morphoelectric and transcriptomic characterization of individual GABAergic cortical neurons". In:

Grünert, Ulrike and Paul R Martin (2020). "Cell types and cell circuits in human and non-human primate retina". In: *Progress in retinal and eye research*, p. 100844.

Kipf, Thomas N and Max Welling (2016). "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907*.

Qi, Charles R, Hao Su, Kaichun Mo, and Leonidas J Guibas (2017). "Pointnet: Deep learning on point sets for 3d classification and segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660.

Riegler, Gernot, Ali Osman Ulusoy, and Andreas Geiger (2017). "Octnet: Learning deep 3d representations at high resolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3577–3586.

Mescheder, Lars, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger (2019). "Occupancy networks: Learning 3d reconstruction in function space". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4460–4470.

Graham, Benjamin, Martin Engelcke, and Laurens Van Der Maaten (2018). "3d semantic segmentation with submanifold sparse convolutional networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9224–9232.

Neniskyte, Urte and Cornelius T Gross (2017). "Errant gardeners: glial-cell-dependent synaptic pruning and neurodevelopmental disorders". In: *Nature Reviews Neuroscience* 18.11, pp. 658–670.

Hubel, David H and Torsten N Wiesel (1968). "Receptive fields and functional architecture of monkey striate cortex". In: *The Journal of physiology* 195.1, pp. 215–243.

Antonini, Antonella, Michela Fagiolini, and Michael P Stryker (1999). "Anatomical correlates of functional plasticity in mouse visual cortex". In: *Journal of Neuroscience* 19.11, pp. 4388–4406.

Antonini, Antonella and Michael P Stryker (1993). "Rapid remodeling of axonal arbors in the visual cortex". In: *Science* 260.5115, pp. 1819–1821.

Bourgeois, Jean-Pierre and Pasko Rakic (1993). "Changes of synaptic density in the primary visual cortex of the macaque monkey from fetal to adult stage". In: *Journal of Neuroscience* 13.7, pp. 2801–2820.

Guido, William (2008). "Refinement of the retinogeniculate pathway". In: *The Journal of physiology* 586.18, pp. 4357–4362.

Hashimoto, Kouichi and Masanobu Kano (2013). "Synapse elimination in the developing cerebellum". In: *Cellular and molecular life sciences* 70.24, pp. 4667–4680.

D'Orazi, Florence D, Sachihiro C Suzuki, and Rachel O Wong (2014). "Neuronal remodeling in retinal circuit assembly, disassembly, and reassembly". In: *Trends in neurosciences* 37.10, pp. 594–603.

Buchanan, JoAnn, Leila Elabbady, Forrest Collman, Nikolas L Jorstad, Trygve E Bakken, Carolyn Ott, Jenna Glatzer, Adam A Bleckert, Agnes L Bodor, Derrick Brittan, et al. (2021). "Oligodendrocyte precursor cells prune axons in the mouse neocortex". In: *bioRxiv*.

Crowley, Justin C and Lawrence C Katz (2000). "Early development of ocular dominance columns". In: *Science* 290.5495, pp. 1321–1324.

Moyer, Caitlin E and Yi Zuo (2018). "Cortical dendritic spine development and plasticity: insights from in vivo imaging". In: *Current opinion in neurobiology* 53, pp. 76–82.

Kasthuri, Narayanan and Jeff W Lichtman (2004). "Structural dynamics of synapses in living animals". In: *Current opinion in neurobiology* 14.1, pp. 105–111.

Horton, Jonathan C and Davina R Hocking (1996). "An adult-like pattern of ocular dominance columns in striate cortex of newborn monkeys prior to visual experience". In: *Journal of Neuroscience* 16.5, pp. 1791–1807.

Grutzendler, Jaime, Narayanan Kasthuri, and Wen-Biao Gan (2002). "Long-term dendritic spine stability in the adult cortex". In: *Nature* 420.6917, pp. 812–816.

Witvliet, Daniel, Ben Mulcahy, James K Mitchell, Yaron Meirovitch, Daniel R Berger, Yuelong Wu, Yufang Liu, Wan Xian Koh, Rajeev Parvathala, Douglas Holmyard, et al. (2021). "Connectomes across development reveal principles of brain maturation". In: *BioRxiv*, pp. 2020–04.

Lein, Ed S, Michael J Hawrylycz, Nancy Ao, Mikael Ayres, Amy Bensinger, Amy Bernard, Andrew F Boe, Mark S Boguski, Kevin S Brockway, Emi J Byrnes, et al. (2007). "Genome-wide atlas of gene expression in the adult mouse brain". In: *Nature* 445.7124, pp. 168–176.

Brain Science, Allen Institute for (2007). *Allen Mouse Brain Atlas*. atlas.brain-map.org.

Majewska, Ania and Mriganka Sur (2003). "Motility of dendritic spines in visual cortex in vivo: changes during the critical period and effects of visual deprivation". In: *Proceedings of the National Academy of Sciences* 100.26, pp. 16024–16029.

Tapia, Juan C, John D Wylie, Narayanan Kasthuri, Kenneth J Hayworth, Richard Schalek, Daniel R Berger, Cristina Guatimosim, H Sebastian Seung, and Jeff W Lichtman (2012). "Pervasive synaptic branch removal in the mammalian neuromuscular system at birth". In: *Neuron* 74.5, pp. 816–829.

Ofer, Netanel, Daniel R Berger, Narayanan Kasthuri, Jeff W Lichtman, and Rafael Yuste (2021). "Ultrastructural analysis of dendritic spine necks reveals a continuum of spine morphologies". In: *Developmental Neurobiology*.

Yuste, Rafael and Tobias Bonhoeffer (2004). "Genesis of dendritic spines: insights from ultrastructural and imaging studies". In: *Nature Reviews Neuroscience* 5.1, pp. 24–34.

Miller, Michael and Alan Peters (1981). "Maturation of rat visual cortex. II. A combined Golgi-electron microscope study of pyramidal neurons". In: *Journal of Comparative Neurology* 203.4, pp. 555–573.

Konur, Sila and Rafael Yuste (2004). "Developmental regulation of spine and filopodial motility in primary visual cortex: reduced effects of activity and sensory deprivation". In: *Journal of neurobiology* 59.2, pp. 236–246.

Divakaruni, Sai Sachin, Adam M Van Dyke, Ramesh Chandra, Tara A LeGates, Minerva Contreras, Poorna A Dharmasri, Henry N Higgs, Mary Kay Lobo, Scott M Thompson, and Thomas A Blanpied (2018). "Long-term potentiation requires a rapid burst of dendritic mitochondrial fission during induction". In: *Neuron* 100.4, pp. 860–875.

Rangaraju, Vidhya, Marcel Lauterbach, and Erin M Schuman (2019). "Spatially stable mitochondrial compartments fuel local translation during plasticity". In: *Cell* 176.1-2, pp. 73–84.

Martell, Jeffrey D, Thomas J Deerinck, Stephanie S Lam, Mark H Ellisman, and Alice Y Ting (2017). "Electron microscopy using the genetically encoded APEX2 tag in cultured mammalian cells". In: *Nature protocols* 12.9, pp. 1792–1816.

Sampathkumar, Vandana, Andrew Miller-Hansen, S Murray Sherman, and Narayanan Kasthuri (2021). "An ultrastructural connectomic analysis of a higher-order thalamocortical circuit in the mouse". In: *European Journal of Neuroscience* 53.3, pp. 750–762.

Wildenberg, Gregg, MR Rosen, Jack Lundell, Dawn Paukner, David J Freedman, and Narayanan Kasthuri (2020). "Primate neuronal connections are sparse as compared to mouse". In: *Available at SSRN 3757075*.

Schindelin, Johannes, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, et al. (2012). "Fiji: an open-source platform for biological-image analysis". In: *Nature methods* 9.7, pp. 676–682.

Sergeev, Alexander and Mike Del Balso (2018). "Horovod: fast and easy distributed deep learning in TensorFlow". In: *arXiv preprint arXiv:1802.05799*.

Li, Hanyu (2020a). *EM_mask*. github.com/Hanyu-Li/EM_mask.

— (2020b). *klab_utils*. github.com/Hanyu-Li/klab_utils.

Januszewski, Michał, Jeremy Maitin-Shepard, Peter Li, Jörgen Kornfeld, Winfried Denk, and Viren Jain (2016). "Flood-filling networks". In: *arXiv preprint arXiv:1611.00421*.

Silversmith, William and J. Alexander Bae (2020). *Kimimaro: Skeletonize densely labeled 3D image segmentations*. github.com/seung-lab/kimimaro.

Wei, Donglai, Kisuk Lee, Hanyu Li, Ran Lu, J Alexander Bae, Zequan Liu, Lifu Zhang, Márcia dos Santos, Zudi Lin, Thomas Uram, et al. (2021). "AxonEM Dataset: 3D Axon Instance Segmentation of Brain Cortical Regions". In: *arXiv preprint arXiv:2107.05451*.

Sarma, Gopal P, Chee Wai Lee, Tom Portegys, Vahid Ghayoomie, Travis Jacobs, Bradly Alicea, Matteo Cantarelli, Michael Currie, Richard C Gerkin, Shane Gingell, et al. (2018). "OpenWorm: overview and recent advances in integrative biological simulation of Caenorhabditis elegans". In: *Philosophical Transactions of the Royal Society B* 373.1758, p. 20170382.

Scheffer, Louis K, C Shan Xu, Michal Januszewski, Zhiyuan Lu, Shin-ya Takemura, Kenneth J Hayworth, Gary B Huang, Kazunori Shinomiya, Jeremy Maitlin-Shepard, Stuart Berg, et al. (2020). "A connectome and analysis of the adult Drosophila central brain". In: *Elife* 9, e57443.

Zheng, Zhihao, Feng Li, Corey Fisher, Iqbal J Ali, Nadiya Sharifi, Steven Calle-Schuler, Joseph Hsu, Najla Masoodpanah, Lucia Kmecova, Tom Kazimiers, et al. (2020). "Structured sampling of olfactory input by the fly mushroom body". In: *BioRxiv*.

Bates, Alexander S, Philipp Schlegel, Ruairi JV Roberts, Nikolas Drummond, Imaan FM Tamimi, Robert Turnbull, Xincheng Zhao, Elizabeth C Marin, Patricia D Popovici, Serene Dhawan, et al. (2020). "Complete connectomic reconstruction of olfactory projection neurons in the fly brain". In: *Current Biology* 30.16, pp. 3183–3199.

Otto, Nils, Markus W Pleijzier, Isabel C Morgan, Amelia J Edmondson-Stait, Konrad J Heinz, Ildiko Stark, Georgia Dempsey, Masayoshi Ito, Ishaan Kapoor, Joseph Hsu, et al. (2020). "Input connectivity reveals additional heterogeneity of dopaminergic reinforcement in Drosophila". In: *Current Biology* 30.16, pp. 3200–3211.

Marin, Elizabeth C, Laurin Büld, Maria Theiss, Tatevik Sarkissian, Ruairı JV Roberts, Robert Turnbull, Imaan FM Tamimi, Markus W Pleijzier, Willem J Laursen, Nik Drummond, et al. (2020). "Connectomics analysis reveals first-, second-, and third-order thermosensory and hygrosensory neurons in the adult Drosophila brain". In: *Current Biology* 30.16, pp. 3167–3182.

Minnen, David Charles, Michal Januszewski, Alex Shapson-Coe, Richard L Schalek, Johannes Ballé, Jeff W Lichtman, and Viren Jain (2021). "Denoising-based Image Compression for Connectomics". In:

Januszewski, Michał and Viren Jain (2019). "Segmentation-enhanced CycleGAN". In: *bioRxiv*, p. 548081.

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žıdek, Anna Potapenko, et al. (2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature*, pp. 1–11.

Vogelstein, Joshua T, Eric Perlman, Benjamin Falk, Alex Baden, William Gray Roncal, Vikram Chandrashekhar, Forrest Collman, Sharmishtaa Seshamani, Jesse L Patsolic, Kunal Lillaney, et al. (2018). "A community-developed open-source computational ecosystem for big neuro data". In: *Nature methods* 15.11, pp. 846–847.

Turing, Alan M (1951). "Can digital computers think?" In: *The Turing Test: Verbal Behavior as the Hallmark of Intelligence*.

Turing, Alan Mathison (1990). "The chemical basis of morphogenesis". In: *Bulletin of mathematical biology* 52.1, pp. 153–197.

Erdos, Paul, Alfréd Rényi, et al. (1960). "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1, pp. 17–60.

Markram, Henry (2006). "The blue brain project". In: *Nature Reviews Neuroscience* 7.2, pp. 153–160.

Givon, Lev E and Aurel A Lazar (2016). "Neurokernel: an open source platform for emulating the fruit fly brain". In: *PloS one* 11.1, e0146581.

Frégnac, Yves and Gilles Laurent (2014). "Neuroscience: Where is the brain in the Human Brain Project?" In: *Nature News* 513.7516, p. 27.

Mainen, Zachary F, Michael Häusser, and Alexandre Pouget (2016). "A better way to crack the brain". In: *Nature News* 539.7628, p. 159.

Rosenblatt, Frank (1957). *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.

Yamins, Daniel LK and James J DiCarlo (2016). "Using goal-driven deep learning models to understand sensory cortex". In: *Nature neuroscience* 19.3, pp. 356–365.

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. (2015). "Human-level control through deep reinforcement learning". In: *nature* 518.7540, pp. 529–533.

Banino, Andrea, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J Chadwick, Thomas Degris, Joseph Modayil, et al. (2018). "Vector-based navigation using grid-like representations in artificial agents". In: *Nature* 557.7705, pp. 429–433.

Lillicrap, Timothy P, Adam Santoro, Luke Marris, Colin J Akerman, and Geoffrey Hinton (2020). "Backpropagation and the brain". In: *Nature Reviews Neuroscience* 21.6, pp. 335–346.

Buhmann, Julia, Arlo Sheridan, Stephan Gerhard, Renate Krause, Tri Nguyen, Larissa Heinrich, Philipp Schlegel, Wei-Chung Allen Lee, Rachel Wilson, Stephan Saalfeld, et al. (2020). "Automatic detection of synaptic partners in a whole-brain Drosophila EM dataset". In: *bioRxiv*, pp. 2019–12.

Shannon, Claude E. (1948). "A Mathematical Theory of Communication". In: *Bell System Technical Journal* 27.3, pp. 379–423.

Kluyver, Thomas, Benjamin Ragan-Kelley, Fernando Pérez, Brian E Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica B Hamrick, Jason Grout, Sylvain Corlay,

et al. (2016). “Jupyter Notebooks-a publishing format for reproducible computational workflows.” In: *ELPUB*, pp. 87–90.

Hadash, Guy, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi (2018). “Estimate and Replace: A Novel Approach to Integrating Deep Neural Networks with Existing Applications”. In: *arXiv preprint arXiv:1804.09028*.

Chen, Chia-Chien, Ju Lu, and Yi Zuo (2014). “Spatiotemporal dynamics of dendritic spines in the living brain”. In: *Frontiers in neuroanatomy* 8, p. 28.

Holler, Simone, German Köstinger, Kevan AC Martin, Gregor FP Schuhknecht, and Ken J Stratford (2021). “Structure and function of a neocortical synapse”. In: *Nature* 591.7848, pp. 111–116.

De Vivo, Luisa, Michele Bellesi, William Marshall, Eric A Bushong, Mark H Ellisman, Giulio Tononi, and Chiara Cirelli (2017). “Ultrastructural evidence for synaptic scaling across the wake/sleep cycle”. In: *Science* 355.6324, pp. 507–510.