2023

Multimodal Neuron Classification based on Morphology and Electrophysiology

Aqib Ahmad
West Virginia University, aa00057@mix.wvu.edu

Multimodal Neuron Classification based on Morphology and Electrophysiology

Aqib Ahmad

Thesis submitted to the
Statler College of Engineering and Mineral Resources
at West Virginia University

In partial fulfillment of the requirements for the degree of
Master of Science in Computer Science

Gianfranco Doretto, Ph.D., Chair
Donald Adjeroh, Ph.D.
Katerina Goseva-Popstojanova, Ph.D.

LANE DEPARTMENT OF COMPUTER SCIENCE
AND ELECTRICAL ENGINEERING

Morgantown, WV
2023

Keywords: neuron classification, multimodality, morphology, electrophysiology

Abstract

Multimodal Neuron Classification based on Morphology and Electrophysiology

by Aqib Ahmad

Categorizing neurons into different types to understand neural circuits and ultimately brain function is a major challenge in neuroscience. While electrical properties are critical in defining a neuron, its morphology is equally important. Advancements in single-cell analysis methods have allowed neuroscientists to simultaneously capture multiple data modalities from a neuron. We propose a method to classify neurons using both morphological structure and electrophysiology. Current approaches are based on a limited analysis of morphological features. We propose to use a new graph neural network to learn representations that more comprehensively account for the complexity of the shape of neuronal structures. In addition, we design a self-supervised approach for learning representations of electrophysiology data with a convolutional neural network. Morphological and electrophysiology representations are then fused in different ways in an end-to-end approach. Our methods are tested on multiple datasets, and we show a performance that exceeds the state of the art in the single modalities, while we also establish the first baseline for the combined modalities.

Acknowledgements

I would first and foremost like to thank my thesis advisor, Dr. Gianfranco Doretto, for his support throughout my time as a Master's student in his lab. In only two years' time, I gained an immense amount of knowledge on machine learning, computer vision, and most vitally, how to pursue meaningful questions and conduct deep, thoughtful, and valuable research under his guidance.

Additionally, I would like to express my gratitude to Dr. Katerina Goseva-Popstojanova and Dr. Donald Adjeroh. As members of the advisory and examining committee, their valuable direction and efforts have not only made this work more rewarding but have also honed my research skills in the domain. I am also obliged to Dr. Brian Powell, who provided me with an opportunity to serve as a Graduate Teaching Assistant in LCSEE.

The whole of life, from the moment you are born to the moment you die, is a process of learning. Throughout the eighteen years of my formal education, I have learned from my parents, my family, my teachers, from my mentors, my peers and my colleagues. In the famous words of Ali ibn Abi Talib, "If a person teaches me one single word, he has made me his servant for a lifetime". Indeed, I am indebted to all the people who made me what I am today.

Especially, I would like to mention my parents who left no stone unturned in my upbringing. Not only did they ensure that I always received quality education, they helped me become a better character by instilling moral and ethical values. My brothers, who have excelled in their disciplines, have been ideal role models for me. Their emotional support and technical guidance brought me all the way from Pakistan to the United States to pursue graduate education. Lastly, I would like to acknowledge my friends who have always appreciated my achievements and have been a constant source of joy for me over the years.

Research reported in this publication was supported by the National Institute on Deafness and Other Communication Disorders (NIDCD) of the National Institutes of Health under Award Number R01DC015901. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This material is also based upon work supported in part by the National Science Foundation under Grants No. 1920920 and 2125872. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Contents

List of Figures

List of Tables

List of Abbreviations

AP	**A**ction **P**otential
DNA	**D**eoxyribonucleic **A**cid

1 Introduction

1.1 Background

As part of the background, we will first introduce and describe neurons from a neuroscience aspect. After that, we will take a brief look at the work done in machine learning for neuron classification. Neurons are the basic building block of an animal nervous system. A nervous system is the command center and a critical part of an animal body. In addition to reacting to stimuli in the environment, it coordinates multiple body functions. For instance, when you touch something sharp like a needle, sensory neurons in the fingertips will sense the sharp stimulus and generate a signal. With the help of intermediate neurons (interneurons), this signal is passed to the motor neurons in your hand. Motor neurons control movement and they will allow you to move your hand away.

Neurons use electrochemical signals to communicate information between different areas of the body. A neuron has three main parts: cell body (soma), axon, and dendrites. The soma or cell body is the central part of a neuron that contains the cell nucleus and DNA. It is the part where an electrical signal is initially generated. Axon is the output structure of a neuron. The generated electrical signal is propagated through the axon which is connected to the dendrites of other neurons. A dendrite, hence, is where a neuron receives input signals from other cells. The connection point between an axon of one neuron and the dendrite of another neuron is called a synapse. At a synapse, the electrical signal is converted into a chemical signal.

To summarize, neurons fire electrical signals (action potentials) that cause them to release a chemical signal (neurotransmitter). The action potential is actually a rapid change in the cell membrane voltage. Neuroscientists often call action potentials as 'spikes', referring to the shape of an actional potential when recorded in a digital waveform. The neurotransmitter released is a chemical molecule that promotes the flow of either positively charged ions or negatively charged ions across the cell membrane of the connected neuron. Depending on this, it can either help excite or inhibit the next neuron from firing its own action potential. The sum of all excitatory and inhibitory inputs to a single neuron will determine whether an action potential will occur or not. If the sum of inputs makes neuron's membrane potential reach the action potential threshold value (typically around -50 mV), then an actional potential will be generated.

Nervous system is critical to an animal body. Therefore, it is important that

we establish a good understanding of the brain and other nervous system organs. However, it is not that easy because of the complexity of their building blocks, the neurons. We gave an example of how the hand sensed a sharp needle and moved away using a combination of three neurons types: sensory, inter and motor neuron types. These types, at a very basic level, classify neurons according to their function. However, they are not sufficient for developing an in-depth understanding of neurons.

This is because neurons will vary extensively in their properties (such as structural, electrical and molecular characteristics, or neural functions and behaviors) within a broad cell type like sensory neurons. Therefore, there is a need of a taxonomic scheme that can classify neurons into more specific cell types with similar properties. This is a major challenge in neuroscience. While new and better cell types have been discovered, a single cell type still contains varying neurons. Even today, neuroscientists have been quoted saying, "We know very little about the brain".

In addition to gaining an understanding of the nervous system, neuron classification is also important for studying and treating brain disorders. We can treat nervous system disorders if we can find the neuron cell types that they target. For instance, amyotrophic lateral sclerosis affects upper and lower motor neurons, and congenital nystagmus affects starburst amacrine cells. Consequently, developing a superior classification method can help tackle a variety of diseases.

Neuroscientists have utilized various modalities for neuron classification task. This includes morphology, electrophysiology and transcriptomy. Morphology cell types are defined on the basis of form or structure of neurons. For instance, basket neurons are named after their basket-like axon branching pattern. The axon branching of chandelier neurons resembles a candelabrum. Electrophysiology cell types are defined according to the electrical properties of neurons. For instance, fast-spiking (FS) neurons are characterized as firing periodic trains of action potentials with extremely high frequency. Low-threshold spiking (LTS) neurons have a low action potential threshold value. Transcriptomic cell types are defined on the basis of genetic makeup of neurons. For instance, parvalbumin-expressing (Pvalb), and the somatostatin-expressing (Sst) cells are named after their characterizing genes, Pvalb and Sst.

We have seen the importance of categorizing neurons into appropriate cell types. However, manually classifying each neuron requires laborious study of neuronal data. To classify on basis of morphology, a neuroscientist would need to carefully inspect the branching pattern before making a decision. To classify on basis of electrophysioloy, a neuroscientist would need to visualize the digital waveform for each neuron. With the introduction of high-throughput neuron data extraction techniques, manual classification has become a bottleneck for neuron analysis. There is a need of an automatic classification method that can directly consume neuronal data and accurately assign a cell-type to the input

neuron without any human intervention. This will greatly benefit neuroscientists and save them time and effort. Ultimately, this will speed up our efforts in understanding the brain.

Fortunately, machine learning algorithms have been used for similar purpose in other domains and can be applied here as well. Moreover, such algorithms are not biased toward certain datasets, like human beings, who may have personal biases that can distort their judgment. As a result, machine learning can help reduce bias and outperform neuroscientists in neuron classification accuracy. We will take a brief look here at the work done in the machine learning domain for neuron classification in all three modalities.

The recent brain microscopy and neuron fiber tracing methods have led to an abundance of reconstructed neuronal morphology data. NeuroMorpho.org [1] is an online repository of publicly shared neuron reconstructions. Traditionally for classification purposes, the three-dimensional spatial and geometric structure of the neurons itself was ignored. Instead, using the L-measure [2] tool, this data was converted to a set of morphological features, also referred to as morphometrics. The performance of such classifiers is limited because a small difference in the spatial structure may change these morphometrics substantially.

Recently, by treating a single neuron sample as a point cloud, [3] experimented with applying point cloud networks for morphological classification and came up with an improved model called MorphoGNN. This approach leverages the DGCNN [4] architecture with some key modifications like dense EdgeConv connections, increasing K value for local KNN graph, joint triplet and cross-entropy loss function, and a double pooling module. MorphoGNN outperforms DGCNN and traditional morphometrics by 6.25% and 12.6% in overall accuracy respectively.

The patch clamp technique has allowed scientists to create datasets of electrophysiology recordings [5]. A stimulus of electric current is injected and the membrane potential response is recorded in the form of a digital waveform. However, traditionally for classification purposes, the raw waveform was converted to a set of hand-crafted features [6], [7]. These features describe the action potential (AP) peak present in the recorded voltage response. Examples of these features are upstroke-downstroke ratio, the AP width and AP threshold.

The raw waveform contains a wealth of information that is difficult to be expressed using a limited number of hand-crafted features. Moreover, manual feature extraction will suffer from inefficient features that can lead to bad class separability. Therefore, the performance of classifiers using these AP features will always be limited.

Recently, the single-cell RNA sequencing (scRNA-seq) technique has allowed researchers to express the transcriptomics (genetic makeup) of cells. These transcriptomics are defined by the gene expression levels that are measured in counts per million (cpm). According to the expression levels, a transcriptomic (T-type) cell type is assigned to the neuron. The most commonly

used transcriptomic sub-types are Sst (somatostatin), Pvalb (parvalbumin) and Vip (vasoactive intestinal peptide).

1.2 Current Challenges and Motivation

Neurons are critical to an animal nervous system so it is important that we establish a solid understanding of how they work. To do this, researchers have devoted significant efforts to classifying neurons into various cell types in order to relate the cell types with certain properties, functions and behaviors. For instance, a neural circuit has been identified in mice brain that processes fear and anxiety. Such neurons have been found to have distinct axonal patterns in their morphology. Another example is of pyramidal neuron which is a morphology-based cell type. All such neurons are excitatory in nature, meaning they release an exciting neurotransmitter.

In addition to gaining an understanding of the nervous system, neuron classification is also important for studying and treating brain disorders. We can treat nervous system disorders if we can find the neuron cell types that they target. For instance, amyotrophic lateral sclerosis affects upper and lower motor neurons, and congenital nystagmus affects starburst amacrine cells.

We saw the importance of categorizing neurons into appropriate cell types. However, manually classifying each neuron requires laborious study of neuronal data. To classify on basis of morphology, a neuroscientist would need to carefully inspect the branching pattern before making a decision. To classify on basis of electrophysioloy, a neuroscientist would need to visualize the digital waveform for each neuron.

With the introduction of high-throughput neuron data extraction techniques, manual classification has become a bottleneck for neuron analysis. There is a need of an automatic classification method that can directly consume neuronal data and accurately assign a cell-type to the input neuron without any human intervention. This will greatly benefit neuroscientists and save them time and effort. Ultimately, this will speed up our efforts in understanding the brain.

Fortunately, machine learning algorithms have been used for similar purpose in other domains and can be applied here as well. Moreover, such algorithms are not biased toward certain datasets, like human beings, who may have personal biases that can distort their judgment. As a result, machine learning can help reduce bias and outperform neuroscientists in neuron classification accuracy.

We have seen the use of individual modalities to classify and understand neurons. However, there are many cases where within a given cell type based on a particular modality, neurons will vary widely in the properties of other modalities. For instance, neurons within the Vip transcriptomic sub-type can show both Regular Spiking (RS) or Low Threshold Spiking (LTS) electrophysiology. Hence, there is a strong need to use more than one modality to define

neuron types. Primarily, the reason behind lack of multimodal analysis has been the inability to extract multiple modalities from a single neuron at scale.

Recently, the patch-seq technique has enabled the extraction of multiple modalities simultaneously from a neuron. Using this method, [5] has provided a dataset consisting of morphologies and electrophysiologies of 517 neurons. Through clustering analysis, they have also defined 28 MET-types that have aligned morphological, electrophysiological, and transcriptomic properties. Such data can be leveraged for multimodal neuron classification. This can be really beneficial for neuroscientists as an incoming neuron can be automatically classified into its cell type to get an idea about its morphological, electrophysiological, and transcriptomic properties at once.

We also note the use of manually crafted features for unimodal analysis in the literature. These features are a limited representation of the input modality and can lead to bad class separation and eventually, bad performance. The challenge is to create an approach that can learn powerful representations of the modalities by itself. For image data, we have seen how CNNs have replaced traditional feature extractors by learning efficient features that represent the input in much more detail. Intuitively, the idea is to have a model that can learn a morphological or electrophysiology feature representation directly from the raw input. This is not straightforward as it requires dedicated approaches that can exploit the unique structure of these neuronal modalities.

1.3 Thesis Contribution

In this work, we propose a deep learning based approach that can learn powerful representations of morphology and electrophysiology of neurons and use both these representations to classify neurons in an end-to-end fashion. Our method entirely eliminates the use of hand-crafted features in the classification pipeline and helps neuroscientists map an input neuron to its morphological, electrophysiological, and transcriptomic properties at once. This eliminates the need for laborius manual study of neuronal data to get an idea about its properties.

For simplicity, we will divide our contributions under three major headings: morphology classification, electrophysiology classification and multimodal classification. For morphology classification, we take a state-of-the-art graph neural network and modify it to learn morphological representation directly from the three-dimensional structure of neuron. For electrophysiology, we devise a self-supervised classification pipeline that learns representation directly from the raw waveform. For multimodal classification, morphology and electrophysiology representations are fused in different ways in an end-to-end approach. Our methods are tested on multiple datasets, and we show a performance that exceeds the state of the art in the single modalities, while we also establish the first baseline for the combined modalities.

2 Literature Review

2.1 Morphology

After images of a neuron are obtained through a microscope, they can be converted to a digital representation by obtaining the 3D coordinates of neuronal pathways (axons and dendrites). This process is called neuron tracing (or neuron reconstruction) and it is a fundamental task in computational neuroscience. Neurons may be traced manually by hand-painting the trajectory of axons and dendrites. Automated reconstruction can also be done by fitting compartments (spheres or tubes) on the neuronal pathways.

The reconstructed morphology data then consists of series of discretized neuronal compartments that are connected together in space. The data file format is called SWC and each row of this file identifies a particular neuronal compartment. The attributes included are type of compartment, 3D coordinates, radius and parent of the compartment.

Instead of analyzing the 3D structure itself, traditionally, SWC data was converted to a set of morphological features for neuron classification purposes. The L-Measure tool [2] has been used by various researchers to convert 3D structure into quantitative measurements. A large feature set can be generated including features like the number of branches, surface area of a compartment, distance of a compartment with respect to soma, ratio between the distance and path length.

We know that a process of manual feature extraction is bound to suffer from inefficient features that can lead to bad class separability. Therefore, the performance of classifiers using these morphometrics will be limited. Moreover, we have seen in the image domain, how CNNs have replaced traditional feature extractors by learning efficient features that represent the input in much more detail. Intuitively, the idea is to have a model that can learn to classify neurons directly from the 3D structure.

If we discard other attributes and keep only the x, y and z coordinates from the SWC data, the neuron can be represented as a point cloud. Recently, there has been a focus on building deep neural networks that directly consume point cloud data for either 3D classification or segmentation. As point clouds have an irregular structure, the task is not straightforward. PointNet [8] achieves this task by incorporating three key modules in its network architecture. A joint alignment network for transformation invariance, max pooling layer for permutation invariance and local and global feature aggregation for segmentation

tasks.

PointNet processes each point independently to compute per-point features and finally aggregates all the individual point features to a global signature. The resultant model learns to summarize the input point cloud by a sparse set of key points. It shows impressive results, however without any convolution-like operation, it fails to capture the local neighborhood structure. PointNet++ [9] works by running PointNet in a hierarchical fashion that allows to extract local features at increasing contextual scales. Hence, this improves upon the performance of the vanilla model.

This technique still processes the points independently at the different hierarchy stages. Thus, it neglects the local geometric structure of points. DGCNN [4] implements a graph convolutional neural network to address this drawback. In particular, it applies a new convolution-like operation called EdgeConv on the edges connecting a point to its neighbors in a local graph. It generates edge features that describe the relationships between a point and its neighbors, thereby capturing the local geometric structure. DGCNN outperforms PointNet++ by 2.2% in overall accuracy.

All of the previously mentioned networks were trained on non-neuron point cloud datasets by the authors. Recently, [3] experimented with training these networks on a neuron point cloud dataset and came up with an improved model as well called MorphoGNN. This approach leverages the DGCNN architecture with some key modifications like dense EdgeConv connections, increasing K value for local KNN graph, joint triplet and cross-entropy loss function, and a double pooling module. MorphoGNN outperforms DGCNN by 6.25% and 8.84% in overall accuracy and mean class accuracy respectively. The neuron dataset for this experiment was collected from [1]. It is divided into seven classes based on the neuron morphology.

2.2 Electrophysiology

Neuron electrophysiology is described by the action potential induced in response to injecting a stimulus of electric current. The use of hand-crafted features to describe the action potential has been common. Some examples of these features are upstroke-downstroke ratio, inter-spike interval and average firing rate. The amount of information that is contained within a raw waveform can not be completely expressed using a limited number of hand-crafted features. While these features work well to classify neurons into the major transcriptomic subclasses, their performance will be limited in more complex tasks such as determining the T-type from the electrophysiology. Therefore, there is a need of learned representation that can more accurately depict the electrophysiology.

[5] has provided a dataset of electrophysiology for 4,435 cells from the mouse visual cortex. Based on their transcriptomes, these cells were assigned to one of the 60 transcriptomically defined cell types (T-type label). Unfortunately, this

dataset is heavily imbalanced with some classes having less than 5 samples. [7] uses the hand-crafted electrophysiological features to find four distinct classes of neurons in the avian brain region. Type 1 neurons had longer time-to-peak of an afterhyperpolarization. Type 2 had a more negative resting membrane potential than the other classes. Type 3 neurons had a shorter action-potential duration and Type 4 neurons had a much larger input resistance.

2.3 Multimodal Analysis

With the introduction of the Patch-seq [10] technique, researchers have been able to extract multiple modalities simultaneously from a single neuron. Recently, [5] has provided a multimodal dataset consisting of morphologies and electrophysiologies of 517 neurons. Through clustering analysis, they also define 28 met-types that have congruent morphological, electrophysiological, and transcriptomic properties. This data can be leveraged for multimodal neuron classification.

[11] explains the different data fusion techniques used in biomedical deep learning. These fusion strategies are grouped under the three major categories of early fusion, intermediate fusion and late fusion. Early fusion involves the concatenation of features of different modalities at the input level. An autoencoder may be used to map the input features onto a lower dimensional space that is more informative. The resulting input vector is fed to a deep learning architecture. In this approach, cross-modality and within-modality correlations are learned simultaneously at a low-level of abstraction.

Intermediate fusion first involves learning features separately for the different modalities to discover within-modality correlations. These learned features are also called marginal representations. The marginal representations are then fused for either direct classification or they are further fed to a multi-layer perceptron for joint representation learning. The joint representation will discover the cross-modality correlations.

Late fusion is a passive strategy as it does not involve representation learning but it is based on the process of averaging the outputs of the models for different modalities. For classification purposes, this can be as simple as averaging probabilities from the softmax layer for each class. This will obviously not learn the cross-modality interactions. However, it can be beneficial when trying to fuse heterogenous data or models.

3 Problem Definition

We are interested in building a model to classify neurons into their cell types. Cells assigned to a single-modality based cell type show a lot of variation in the used modality. Therefore, machine learning based approaches that utilize hand-crafted features are limited in their performance. There is a need for an approach that can learn accurate feature representations itself to account for the complexities in neuronal data. Moreover, cells assigned to a single-modality based cell type will also show variation across other neuronal modalities. This makes the classification task complex and requires the use of multiple modalities to define and classify a neuron.

It is time consuming to manually study properties of neurons by examining data for different modalities like morphological structure and electrophysiology action potential. It will be really helpful to develop an automated technique to classify neurons into their cell types to get an idea about their properties. We define the problem of multimodal neuron classification. Specifically, the problem requires a data sample consisting of both morphology $\mathcal{M}$ and electrophysiology $\mathcal{E}$. We then define two feature extractors F_M and F_E that will map $\mathcal{M}$ and $\mathcal{E}$ to learned feature representations $\mathcal{M}_L$ and $\mathcal{E}_L$ respectively. $*$ is defined as a fusion operator. $\mathcal{M}_L * \mathcal{E}_L$ fuses the learned representations together to an intermediate embedding. Finally, a classifier C takes the fusion output and generates the class scores by $C(\mathcal{M}_L * \mathcal{E}_L)$.

Given the specific setting of our problem, to the best of our knowledge, this is the first work to propose a deep learning based method for the task of multimodal neuron classification.

4 Proposed Methods

4.1 Morphological Classification

For morphological classification, we use the graph neural network introduced in [3], also called MorphoGNN. The approach treats the neuron morphology as a point cloud by only considering the spatial coordinates of neuronal compartments. Hence, the input to the network is the point cloud P as described in Eq. (4.1):

$$P = \left\{ P_i \in R^3 = \{x_i, y_i, z_i\} \;\; | i = 1, 2, ..., n \right\} \tag{4.1}$$

The morphology file format (SWC) stores the neuron as a tree structure and therefore each node (compartment) has a parent attribute assigned to it. However, shape of neurons is determined by the positioning of nodes in space, regardless of these local connections. Therefore, it is safe to omit the parent attribute from input. Additionally, there is a radius attribute. However, the radius measurements obtained are related to the parameters of laboratory techniques like fluorescent strength, and not with the morphology. Therefore, that is ignored from analysis as well.

In the graph neural network, the points in the point cloud form a local graph with the nearest K points. New features of each node are learned from the neighboring nodes. As node features are updated, the graph is dynamically recomputed between layers. The initial feature of each neuronal node is a 3D (x_i, y_i, z_i) vector, and the local features are updated to a higher dimension via a EdgeConv [4] layer.

EdgeConv is a convolution-like operation that is applied on the edges connecting a point to its neighbors in a local graph. It generates edge features that describe the relationships between a point and its neighbors, thereby capturing the local geometric structure. The updated local feature x'_c of a point p_c is represented by Eq. (4.2):

$$\begin{aligned} x'_c &= max\left\{h(e_1), h(e_2), ..., h(e_K)\right\} \\ h(e_i) &= LeakyReLU(\Phi.x_c + \Theta.(x_i - x_c)) \end{aligned} \tag{4.2}$$

Φ and Θ are trainable matrices. $h(e_i)$ is the vector of the edge e_i from the neighboring node p_i to the updating node p_c. The updated local feature x'_c is obtained

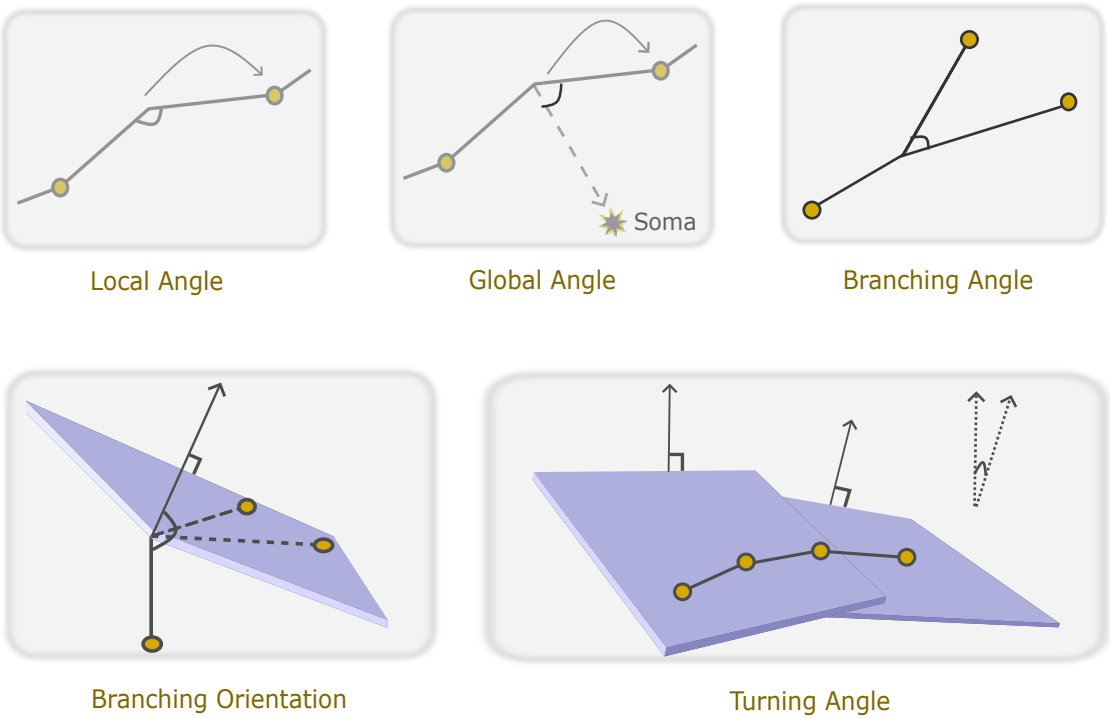

FIGURE 4.1: **Geometric Angles:** A set of five angles to help represent the local neighborhood structure of neuronal morphology

after a max-pooling operation on all edge vectors.

Typically, a cross-entropy loss is used for multi-class classification tasks. In this case, however, the discriminating power of the input features is affected due to difference in morphology reconstruction methods, imaging resolution, as well as parameter settings. Thus, a network trained with simply cross-entropy may suffer from inferior class separation. Triplet loss compares a reference input (anchor) to a matching input (positive) and a non-matching input (negative). The distance from the anchor **A** to the positive **P** is minimized, and the distance from the anchor **A** to the negative **N** is maximized. Applied to a classification problem, the anchor and positive will belong to the same class while the negatives will come from other classes. In this way, tripet loss produces better class separability by enforcing that the distance between the samples from the same class is smaller than that between the samples from different classes. Thus, a joint triplet and cross-entropy loss function is used. The triplet loss L_T is represented by:

$$L_T = max(||f(A) - f(P)||_2 - ||f(A) - f(N)||_2 + \alpha, 0), \tag{4.3}$$

where f is the embedding generated by MorphoGNN and α is the margin value. The cross-entropy loss L_{CE} is represented by:

$$L_{CE} = -\sum_{i=1}^{n} t_i \log(p_i), \tag{4.4}$$

where n is the number of classes, t_i is the truth label and p_i is the Softmax probability for the i^{th} class. The final loss function is expressed by:

$$\mathcal{L} = L_T + L_{CE} \tag{4.5}$$

While the approach delivers good results, the reduction of neuron morphology to mere spatial coordinates is a simplification that needs to be addressed. The three-dimensional geometry of neuron contains valuable angles that can help represent the orientations of neuronal compartments in space. These angles, in addition to the spatial coordinates, can help better represent the neuronal morphology. Therefore, we propose a set of five geometric angles to modify the MorphoGNN approach at the input level. These are depicted in Figure 4.1.

The first of these angles, the local angle, measures the angle between the vector connecting a node to its parent and the vector connecting the node to its child. Effectively, it is a measure of how straight the neurites of the neurons are. The global angle measures the angle between the vector connecting a node to its child and the vector connecting the node to the soma. It is a measure of how much the neurites point away from the soma. In the situation where a node has two children (branching point), it is non-obvious how to set up the local and global angles. To solve this problem, we propagate the angle of a node to the next (child) node. In this case, every node will have a local and global angle assigned to it, except of course the soma, its child and grandchild.

Third, the branching angle measures the angle between the vector connecting a node to its first child and the vector connecting the node to its second child. It measures how wide or narrow the branching is. Fourth, the branch orientation measures the angle between the vector connecting a node to its parent and the normal vector of the plane formed by the two node-to-children vectors. Hence, it reflects the orientation of a neuron branching in space. Obviously, both of these angles can only be assigned to branching nodes.

Finally, the turning angle represents the turning orientations of neurites in space. To calculate this, we measure the angle between the normal vectors of two planes. The first plane is formed by the vector connecting a node's grandparent to its parent and the vector connecting a node's grandparent to its great-grandparent. The second plane is formed by the vector connecting a node's parent to its grandparent and the vector connecting a node's parent to the node itself. This angle will be assigned to all nodes that possess a great-grandparent.

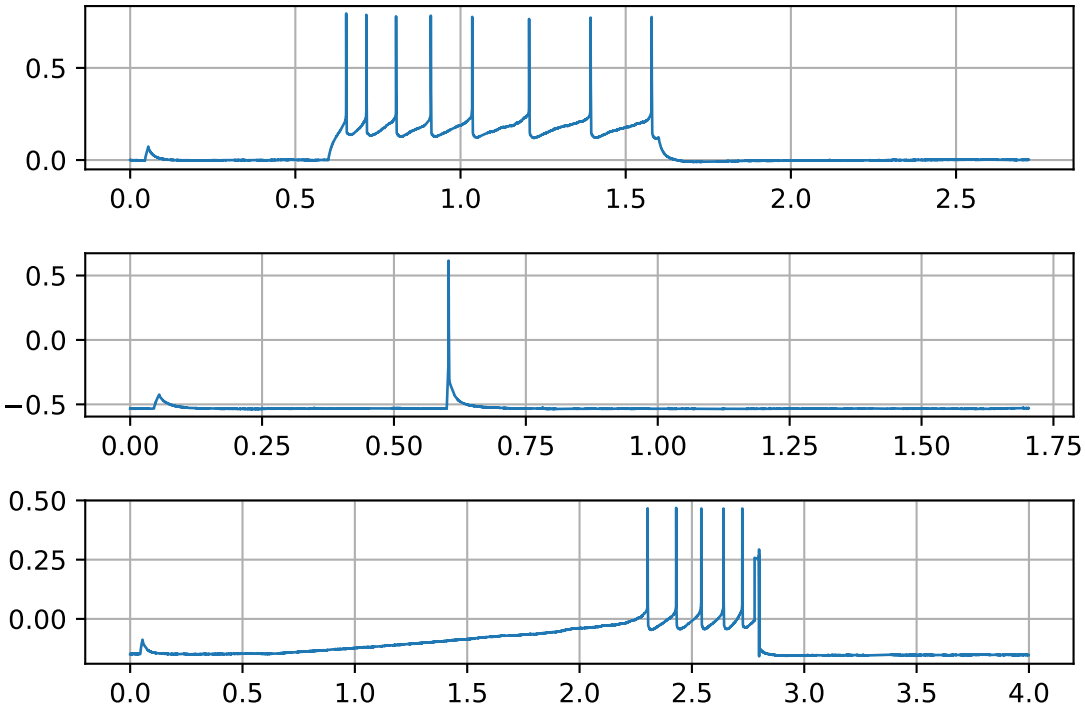

FIGURE 4.2: **Electrophysiology Waveform:** Normalized 3-channel input for electrophysiology model. The channels correspond to the long square, short square and ramp stimuli.

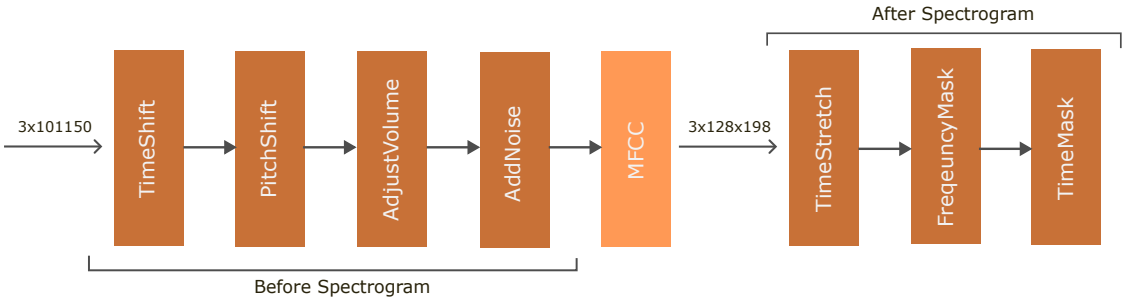

FIGURE 4.3: **Data augmentation pipeline:** Random transformations are applied to generate augmented batches for self-supervised learning.

4.2 Electrophysiology Classification

For electrophysiology input representation, we take the raw waveforms provided by [5]. Most samples in this dataset have waveform recordings against three different stimuli: the short square, long square and ramp stimuli. For a comprehensive representation, we take the responses to all three stimuli and append them along the channel dimension. This is shown in Figure 4.2. All the recordings are originally sampled at 50 kHz. However, since they are of varying

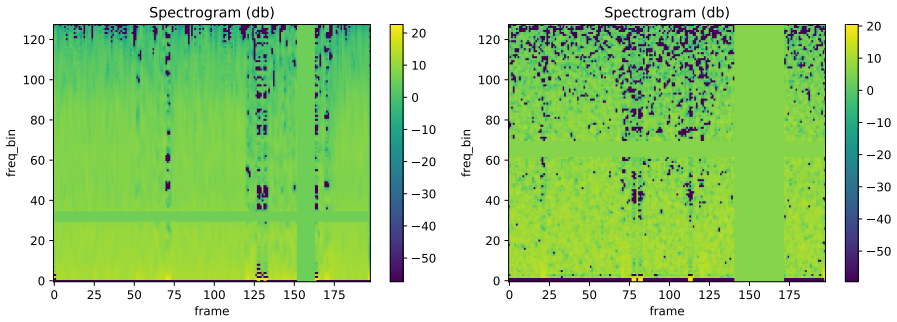

FIGURE 4.4: **Augmented Spectrograms:** Transformed spectrograms of the same electrophysiology sample. Time shift, time masking and frequency masking are visible from the naked eye while other transforms are also present.

time lengths, they cannot be appended together in an array. Therefore, we resample all the recordings to 101150 samples. This number is the lowest number of samples a recording has in the dataset.

Due to the lack of a balanced dataset with sufficient number of samples per class, it is difficult to train an electrophysiology model in a supervised fashion. Therefore, we resort to self-supervised learning. Traditionally, self-supervised learning was done in a contrastive manner. The objective is simple: minimize the distance in representation space between positive sample pairs while maximizing the distance between negative sample pairs. This requires comparing augmented views of the same sample with many negative examples and causes a significant memory footprint. Furthermore, given the random sampling, false negatives can cause a degradation in the representation quality.

Recently, non-contrastive loss functions have appeared that only require positive samples and minimize the distance in space between them. Simply using invariance (mean-squared error) as a non-contrastive loss function causes the learned representations to collapse to a trivial constant solution. Some non-contrastive self-supervised approaches leverage complicated techniques like weight sharing between the branches, batch normalization, feature-wise normalization, output quantization, stop gradient and memory banks to solve this problem. We use the VICReg [12] approach because of its simplicity and state-of-the art results. It uses a non-contrastive loss function and does not rely on techniques like memory banks, large batch sizes and stop-gradient. The principle behind VICReg's approach is to use a loss function with three terms: invariance, variance and covariance. The variance and covariance terms help avoid the embedding collapse.

The data samples are processed in batches, and $Z = [z_1, ..., z_n]$ and $Z' = [z'_1, ..., z'_n]$ are the two batches composed of n vectors of dimension d, of embeddings corresponding to the two batches of augmented views of samples. z^j denotes the vector composed of each value at dimension j in all vectors in Z.

The invariance term s is simply the mean-squared euclidean distance between the embeddings of augmented views of the same samples:

$$s(Z, Z') = \frac{1}{n} \sum_i ||z_i - z'_i||_2^2 \tag{4.6}$$

The variance regularization term v is a hinge loss to maintain the standard deviation (over a batch) of each variable of the embedding above a given threshold. This term forces the embedding vectors of samples within a batch to be different:

$$v(Z) = \frac{1}{d} \sum_{j=1}^{d} \max(0, \gamma - S(z^j, \epsilon)), \tag{4.7}$$

where S is the regularized standard deviation:

$$S(x, \epsilon) = \sqrt{Var(x) + \epsilon} \tag{4.8}$$

The covariance regularization term c attracts the covariances (over a batch) between every pair of embedding variables towards zero. This decorrelates the variables of each embedding and prevents an informational collapse in which the variables would vary together or be highly correlated. If the covariance matrix of Z is represented by $C(Z)$, then c encourages the off-diagonal coefficients of $C(Z)$ to be close to 0, decorrelating the different dimensions of the embeddings and preventing them from encoding similar information:

$$c(Z) = \frac{1}{d} \sum_{i \neq j} [C(Z)]_{i,j}^2 \tag{4.9}$$

The final loss function is a weighted average of the invariance, variance and covariance terms:

$$\mathcal{L} = \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + \alpha[c(Z) + c(Z')] \tag{4.10}$$

For the backbone network, we use a ResNet-50 CNN. To feed a waveform to a 2D CNN, it has to be converted to a spectrogram first. We experiment with different types of spectrograms and find Mel Frequency Cepstral Coefficients

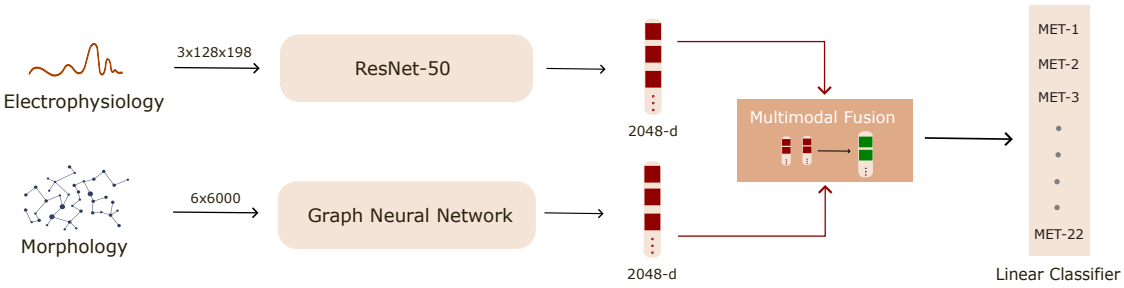

FIGURE 4.5: **Network architecture:** The two networks learn 2048-d marginal representations that are fused together for multimodal neuron classification task.

(MFCC) to deliver the best results. The input to the CNN is a 3 channel 128x198 dimensions spectrogram.

We also design a data augmentation pipeline to prepare augmented batches for the self-supervised learning. This pipeline is divided into two stages as shown in Figure 4.3. A few of the transformations are applied on the raw waveform. Next, it is converted to the MFCC spectrogram. After that, a series of augmentations is applied on the spectrogram as well. Since the performance of a self-supervised model depends on the augmentations as well, it is necessary to ensure that a variety of augmentations are included.

4.3 Multimodal Fusion

First, both the morphology and electrophysiology networks are trained separately to learn the marginal representations on their individual modalities. After that, these networks are frozen and the ouputs are fused together in multiple ways before finally feeding them to a linear classifier. The network architecture is shown in Figure 4.5.

Both networks output a 2048-dimensional embedding. We try different fusion methods like multiplication, addition, simple concatenation and the tensor fusion layer [13]. We avoid the use of an MLP and use a linear classifier instead because of the low number of multimodal samples available. However, this means we cannot do joint-representation learning and hence learn the cross-modality correlations. The tensor fusion layer helps in this regard as it expresses the bimodal interactions by doing an outer product of marginal representations. It also has no trainable parameters. Ideally, we would want to fine-tune both the networks along with training the classifier. However, the 517 neurons available are divided into 28 classes. This is a heavily imbalanced dataset with low number of samples per class. Therefore, fine-tuning with the MET-labels does not deliver improvement.

The loss function is a cross-entropy loss expressed by:

$$\mathcal{L} = -\sum_{i=1}^{n} t_i \log(p_i), \tag{4.11}$$

where n is the number of classes, t_i is the truth label and p_i is the Softmax probability for the i^{th} class.

The multimodal MET samples also have a T-type label assigned to them. Most T-types belong to a single MET-type. Therefore, it is possible to re-label the imbalanced 4,435 cells electrophysiology dataset by merging the T-types that belong to a single MET-type. This re-labelling can be used to fine-tune the electrophysiology network. Unfortunately, the morphology network cannot be fine-tuned as morphologies are only available for the 517 neurons.

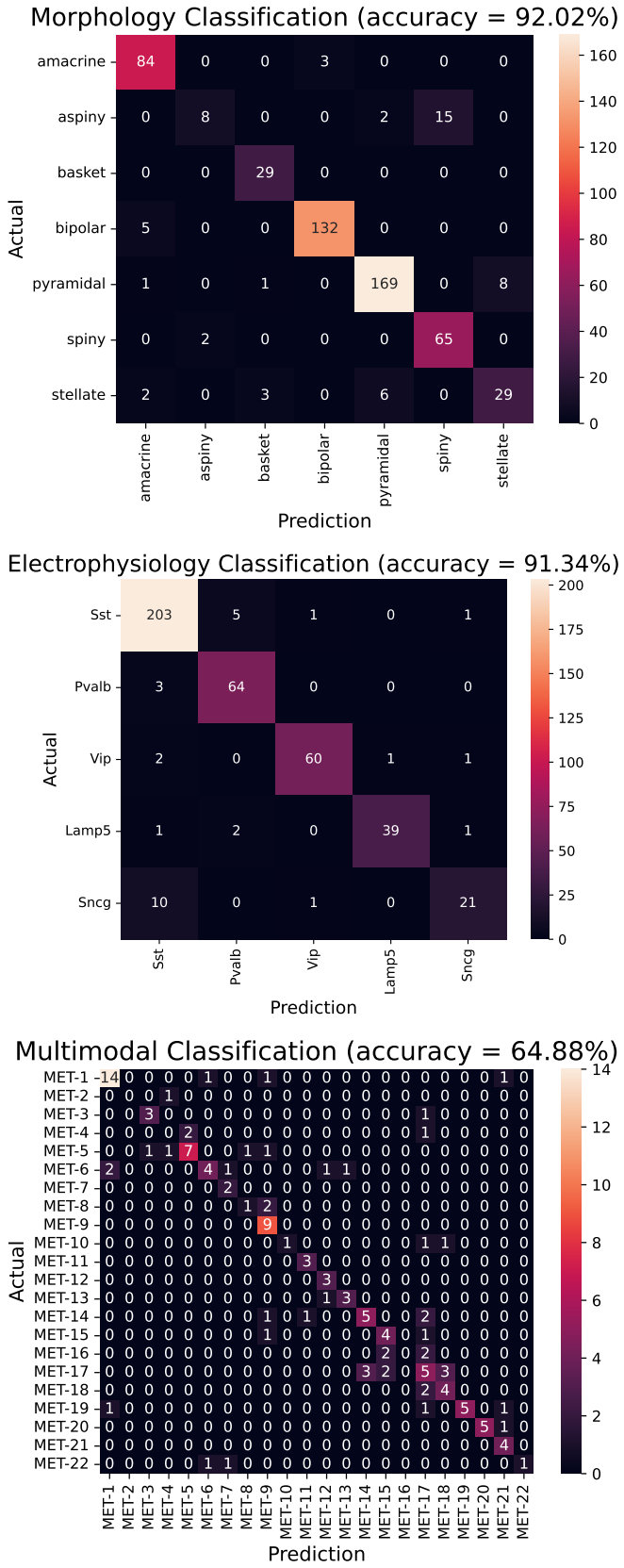

FIGURE 4.6: **Confusion Matrices:** Summarizing the results for unimodal and multimodal classification through confusion matrices.

5 Results

5.1 Unimodal

TABLE 5.1: Morphology classification on 7-class dataset

Method	Point Features Used	OA(%)	MA(%)
DGCNN	3D Coordinates	78.61	63.38
MorphoGNN	3D Coordinates	82.8	71.19
Ours	3D Coordinates, Local Angle	88.29	82.23
Ours	3D Coordinates, Local Angle, Global Angle	90.07	85.07
Ours	3D Coordinates, Local Angle, Global Angle, Branch Angle	90.24	83.11
Ours	3D Coordinates, Local Angle, Global Angle, Branch Orientation	90.07	84.09
Ours	3D Coordinates, Local Angle, Global Angle, Turning Angle	**92.02**	**87.1**

The morphology model is trained and evaluated on a 7-class dataset collected from NeuroMorpho.org [1]. For forming the dataset, we collect samples from multiple research contributions. The aspiny neurons come from the two groups [14] and [15]. Spiny neurons are contributed by [14]. The amacrine and bipolar neurons come from [16]. The stellate neurons are provided by [17], [18] and [19]. The basket neurons come from the three groups [20], [21] and [22]. The pyramidal neurons come from [17], [18], [23], [24], [25], [26].

In total, we get a data of 1890 samples. A 7:3 train-validation split is done. Neurons that have less than 6000 points are zero-padded to 6000. Morphologies that have more than 6000 points are uniformly downsampled. The spatial coordinates are normalized to be in the range $(-1, 1)$. We train the model for 70 epochs and the rest of the training implementation is the same as in [3].

The evaluation results are shown in Table 5.1. We try different combinations of input node-based features and find that the local, global and turning angles in addition to the 3D coordinates deliver the best peformance (in bold). Our method outperforms the current state-of-the-art MorphoGNN by 9.22% in overall accuracy and 15.91% in mean-class accuracy.

We also compare the train and inference times between MorphoGNN and our approach on an Nvidia Titan RTX GPU. Our method does not take any noticeable additional time. This is due to the fact that the angles can be extracted during data pre-processing and do not add any overhead cost during execution. Moreover, the only architectural modification we do is change the number of filters of the first 1x1 convolution layer from 6 to 12. This is done to accomodate the additional angles (channels) added. With this change, the total number of trainable parameters only increase by 192. Both approaches take around 3 hours 32 minutes for training (70 epochs), and around 90 milliseconds for inference.

As apparent in Figure 4.6, the misclassification rate for aspiny neurons is high and they are mostly being confused as spiny neurons. This is partly because of the fact that the number of samples available for the aspiny class is the lowest compared to other classes. However, it also points out that spiny and aspiny neurons are more similar in their morphologies than other classes and that it was difficult for the model to produce a good class separation between them given the current data.

The electrophysiology self-supervised model is trained on the patch-seq dataset provided by [5]. Out of the 4,435 cells, we ignore the 517 neurons that have a multimodal label assigned to them. This is done to ensure the model does not see the data it is going to be evaluated on later during multimodal analysis. The raw waveforms are normalized channel-wise to be in the range $(-1, 1)$. We train the model for 1000 epochs with a batch size of 512 samples. The final loss value comes out to 15.081. A base learning-rate of 0.5 was set and the rest of the training implementation is the same as in [12].

TABLE 5.2: Electrophysiology classification on 5-class dataset

Method	OA(%)	MA(%)
Supervised learning	86.53	82.39
Frozen backbone + Linear Classifier	82.93	74.05
Fine-tuned backbone + Linear Classifier	**91.34**	**87.78**

To evaluate the effectiveness of a self-supervised approach, we require a dataset with sufficient number of samples per class. Unfortunately, the patch-seq dataset has some classes having less than 5 samples. To solve this problem, we merge the 60 available transcriptomic (T-type) labels down to the 5 transcriptomic sub-types (Sst, Pvalb, Vip, Lamp5, Sncg). We then evaluate our method on this dataset.

The evaluation results are shown in Table 5.2. We compare our self-supervised method with traditional supervised learning. The frozen backbone delivers good results attesting to the effectiveness of self-supervised representations. In addition to this, the fine-tuned representations beat supervised learning by 4.81% in overall accuracy and by 5.39% in mean-class accuracy.

Figure 4.6 shows good per-class performance for all transcriptomic sub-types using fine-tuned representations. Some Sncg neurons are being confused as Sst neurons but that is mainly because of the low number of samples available for the Sncg class.

5.2 Multimodal

TABLE 5.3: Comparison of fusion methods for multimodal classification

Fusion Method	OA(%)	MA(%)
Multiplication	61.83	56.55
Addition	62.59	58.34
Concatenation	**64.88**	**60.86**
Tensor Fusion Layer	60.3	51.9

TABLE 5.4: Performance of modalities for classification on 22 MET classes

Modality Used	OA(%)	MA(%)
Electrophysiology-only	50.38	41.73
Morphology-only	35.11	32.27
Both	**64.88**	**60.86**

The multimodal classifier is trained and evaluated on the dataset provided by [5]. This data contains 517 neurons for which both morphologies and electrophysiologies are available. These neurons have been assigned MET (morphological-electrophysiological-transcriptomic) cell type labels. Within a single MET-type, the neurons have aligned morphological, electrophysiological and transcriptomic properties. Out of the 28 MET-types, we only use the 22 MET-types that have at least 5 samples available. A 7:3 train-validation split is done.

The performance comparison between different fusion techniques is shown in Table 5.3. Simple concatenation delivers the best results (in bold). We achieve an overall accuracy of 64.88% and a mean-class accuracy of 60.86%.

Figure 4.6 shows good per-class performance for all classes except MET types 14 through 17. This is because of scarcity of data and more similarity among these classes.

As shown in Table 5.4, the use of both modalities outperforms the use of a single modality (either morphology or electrophysiology) for multimodal classification task. Morphology performs the lowest and it is understandable knowing the morphology model could not be fine-tuned.

For training the unimodal classifiers, we had sufficient data available. For morphology, we have 1890 samples divided into 7 classes. For electrophysiology, we have around 4000 samples divided in 5 classes. Unfortunately, for multimodal classification, data is scarce and this is the reason why multimodal results are not comparable with unimodal. In this case, we have only 517 neurons divided into 22 classes. Therefore, the number of samples per class is quite low.

6 Conclusions and Future Work

6.1 Conclusions

In this work, we have devised a deep-learning based approach for both unimodal and multimodal classification. By avoiding the traditional machine learning models that make use of hand-crafted features, we have been able to generate accurate representations that deliver state-of-the-art performance. This is the first application of deep learning to make use of multiple neuronal modalities. Defining a neuronal type requires more than one modality and our method can map an input neuron to its morphological, electrophysiological, and transcriptomic properties at once.

During our analysis, we note how the use of geometric angles enriches the morphological representation and more comprehensively accounts for the complexity of the shape of neuronal structures. It is indeed true that these angles help in defining the morphology in addition to the 3D coordinates. Moreover, we note how well a self-supervised approach can learn to differentiate between neuronal electrophysiologies. Not only does it deliver good results, it also beats the supervised method by learning a richer feature representation. Additionally, the use of both modalities outperforms the use of a single modality (either morphology or electrophysiology) for multimodal classification task.

6.2 Future Work

Subsequent efforts could make use of a multimodal dataset that has more samples available. Deep learning approaches always benefit from more data and that will help improve the multimodal classification accuracy. An ideal strategy would be to learn self-supervised representations for both modalities, fine-tune them with the multimodal data, and train a linear classifier to predict the cell type. Unfortunately, we were dealing with a limited data and could not follow this approach.

We did not use transcriptomics as a third modality for classification. However, it could be experimented with using other multimodal data in future. Moreover, new modalities like molecular properties could also appear in future. For multimodal fusion, late fusion or decision fusion can also be tried and compared with intermediate fusion.

Another thing that could be looked at is the graph neural network for morphological representation learning. Experimentation with newer and lighter networks may yield better results. Moreover, training it in a self-supervised manner is also a possibility. This requires thinking to devise a robust data augmentation pipeline. The geometric angles are rotation and translation invariant. So, to augment these angles, a transformation in addition to simple rotation and translation will have to be implemented.

References

[1] G. A. Ascoli, D. E. Donohue, and M. Halavi, "Neuromorpho.org: A central resource for neuronal morphologies," *Journal of Neuroscience*, vol. 27, no. 35, pp. 9247–9251, 2007.

[2] R. Scorcioni, S. Polavaram, and G. A. Ascoli, "L-measure: A web-accessible tool for the analysis, comparison and search of digital reconstructions of neuronal morphologies.," *Nat Protoc*, vol. 3, no. 5, pp. 866–876, 2008.

[3] T. Zhu, G. Yao, D. Hu, C. Xie, H. Gong, and A. Li, "MorphoGNN: Morphological embedding for single neuron with graph neural networks," en, May 2022.

[4] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, 146:1–146:12, 2019.

[5] N. W. Gouwens, S. A. Sorensen, F. Baftizadeh, *et al.*, "Integrated morphoelectric and transcriptomic classification of cortical GABAergic cells," en, *Cell*, vol. 183, no. 4, 935–953.e19, Nov. 2020.

[6] N. W. Gouwens, S. A. Sorensen, J. Berg, *et al.*, "Classification of electrophysiological and morphological neuron types in the mouse visual cortex," en, *Nat. Neurosci.*, vol. 22, no. 7, pp. 1182–1195, Jul. 2019.

[7] M. Kubota and I. Taniguchi, "Electrophysiological characteristics of classes of neuron in the hvc of the zebra finch.," *J Neurophysiol*, vol. 80, no. 2, pp. 914–923, Aug. 1998.

[8] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017.

[9] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA, 2017, pp. 5105–5114.

[10] M. Lipovsek, C. Bardy, C. R. Cadwell, K. Hadley, D. Kobak, and S. J. Tripathy, "Patch-seq: Past, present, and future," *Journal of Neuroscience*, vol. 41, no. 5, pp. 937–946, 2021, ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.1653-20.2020. eprint: https://www.jneurosci.org/content/41/5/937.full.pdf. [Online]. Available: https://www.jneurosci.org/content/41/5/937.

[11] S. R. Stahlschmidt, B. Ulfenborg, and J. Synnergren, "Multimodal deep learning for biomedical data fusion: a review," *Briefings in Bioinformatics*, vol. 23, no. 2, Jan. 2022, bbab569, ISSN: 1477-4054. DOI: 10.1093/bib/bbab569. eprint: https://academic.oup.com/bib/article-pdf/23/2/bbab569/42805085/bbab569.pdf. [Online]. Available: https://doi.org/10.1093/bib/bbab569.

[12] A. Bardes, J. Ponce, and Y. LeCun, "VICReg: Variance-invariance-covariance regularization for self-supervised learning," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=xm6YD62D1Ub.

[13] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L. Morency, "Tensor fusion network for multimodal sentiment analysis," in *EMNLP*, Association for Computational Linguistics, 2017, pp. 1103–1114.

[14] L. Madisen, A. R. Garner, D. Shimaoka, *et al.*, "Transgenic mice for intersectional targeting of neural sensors and effectors with high specificity and performance," *Neuron*, vol. 85, no. 5, pp. 942–958, 2015, ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2015.02.022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0896627315001373.

[15] M. W. Bagnall, C. Hull, E. A. Bushong, M. H. Ellisman, and M. Scanziani, "Multiple clusters of release sites formed by individual thalamic afferents onto cortical interneurons ensure reliable transmission," *Neuron*, vol. 71, no. 1, pp. 180–194, 2011, ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2011.05.032. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0896627311004831.

[16] M. Helmstaedter, K. L. Briggman, S. C. Turaga, V. Jain, H. S. Seung, and W. Denk, "Connectomic reconstruction of the inner plexiform layer in the mouse retina," *Nature*, vol. 500, no. 7461, pp. 168–174, 2013. DOI: 10.1038/nature12346. [Online]. Available: https://doi.org/10.1038/nature12346.

[17] T. J. Ellender, S. V. Avery, K. Mahfooz, *et al.*, "Embryonic progenitor pools generate diversity in fine-scale excitatory cortical subnetworks," *Nature Communications*, vol. 10, no. 1, p. 5224, 2019. DOI: 10.1038/s41467-019-13206-1. [Online]. Available: https://doi.org/10.1038/s41467-019-13206-1.

[18] T. Guillamon-Vivancos, W. A. Tyler, M. Medalla, *et al.*, "Distinct Neocortical Progenitor Lineages Fine-tune Neuronal Diversity in a Layer-specific Manner," *Cerebral Cortex*, vol. 29, no. 3, pp. 1121–1138, Feb. 2018, ISSN: 1047-3211. DOI: 10.1093/cercor/bhy019. eprint: https://academic.oup.com/cercor/article-pdf/29/3/1121/27768892/bhy019.pdf. [Online]. Available: https://doi.org/10.1093/cercor/bhy019.

[19] D. L. F. Garden, P. D. Dodson, C. O'Donnell, M. D. White, and M. F. Nolan, "Tuning of synaptic integration in the medial entorhinal cortex to the organization of grid cell firing fields.," *Neuron*, vol. 60, pp. 875–889, 2008, ISSN: 1097-4199 (Electronic); 0896-6273 (Linking). DOI: 10.1016/j.neuron.2008.10.044.

[20] T. Lalanne, J. Oyrer, A. Mancino, *et al.*, "Synapse-specific expression of calcium-permeable ampa receptors in neocortical layer 5," *The Journal of Physiology*, vol. 594, no. 4, pp. 837–861, 2016. DOI: https://doi.org/10.1113/JP271394. eprint: https://physoc.onlinelibrary.wiley.com/doi/pdf/10.1113/JP271394. [Online]. Available: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/JP271394.

[21] V. K. Vereczki, J. M. Veres, K. Müller, *et al.*, "Synaptic organization of perisomatic gabaergic inputs onto the principal cells of the mouse basolateral amygdala," *Frontiers in Neuroanatomy*, vol. 10, 2016, ISSN: 1662-5129. DOI: 10.3389/fnana.2016.00020. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnana.2016.00020.

[22] L. Rovira-Esteban, Z. Péterfi, A. Vikór, Z. Máté, G. Szabó, and N. Hájos, "Morphological and physiological properties of cck/cb1r-expressing interneurons in the basal amygdala," *Brain Structure and Function*, vol. 222, no. 8, pp. 3543–3565, 2017. DOI: 10.1007/s00429-017-1417-z. [Online]. Available: https://doi.org/10.1007/s00429-017-1417-z.

[23] E. K. W. Brennan, S. K. Sudhakar, I. Jedrasiak-Cape, T. T. John, and O. J. Ahmed, "Hyperexcitable neurons enable precise and persistent information encoding in the superficial retrosplenial cortex.," *Cell Rep*, vol. 30, no. 5, pp. 1598–1612, Feb. 2020.

[24] M. P. Fiske, M. Anstötz, L. J. Welty, and G. Maccaferri, "The intrinsic cell type-specific excitatory connectivity of the developing mouse subiculum is sufficient to generate synchronous epileptiform activity," *The Journal of Physiology*, vol. 598, no. 10, pp. 1965–1985, 2020. DOI: https://doi.org/10.1113/JP279561. eprint: https://physoc.onlinelibrary.wiley.com/doi/pdf/10.1113/JP279561. [Online]. Available: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/JP279561.

[25] S. C. Borrie, S. B. Sartori, J. Lehmann, A. Sah, N. Singewald, and C. E. Bandtlow, "Loss of nogo receptor homolog ngr2 alters spine morphology of ca1 neurons and emotionality in adult mice," *Frontiers in Behavioral Neuroscience*, vol. 8, 2014, ISSN: 1662-5153. DOI: 10.3389/fnbeh.2014.00175. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbeh.2014.00175.

[26] Z. Bertels, H. Singh, I. Dripps, *et al.*, "Neuronal complexity is attenuated in preclinical models of migraine and restored by hdac6 inhibition," *eLife*, vol. 10, A. Basbaum, K. M. Wassum, and M. Waung, Eds., e63076, Apr.

2021, ISSN: 2050-084X. DOI: 10.7554/eLife.63076. [Online]. Available: https://doi.org/10.7554/eLife.63076.