DENDRITIC SPINE SHAPE ANALYSIS BASED ON TWO-PHOTON MICROSCOPY IMAGES

by

Muhammad Usman Ghani

Submitted to the Graduate School of Engineering and Natural Sciences

in partial fulfillment of

the requirements for the degree of

Master of Science

Sabancı University

July 2016

DENDRITIC SPINE SHAPE ANALYSIS BASED ON
TWO-PHOTON MICROSCOPY IMAGES

APPROVED BY

Assoc. Prof. Dr. Müjdat Çetin .....
(Thesis Supervisor)

Assoc. Prof. Dr. Tolga Taşdizen .....

Assoc. Prof. Dr. Devrim Ünay .....

DATE OF APPROVAL: July 25, 2016

to my family

Acknowledgments

Praise be to ALLAH, the Cherisher and the Sustainer of the worlds. (Holy Quran 1:1)

The accomplishments in this thesis would not have been possible without the help and support of many colleagues, friends, and my family. I am grateful to Müjdat Çetin and Devrim Ünay that they selected me to work on microscopic image analysis project. I would like to express my sincere gratitude to my advisor, Müjdat Çetin, for not only his continuous guidance during these two years but also for providing a perfect example of an academician, a thesis advisor, and a humble human being. I am grateful to Tolga Taşdizen, his experience with machine learning problems have been instrumental to tackle many problems. I was very fortunate to have Sumeyra Demir Kanık as a continuous resource, her in-depth analytic skills have greatly influenced this thesis. This work would not have been possible without Ali Özgür Argunşah, his expertise on neuroscience as well as signal processing problems have played vital role in this work. I am thankful to members of my thesis committee, Devrim Ünay and Tolga Taşdizen, for careful evaluation of my work and providing useful comments to improve the quality of this thesis. I am also grateful to the Scientific and Technological Research Council of Turkey (TUBITAK) for supporting me under under the Grant $113E603$. I wish to acknowledge support from Faculty of Engineering and Natural Sciences at Sabanci University in form of tuition fee and dormitory fee waiver, and partial support to attend the ISBI 2016. I would also like to acknowledge partial support from *Erasmus+* for visiting Champalimaud Foundation, Lisbon, Portugal, during summer of $2014 - 15$. I was fortunate to work with Inbal Israely and other members of her group, Neuronal Structure and Function, at Champalimaud Foundation. I am also thankful to Fitsum

Mesadi and Ertunç Erdil for their interaction and insightful discussions which greatly contributed to this work.

I am profoundly indebted to many of my great friends including Muhammad Sohaib Amjad, Fiaz Ahmad, Mansoor Ahmad, Faran Ahmad, Fahad Sohrab, and Waqar Ahmad for their countless favors. I am grateful to Sohaib and Mansoor for cooking lessons, which helped me survive in Lisbon. Support and guidance by Sohaib, Mansoor, and Fiaz to handle various distractions on the way have been phenomenal. I am thankful to Sohaib, Mansoor, Fiaz, and Faran for being good listeners and late-night discussions on various intellectual topics of common interest. I want to thank Fiaz, Sohaib, Mansoor, and Waqar for memorable games of "Ludo" and "Rung". Numerous quality dinners prepared by Sohaib, Mansoor, and Waqar have also greatly contributed to this work. I am grateful to all of them for their confidence in my work.

I am thankful to many of my friends at SPIS lab including Muhammad Burak Alver, Abdullahi Adamu, Majed Elwardy, and Sezen Yağmur Günay. I want to thank Burak for numerous cups of Turkish Çay, Burak's Çay making skills are tremendous. I am grateful to Abdullahi for introducing me to "Antep Fıstıklı Baklava". I am thankful to Majed for being my roommate for first semester. I thank Yağmur for her company at the SIU 2016 and making this tour full of fun. I am also grateful to SPIS lab administrator, Osman Rahmi Ficici, for his continuous support in terms of computing facilities at lab. I thank Anaguli Abulizi (Anar) for being a nice friend and for thoughtful discussions on philosophical aspects of life.

I am grateful to my parents, brothers, and sisters, for their everlasting encouragement, emotional support, trust, and pure love. I thank all of them to support my decision of spending these two years away from them.

I want to recognize many individuals whose names I could not mention here, who have profoundly affected me and helped me shape my thoughts and believes.

DENDRITIC SPINE SHAPE ANALYSIS BASED ON TWO-PHOTON MICROSCOPY IMAGES

Muhammad Usman Ghani

CS, M.Sc. Thesis, 2016

Thesis Supervisor: Müjdat Çetin

Keywords: Dendritic spines, classification, clustering, Disjunctive Normal Shape Model, HOG, shape analysis, Kernel Density Estimation, microscopy, neuroimaging.

Abstract

Neuronal morphology and function are highly coupled. In particular, dendritic spine morphology is strongly governed by the incoming neuronal activity. Previously, volumes of dendritic spines have been considered as a primary parameter to study spine morphology and gain insight into structure-function coupling. However, this reductionist approach fails to incorporate the broad spine structure repertoire. First step towards integrating the rich spine morphology information into functional coupling is to classify spine shapes into main spine types suggested in the literature. Due to the lack of reliable automated analysis tools, classification is currently performed manually, which is a time-intensive task and prone to subjectivity. Availability of automated spine shape analysis tools can accelerate this process and help neuroscientists understand underlying structure and function relationship. Several studies on spine shape classification have been reported in the literature, however, there is an on-going debate on whether distinct spine shape classes exist or whether spines should be modeled through a continuum of shape variations. Another challenge is the subjectivity and bias that is introduced due to the supervised nature of classification approaches. This thesis focuses on morphological, shape, and appearance features based methods to perform dendritic spine shape analysis using both clustering and classification approaches. We apply manifold learning methods for dendritic spine classification and observe that ISOMAP implicitly computes prominent features suitable for classification purposes. We also apply linear representation based approach for spine classification and conclude that sparse representation provides slightly better classification performance. We propose 2D and 3D

morphological features based approach for spine shape analysis and demonstrate the advantage of 3D morphological features. We also use a deep learning based approach for spine classification and show that mid-level features extracted from Convolutional Neural Networks (CNNs) perform as well as hand-crafted features. We propose a kernel density estimation (KDE) based framework for dendritic spine classification. We evaluate our proposed approaches by comparing labels assigned by a neuroscience expert. Our KDE based framework also enables neuroscientists to analyze separability of spine shape classes in the likelihood ratio space, which leads to further insights about the nature of the spine shape analysis problem. Furthermore, we also propose a methodology for unsupervised learning and clustering of spine shapes. In particular, we use x-means to perform cluster analysis that selects the number of clusters automatically using the Bayesian information criterion (BIC). The objective of clustering in this context is two-fold: confirm the hypothesis of some distinct shape classes and discover new natural groups. We observe that although there are many spines which easily fit into the definition of standard shape types (confirming the hypothesis), there are also a significant number of others which do not comply with standard shape types and demonstrate intermediate properties.

İKI FOTON MIKROSKOBIK GÖRÜNTÜLERI KULLANARAK DENDRITIK DIKEN ŞEKIL ANALIZI

Muhammad Usman Ghani

CS, Yüksek Lisans Tezi, 2016

Tez Danışmanı: Müjdat Çetin

Anahtar Kelimeler: Dendritik dikenler, sınıflandırma, kümeleme, Ayrık Normal Şekil Modeli, HOG, şekil analizi, Çekirdek Yoğunluk Tahmini, mikroskop, nörogörüntüleme.

Özet

Sinirsel morfoloji ve fonksiyon birbiriyle oldukça ilintilidir. Özellikle dendritik diken morfolojisi güçlü bir şekilde gelen sinirsel aktivite ile yönetilir. Önceki çalışmalarda dendritik dikenlerin hacminin diken morfolojisini incelemek ve yapı-fonksiyon ilişkisini anlamak için temel parametre olduğu düşünülüyordu. Fakat bu indirgemeci yaklaşım dikenlerin kapsamlı yapı dağarcığını içermemektedir. Zengin diken morfoloji bilgisini fonksiyonel eşleşmeyle bütünleştirmenin ilk adımı, diken şekillerinin literatürde önerilen temel şekil sınıflarına göre sınıflandırılmasıdır. Yeterli seviyede güvenilir otomatik analiz araçlarının olmaması nedeniyle sınıflandırma işlemi elle yapılmaktadır. Bu da analizin öznel ve zaman isteyen bir işlem olmasına yol açmaktadır. Otomatik diken şekil analiz araçları bu işlemi hızlandırarak sinirbilimcilerin altta yatan yapı ve fonksiyon ilişkisini anlamasına yardımcı olacaktır. Literatürde diken şekil sınıflandırması ile ilgili birçok çalışma yer almaktadır. Fakat diken şekillerinin ayrı sınıflar halinde mi yoksa bir şekil değişim süreci olarak mı ele alınması gerektiği konusunda bir fikir birliğine varılmamıştır. Bu problemde karşımıza çıkan bir diğer güçlük sınıflandırma yaklaşımlarının güdümlü yapısının getirdiği öznellik ve yanlılıktır. Bu tez, hem kümeleme hem de sınıflandırma yaklaşımlarını morfolojik, şekil ve görüntü öznitelikleriyle kullanarak dendritik diken şekil analizi gerçekleştirme üzerine kurulmuştur. Dendritik diken sınıflandırma problemine çok katlı (manifold) öğrenme yöntemlerini uyguladığımızda ISOMAP'in dolaylı olarak sınıflandırma için önemli öznitelikleri hesapladığını gözlemledik. Sınıflandırma amacıyla doğrusal temsil yaklaşımına başvurduğumuzda seyrek temsilin kısmen daha iyi bir sınıflandırma performansı sağladığını gördük. 2 boyutlu ve

3 boyutlu morfolojik özniteliklere dayalı diken şekil analizi yaklaşımında 3 boyutlu morfolojik özniteliklerin sağladığı avantajları gösterdik. Derin öğrenmeye dayanan sınıflandırma yaklaşımında Konvolüsyonel Sinir Ağlarından (CNNs) çıkarılan orta seviye özniteliklerin özel çıkarılmış öznitelikler kadar iyi performans gösterdiğine şahit olduk. Dendritik diken sınıflandırması için çekirdek yoğunluk tahminine (KDE) bağlı bir çerçeve tasarladık. Önerdiğimiz yaklaşımları sinirbilimci bir uzmanın belirlediği etiketlerle karşılaştırdık. Çekirdek yoğunluk tahminine bağlı çerçeve sinirbilimcilerin dikenlerin şekil sınıflarının ayrılabilirliğini olabilirlik oranı uzayında incelemelerine olanak vererek şekil analiz problemine daha derinden bakabilmelerini sağlayabilir. Bunlara ek olarak diken şekillerini güdümsüz öğrenme ve kümeleme yöntemleri üzerinde çalışmalar yaptık. Bayes bilgi kıstasını kullanarak küme sayısını veriden otomatik seçen x-ortalama (x-means) tekniğini kullandık. Bu bağlamda kümelemenin iki farklı amaçla kullanılmasından söz edilebilir: ayrı şekil sınıflarının varlığına dair hipotezi doğrulamak ve yeni gruplar keşfetmek. Elimizdeki veride çok sayıda diken standart şekil sınıfları içinde değerlendirilebilse de, önemli sayıda dikenin bu sınıflara uymadığını ve ara niteliklere sahip olduğunu gözlemledik.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis presents new dendritic spine shape analysis approaches based on probabilistic and machine learning methods. In this chapter, we start with defining the spine analysis problem and motivation behind this research. Further, we provide an overview of the contributions of this thesis and then discuss the structure of this thesis.

1.1 Problem Definition and Motivation

Dendritic spines, small protrusions of the dendritic shaft, are one of the most important structures of neurons. Ramón y Cajal first identified spines in the 19th century and suggested that changes in neuronal activity modify the spine morphology [2, 3]. This claim has been supported by several studies reporting changes in the morphology and density with changes in neuronal activity (such as learning and memory), and neurodegenerative diseases (e.g., Alzheimer's and Parkinson) [4, 5, 6, 7]. Therefore, understanding the structure-function relationships might provide a way to interpret how our brain learns and stores new information. It might also enhance our understanding of various neurogenerative and neurodevelopmental disorders. The first step towards understanding the structure-function relationships is to classify spines into main shape types reported in the literature.

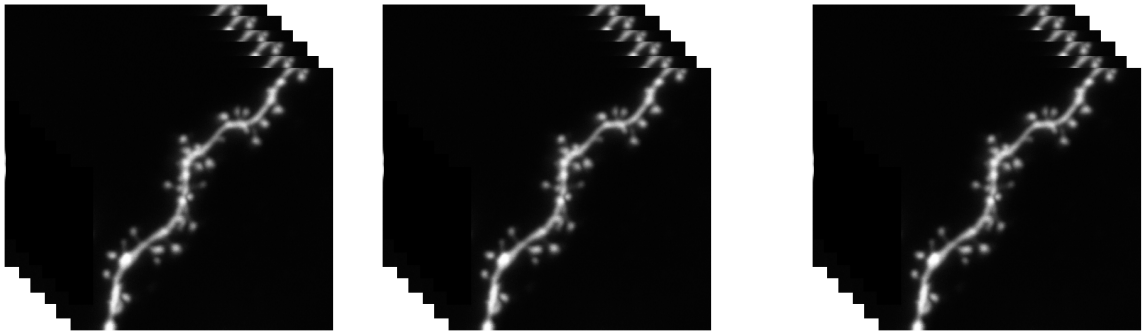
Recent findings on the structure-function links and availability of modern neuron imaging technology has attracted many researchers, this has led to the collection of

vast amount of datasets which are mostly analyzed manually due to unavailability of automated analysis tools. Manual analysis is a tedious, time-intensive, and most importantly subjective task. Availability of reliable automated analysis tools can expedite research in this domain and assist neuroscientists decode the underlying relationship between neuron function and structure.

Most of the studies on spine analysis consider confocal laser scanning microscopy (CLSM) images which does not allow imaging of living cells and therefore cannot capture dynamic data. Two-photon laser scanning microscopy (2PLSM) has the capability to image living cells thus can produce dynamic data, which would capture shape transitions during synaptic process, allowing the analysis of tissues over time[8, 9]. However, the signal-to-noise ratio (SNR) of data collected using 2PLSM is very low as compared to CLSM. Additionally, following the Abbe's law [10], resolution of 2PLSM images is half of the CLSM images. Additionally, experiments with 2PLSM involve imaging cells over prolonged periods of time which produces large amounts of data (MxNxZxT), as illustrated in Figure 1.1. This makes the process of manual analysis even more difficult. Another challenge is subjectivity introduced due to manual analysis, however, this still has effects on automated systems as well due to supervised nature of classification systems. Additionally, there is an on-going discussion in the literature whether to model the spines as distinct classes or continuum of shape variations. Accordingly this thesis attempts to fill this gap presenting new probabilistic and machine learning based methods for 2PLSM images to perform dendritic spine shape analysis automatically.

1.2 Contributions of this Thesis

The first major contribution of this thesis is the development of a shape and appearance features based classification method. Disjunctive Normal Shape Models (DNSM) [11] is a recently proposed parametric shape representation. We start with automatically segmenting spine images by applying DNSM and use achieved representation as shape features. Histogram of oriented gradients (HOG) [12] has been widely used for object detection and recognition tasks in computer vision, we use HOG to compute



(a) t = 1 (b) t = 2 (c) t = T

Figure 1.1: Stacks of dendritic branches (MxNxZ) captured using 2PLSM from time $t = 1$ to $t = T$.

appearance features. We perform non-parametric kernel density estimation (KDE) and apply likelihood ratio test (LRT) to classify test images [13]. Our KDE based classification approach provides likelihoods of class membership, it can be used to examine the question of continuum of shape variations in a principled manner.

The second contribution of this thesis is using morphological features with state-of-the-art classification techniques and report importance of morphological features for classification of spines. Many important morphological features of dendritic spines, e.g., head diameter, neck length, etc., are estimated and results are reported with complete and a subset of features.

The third contribution of this thesis is a clustering based approach for spine shape analysis. We use morphological, shape, and appearance based features to perform cluster analysis of dendritic spines. The advantages of adopting a clustering approach are: such an approach would not suffer from subjectivity, analysis time would be reduced by avoiding manual labeling tasks, and it would help us confirm existing hypothesis regarding spine shapes as well as discover new patterns.

The fourth contribution of this thesis is to apply manifold learning based approach for spine classification. Manifold learning techniques uncover the intrinsic dimensionality of datasets, we applied different manifold learning approaches on spines dataset and classified spines from extracted features. We observed that ISOMAP [14] implicitly

computes prominent features suitable for classification of dendritic spines.

The fifth contribution of this is the application of linear representation based methods for spine classification. We apply the ℓ_1-norm based approach discussed in [15] and compare the classification results with the least squares method and ℓ_2-norm method [16].

The sixth contribution of this thesis is the development of an approach for $3D$ neck length estimation. Neck length is an important feature of dendritic spines. We have developed an approach to estimate $3D$ neck length of spines by estimating geodesic distance between dendrite surface and spine head. We estimate several other features important for different applications; neck base point: important for spines tracking; spine head to dendrite angle: important to study spine motility. Additionally, we propose a neck-path features based spine classification approach.

The final contribution of this thesis is the application of a deep learning based approach for spine analysis. We use a pre-trained network trained on natural images as a feature extractor as well as fine-tune this network on our spines dataset.

Overall, this thesis presents several approaches to perform spine analysis from $2D$ and $3D$ images. Our KDE based framework enables neuroscientists to study the question of continuum of shape variations in a principled manner. Our shape and appearance features based approach is efficient, robust, and accurate feature extraction scheme. Additionally, our cluster analysis approach allows neuroscientists to analyze large datasets without the need of labeling spines and discover possibly new patterns.

1.3 Organization of the Thesis

This thesis is organized as follows. Chapter 2 presents background of dendritic spine shape analysis. This includes introduction to dendritic spines, relationship between neuronal function and spine structure, spine shape analysis, an overview of related work reported in the literature, and background of approaches applied for spine analysis in this thesis. Details of $2D$ and $3D$ morphological feature based analysis are presented in Chapter 3. Feature extraction approaches applied in this thesis, KDE based clas-

sification framework, and results for proposed classification techniques are presented and discussed in Chapter 4. Our clustering approach for unsupervised shape analysis and the corresponding results achieved with different feature extraction techniques are discussed in Chapter 5. Chapter 6 provides a summary of the main findings of this thesis and future work suggestions.

Chapter 2

Background

In this chapter, we present background of dendritic spine shape analysis. We begin with introducing dendritic spines and relationship between their structural changes and neuronal activity changes. Further, we describe the dendritic spine shape analysis problem and present an overview of some existing spine shape analysis studies.

2.1 Dendritic Spines

Dendritic spines, bulbous protrusions of the dendritic shaft, are important features of neurons. Spines were first identified by Ramón y Cajal in the 19^{th} century and reported to change their morphology with variations in neuronal activity [2, 3]. Studies supported this proposal and reported changes in spine density and morphology with neuronal activity [4, 5, 6, 7]. Spines from hippocampal neurons are related to short-term memory, learning, and neuro-degenerative diseases; for instance Parkinson's and Alzheimer's [17, 18]. It is also found that Alzheimer's disease cause decrease in spines density and dendrite deformation [18]. Spines act as post-synaptic part of synapses [19], and are main receivers for synaptic input [3]. Spines make synaptic connections with neurotransmitters from axon terminals and receive excitatory inputs transmitted by the central nervous system [17]. Dendritic spines keep storing the synaptic strength and assist the transmission of electrical signals to the neuron's cell body. Spacek and Hartmann [20] identified the correlation between the surface area of a synapse and

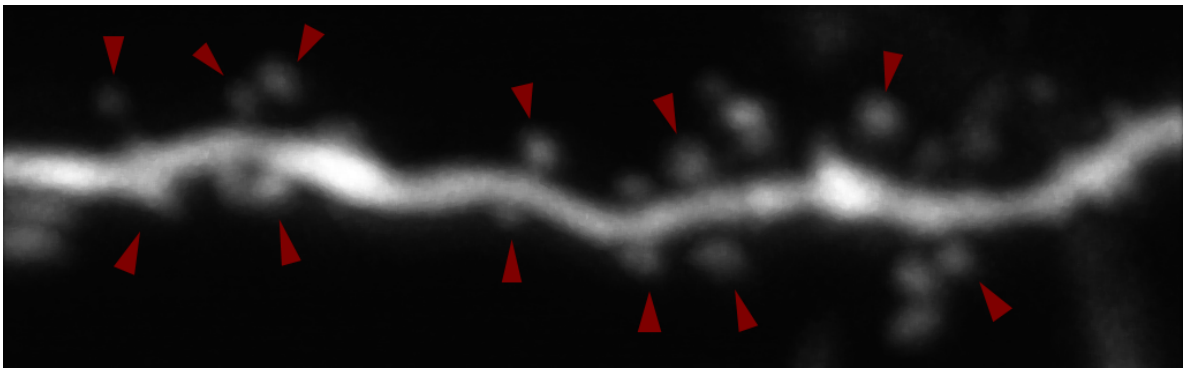


Figure 2.1: A dendritic branch with several spines imaged using a two-photon laser scanning microscope (2PLSM) is shown, arrows point at some of the spines attached to the dendritic branch.

spine's volume, but they did not explicitly study the role of the spine neck and head.

A dendritic branch with several spines captured using 2PLSM is shown in Figure 2.1. Each dendritic spine has a small bulbous head that is connected to the parent dendritic shaft through a narrow neck [8]. Spine head and neck both have different functions and collaborate with each other to transfer synapses received from axons to the dendritic branch. Postsynaptic density (PSD) area is found to be correlated with spine head diameter and number of postsynaptic receivers [3, 21]. Additionally, the neck length of a spine is also reported to be proportional to its functional properties [21], its impedance enables filtering of membrane electrical potentials [3, 21]. Spine neck diameter and length are also reported to affect the diffusional coupling between spine and dendrite [22, 23]. Morphological properties of spine neck and head are usually not proportional to each other, the spine neck diameter and neck length are not related either [3]. Spines have been known to show extraordinary diversity since their discovery [24]. They are reported to have different density and size across different brain areas, cell types, and animal species [3]. Even within a particular cell, spines exhibit a great variety in spine neck and head dimensions [3].

(a) Intensity images collected using 2PLSM

(b) Manual annotations

Figure 2.2: Spine Classes: Mushroom, Stubby, Thin, Filopodia (Left to Right). Intensity and corresponding manually annotated images are shown for each shape class.

2.2 Shape Analysis

There is an on-going debate in the literature whether spine shapes represent distinct classes or should be modeled through a continuum of shape variations. This section discuss both distinct classes and continuum of shape variations perspectives.

2.2.1 Classification

Dendritic spines have different shape types, and researchers believe these different morphological variations could be proportional to various functional roles or developmental stages [25]. Traditionally, dendritic spines in the literature are grouped into four classes: mushroom, stubby, thin, and filopodia [3, 17, 22, 26, 27]. An example of each look of these classes is given in Figure 2.2. Mushroom spines have large bulbous head and long neck, thin spines have small head and thin long neck, whereas neck in stubby spines is either missing or very small, and filopodia are found to have longer necks and generally do not have clear head [3]. As discussed earlier, the distribution of different types of spines varies in different parts of brain. It is also dependent upon age of the animal being imaged. For instance stubby spines are known to be dominant during early postnatal development but they are found in adult animals as well [3]. Grutzendler et al. [28] observed abundance of filopodia type spines in young animals and their absence in adults. Dendritic spine plasticity is greatly reduced in adulthood and long-term memory capability is achieved [29].

2.2.2 Continuum of Shape Variations

The classification of shapes described in previous subsection has been widely applied in most of studies, however, there is still an open question whether distinct spine classes exist or these should be modeled through a continuum of shapes. Peters and Kaiserman-Abramof [27] also pointed that some spines in their dataset had intermediate shapes and were difficult to be assigned to one of the standard classes. Parnass et al. [25] suggested that morphological groups of spines do not depict inherent distinct classes of spines, instead they represent different variations a spine shape can take at different times through its lifetime. Morphological changes in dendritic spines are related to the synaptic function and neuronal activities [17]; Bourne and Harris [30] reported enlargement of thin spines and their transition to mushroom spines upon synaptic enhancement.

Whether to view spine analysis as a classification problem or whether to model them through a continuum of morphological variations is still an open question and being studied extensively in the literature. Arellano et al. [21] reported that classification into traditional spine shape types was not possible due to existence of several spines with intermediate morphological characteristics; they applied morphological features for this characterization. Spacek and Hartman [20] additionally added two intermediate spine classes between stubby, and mushroom; and mushroom, and thin spines. Ruszczycki et. al. [24] suggests that identifying spines in 2 groups (large and small) instead of classifying into different classes results in better sensitivity. Basu et al. [31] reported that human expert was not sure while assigning labels for some of the the spines in their dataset.

Wallace and Bear [32] claimed that morphological measurements of spines acquired from their data do not support the idea of existence of distinct spine classes. They studied spine head diameter, and length; and reported to have a continuous distribution. Mancuso et al. [33] suggested to perform quantitative analysis of spines based on morphological parameters; divide them into natural groups and count spines in different groups. Ruszczycki et. al. [24] believes that there is no standard classification rule, and different researchers may use different criteria. We have also noted similar observations

in the literature that there is no standard for classification and that each group defines the classes based on single or multiple expert/experts they work with, which causes analysis results to suffer from subjectivity.

2.3 Related Work

This section presents a brief summary of some of the studies reported on dendritic spine classification. Although many different algorithms are proposed to segment the dendritic spines automatically, there are a few studies in the literature focused on automated classification of dendritic spines. Rodriguez et al. [22] reported a study on spine classification based on 3D images acquired by confocal laser scanning microscopy (CLSM) and computed head to neck ratio, neck length, head diameter, and aspect ratio. They performed classification using a decision tree and used manual labels assigned by human expert operators to validate performance of their approach. They reported intra-operator and inter-operator variability while assigning the labels. Son et al. [17] used neck diameter, head diameter, shape criteria, area, length, and perimeter with a decision tree to classify spines. They also used CLSM for imaging and human expert assigned labels for evaluation. Shi et al. [19] developed a semi-supervised learning approach based on 3D images acquired using CLSM and used a weighted feature set consisting of neck diameter, head diameter, volume, and length for classification of spines. A recent study on spine classification based on CLSM images extracted morphological features and used a rule-based classification approach [31].

Koh et al. [8] developed a classification approach based on ratio criteria inspired by Harris et al. [34] using the ratio of spine length to neck diameter, and ratio of head diameter to neck diameter. They used 2PLSM to acquire images. Erdil et al. [35] suggests that intensity information in the regions in which a potential neck is likely to be contained can be used to differentiate spine classes. Erdil et al. [35] applied intensity based features to perform classification of spines from 2PLSM intensity images.

Most of the studies on spine analysis focus on CLSM images, there are only a few studies that considered 2PLSM images. Another observation is that most of the stud-

ies considered morphological features and rule based classifiers. This thesis attempts to fill this gap and propose new probabilistic and machine learning methods based classification approaches.

Also it can be noticed from a small subset of studies on classification summarized here, most of the groups use one or more human experts to assign class labels which are later used to evaluate the performance of their supervised classification approaches. Even though using the manually extracted labels as ground truth is a viable approach for this problem, it introduces subjectivity. We attempt to address this issue by presenting a clustering approach aiming to discover natural groups of spine shapes in an unsupervised fashion using various feature representations.

2.4 Manifold Learning

Manifold learning is an important methodology with applications in a wide range of areas including data compression, pattern recognition, and machine learning [36]. Manifold learning can be seen as a dimensionality reduction problem, with the goal of producing a compressed representation of high-dimensional data. It can also be viewed as an algorithm to compute degrees of freedom that would be sufficient to reproduce most of the variability in data [36]. Mathematically, we can formulate the dimensionality reduction or manifold learning problem as follows: given an N-dimensional random variable $\mathbf{x} = (x_1, x_2, ..., x_N)^T$, compute its low dimensional representation, $\mathbf{y} = (y_1, y_2, ..., y_D)^T$ such that $D \leq N$, keeping maximum information from original high-dimensional data according to some criterion [37]. Different algorithms apply different criterion to reduce dimensionality, e.g., principal component analysis (PCA) uses maximum variance as criteria.

Many dimensionality reduction techniques have been developed with application in several areas. These techniques are broadly categorized into linear and non-linear dimensionality reduction techniques. While all of these approaches share a similar objective: reduce dimensionality, approaches applied are different. The reason behind their success is the inherent redundancy in most natural images and the fact that natural

images having high-dimensional data mostly lie near a low-dimensional manifold [36].

PCA is a widely used classical technique that provides a transformed lower dimensional representation attempting to preserve maximum variance, but it is not very effective in various application due to its global linearity property [38]. Multidimensional scaling (MDS) provides a lower dimensional representation attempting to preserve the distance between points, but it suffers from similar problems as PCA [39]. Locally linear embedding (LLE) is a nonlinear dimensionality reduction approach that finds the low-dimensional representation striving to keep embedding of high-dimensional data [40].

ISOMAP is another non-linear dimensionality reduction approach that possesses the best features of PCA and MDS [14]. It can be viewed as an extension of MDS by replacing the Euclidean distance metric with geodesic distance. The Laplacian eigenmaps method constructs a graph by applying the K-nearest neighbors (KNN) and computes its weights in such a way that the norm of the gradient is minimized in the least squares sense [41]. Local Tangent Space Alignment (LTSA) also constructs the graph using KNN and for dimensionality reduction it applies an approximation to local tangent spaces for each neighborhood [42].

2.5 Linear Representation

Wright et al. [15] presented a sparse representation based classification approach for face recognition. The approach consists of two ideas, one is to represent incoming test image as a linear combination of training images, other is to achieve this representation by imposing ℓ_1-norm constraint (sparsity).

2.5.1 The ℓ_1-Norm Method

Sparse representation attempts to compute the sparse decomposition of signals in a dictionary [43]. Sparse representation has proven to be successful in a wide range of applications; from signal representation to acquisition and compression of high dimensional signals [44]. It has also offered effective solutions to computer vision problems

such as face recognition [15] and image classification [45]. It has been claimed that this approach uses the inherent property of most natural images; images from the same class demonstrate degenerate structure [44].

The assumption behind sparsity based classification is that spine shapes from the same class lie on a low-dimensional linear subspace. The idea is to represent incoming test spine image as a linear combination of spines from the training data. The sparse coefficients produced by this representation can then be used for classification [15]. Sparsity requires these coefficients to be dominant for one class and zero for all other classes. This can be achieved using ℓ_0 minimization but for many applications that is an NP-hard problem [15]. However, if the ℓ_0 solution is fairly sparse, it is equivalent to solving the ℓ_1 minimization problem [15].

We construct the matrix $A_i = [s_{i,1}, s_{i,2}, ...s_{i,n_i}] \in \mathbb{R}^{m \times n_i}$ with n_i training samples from the ith class, where each training sample represents a column of A_i by stacking columns of each training image. Hence, each column s_i has $m = width \times height$ rows. Now, assuming sufficient samples are available for training, any new image ($t \in \mathbb{R}^m$) from the ith class can be linearly represented in terms of training images of the same class using Equation 3.1.

$$t = \zeta_{i,1} s_{i,1} + \zeta_{i,2} s_{i,2} + ... + \zeta_{i,n_i} s_{i,n_i} \tag{2.1}$$

where, $\zeta_{i,j} \in \mathbb{R}$ is a scalar $\forall j$. For classifying, as the class membership is initially unknown, we construct a new matrix A containing complete n training data available for all k classes, as illustrated in Equation 3.2.

$$A = [A_1, A_2, ...A_k] = [s_{1,1}, s_{1,2}, ...s_{k,n_k}] \tag{2.2}$$

The linear representation would be modified to the form of Equation 2.3.

$$s = Ax_0 \in \mathbb{R}^m \tag{2.3}$$

Where, $x_0 = [0, ..., \zeta_{i,1}, \zeta_{i,2}, ..., \zeta_{i,n_i}, 0, ..., 0]^T \in \mathbb{R}^n$ is the sparse coefficients vector, ideally with all zero elements except the ones associated with the ith class. Here, class information is encoded in entries of vector x_0, which can be easily exploited to perform classification. As discussed earlier, ℓ_0 minimization problem is NP-hard and it is

equivalent to ℓ_1 solution assuming that it is sufficiently sparse. The solution for this problem can be achieved in polynomial time and there are several solutions reported in the literature.

Minimum residuals based classification (also referred as sparse representation-based classification algorithm, SRC) has been introduced by Wright et. al. [15] to perform classification when we represent a test image as a linear combination of training images. As name suggests, it performs classification based on minimum residuals (as illustrated in Equation 2.4). Where, $\delta_i(\hat{x}_i)$ represent residuals for the ith class. The idea is that a test image would ideally be represented by its representative class, which is not practically achievable due to noise and other artifacts in real images. In any case, while there might be some representation coefficients belonging to wrong class, most of the coefficients should come from true class.

$$Class(t) = argmin_i \|t - A\delta_i(\hat{x}_i)\|_2 \tag{2.4}$$

2.5.2 The Least Squares Method

The idea behind the least squares method is similar to ℓ_1 approach, i.e., represent test image as a linear combination of training images. However, in comparison to the ℓ_1 case, ζ are estimated by applying the least squares method using Equation 2.5 [16].

$$\hat{\zeta} = argmin_{\alpha \in \mathbb{R}^n} \|t - A\zeta\|_2 \tag{2.5}$$

Solution of Equation 2.5 can be found by re-formulating the psuedo-inverse. We can perform QR factorization, since our input data is real $\mathbf{Q}$ would form an orthonormal basis, and $\mathbf{R}$ an upper triangle matrix. Using this approach we can estimate representation coefficients $\hat{\zeta}$, as given in Equation 2.6. Once the representation coefficients have been estimated SRC algorithm can be applied to perform classification.

$$\begin{aligned} &\text{Compute } \mathbf{QR} = \mathbf{A} \\ &\hat{\zeta} = \mathbf{R}^{-1}\mathbf{Q}^T\mathbf{t} \end{aligned} \tag{2.6}$$

2.5.3 The ℓ_2-Norm Method

Using this approach, we represent the test image as a linear combination of training images, however, we use Tikhonov regularization to achieve this representation. We estimate the representation coefficients, $\hat{\zeta}$, by applying ℓ_2-norm constraint on coefficients, as illustrated in Equation 2.7. In order to perform classification SRC algorithm is applied on achieved representation.

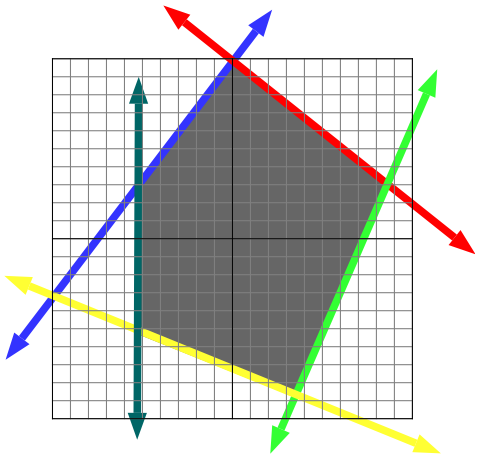
$$minimize \left\| Ax - t \right\|_2^2 + \lambda^2 \left\| x \right\|_2 \tag{2.7}$$

2.6 Disjunctive Normal Shape Models (DNSM)

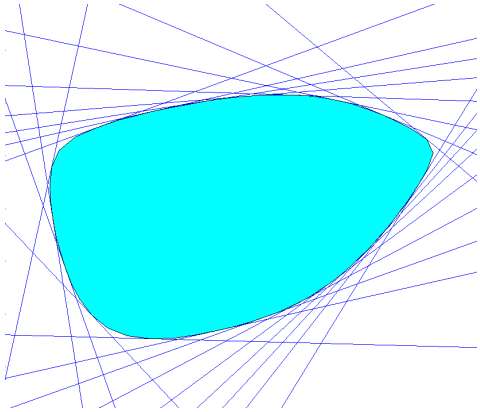
Disjunctive Normal Shape Models (DNSM) is a recently proposed shape model; we exploit its parametric nature and use it as a feature extraction approach. DNSM [11] is an implicit model that represents a shape by union of convex polytopes that are constructed by intersections of half spaces. Mesadi et al. [1] introduced DNSM-based shape and appearance priors and tested their potential on various segmentation problems. This approach has proven successful, it provides better segmentation as compared to the state-of-the-art approaches.

Shapes can be represented using a characteristic function, and DNSM approximates shape characteristic function by a union of N convex polytopes. These polytopes are constructed by intersection of M half-spaces, as illustrated in Figure 2.3(a). These half-spaces are further relaxed using sigmoid function, a smoothed polytope is shown in Figure 2.3(b). The resulting DNSM approximation to the characteristic function of shape is presented in Equation 2.8. In the following equation, $\mathbf{x} = \{x, y, 1\}$, and $D = 2$ for 2-dimensional ($2D$) shapes. w_{ijk} are the only free parameters in the DNSM and they determine the position and orientation of half-spaces (discriminants). For further details of the DNSM, readers are referred to [11].

$$f(\mathbf{x}) = 1 - \prod_{i=1}^{N} \left(1 - \prod_{j=1}^{M} \frac{1}{1 + e^{\sum_{k=1}^{D+1} w_{ijk} x_k}} \right) \tag{2.8}$$



(a) Sharp Polytopes



(b) Smooth Polytopes

Figure 2.3: Sharp and Smoothed polytopes to illustrate shape representation using DNSM.

Further, Mesadi et al. [1] introduced DNSM based shape and appearance priors to improve segmentation. In this paper, we apply this DNSM based approach to segment dendritic spines. This approach exploits the parametric nature of DNSM, and learns shape and appearance features from training data to segment test images. DNSM shape and appearance priors based approach has two stages for segmentation of spines: training, and testing.

The training stage consists of two steps, first represent the manually segmented (binary) image using DNSM parameters. During second step, we construct local appearance and shape priors from training intensity and binary images. This method gen-

erates an ample amount of shape variations by local combinations of training shapes, which enables this method to produce good segmentation results even with limited training data. Additionally, local appearance priors constructed by intensity statistics around each half-plane equips this method with better expressive capability to represent training data.

Images are segmented in the testing stage by minimizing the weighted average of appearance and shape energy terms. Weights w_{ijk} are updated in each iteration using the gradient descent, as illustrated in Equation 2.9, where α, and γ are the levels of contributions from shape and appearance terms in updating the weights, w_{ijk}.

$$w_{ijk} \leftarrow w_{ijk} - \alpha \frac{\partial E_{Shape}}{\partial w_{ijk}} - \gamma \frac{\partial E_{Appr}}{\partial w_{ijk}} \tag{2.9}$$

2.7 Histogram of Oriented Gradients (HOG)

Histogram of oriented gradients (HOG) [12], as the name suggests computes histograms of gradient directions and applies contrast normalization. HOG has been widely used for object detection and recognition tasks in computer vision. HOG [12] characterizes the local appearance features by computing 1-D histograms of gradient orientations. HOG has been studied extensively in the literature and has been successful in various object detection and recognition tasks.

The HOG representation is achieved in several steps: first step involves dividing the image into small spatial regions called “cells" and computing gradient orientations in each cell. Then gradient orientations are divided into smaller regions called “bins". A 1-D histogram is constructed for each bin by accumulating corresponding gradient directions. Further, it is suggested by Dalal and Triggs [12] to apply contrast normalization in order to achieve illumination in variance properties for achieved descriptors. Contrast normalization is applied using relatively large sized regions called “blocks" and normalizing the cell histograms by block histograms. It is suggested to use overlapping blocks for sufficient contrast normalization. Final step involves constructing a single 1-D descriptor by combining all histograms.

2.8 Intensity Profile Based Features

Erdil et al. [35] has proposed a joint classification and segmentation approach for dendritic spine segmentation in 2PLSM images. That study suggests that intensity information in the regions in which a potential neck is likely to be contained in can be used to detect spine classes. Regions where the neck might appear is found using the assumption that the spine neck lies below the spine head. Once the spine head is found by minimizing an intensity-based energy function using active contours [46], the proposed approach creates two rectangular regions below the spine head as shown in Figure 2.4. The first region shown in Figure 2.4(a) is constructed such that the bottom point of the spine head (shown by a red cross) lies at the center of the rectangle. The second rectangular region shown in Figure 2.4(b) is a narrower one and is drawn such that it is located just below the spine head. Erdil et al. extract three sets of feature vectors by exploiting intensities in these rectangular regions. The first set of feature vectors is obtained by summing up the intensities in the first rectangle horizontally. Similarly, the second set of feature vectors are obtained by vertical summation of the intensities in the corresponding rectangle. The final set of feature vectors are the histograms of intensities in the second rectangular region. Erdil et al. [35] used these feature vectors for classification of mushroom and stubby spines and report their effectiveness. In this thesis, we investigate the performance of these feature vectors in clustering of dendritic spines.

(a) First region (b) Second region

Figure 2.4: Regions in which a potential neck is likely to be contained.

2.9 Deep Learning

The recent success of convolutional neural network (CNNs) in various image classification tasks have a tremendous impact on machine learning research. The reasons behind their success are availability of large datasets and their ability to automatically extract reliable mid-level features. Conventional methods involve a feature extraction step before training a classifier to be able to classify new images. This typically involves designing features specialized for a domain, which is a demanding task and requires domain knowledge. For instance, most of the studies on spine classification extract morphological features of dendritic spines. While designing these features is a comprehensive task, these hand crafted features become over-specialized to specific datasets. It becomes worse when these over-specialized features are combined with rule based classifiers, as it is practiced in most of the spine classification studies [22, 17, 19, 31, 8]. In contrast, deep learning methods learn mid-level features during the training stage which are not manually designed. However, deep learning methods require large amount of dataset during the training process, which is not achievable in many biomedical image analysis techniques. Therefore, we apply transfer learning with CNNs to cope with small dataset problem.

Transfer learning efforts to transfer information learned from source task(s) to improve learning of a target task; source and target tasks are mostly related to each other [47]. With recent success of CNNs in various image classification tasks, transfer learning has been successfully applied with CNNs to learn a target task which in some cases is very different from source task. The reason behind applying transfer learning with CNNs is that CNNs generally require large datasets for training from scratch. In this context, transfer learning enables researchers to use CNNs as a feature extraction technique as well as fine-tune a model trained on source task with target task dataset. However, transferability depends upon the distance between source and target tasks [48].

AlexNet [49] consists of eight layers, the first 5 layers are convolutional and the last 3 layers are fully-connected. An N-dimensional softmax is followed by the last layer which

produces a distribution for each class. AlexNet uses the multinomial logistic regression as the objective function for classification. AlexNet won the ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC) [50]. ImageNet [51] is a large dataset consisting of millions of high-resolution images including thousands of object categories and ILSVRC deals with recognizing objects in this dataset. Success of AlexNet in ILSVRC 2012 confirms its robustness and reliability of its feature extraction process. As mentioned earlier, transfer learning can be applied in two ways with CNNs: (i) use trained CNNs as a feature extractor or (ii) Use trained CNNs as initialization and fine-tune network weights on target dataset.

Chapter 3

Morphological Features for Spine Shape Analysis

We have proposed morphological features based approach for both 2D projections and 3D data. We describe both of the morphological feature extraction methods in this chapter.

3.1 Morphological Analysis in 2D (Morph2D)

We have developed procedures to extract features from 2D projections, that are informative about the spine shape classes. Basic image processing techniques are applied to compute morphological features of dendritic spines. We start by segmenting spines using DNSM and use segmented images to extract morphological features. The features used in this study are listed below:

- Head Diameter
- Neck Length
- Area (No. of Pixels in foreground)
- Perimeter
- Height of bounding box

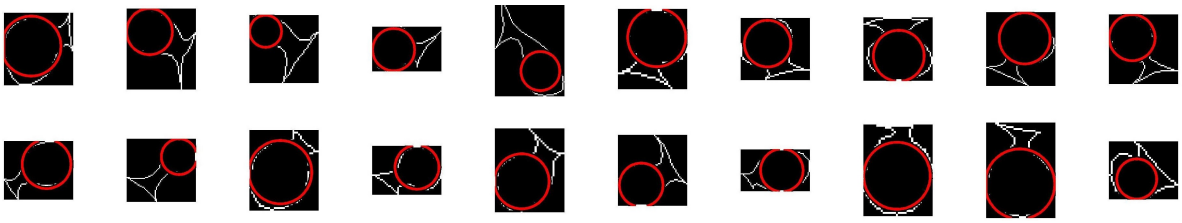


Figure 3.1: Circle fitting results for some of the spines.

- Width of bounding box
- Neck Length to Head Diameter Ratio
- Circularity
- White to Black Pixels ratio in bounding box
- Shape Factor

In order to compute the head diameter, Hough Circle Transform (HCT) [52] is applied to fit the biggest circle inside the spine. For some of the spines, HCT fails to fit a circle in the spine head. In this case, the ellipse fitting algorithm of [53] is applied. Finally head diameter is computed from the diameter of the circle or the axes of the ellipse fitted in the spine head. The results of the circle fitting algorithm are presented in Figure 3.1 for some of the spines. Circularity is computed using perimeter and area as shown in Equation 3.1.

$$Circularity = \frac{Perimeter^2}{4\pi \times Area} \tag{3.1}$$

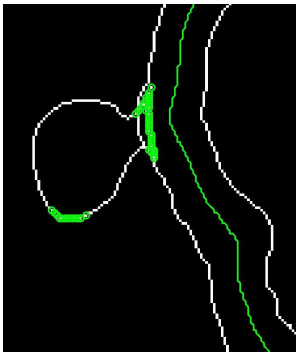
Neck length computation is a challenging process. First, dendrite perimeter and medial axis are extracted from maximum intensity projection image, to be used at later stage as reference point. First we applied Otsu thresholding [54] get a rough segmentation of the dendrite (which included spines as well), and skeletonized this segment using a fast marching distance transform approach [55]. Then in order to exclude spines from the dendrite we applied erosion with a locally-adaptive sized, disk-shaped structuring element that runs over the medial axis. To achieve size variation,

Figure 3.2: If circle fitted on spine head intersects with dendrite ⇒ NeckLength = 0.

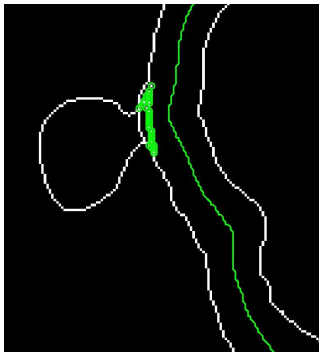
at every medial axis location diameter of the structuring element was adapted to the measured width of the segment.

Based on manual analysis of stubby spines, a heuristic is applied, if the circle fitted on spine head intersects with dendrite, it is concluded that the spine does not have a neck, as shown in Figure3.2. Otherwise, neck length computation algorithm is applied. Then the algorithm computes the distance from spine boundary points to the center of head, and selects top N points with maximum distance, as illustrated in Figure 3.3(a). Subsequently the distance is calculated between sorted spine points and dendrite medial axis. A threshold (T_m, maximum allowed distance) is applied to the distance between these N points and the dendrite medial axis. T_m is computed as follows: $T_m = meanDistance + 2 \times StandardDeviation$, where $meanDistance$ and $StandardDeviation$ represent mean and standard deviation of distance from each sorted spine point to dendrite medial axis respectively. Pixels below T_m are selected as candidate pixels for base points, as depicted in Figure 3.3(b). Base points are the pixels where the spine is connected to the dendrite surface. This approach allows us to locate the pixels closest to the dendrite and furthest from the spine head.

Among the candidate pixels, the two pixels with maximum distance from each other under the condition distance $\leq 3 \times headRadius$ (as shown in Figure 3.4) are selected to be the base pixels of the spine, here $headRadius$ represent radius of spine head. Finally, Multistencil Fast Marching (MSFM) method [56] is used to construct a distance map. This map is used as an input for the Runge-Kutta algorithm [57] to calculate the shortest path between center point of the spine head and the target point (center point between base pixels). Shortest path results for neck length computation for a few images are depicted in Figure 3.5. Neck length is measured by subtracting the radius of the head from shortest path length ($Dist$ = shortest path length), as described in Equation 3.2.



(a) N points are selected with top distance from boundary points to head center.



(b) Points with $distance \leq T_m$ from dendrite are selected.

Figure 3.3: Selecting candidates points for neck base.

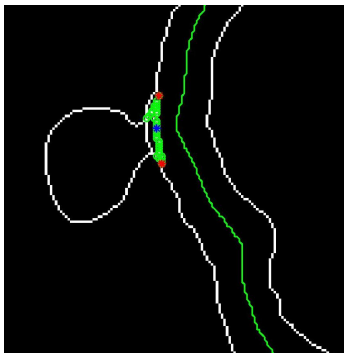


Figure 3.4: Neck base points selection.

$$NeckLength = Dist - headRadius \tag{3.2}$$

To compute shape factor, which consists of three features, the algorithm fits a circle inside the bounding box of the spine with $radius = (NeckLength + HeadDiameter)/2$. Then white pixels inside the circle, white pixels outside the circle, black pixels inside the circle are calculated and serve as the three features of the shape factor. Classification results using this approach has been reported on a small dataset in [58].

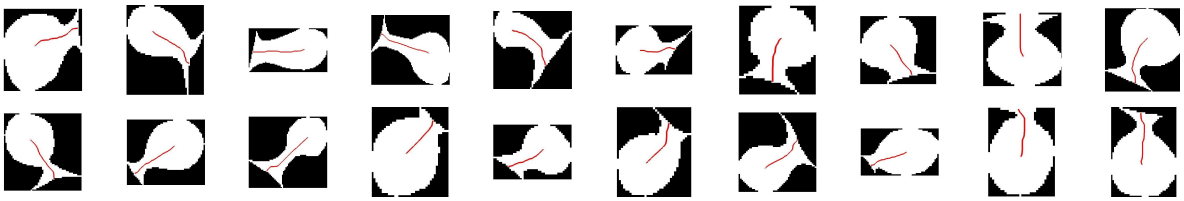


Figure 3.5: Shortest Paths for Neck Length Computation

3.2 Morphological Analysis in 3D (Morph3D)

We present a new approach for analysis of dendritic spine shapes using 3D information without the need to segment spine images. We compute $3D$ neck length, align neck paths and extract shape and appearance features using neck path information.

We start the process by manually selecting a region of interest (ROI) around the spine. The ROI is selected in such a way that spine head is located around center of it. Further, we use water-shed based segmentation approach to segment spine head. Once the head of spine of interest has been segmented, a fast marching algorithm [56] computes the spine neck path from the center of the head of the segmented spine to a number of candidate target locations on the proximal surface of the segmented dendrite, which results into a neck path for each target location. Further, we apply three constraints to select the neck path from these candidate paths. These constraints are neck path length, path complexity (ℓ_1-norm of path derivatives), and path smoothness (ℓ_1-norm of image intensities along the path). We select the neck path that has collectively the lowest value for these three constraints.

3.2.1 Spine Neck Path and Length

Neck length computation is a challenging task due to spine shape variations and neck motility. We begin with partial segmentation of spine head by applying watershed segmentation using $k = 1$. This is further used to compute the center of spine head by finding its center of mass. Further, dendrite skeleton and segmentation is computed in $2D$ using techniques described earlier in the paper. In order to map the dendrite on z-axis, we construct a vector with intensity values for all slices on z-axis at each

skeleton point and fit a Gaussian. The mean value of fitted Gaussian corresponds to coordinate of dendrite in z-direction. These observations are noisy due to the fact that often there are spines on dendrites (along z-direction). To cope with this noise, median of all z-coordinate values is computed. Although this assumption is not always true globally (for entire dendritic branch), however, this approximation holds locally (in the region of interest). Similar approach is used to map center of spine head on z-axis.

Each slice of dendritic branch image is eroded with a disk-structuring element to reduce the spurious paths. Multi stencil fast marching (MSFM) method [56] is applied to compute the $3D$ distance map using spine head center as source point. The Runge-Kutta algorithm [57] is applied on $3D$ distance map to compute the shortest paths (geodesic) from N point on dendrite perimeter to the spine head center. These points are selected by finding N nearest points from spine head center to dendrite perimeter (using Euclidean distance as metric).

Finally, selection of the correct neck path is the crucial step. A simple approach would be to select the path with minimum length (Equation 3.3), but it would fail in this scenario because of motile nature of spine necks. Therefore, mere path length constraint is not enough. We introduced two additional constraints to select the path with best geodesic approximation. The first additional constraint is path complexity (Equation 3.4), i.e. path should be as simple as possible. Other constraint is smoothness of image intensities on the path (Equation 3.5), i.e. intensity changes on the path should be as minimal as possible. Equation 3.6 is applied to find the correct neck path. Final neck paths for some of the spines are shown in Figure 3.6.

$$L_P = \oint_{\vec{P}} \mathrm{d}s \tag{3.3}$$

$$C_P = \left\| \frac{\partial P}{\partial x} \right\|_1 + \left\| \frac{\partial P}{\partial y} \right\|_1 + \left\| \frac{\partial P}{\partial z} \right\|_1 \tag{3.4}$$

$$S_P = \left\| \frac{\mathrm{d}V(x_P, y_P, z_P)}{\mathrm{d}I} \right\|_1 \tag{3.5}$$

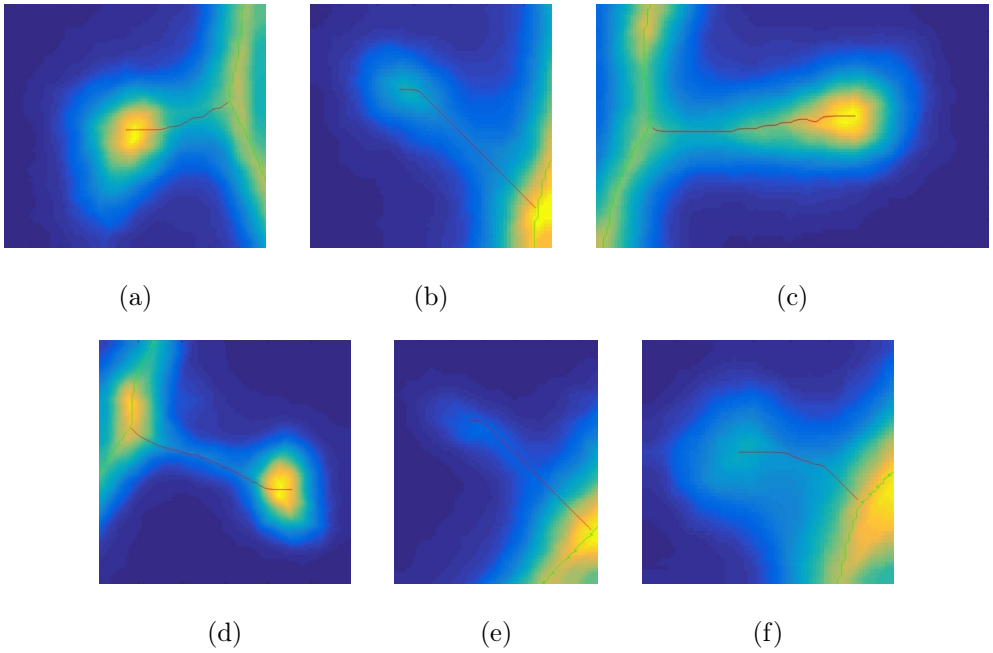


Figure 3.6: Neck paths for some of the spines.

$$NeckPath = argmin_P \left(\frac{L_P}{max(L_p)} + \frac{C_P}{max(C_p)} + \frac{S_P}{max(S_p)} \right) \tag{3.6}$$

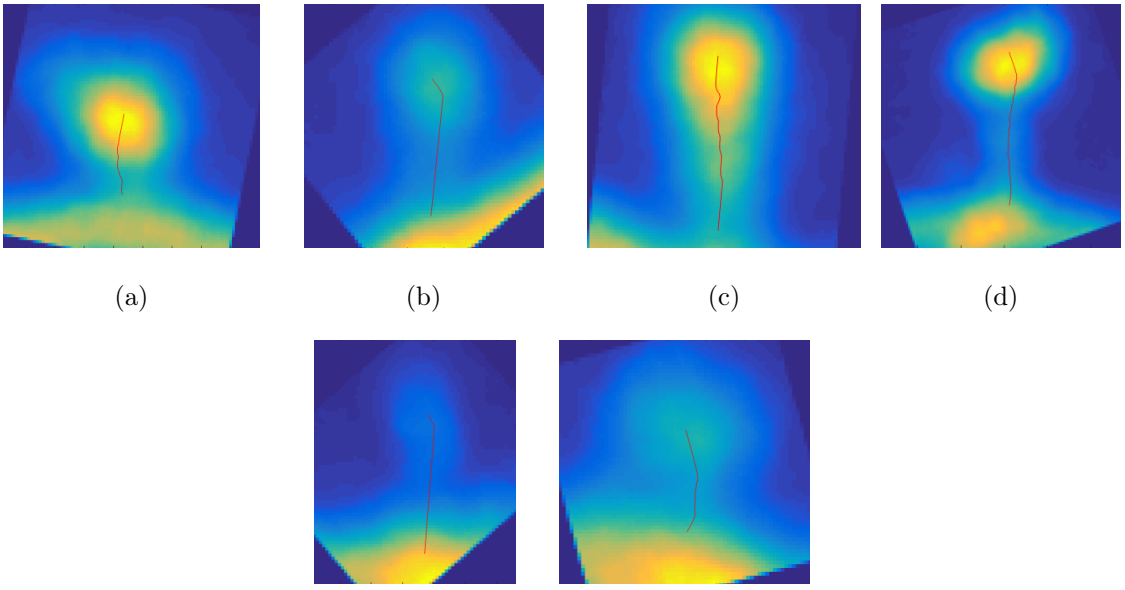
Equation 3.3 corresponds to the path length from dendrite surface to spine head center. To compute neck length, we first compute the radius of spine head by fitting a circle using Hough Circle Transform (as suggested in [58]) on watershed segmented spine head with $k = 5$ and then use Equation 3.7.

$$NeckLength = L_P - radius \tag{3.7}$$

3.2.2 Neck Shape Representation Using Neck Path

The $3D$ neck paths that our approach finds provide a representation of neck shape, if we can align all spines and neck paths according to a common reference, these paths can serve a nice representation of spine neck shape which can further be used for spine classification. Spines can be aligned as explained below:

- Find angle between neck base and center of the spine head.



(a) (b) (c) (d)

(e) (f)

Figure 3.7: Aligned neck paths for some of the spines.

- Compute the alignment angle based on position of spine with respect to surface of dendrite.
- Align path by applying geometric transform to rotate the path according to alignment angle.

We apply this approach to align neck paths, results are shown in Figure 3.7, as it can be seen it produces reasonable results. Spine neck path can be used a representation of spine neck shape, basic geometric features computed from this path can be used for classification δx, δy, δz. We construct feature vector for classification consisting of head diameter, neck length, neck path shape features (δx, δy, δz) and neck path appearance features (gradient of intensities on the path).

Chapter 4

Classification

Several feature extraction techniques have been proposed to apply for dendritic spine shape analysis, this chapter describes these feature extraction techniques, kernel density estimation (KDE) based classification framework, and concludes with classification results. We start by explaining the data collection process performed using 2PLSM and then discuss each feature extraction method applied in this thesis. To perform classification, we use KDE, this non-parametric approach intrinsically provides the likelihood of membership for each spine class compared to other approaches that apply different techniques to provide scores which can be interpreted as probabilities. Hence, our KDE-based approach has the potential to represent complicated shape distributions well. Additionally, it provides a simplified framework that enables us to examine the shape distributions, including the question of whether the spine shapes constitute a continuum across classes.

4.1 Data Acquisition

In order to be imaged under 2PLSM, hippocampal neurons from mouse organotypic slice cultures postnatal day 7-10[1] were transfected using biolistic gene transfer with gold beads (10 mg, 1.6 um diameter, Biorad) coated with Dendra-2 (Evrogen) plasmid DNA ($100\mu g$) or AFP using a Biorad Helios gene gun after 4 or 7 days in vitro.

[1] All animal experiments are carried out in accordance with European Union regulations on animal care and use, and with the approval of the Portuguese Veterinary Authority (DGV).

Imaging experiments were performed 2 to 5 days post-transfection. Slices were perfused with artificial cerebrospinal fluid (ACSF) containing 127 mM NaCl, 2.5 mM KCl, 25 mM $NaHCO_3$, 1.25 mM NaH_2PO_4, 25 mM D-glucose, 2 mM $CaCl_2$ and 1 mM $MgCl_2$ (equilibrated with O2 95%, CO2 5%) at room temperature at a rate of 1.5 ml/min. Two-photon imaging was performed using a galvanometer-based scanning system (Prairie Technologies, acquired by Bruker Inc.) on an Olympus BX61WI equipped with $60X$ water immersion objective (0.9 NA), using a Ti:sapphire laser (Coherent Inc.) controlled by PrairieView software. Z-stacks ($0.3\mu m$ axial spacing) from secondary or tertiary dendrites from CA1 neurons were collected every 5 minutes up to 4 hours. Field of view was $19.8 \times 19.8\mu m$ at 1024×1024 pixels.

This study is based on 2PLSM images. The reason behind using 2PLSM images is that it allows imaging living cells. This is possible due to the property of 2PLSM that it attempts to minimize photo-damage and photobleaching; two of the major limitations of fluorescence microscopy of living tissues and cells [9].

We acquired 3D stacks of 40 dendritic branches. Further, we project 3D images to 2D using Maximum Intensity Projection (MIP,also known as Maximum Activity Projection) [59] and apply median filtering to reduce noise. The ground truth for segmentation and classification has been prepared by an expert from 2D images. We selected 456 dendritic spines including: 288 mushroom, 113 stubby, and 55 thin type spines.

4.2 Feature Selection

DNSM, HOG, $3D$ neck shape based method, and AlexNet produce high-dimensional feature vectorss for each spine image. We are considering a 3 class problem here, some of the features might be redundant or less relevant for classification. This is why it would be interesting experiment to apply some feature selection techniques and perform classification on reduced features. We consider two feature selection techniques here: (i) correlation based feature selection (CFS) [60], and (ii) information gain based feature selection (IG) [61].

CFS [60] selects the features based on correlation; prefers the features with high correlation to a class and low intercorrelation. It ignores (irrelevant) features which have low correlation with class. The features having high intercorrelation are considered redundant and ignored. Hence, CFS accepts a feature if it has a high correlation to class and another feature does not have high correlation in that area of feature space.

IG performs feature selection based on information gain with respect to a class. It computes information gain for all combinations of classes and features using equation 4.1 [61], here H represents entropy that is used to measure information in a process [62]. It computes the change in information when we are provided with knowledge of a particular feature with respect to that class. We select the features with IG score of 0.1 and greater.

$$InfoGain(Class, Feature) = H(Class) - H(Class|Feature) \tag{4.1}$$

4.3 Kernel Density Estimation

We estimate non-parametric density using kernel density estimation (KDE) and apply a likelihood ration (LRT) to perform classification. Our non-parametric density estimation approach is similar to [63]. Assume we have m-dimensional feature vector: $x_1, x_2, \ldots, x_m$, sampled from n-dimensional density function $p(x)$. The Parzen density can be estimated by applying Equation 4.2.

$$\hat{p}(x) = \frac{1}{m} \sum_{i=1}^{m} k(x - x_i, \Sigma) \tag{4.2}$$

Where, $k(x, \Sigma) = \mathcal{N}(x; 0, \Sigma^T \Sigma)$ is an n-dimensional kernel, which can be simplified using the assumption that kernel is spherical, i.e., $\Sigma = \sigma I$. Applying this assumption Equation 4.2 can be simplified, as given in Equation 4.3.

$$\hat{p}(x) = \frac{1}{m} \sum_{i=1}^{m} k(d(x, x_i), \sigma) \tag{4.3}$$

Where $d(x, x_i)$ is the ℓ_2 distance between x and x_i in $\mathbb{R}^n$ and $k(x, \sigma) = \mathcal{N}(x; 0, \sigma^2)$

is the 1D Gaussian kernel. Kernel size (σ) is estimated by the bracket method (also known as the bisection method) [64]. First, we compute 1D kernel size from each feature vector and use this m dimensional kernel size vector to compute minimum (σ_{min}) and maximum kernel size (σ_{max}). Finally, we apply the bracket method to compute the optimal kernel size in $[\sigma_{min}, \sigma_{max}]$ range by iteratively bisecting the interval and selecting the subinterval that contains the optimal kernel size.

Once we have estimated the likelihood of an image belonging to Mushroom (l_m), Stubby (l_s), and Thin (l_t) classes, we can perform classification using the LRT. This is a 3-class problem and requires multiple likelihood comparisons; we define 2 likelihood ratios, as depicted in Equation 4.4, where $\mathcal{L}_s$ stands for stubby and $\mathcal{L}_t$ for thin spines.

$$\begin{aligned} \mathcal{L}_s &= \frac{l_s}{l_m} \\ \mathcal{L}_t &= \frac{l_t}{l_m} \end{aligned} \tag{4.4}$$

Finally, we can compare these likelihood ratios to perform classification, as illustrated in Equation 4.5, Equation 4.6, and Equation 4.7. Here “Not M" means do not decide Mushroom as classification decision, “Not S" denotes do not decide Stubby, and “Not T" depicts do not decide Thin as classification decision. In this manner we use reductionist approach until we are left with only one possible class that is used as the decision. This approach simplifies the classification process by mapping an n-dimensional classification problem to 2D problem, specifying the problem in terms of likelihood ratios. Figure 4.1 illustrates the decision regions for classification in the 2D likelihood ratio space.

$$\mathcal{L}_s \underset{\text{Not S}}{\overset{\text{Not M}}{\gtrless}} 1 \tag{4.5}$$

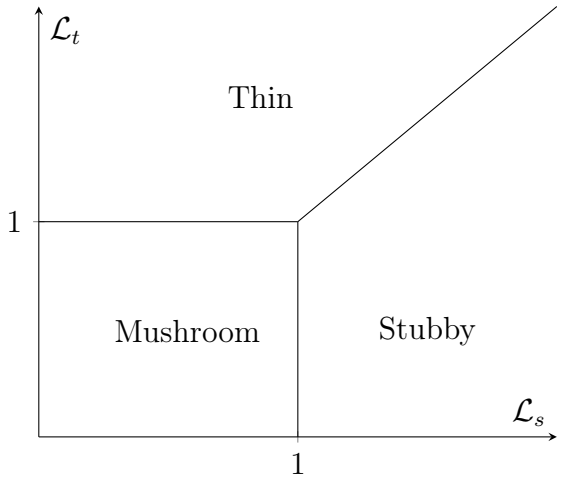


Figure 4.1: Decision regions for classification in $2D$ likelihood ratio space

$$\begin{matrix}\text{Not T}\\ \mathcal{L}_s \quad \gtrless \quad \mathcal{L}_t \\ \text{Not S}\end{matrix} \tag{4.6}$$

$$\begin{matrix}\text{Not M}\\ \mathcal{L}_t \quad \gtrless \quad 1 \\ \text{Not T}\end{matrix} \tag{4.7}$$

4.4 Shape and Appearance Features Based Approach

We use DNSM to perform segmentation and shape feature extraction. This method requires manually segmented (binary) images to train DNSM shape priors. We perform the following procedure to prepare the aligned dataset. Firstly, we choose a region of interest (ROI) in the projected $2D$ image for each spine. The ROI is selected in a way that spine head center is placed approximately at the center of the ROI. Further, each spine image is scaled to 250×250 pixels. In order to keep the aspect ratio same (so that scaling does not affect the shape), it is made sure that selected ROI is a square. Finally, we rotate each spine image such that spine neck is perpendicular to horizontal axis.

This process currently involves manual procedures. However, the process can be automated by applying Hough Circle Transform [52] to locate the spine head center

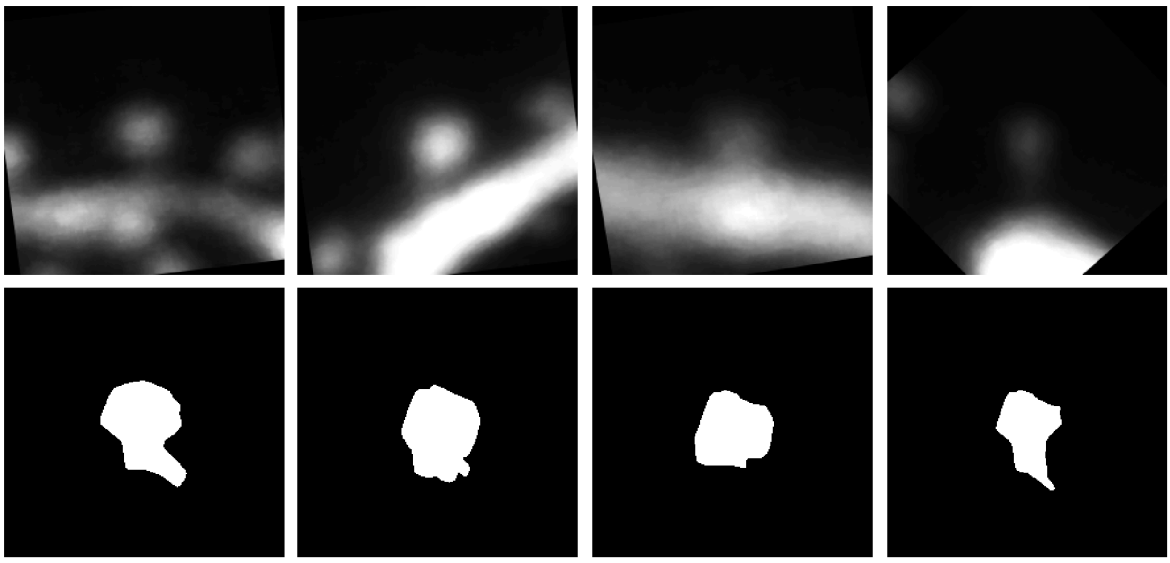


Figure 4.2: A few images from dataset, without segmentation (above) and segmented images (below). First 2 spines are labeled as Mushroom, 3rd spine as Stubby, and 4th as Thin. Spines are segmented using DNSM shape and appearance priors based approach [1]. Automated segmentation results are not perfect; sometimes suffered from over-segmentation and in some cases under-segmentation. But these segmentation results fairly represent the shape types and can be used for classification.

and by computing spine neck path to detect its orientation with respect to the dendrite surface [58]. We apply this approach to segment the dendritic spines from MIP images and use its parameters as shape feature.

DNSM shape and appearance priors based approach [1] has two stages for segmentation of spines: training, and testing. For this purpose, we divided our data into training and testing sets using 10 folds cross validation procedure. This approach uses local shape and appearance priors, use of local appearance priors is especially important for spine segmentation, since it requires to distinguish between dendrite and spine regions. Since intensity level for spine and dendrite regions are similar in 2PLSM images, it becomes crucial to use local appearance priors in order to provide good spine segmentation.

This approach has several parameters which must be tuned for different applications: number of half spaces M, number of polytopes N; and level of contribution from shape

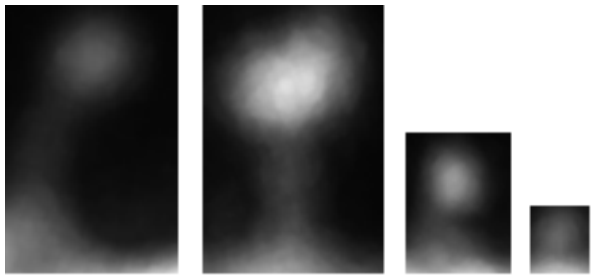


Figure 4.3: A few images from dataset prepared for HOG. First spine is labeled as Thin, 2nd and 3rd as Mushroom, and 4th as Stubby (from left to right).

priors α, and appearance priors γ. We used: $M = 16$, $N = 8$, $\alpha = 0.05$, and $\gamma = 0.5$, these values were empirically found. Segmentation results for a few spines are given in Figure 4.2. As a result of segmentation, we have achieved the approximated form of characteristic function of each spine shape in terms of DNSM parameters. The DNSM represents each segmented image using $M \times N \times 3$ parameters. As discussed earlier, we exploit DNSM's parametric nature and use DNSM parameters as feature vectors to train the classifier to perform spine classification. Classification results using DNSM features has been reported on a small dataset in [13].

We realized from intensity images dataset analysis that intensity distribution inside spines also contain some useful information which could be used for classification. However, we also noticed that intensity level vary with dendritic branch and datasets, that points towards using gradients rather than absolute intensity information. We observed in our data that spines have uniform intensities inside the head, however, this is not true for the neck. A decreasing intensity pattern can be noticed in the neck part of spines. Hence using this local appearance information would assist us in the spine classification task. Therefore, we decided to apply HOG to compute appearance features.

In order to compute HOG features, we select a region of interest (ROI) in intensity images such that the spine is completely inside the ROI, this does not require ROI for all spines to have same dimensions. Further, we rotate the ROI in the sense that spine necks are vertically aligned. A few images from this dataset are shown in Figure 4.3.

As discussed earlier, HOG has different components and changing their values affect the descriptors, these parameters include: cell size, block size, block overlap ratio, and

number of oriented histogram bins. Choice for value of cell size depends upon if we are interested on global/large-scale features or local features. If we are interested in large-scale features, a large cell size value would be desired. Because if we increase the cell size, it will capture more pixels, as a result we will loose local level or small-scale changes and will only be able to keep global information. Similar to cell size, a small block size will allow us to keep track of local illumination changes. If we want to perform contrast normalization, a block size value greater than cell size would be required. Contrast normalization is also controlled through block overlap, the number of overlapping cells in adjacent blocks. If we use unsigned gradient directions, we cannot differentiate between dark to light, and light to dark intensity transitions. The number of histogram orientation bins controls the size of feature vector; a large value increases the feature vector size but enables capturing finer intensity changes. We did experiment with different values of HOG features and empirically selected the values of HOG parameters to be, $CellSize = \big[height/5, width/5\big]$, $BlockSize = 2 \times CellSize$, $BlockOverlap = 1$, and $Bins = 9$.

4.5 Linear Representation Based Approach

The motivation behind applying this method is the success of [15] for face recognition problem. It presents two ideas, one is to represent incoming test image as a linear combination of training images, other is to achieve this representation by imposing ℓ_1-norm constraint (sparsity). We perform different experiments to analyze performance boost is achieved due to the idea of linear representation or sparsity constraint. To best of the our knowledge, this approach has not been applied previously for spine analysis, therefore, it is an interesting experiment to find out if the sparsity assumption holds for dendritic spines.

We applied the DNSM [11] based algorithm to automatically segment the dendritic spines. Further, we apply the least squares method, the ℓ_2-norm method, and the ℓ_1-norm method to represent test images and apply SRC algorithm to classify spines. We use ℓ_1-minimization toolbox proposed in [65] to achieve ℓ_1 representation coefficients.

We use 10-fold cross validation approach to automatically segment the spines using DNSM. Same training and testing folds are used during classification.

To achieve ℓ_1 representation, we use an ℓ_1 regularized least squares problem (LSP) solution proposed by Kim et al.[66], as presented in Equation 4.8.

$$minimize \left\|Ax - t\right\|^2 + \lambda \left\|x\right\|_1 \tag{4.8}$$

where $x \in \mathbb{R}^n$ is the variable, λ is the regularization parameter and $t \in \mathbb{R}^m$.

Optimizing regularization parameter λ is important to achieve sufficiently sparse solutions using Equation 4.8. For this purpose, we identify a sparsity measure and optimize sparsity for both of these techniques. Hurley and Rickard [67] compared different sparsity measures and declared Gini index (GI) to be performing best based on several intuitive attributes. GI is widely used as a sparsity measure; have various advantages over other methods, GI is normalized, an index value of 0 means least sparse solution and 1 means the sparsest solution. We use bisection method (bracket method) to optimize the regularization parameter λ. Optimized value of λ for ℓ_1-based solution (computed using Equation 4.8) is found to be 401.45 with average GI value of 0.971.

For an overdetermined case, $m > n$, the solution of linear system of equations, x_0, is mostly unique. Here, n is the number of training images and m is the size of image ($width \times height$). Hence, uniqueness of solution depends upon number of training images and their dimensions. For spine classification problem, $m = 456$ and $n = 62,500$, therefore it is an overdetermined system.

For ℓ_2-norm method, we estimate the representation coefficients, $\hat{\zeta}$, by applying ℓ_2-norm constraint on coefficients, as illustrated in Equation 2.7. We optimized the value of the regularization parameter,λ, using the L-curve method [68]. L-curve is a plot of the size of regularized solution versus the residual. Classification results using this approach has been reported on a small dataset in [69].

For the least squares method, we estimate the representation coefficients, $\hat{\zeta}$, using Equation 2.5. Representation coefficients for the least squares method, the ℓ_2-norm method, and the ℓ_1-norm method have been achieved using mentioned methods and representation coefficients for one of the spines are presented in Figure 4.4.

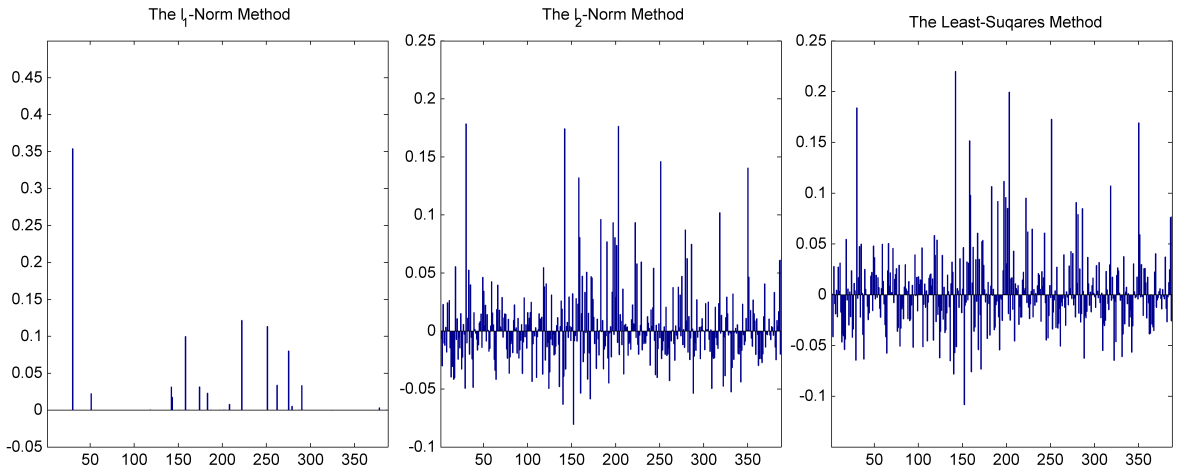


Figure 4.4: Linear representation coefficients using different representation algorithms.

4.6 Manifold Learning Methods Based Approach

In this study, we use several manifold learning techniques for spine classification and compare their performance. Manifold learning approaches (listed below) applied in this study to reduce dimensionality of data are being discussed briefly in this section.

- Principal Component Analysis (PCA) [38]
- Multi-Dimensional Scaling (MDS) [39]
- Locally Linear Embedding (LLE) [40]
- Laplacian eigenmap [41]
- Local Tangent Space Alignment [42]
- ISOMAP [14]

We start with automatically segmented data (segmented using DNSM) and construct our feature vector by stacking and concatenating the columns of images. Firstly, the segmented spine images were used to construct $62,500$ dimensional feature vectors by concatenating the stacked columns of each spine image. These feature vectors were further used to construct the feature matrix. Finally, manifold learning algorithms were applied on this feature matrix to produce 25-dimensional feature vectors. Classification results using this approach has been reported on a small dataset in [70].

4.6.1 ISOMAP-Space Analysis

ISOMAP [14] is known to compute distinctive features from a dataset. Samples from two-dimensional ISOMAP feature space are presented in Figure 4.5. Visual analysis of feature space results in interesting observation, the head diameter of spines varies along the vertical axis and the neck length along the horizontal axis. This validates our claim [58] that head diameter and neck length are the most important features for the classification of spines. This leads to an important finding that ISOMAP implicitly computes degrees of freedom of a dataset, in this case it is 2. A similar analysis has been previously performed on faces and digits dataset [14].

4.7 Deep Learning Based Approach

We use AlexNet [49], which is a well-known CNNs architecture, and apply transfer learning to use a network trained on natural images as a feature extractor. Further, we also fine-tune this network on our dendritic spines dataset. We use Caffe [71] library for feature extraction, training, and testing of CNNs. To the best of our knowledge transfer learning or any of the deep learning methods have not been applied for dendritic spine classification. Therefore, it is an interesting experiment to apply transfer learning with deep learning methods and compare their performance with state-of-the-art morphological feature based methods and other methods proposed in this thesis.

In order to extract spines, we select a region of interest (ROI) in intensity images such that the spine is completely inside the ROI, this does not require ROI for all spines to have same dimensions. Further, we rotate the ROI in the sense that spine necks are vertically aligned. This results in a dataset having arbitrary dimensions, however, AlexNet [49] (CNNs architecture that we use for this study) requires fixed size input images. Therefore, we scaled the shorter dimension of the ROI to 256 and then cropped a 256×256 patch from center of the ROI. A few samples from this scaled and center cropped dataset are shown in Figure 4.6.

We apply transfer learning for dendritic spine classification and compare performance of feature extraction and fine-tuning approaches. We use AlexNet that has been

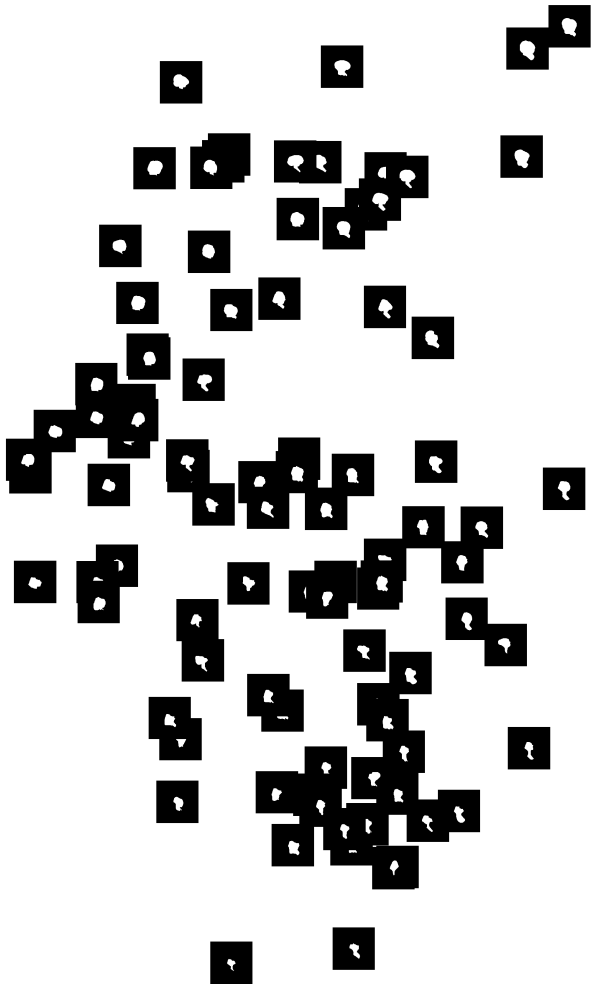


Figure 4.5: ISOMAP $2D$ features: Spine head diameter varies along y-axis and neck length changes along x-axis. DNSM segmentation results of some of the spine from our dataset are shown.

trained on ImageNet dataset with $1,000$ categories. In order to use it as a feature extractor we cut last one ($fc8$) or two fully connected layers ($fc7$ and $fc8$) and compute $4,096$-dimensional feature vectors on dendritic spines dataset from 7^{th} layer ($fc7$) or 6^{th} layer ($fc6$) since CNNs tend to learn generic features in their initial layers and task specific features in last layers [48].

Other method to apply transfer learning with CNNs is to fine-tune a trained network. We use AlexNet trained on ImageNet with $1,000$ categories as initialization, modify $fc8$ by changing its number of outputs to 3 (since we have a 3 class problem here). Hence, $fc8$ would start training from random weights. Since AlexNet is already trained

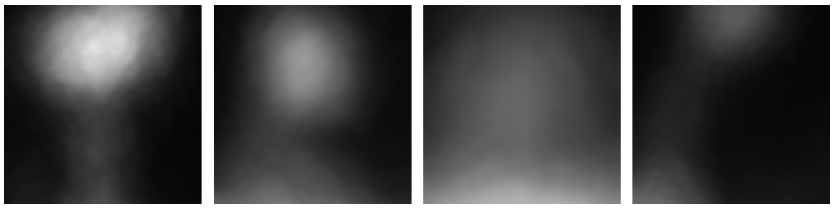


Figure 4.6: A few images from scaled and center cropped dataset. First and 2^{nd} spines are as Mushroom, 3^{rd} as Stubby, and 4^{th} as Thin (from left to right).

on a large dataset, we reduce the overall network learning rate to 0.00001 to let the network weights change slowly. Another approach is to freeze the weights of initial layers considering the generic nature of their learned features and only train the fully connected layers on new task. We use both strategies: fine-tune all layers (FT $fc1-fc7$ AlexNet) and freeze first 2 convolutional layers (FT $fc3-fc7$ AlexNet) and train the network for $3,000$ iterations. After fine-tuning the AlexNet with both configurations, we extract features from $fc7$. We also use classification results produced by final layer of fine-tuned AlexNet. In order to evaluate the performance of proposed approach, we use 10-fold cross validation approach, i.e. we fine-tuned 10 networks and used those to extract features and perform classification for corresponding test fold.

4.8 Classification Results and Discussion

We extract features as we discuss in respective sections and perform classification. In order to evaluate the classification results, we use labels assigned by a human expert. We compare classification results of our proposed approaches to three morphological feature based approaches: Koh et al. [8], NeuronStudio [22], and 2dSpAn [31]. NeuronStudio uses an automated spine detection algorithm; their approach misses 83 of the spines from our complete dataset. 2dSpAn tool uses a global thresholding based approach for dendrite and spine segmentation, it fails to segment 197 of the spines from our complete dataset. For fair comparison, we present results on four datasets; DataA: complete dataset, DataB: which includes only the spines that are successfully detected by NeuronStudio spine detection algorithm, DataC: which includes only the spine that

Table 4.1: Ratio criteria for classification of dendritic spines.

L/d_n	d_h/d_n		
	[0,1.3)	[1.3,3)	[3,∞)
[0,2/3)	Stubby	Mushroom	Mushroom
[2/3,2)	Stubby	Stubby	Stubby
[2,3)	Stubby	Mushroom	Mushroom
[3,5)	Thin	Mushroom	Mushroom
[5,∞)	Thin	Thin	Thin

are segmented by 2dSpAn segmentation approach, and DataD: that consists of only the spines which are correctly segmented by Morph3D approach. DataA consists of 456 spines including 288 mushroom, 113 stubby and 55 thin type spines. DataB consists of 373 spines including 251 mushroom, 96 stubby, and 26 thin type spines. DataC consists of 259 spines including 182 mushroom, 71 stubby, and 6 thin spines. DataD consists of 428 spines including 290 mushroom, 84 stubby, and 54 thin spines.

We use features proposed by Koh et al. [8] to perform classification using ratio criteria that they used in their paper. Ratio criteria used by Koh et al. [8] was originally itroduced by Harris et. al. [34], it is given in Table 4.1. Here, L depicts spine length, d_n represents spine neck diameter, and d_h represents head diameter. However, these criterion seem to be very specialized for specific data, this is why their performance is very poor on our dataset. In order to test the potential of their proposed features, we trained KDE, SVM, and NN, and performed classification using same settings used for other methods. This performance is closer to other morphological features based approaches, which means that ratio criterion based classification approach is very specialized to some datasets and its use must be discouraged.

In order to perform comparison to NeuronStudio [22], we downloaded the latest version of NeuronStudio[2] and computed classification results. After loading and setting dataset parameters, we applied median filtering to reduce noise. We used semi-

[2]http://research.mssm.edu/cnic/tools-ns.html

automatic dendrite tracing by providing manual seed. After dendrite tracing, we used NeuronStudio built-in spine detection and classification tools to compute classification results on DatasetB.

2dSpAn [31] is a recently developed tool for spine classification. Initially, users clicks at two points in the image to select ROI. It uses global thresholding to perform segmentation.Threshold can be manually adjusted, however, due to the challenging nature of 2PLSM data it is less probable to segment all the spines using global threshold. Therefore, we select the threshold manually so that the maximum number of spines has been segmented. Further, spine detection is performed manually; user clicks on each spine, the tool computes the morphological features and perform classification using rule based approach.

We compare the performance of our approaches using three classifiers: KDE, support vector machines (SVM) and Neural Networks (NN). We used the linear kernel for SVM and 2-layer network with $\frac{\textbf{No. of features} + \textbf{No. of Classes}}{2}$ nodes in each layer for Neural Network. We compare feature extraction schemes for all three classifiers using 10 fold cross validation; results are presented in Table 4.2.

As we can see from Table 4.2, the AlexNet, and shape and appearance features based method outperform all other approaches considered in this thesis. It is also clear from the presented results that most of our proposed approaches outperform commonly used morphological features based methods [8, 22, 31]. It confirms the robustness and reliability of deep learning methods based features, it performs better than several hand-crafted features, some of these were even specially designed for dendritic spine classification (i.e. morphological features). For pre-trained AlexNet, best performance is achieved with features extracted from $fc6$ and selected with CFS based method; it confirms the observation reported in [72, 73] that $fc6$ features perform better than $fc7$ features. For fine-tuned AlexNet, best performance is achieved with AlexNet final layer classifier when first 2 convolutional layers were frozen during fine-tuning. It is an interesting observation since generally fine-tuning a network improves classification performance as compared to pre-trained network [48]. However, it is also important to note that the distance between ImageNet dataset and dendritic spines is quite high,

Table 4.2: Classification Results, comparison of feature extraction and classification approaches.

Classifier	**Features**	**DataA**	**DataB**	**DataC**	**DataD**
KDE	Koh et al. [8]	54.17%	54.16%	52.12%	54.21%
	Morph2D	61.18%	61.39%	64.87%	60.51%
	Morph3D	-	-	-	77.10%
	Morph3D+CFS	-	-	-	80.37%
	Morph3D+IG	-	-	-	77.34%
	ISOMAP	68.64%	70.51%	74.90%	69.86%
	Laplacian	68.86%	73.99%	73.75%	68.46%
	LLE	71.05%	73.19%	69.88%	71.26%
	LTSA	50.22%	56.84%	57.14%	55.38%
	MDS	68.86%	73.19%	74.90%	69.16%
	PCA	69.96%	71.58%	76.06%	69.39%
	DNSM	75.60%	75.34%	75.29%	73.13%
	HOG	83.11%	84.45%	86.87%	81.08%
	DNSM+HOG	82.02%	83.65%	84.94%	79.44%
	DNSM+HOG+CFS	84.87%	85.26%	87.26%	81.08%
	DNSM+HOG+IG	83.33%	**85.79%**	85.71%	81.78%
	Pre-trained AlexNet	83.77%	80.70%	84.94%	82.25%
	FT $fc1$-$fc7$ AlexNet	**85.53%**	82.84%	**87.65%**	82.24%
	FT $fc3$-$fc7$ AlexNet	84.43%	80.96%	85.33%	**82.48%**
SVM	Koh et al. [8]	69.52%	72.92%	76.06%	71.03%
	Morph2D	69.30%	75.60%	74.13%	72.20%
	Morph3D	-	-	-	81.08%
	Morph3D+CFS	-	-	-	80.84%
	Morph3D+IG	-	-	-	**83.88%**
	ISOMAP	78.29%	80.16%	77.99%	75.70%
	Laplacian	76.32%	78.02%	81.08%	75.47%

Table 4.2 – continued from previous page

Classifier	Features	DataA	DataB	DataC	DataD
	LLE	74.56%	78.28%	78.38%	77.80%
	LTSA	63.16%	67.29%	67.95%	67.76%
	MDS	78.51%	78.89%	81.47%	78.97%
	PCA	75.66%	78.02%	79.54%	77.10%
	DNSM	74.12%	78.82%	74.90%	72.90%
	HOG	82.46%	82.04%	85.71%	81.08%
	DNSM+HOG	81.36%	82.57%	86.10%	80.61%
	DNSM+HOG+CFS	80.48%	81.50%	84.94%	78.27%
	DNSM+HOG+IG	79.83%	81.77%	85.33%	78.51%
	Pre-trained AlexNet	84.87%	83.91%	86.49%	82.25%
	FT $fc1$-$fc7$ AlexNet	**87.50%**	**86.86%**	**89.19%**	83.65%
	FT $fc3$-$fc7$ AlexNet	87.06%	83.11%	83.01%	82.01%
NN	Koh et al. [8]	67.54%	70.78%	72.97%	67.76%
	Morph2D	69.96%	71.31%	74.90%	68.46%
	Morph3D	-	-	-	83.88%
	Morph3D+CFS	-	-	-	84.81%
	Morph3D+IG	-	-	-	83.88%
	ISOMAP	77.63%	80.16%	80.69%	76.64%
	Laplacian	76.32%	80.97%	82.24%	76.87%
	LLE	77.85%	79.29%	81.08%	76.87%
	LTSA	58.11%	63.00%	62.16%	64.95%
	MDS	76.75%	79.36%	80.70%	75.23%
	PCA	78.95%	79.09%	83.78%	76.40%
	DNSM	80.70%	82.04%	84.94%	79.91%
	HOG	84.21%	84.72%	87.26%	83.41%
	DNSM+HOG	85.53%	84.97%	85.33%	83.18%
	DNSM+HOG+CFS	83.55%	86.33%	86.10%	84.35%

Table 4.2 – continued from previous page

Classifier	Features	DataA	DataB	DataC	DataD
	DNSM+HOG+IG	87.06%	**87.13%**	86.87%	83.65%
	Pre-trained AlexNet	**87.72%**	85.52%	**89.58%**	**86.22%**
	FT $fc1$-$fc7$ AlexNet	86.84%	85.19%	87.26%	82.94%
	FT $fc3$-$fc7$ AlexNet	86.40%	83.65%	89.19%	84.11%
SRC	Least-squares method	68.64%	77.21%	78.76%	70.09%
	The ℓ_2-norm method	69.74%	75.34%	77.99%	70.09%
	The ℓ_1-norm method	75.44%	80.43%	83.78%	75.23%
Ratio Criteria	Koh et al. [8]	25.66%	26.27%	19.04%	27.08%
AlexNet	FT $fc1$-$fc7$ AlexNet	87.50%	86.33%	**89.58%**	**86.45%**
	FT $fc3$-$fc7$ AlexNet	**87.72%**	86.60%	**89.58%**	**86.45%**
NeuronStudio [22]		-	60.86%	-	-
2dSpAn [31]		-	-	39.38%	-

also we have a small dataset consisting of only 456 samples.

Among hand-crafted features, our shape and appearance features based method performs best. It is also clear from presented results that HOG based appearance features perform slightly better than DNSM features. Combining DNSM and HOG performs slightly better in some cases. Further applying feature selection on DNSM+HOG results in higher accuracy in some cases. Another conclusion to draw from Table 4.2 is that, shape and appearance features combined with Neural Network outperforms other hand-crafted feature based methods. Given the statistical description in terms of the shape and appearance features, our KDE based approach works using the likelihood ratios, which is the sufficient statistic for the classification problem.

Among morphological features, our $3D$ morphological features (Morph3D) based approach performs best. It outperforms other morphological features based approaches with a great margin. It is important to note that NeuronStudio [22] exploit $3D$ infor-

mation while computing features, whereas Morph2D, Koh et. al. [8], and Basu et. al. [31] use 2D images to compute morphological features. It concludes not only the added value of third dimension improves the performance of our Morph3D approach but also the distinctive nature of Morph3D feature set also improves classification performance.

Among linear representation based methods, the ℓ_1-norm based method performs best. We also visually analyzed the most dominant spines for ℓ_1-norm based representation, they look quite similar. If we perform classification using class label of most dominant spine, it results in 73.60% classification accuracy on DataA which is slightly worse than using SRC for classification. For ℓ_2-norm based method, we select regularization parameter, λ, using L-curve method. However, we can also select regularization parameter using GI method, as we used for the ℓ_1-norm based method, which aims to select sparser solution. Although using GI would not be appropriate to use with the ℓ_2-norm method since sparsity is not the objective here, however, if we use GI with ℓ_2-norm based method, it performs slightly better and classifies 72.37% of spines correctly on DataA.

Another conclusion that can be drawn from results is that SVM, KDE, and NN outperform rule-based classification approaches considered in [22, 8, 31]. The reason behind their failure is the over-specialized nature of these approaches. These rules have been designed on the datasets used in specific studies and are over-specialized for those datasets. We have shown that these rules perform much worse than advanced machine learning based classification algorithms.

It is evident from results presented in Table 4.2 that performance of different feature extraction approaches is dependent upon classifier. It makes sense because different classifiers use different decision criteria. They make an effort to impose a structure on the features; the feature representations produce a feature space, if that matches well with the structure imposed by the classification algorithm it results in good performance. Caruana and Niculescu-Mizil perform an empirical evaluation of different classification approaches and conclude that even the best classifier sometimes result in poor performance and vice versa [74]. LeCun et al. [75] also perform comparison of various classifiers and argue that error rate is a function of the number of training examples

available and this function is different for each classification algorithm. They also claim that many algorithms can achieve a reasonable performance if sufficient training data are available. In this context, we observe different classification performance combining each feature extraction approach with different classifiers, however, we have a small dataset (i.e., 456 spines), which makes it difficult to draw any conclusion about which classifier is best. Our dataset has imbalanced numbers of samples for each class; KDE, SVM, and NN handle this problem reasonably well, however, SRC algorithm fails to do so and mis-classifies all the thin spines. Overall, NN performs slightly better than SVM, which performs slightly better than KDE. However, the difference in performance is not significant given the small size of our dataset. To summarize, we test the classifiability of dendritic spine using combination of different feature extraction and classification algorithms, and conclude that reasonable classification results can be obtained using these methods.

4.9 Likelihood Ratio Space Analysis

As discussed earlier, whether to view spines as belonging to distinct shape classes or to model them through a continuum of shape variations is still an open question. Since our KDE based classification approach gives the likelihood of a spine being member of Mushroom (l_m), Stubby (l_s), and Thin (l_t) classes, it can be used to examine this question in a principled manner. We computed the histogram of likelihood ratios ($\mathcal{L}_s$ and $\mathcal{L}_t$), as given in Figure 4.7 and analyzed whether we see three distinct modes or a continuum of shapes. It is evident from the presented histogram that we see three peaks, however the samples of three classes can be found all over the grid. One might argue that there are three classes, mushroom, stubby, and thin, but there is a significant overlap between their distributions. For analyzing the statistical significance of this framework, we performed Multi-variate Analysis of Variance (MANOVA) with null hypothesis to have same mean for all class distributions. Since we have two dependent variables, $\mathcal{L}_s$ and $\mathcal{L}_t$, and three classes, MANOVA is a fair choice for statistical significance analysis. We applied MANOVA on our data and found that it rejects the null hypothesis at

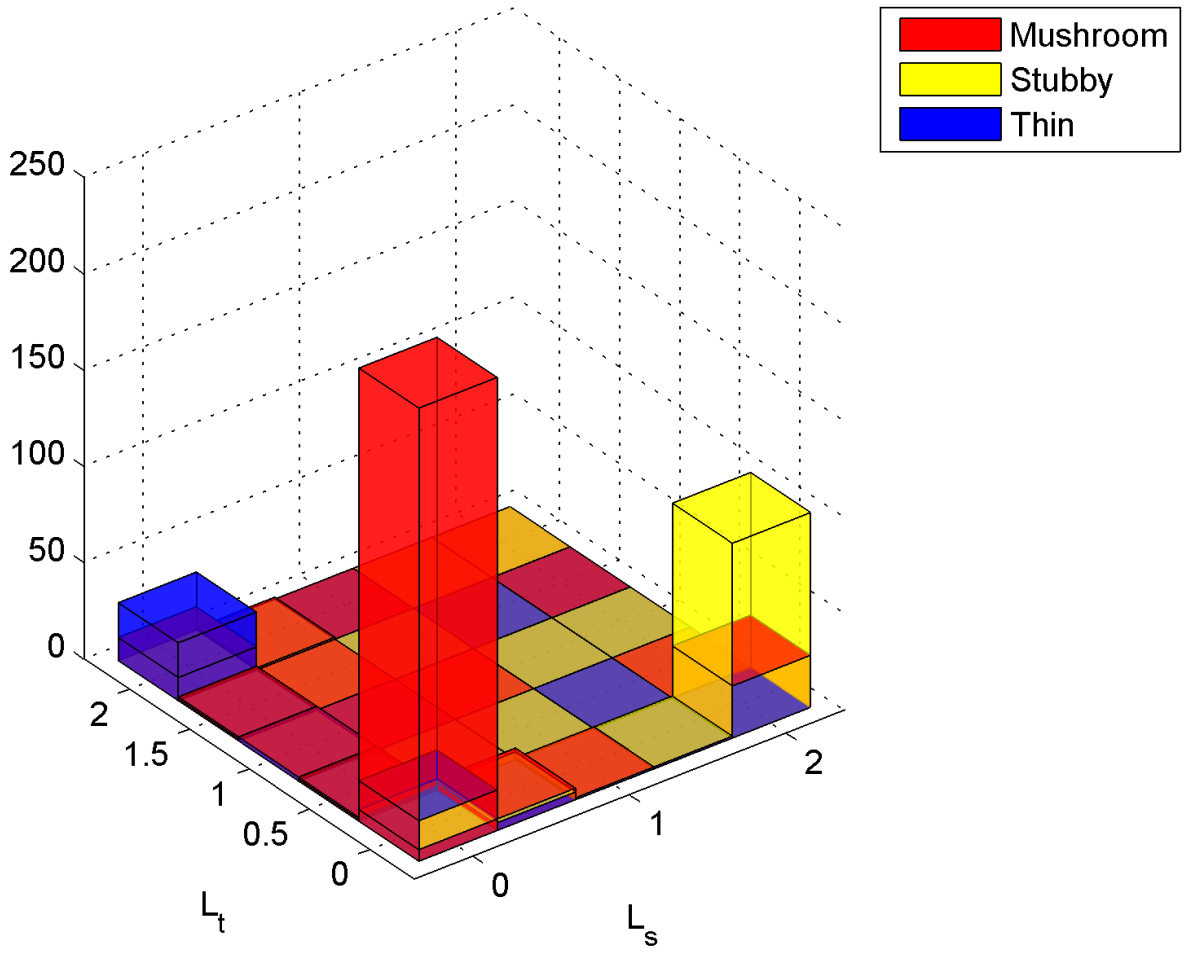


Figure 4.7: $2D$ likelihood ratio space produced using DNSM+HOG+InfoGain on DataA. We have added transparency in the histogram to make the visualization better. We can see three peaks; however, the samples of each shape are distributed all over. With aid of transparency, we can see different shape samples spread over the grid produced as a mixture of different colors such as red and yellow, yellow and blue, etc.

Table 4.3: Two-sample Two-dimensional Kolmogorov-Smirnov Test results for different class separation problems. we use null hypothesis to have same mean for both distributions, that is rejected in all cases, which supports the existence of distinct shape classes.

Test	p-value	Rejected/Accepted
Mushroom vs. Stubby	1.49×10^{-43}	Rejected
Mushroom vs. Thin	3.05×10^{-19}	Rejected
Stubby vs. Thin	1.54×10^{-25}	Rejected

$\alpha = 0.05$ significance level. MANOVA rejects the existence of continuum of shape variations. Further, we apply post-hoc tests to analyze the individual class separability using two-sample bi-variate Kolmogorov-Smirnov test[76] with null hypothesis being same mean for both distributions. Here, we perform three tests to analyze mushroom vs. stubby, mushroom vs. thin, and stubby vs. thin class separations. The test results are given in Table 4.3. It rejects the null hypothesis in all cases that supports the argument of existence of distinct classes. It provides us an insight that class distributions have different mean, meaning that there exist three distinct classes in our dataset. Such low p-value strongly support the significance of our analysis.

If we treat this problem as a classification task, best performance (assuming equal priors and the probability of error as the decision criterion) can be achieved by thresholding 2D likelihood ratio space as illustrated in Figure 4.1 to perform classification. However, classifying the spines lying around these decision boundaries is a difficult decision since values of $\mathcal{L}_s$ and $\mathcal{L}_t$ are very close in these case. Our KDE based framework provides a principled approach to handle such spines, if values of $\mathcal{L}_s$ and $\mathcal{L}_t$ are not very different, one might use the help of neuroscientists to investigate the spines and make a decision manually. Further analysis may also include 3D image evaluation of the spines whose likelihoods for different classes are very close.

Chapter 5

Cluster Analysis

Manual analysis is a laborious, time-intensive, and most importantly subjective task. Rodriguez et al. [22] reported inter-operator and intra-operator variations in the spine type labeling task. One might question why perform clustering rather than treating this as a classification problem. First of all, classification methods use manually provided labels as ground truth and extracting those labels is a time-intensive task. It also introduces subjectivity, which could be reduced by employing several experts and using a majority vote approach but this would make the labeling effort even more time-intensive. Inter-operator and intra-operator variability reported by Rodriguez et al. [22] emphasizes that subjectivity is a major issue in performing classification. Another issue with supervised classification is that it inherently starts from a pre-defined set of classes and does not allow exploration of potential intermediate shapes or possible continuous variation of shapes. Although clustering does not explicitly enable the latter either, it can be viewed as a step in that direction. Furthermore, some existing techniques require manual annotation of spines either to directly use them for feature extraction or for training segmentation algorithms. The objective of clustering in this context is two-fold: confirm the hypothesis of some distinct shape classes and discover new natural groups. We discover natural groups in the data using different features and analyze whether they support the existing hypotheses or add new information to our understanding of spine shapes.

As suggested by Mancuso et al. [33], we present a clustering-based approach for

spine shape analysis. We perform cluster analysis using several feature representations and gain insights by performing analysis of discovered natural groups. We use Disjunctive Normal Shape Models (DNSM) [1], Histogram of Oriented Gradients (HOG) [12], intensity profiles [35], and morphological features [58].

Jain [77] suggests there are two objectives for clustering: (i) exploratory: when there is no existing hypothesis or model, the aim is to discover patterns, and (ii) confirmatory: when a pre-specified model or hypothesis exists, the objective of cluster analysis is to confirm the model on the dataset being used. For dendritic spine analysis, the literature provides a pre-specified model as described in the introduction section. The nature of our analysis is: (i) an attempt to analyze how well a pre-specified model fits our data, (ii) if such a model does not fit our data, discover and explore natural groups within the data.

5.1 Feature Selection

Considering the high-dimensionality of feature representations being used (except morphological features), we apply a feature similarity based unsupervised feature selection algorithm [78]. Mitra et al. [78] introduced the maximum information compression index, which attempts to minimize the information loss while selecting a certain number of features. In this context, correlation coefficient is a well-known parameter to measure similarity between two random variables. Mitra et al. [78] argues that correlation coefficient have several nice properties, however, sensitivity with respect to rotation and invariance with respect to translation and scaling of random variables make it unsuitable for feature selection. Least square regression error is another important parameter that measures the amount of linear dependency between two random variables. However, this measure is not symmetric and is sensitive with respect to rotation [78].

Mitra et al. [78] introduced the maximal information compression index (λ_2), which is a linear dependency measure, however, it possess several advantageous properties which make it favorable for feature selection problem. Assuming Σ denotes covariance of two random variables x and y, λ_2 represents the smallest eigen value of Σ, as illustrated

in Equation 5.1. Its value is zero when x and y are linearly dependent and increases as the dependency between the two decreases. It is important to note that λ_2 is eigen value for the direction perpendicular to principal components direction of x and y and maximum information is compressed when data is projected along principal components direction[79]. In this case, information loss or reconstruction error is equivalent to λ_2. Therefore, this algorithm allows maximal information compression or minimum information loss while selecting a number of features. Here, the aim of feature selection is to aid the clustering algorithm, we select 100 features for each feature representation (except morphological features) and use these selected features to perform clustering.

$$2\lambda_2(x,y) = \text{var}(x) + \text{var}(y) - \sqrt{(\text{var}(x) + \text{var}(y))^2 - 4\text{var}(x)\text{var}(y)(1 - \rho(x,y)^2)} \quad (5.1)$$

5.2 X-means

Jain [77] argues that there is no best clustering algorithm, because every clustering technique implicitly or explicitly imposes a structure on the data, and it gives good results if there is a good match. Jain further emphasizes that it is rather crucial to select the appropriate representation that implicitly or explicitly makes the pattern discovery an easy process. Considering the clustering analysis problem as a selection of appropriate representation rather than selection of a clustering method, we have compared different feature representations in terms of clustering results.

We applied x-means [80], an extended version of k-means, which does not require the number of clusters to be provided. It uses Bayesian Information Criterion (BIC) to automatically select the number of clusters in the available data from a given range of number of clusters, which we set as 2 to 10. It begins with lower bound of given range for number of clusters and continues computing clusters until upper bound for number of clusters have been reached; during this process it also computes BIC score for each cluster assignment. Finally, it selects the number of clusters based on best BIC score.

5.3 Clustering Results

Our dataset consists of 242 dendritic spines selected from 15 dendritic branches for this analysis. These are spines that have been labeled as mushroom or stubby by a human expert. We computed HOG-based appearance, and intensity profile based features from intensity images; DNSM-based shape, and morphological features from automatically segmented spine images. In all cases, we applied x-means to perform cluster analysis that uses BIC to select the number of clusters. Analysis of clusters formed using different feature representations is presented in this section.

5.3.1 HOG Features Based Analysis

Using HOG based appearance feature representation for x-means clustering resulted in 4 clusters. The average image for each cluster is computed by averaging manually segmented binary images in that cluster. The resulting images are shown in Fig. 5.1. There are 49 spines in cluster 1, 93 spines in cluster 2, 72 spines in cluster 3, and 28 spines in cluster 4. As it is evident from the average images, cluster 2 and cluster 3 represent mushroom spines (long neck and big head). However, clusters 1 and 4 appear to consist either of spines from both classes or of spines that may possibly lie in between these two classes in the shape space. When we examine individual samples from these clusters, illustrated in Fig. 5.2, we observe that they exhibit similar characteristics, i.e., have small heads and no necks. However, closer analysis of intensity images shows existence of short necks, i.e., low intensity regions just below the head part. These observations support the produced clusters in the sense that although there are some spines which are easy to be classified (grouped in clusters 2 and 3), even a human expert would have difficult time providing labels for most of the spines in cluster 1 and cluster 4. This analysis also points to what one might call two subclasses (cluster 2 and cluster 3) within the mushroom class.

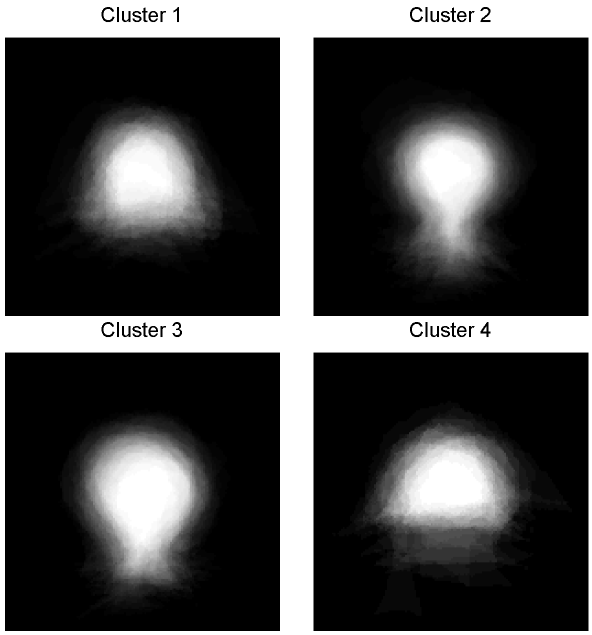
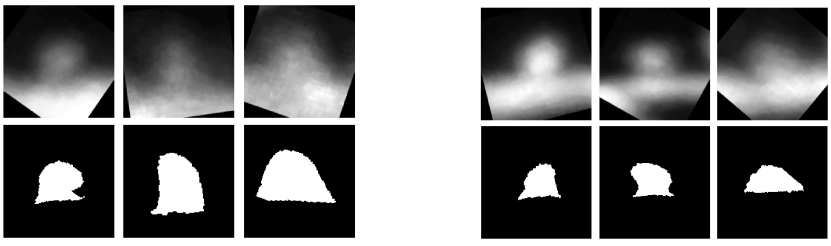


Figure 5.1: Average image for each cluster generated using the HOG features.

5.3.2 DNSM Features Based Analysis

We computed shape features using DNSM and performed clustering on this representation. The algorithm produced 4 clusters consisting of 32, 48, 50, and 112 spines. Average images of these clusters are given in Fig. 5.3. It can be observed from the formed clusters and their average images that the neck length of the spines is increasing in the following order: cluster 1, cluster 3, cluster 4, cluster 2.

Most of the spines in cluster 1 have short or no necks; their head diameter to neck



(a) Cluster 1 (b) Cluster 4

Figure 5.2: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines grouped in cluster 1 and cluster 4 using the HOG features.

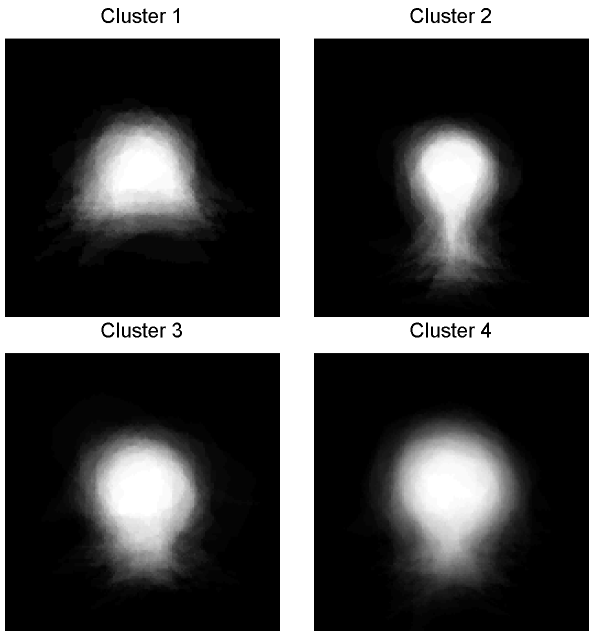


Figure 5.3: Average image for each cluster generated using the DNSM features.

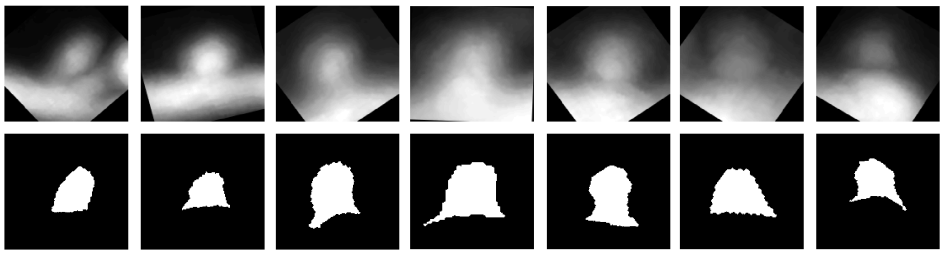


Figure 5.4: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines grouped in cluster 1 using the DNSM representation.

diameter ratio is approximately 1. A few spines from cluster 1 are presented in Fig. 5.4. This cluster appears to contain spines that clearly exhibit the characteristics of stubby spines as well as spines with distinct heads and thick necks. Cluster 2, cluster 3, and cluster 4 are mostly mushroom clusters. Again, cluster 2, cluster 3, and cluster 4 possibly represent different subclasses within mushroom type spines.

5.3.3 Morphological Features Based Analysis

Clustering analysis with morphological features resulted in 4 clusters with sizes: 102, 64, 64, and 12 spines. Average image for each of the produced clusters is given in

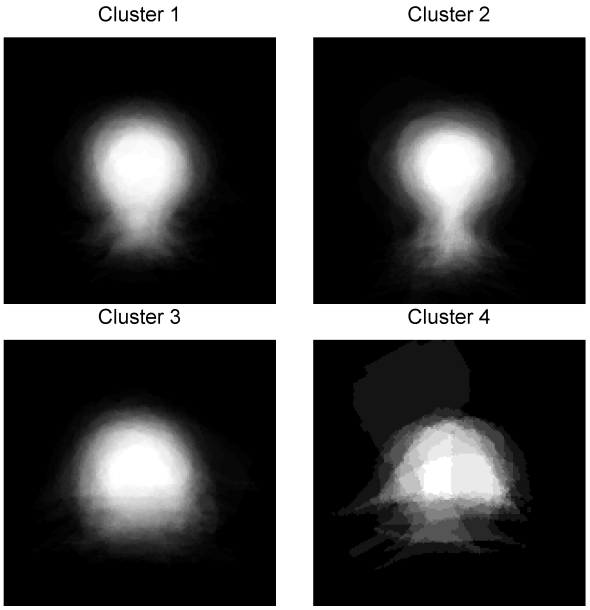
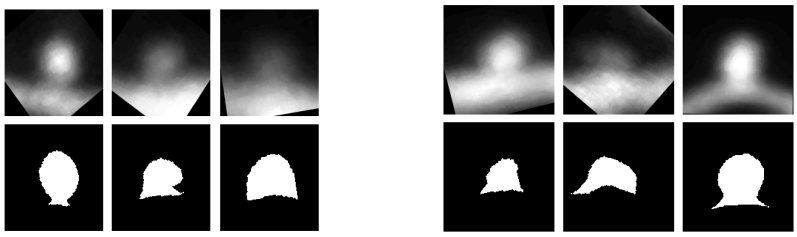


Figure 5.5: Average image for each cluster generated using morphological features.

Fig. 5.5. Examining average images, one can conclude that the neck length of spines is increasing in the following order: cluster 3, cluster 4, cluster 1, cluster 2.

It is clear from Fig. 5.5 that cluster 1, and 2 are mushroom majority clusters. Most of these spines can be easily decided to be mushroom based on their morphological properties, i.e., long necks, and big heads. However, cluster 3 and cluster 4 show a mixed pattern, most of the spines have short thick neck, small head, and most importantly their neck diameters and head diameters are similar. A few spines from cluster 3 and cluster 4 along with their manually annotated images are presented in Fig. 5.6. These cluster appear to contain many stubby spines as well as spines with distinct heads and thick necks. This pattern has been observed in clusters produced by other representations as well. It would be interesting to analyze which features are dominant in the clustering process, which might provide important information to neuroscientists. In this context, we perform an initial analysis using information gain [61] and conclude that neck length is the most dominant feature for data used in this study, which confirms our ISOMAP based analysis.

This provides us an insight that although some spines can easily be classified into standard shape groups, there are many spines which are either difficult to classify or



(a) Cluster 3 (b) Cluster 4

Figure 5.6: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines from cluster 3 and cluster 4 using the morphology based features.

lie somewhere in between these groups.

5.3.4 Intensity Profile Features Based Analysis

Using the intensity profile based features resulted in 4 clusters consisting of 45, 81, 48, and 68 spines. The average image for each of these clusters is presented in Fig. 5.7. It is clear that cluster 1, cluster 2, and cluster 3 are similar and appear to consist mostly of mushroom-like spines, i.e., they have big heads and long necks. Spines in cluster 3 have relatively shorter necks as compared to cluster 2, spines in cluster 4 have big heads and very short or no necks.

Some of the spines clustered in cluster 4 are shown in Fig. 5.8. First observation is that spines in this cluster have similar morphological characteristics. This cluster appears to contain many stubby spines as well as spines with distinct heads and thick necks.

5.3.5 Combined Features Based Analysis

Since, shape and appearance are complementary features, it is intuitive to combine both types of features and perform cluster analysis. We have already selected 100 features from each group using a feature similarity based approach. We combine these selected features to perform clustering in this section.

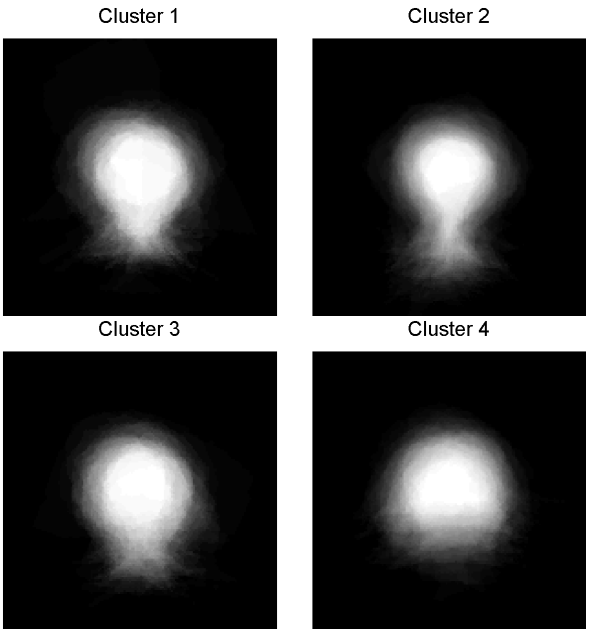


Figure 5.7: Average image for each cluster generated using the intensity profile based features.

HOG and DNSM Features

Using a combination of HOG and DNSM based features results in 4 clusters consisting of 30, 78, 22, and 112 spines. The average image for each of these clusters is presented in Fig. 5.9. It is clear that cluster 2 and cluster 4 are similar and consist most of the mushroom-like spines, i.e., they have big heads and long necks. Spines in cluster 1 and cluster 3 are similar to one another in the sense that they have big heads and very short or no necks, as illustrated in Fig. 5.10. It may be difficult to assign one

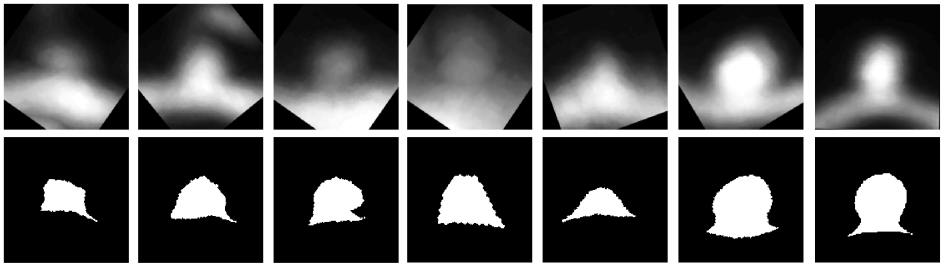


Figure 5.8: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines from cluster 4 generated using the intensity profile based features.

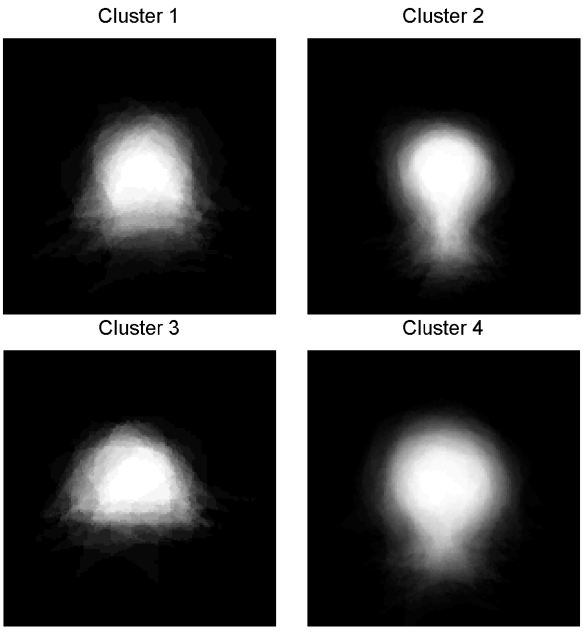
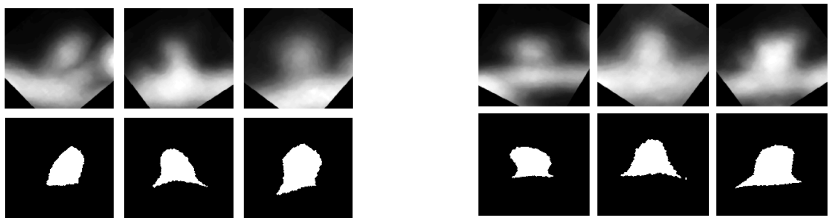


Figure 5.9: Average image for each cluster generated using HOG+DNSM features.

of the known labels to the spines in these clusters, therefore, we might call these mixed or intermediate clusters.

DNSM and Intensity Profile Features

Using a combination of DNSM and intensity profile based features results in 4 clusters consisting of 32, 62, 36, and 112 spines. Average image for each cluster is presented in Fig. 5.11. Cluster 2, cluster 3, and cluster 4 consist of mostly mushroom-like spines, having big heads and long necks. However, cluster 1 consists of spines



(a) Cluster 1 (b) Cluster 3

Figure 5.10: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines from cluster 1 and cluster 3 using HOG+DNSM based features.

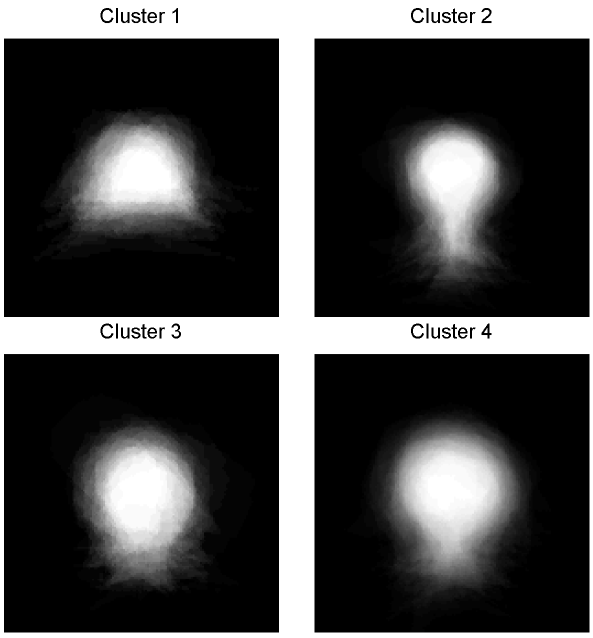


Figure 5.11: Average image for each cluster generated using DNSM+IntensityProfile features.

with intermediate properties: short, thick necks and big heads, as illustrated in Fig. 5.12. These spines have some morphological properties similar to mushroom spines and some similar to stubby spines, therefore, we may call cluster 1 a mixed or intermediate cluster.

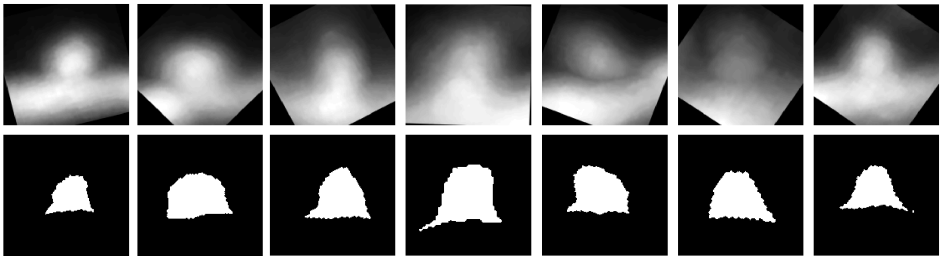


Figure 5.12: Intensity (top) and corresponding manually annotated images (bottom) for some of the spines from cluster 1 generated using DNSM+IntensityProfile features.

Table 5.1: Comparison of clustering results and labels from human expert

Features	Acc.	Class	Clusters			
			1	2	3	4
DNSM	79.34%	m	11	48	38	85
		s	21	0	12	27
Morphology	81.82%	m	88	64	26	4
		s	14	0	38	8
HOG	**88.02%**	m	15	91	68	8
		s	34	2	4	20
IntensityProfile	80.17%	m	39	81	34	28
		s	6	0	14	40
HOG+DNSM	79.34%	m	15	76	6	85
		s	15	2	16	27
DNSM+IntensityProfile	80.17%	m	10	62	25	85
		s	22	0	11	27

5.3.6 Clustering vs. Human Expert

In this section, we compare the clustering results achieved using different representations to the labels assigned by a neuroscience expert. The idea is that similar data samples (belonging to same class) should be clustered in the same group.

There are two challenges in spine shape analysis: (i) separating mushroom spines from stubby spines, and (ii) separating thin spines from filopodia type spines. Because of the developmental age of the animals we use, we see few filopodia in our data, this is why we focused on mushroom vs. stubby problem for this study. Stubby vs. mushroom analysis is a challenging task due to 2PLSM resolution limits. In fact, in stimulated emission depletion (STED) microscopy images, many reported stubby spines look like mushroom spines [81].

A human expert manually labeled 242 spine images, 182 spines as mushroom and 60 as stubby. Table 5.1 shows the class membership of the spines in each of the clusters

formed using each feature type. We observe that some clusters are dominated by shapes from one class whereas other are mixed. We have already analyzed the similarity within each of these clusters in the previous sections, and observed the exploratory nature of our approach pointing to possibly intermediate shapes.

Given the availability of manual labels, let us now carry out an analysis on the confirmatory aspects of our approach. In particular, to evaluate how strongly each clustering approach based on a different feature set confirms the manual shape labels, let us evaluate our clustering results using the manual labels as ground truth. To this end, let us pretend our clustering methods assign each cluster to the shape class with the majority of samples in that cluster. Then we can count the number of "correct and incorrect classifications". Using this approach, we evaluate these feature representations and find out that HOG features perform best on the available data taking the human expert's labels as the ground truth, viewing this it as a classification problem we can achieve 88.02% classification accuracy. If we treat this problem as a supervise classification problem, we achieve 88.84% classification accuracy using HOG features with NN. Therefore, reasonable classification results can be obtained using our proposed clustering approach.

According to expert's labels, clusters 2, 3, and 4 formed with the DNSM representation correspond to the mushroom class, whereas cluster 1 is the stubby majority cluster. Sample images shown in Fig. 5.4 suggests that spines in cluster 1 have similar characteristics, however, the expert has labeled some of these spines as mushroom and others as stubby. This itself depicts the challenging nature of spine analysis and subjective nature of the manual labeling task. We have similar observations on clusters formed through the use of the other features. In particular, we observe both the confirmatory role of the clustering methods through the formation of clusters dominated by one of the classes as labeled by the human expert (e.g., HOG clusters 2 and 3), as well as the exploratory nature of clustering through the generation of clusters with mixed membership (e.g., HOG clusters 1 and 4). Our experimental analysis suggests that the possibility of intermediate shape types in addition to the conventional shape classes should be considered in spine shape analysis. One further step along this direction

could involve efforts to characterize the distribution of spines in a continuous shape space.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we propose several dendritic spine shape analysis approaches based on 2PLSM images. Mainly, we focus on classification and clustering approaches to perform spine shape analysis. We compare classification performance using KDE, SVM, and NN classifiers; deep learning approach performs as well as shape and appearance features based approach. These methods outperform all other spine analysis approaches considered in this thesis including three commonly used morphological features based approaches. Among morphological features based methods, our $3D$ morphological features based approach results in the best classification accuracy. Additionally, our KDE based framework allows to study the question of distinct shape classes vs. continuum of shape variations in a principled manner. We present histogram of likelihood ratios which provides an insight into the challenging nature of the spine shape analysis problem. The likelihood ratio space exhibits three peaks but samples of all classes are spread all over. Further, we apply MANOVA test to check the statistical significance of our analysis and it strongly supports the idea of distinct shape classes.

We also proposed a clustering approach in this thesis to perform spine shape analysis. The advantages of adopting a clustering approach for spine shape analysis are: such an approach would not suffer from subjectivity, and analysis time would be reduced by avoiding manual labeling tasks. To the best of our knowledge an extensive clustering

analysis of spine shapes has not been reported in the literature. We use appearance, shape, and morphological feature based representations to perform clustering and shed some light on this problem. We perform clustering using x-means that uses BIC to select the number of clusters automatically; interestingly it produces 4 clusters for all of the features considered here. Additionally, we have observed that, for the data used in our analysis, although there are many spines which easily fit into the definition of standard shape types (confirming the hypothesis), there are also a significant number of others which do not comply with standard shape types and demonstrate intermediate properties. Existence of intermediate shape types has been observed using all representations. To conclude, the clustering perspective we propose in this thesis can both be used to perform automated spine shape analysis to identify known shape classes as well as to help neuroscientists discover and explore unknown patterns in the shape space.

6.2 Future Work

This thesis provides a new direction to researchers working on development of automated spine analysis tools; we investigated the use of a shape feature extraction technique, DNSM, and an appearance feature extraction technique, HOG. However, one could propose a new shape representation consisting of shape primitives that can compactly represent spine parts such as head and neck, which could seamlessly produce biologically meaningful features such as neck length. Furthermore, we analyzed the question of continuum of shape variations, which should be further investigated on more datasets using our framework which could potentially contribute to the resolution of this on-going debate. On the technical side, tools could be developed to study spine shapes in an unsupervised fashion, which would reduce subjective biases introduced due to supervised nature of classification based analysis. Another challenge is that sometimes human experts are not sure while assigning labels or have a difficult time during the label assignment process, one could provide a confidence level during the label assignment process, so if a classification framework makes a mistake for cases when even a human expert is unsure, it should have less cost. We perform a preliminary level

analysis of dendritic spines using a deep learning approach, it would be interesting to perform an in-depth analysis using deep learning approaches to understand the reason behind their success and biological meaning of learned features.

We provide an initial analysis on clustering that provides clustering perspective on spine analysis and compare it with expert labels, it would also be interesting to use proposed approach to perform an analysis tying clusters to different experimental conditions. It would also be interesting to perform a neuroscientific analysis of produced clusters and understand biological meaning of each cluster produced. Intermediate group of spines have been observed among cluster results produced with different feature representations. The emergence of this phenomenon can be explained in several ways. It is a known fact that dendritic spines exhibit shape type transitions over time, this phenomenon happens over the period of hours. If the spines are captured at these transition periods, for instance a mushroom spine changing to a stubby spine, it might happen to have a short and thick neck and a head diameter to neck diameter ratio close to 1. As some spines in our data demonstrate such properties, it would be difficult to label them as mushroom or stubby. A temporal analysis of several spine shapes would provide more insight into this phenomenon.

It should also be noted that based on the expert labels, the data we used for cluster analysis consists of two shape classes: mushroom and stubby. Including other shapes of spines such as thin and filopodia in the type of analysis we have proposed here might facilitate an even better understanding of the nature of shape classes and distribution. It might also be interesting to pose this as unsupervised regression problem and generate continuous (soft) outputs rather than discrete assignments to study continuous shape variations. It is important to mention that the distribution of spine shapes are dependent on various aspects of the data used, including which anatomical region of the brain the imaged neurons belong to as well as the age of the imaged neurons. This might also contribute towards different conclusions from different studies on spine shapes. Another potential issue might be performing the analysis on 2D projections versus 3D data. Therefore, it would be interesting to perform similar analysis with different 2D projection methods as well as 3D data.

Bibliography

[1] F. Mesadi, M. Cetin, and T. Tasdizen, "Disjunctive normal shape and appearance priors with applications to image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pp. 703–710, Springer, 2015.

[2] J. Lippman and A. Dunaevsky, "Dendritic spine morphogenesis and plasticity," *Journal of neurobiology*, vol. 64, no. 1, pp. 47–57, 2005.

[3] R. Yuste, *Dendritic spines*. MIT Press, 2010.

[4] R. Yuste and B. T., "Morphological changes in dendritic spines associated with long-term synaptic plasticity.," *Annu Rev Neurosci*, vol. 24, p. 1071–1089, 2001.

[5] M. Matsuzaki, N. Honkura, G. C. Ellis-Davies, and H. Kasai, "Structural basis of long-term potentiation in single dendritic spines," *Nature*, vol. 429, no. 6993, pp. 761–766, 2004.

[6] C. D. Harvey and K. Svoboda, "Locally dynamic synaptic learning rules in pyramidal neuron dendrites," *Nature*, vol. 450, no. 7173, pp. 1195–1200, 2007.

[7] A. Govindarajan, I. Israely, S.-Y. Huang, and S. Tonegawa, "The dendritic branch is the preferred integrative unit for protein synthesis-dependent ltp," *Neuron*, vol. 69, no. 1, pp. 132–146, 2011.

[8] I. Y. Koh, W. B. Lindquist, K. Zito, E. A. Nimchinsky, and K. Svoboda, "An image analysis algorithm for dendritic spines," *Neural computation*, vol. 14, no. 6, pp. 1283–1310, 2002.

[9] P. T. So, C. Y. Dong, B. R. Masters, and K. M. Berland, "Two-photon excitation fluorescence microscopy," *Annual review of biomedical engineering*, vol. 2, no. 1, pp. 399–429, 2000.

[10] A. Lipson, S. G. Lipson, and H. Lipson, *Optical physics*. Cambridge University Press, 2010.

[11] N. Ramesh, F. Mesadi, M. Cetin, and T. Tasdizen, "Disjunctive normal shape models," in *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*, pp. 1535–1539, April 2015.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.

[13] M. U. Ghani, F. Mesadi, S. D. Kanik, A. O. Argunşah, I. Israely, D. Unay, T. Tasdizen, and M. Cetin, "Dendritic spine shape analysis using disjunctive normal shape models," in *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*, IEEE, 2016.

[14] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, p. 2319, 2000.

[15] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, pp. 210–227, Feb. 2009.

[16] Q. Shi, A. Eriksson, A. van den Hengel, and C. Shen, "Is face recognition really a compressive sensing problem?," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 553–560, June 2011.

[17] J. Son, S. Song, S. Lee, S. Chang, and M. Kim, "Morphological change tracking of dendritic spines based on structural features," *Journal of microscopy*, vol. 241, no. 3, pp. 261–272, 2011.

[18] X. Xu and S. Wong, "Optical microscopic image processing of dendritic spines morphology," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 132–135, 2006.

[19] P. Shi, X. Zhou, Q. Li, M. Baron, M. A. Teylan, Y. Kim, and S. T. Wong, "Online three-dimensional dendritic spines mophological classification based on semi-supervised learning," in *ISBI'09 IEEE International Symposium on Biomedical Imaging: From Nano to Macro. (pp. 1019-1022)*, 2009.

[20] J. Spacek and M. Hartmann, "Three-dimensional analysis of dendritic spines. i. quantitative observations related to dendritic spine and synaptic morphology in cerebral and cerebellar cortices," *Anat. Embryol.*, vol. 167, p. 289–310, 1983.

[21] J. I. Arellano, R. Benavides-Piccione, J. DeFelipe, and R. Yuste, "Ultrastructure of dendritic spines: correlation between synaptic and spine morphologies," *Frontiers in neuroscience*, vol. 1, no. 1, 2007.

[22] A. Rodriguez, D. B. Ehlenberger, D. L. Dickstein, P. R. Hof, and S. L. Wearne, "Automated three-dimensional detection and shape classification of dendritic spines from fluorescence microscopy images," *PloS one*, vol. 3, no. 4, 2008.

[23] K. M. Harris, "Structure, development, and plasticity of dendritic spines," *Current opinion in neurobiology*, vol. 9, no. 3, pp. 343–348, 1999.

[24] B. Ruszczycki, Z. Szepesi, G. M. Wilczynski, M. Bijata, K. Kalita, L. Kaczmarek, and J. Wlodarczyk, "Sampling issues in quantitative analysis of dendritic spines morphology.," *BMC Bioinformatics*, vol. 13, p. 213, 2012.

[25] Z. Parnass, A. Tashiro, and R. Yuste, "Analysis of spine morphological plasticity in developing hippocampal pyramidal neurons," *Hippocampus*, vol. 10, no. 5, pp. 561–568, 2000.

[26] F. Chang and W. T. Greenough, "Transient and enduring morphological correlates of synaptic activity and efficacy change in the rat hippocampal slice," *Brain Res.*, vol. 309, p. 35–46, 1984.

[27] A. Peters and I. R. Kaiserman-Abramof, "The small pyramidal neuron of the rat cerebral cortex. the perikaryon, dendrites and spines," *Am. J. Anat.*, vol. 127, p. 321–356, 1970.

[28] J. Grutzendler, N. Kasthuri, and W.-B. Gan, "Long-term dendritic spine stability in the adult cortex," *Nature*, vol. 420, no. 6917, pp. 812–816, 2002.

[29] A. Matus, "Actin-based plasticity in dendritic spines," *Science*, vol. 290, no. 5492, pp. 754–758, 2000.

[30] J. Bourne and K. M. Harris, "Do thin spines learn to be mushroom spines that remember?," *Current opinion in neurobiology*, vol. 17, no. 3, pp. 381–386, 2007.

[31] S. Basu, D. Plewczynski, S. Saha, M. Roszkowska, M. Magnowska, E. Baczynska, and J. Wlodarczyk, "2dspan: semiautomated 2-d segmentation, classification and analysis of hippocampal dendritic spine plasticity," *Bioinformatics*, 2016.

[32] W. Wallace and M. F. Bear, "A morphological correlate of synaptic scaling in visual cortex," *Journal of Neuroscience*, vol. 24, no. 31, pp. 6928–6938, 2004.

[33] J. J. Mancuso, Y. Chen, X. Li, Z. Xue, and S. T. Wong, "Methods of dendritic spine detection: from golgi to high-resolution optical imaging," *Neuroscience*, vol. 251, pp. 129–140, 2013.

[34] K. M. Harris, F. E. Jensen, and B. Tsao, "Three-dimensional structure of dendritic spines and synapses in rat hippocampus (ca1) at postnatal day 15 and adult ages: implications for the maturation of synaptic physiology and long-term potentiation [published erratum appears in j neurosci 1992 aug; 12 (8): following table of contents]," *The Journal of neuroscience*, vol. 12, no. 7, pp. 2685–2705, 1992.

[35] E. Erdil, A. O. Argunsah, T. Tasdizen, D. Unay, and M. Cetin, "A joint classification and segmentation approach for dendritic spine segmentation in 2-photon microscopy images," in *Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*, pp. 797–800, IEEE, 2015.

[36] A. Ghodsi, "Dimensionality reduction: A short tutorial," technical report, Department of Statistics and Actuarial Science, University of Waterloo, 2006.

[37] I. Fodor, "A survey of dimension reduction techniques," tech. rep., Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2002.

[38] I. Jolliffe, *Principal Component Analysis*. Springer Verlag, 1986.

[39] T. F. Cox and M. Cox, *Multidimensional Scaling, Second Edition*. Chapman and Hall/CRC, 2 ed., 2000.

[40] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *SCIENCE*, vol. 290, pp. 2323–2326, 2000.

[41] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, pp. 1373–1396, 2002.

[42] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, pp. 313–338, Jan. 2005.

[43] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 3rd ed., 2008.

[44] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.

[45] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*, 2008.

[46] T. F. Chan and L. A. Vese, "Active contours without edges," *Image processing, IEEE transactions on*, vol. 10, no. 2, pp. 266–277, 2001.

[47] L. Torrey and J. Shavlik, "Transfer learning," *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, vol. 1, p. 242, 2009.

[48] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in neural information processing systems*, pp. 3320–3328, 2014.

[49] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[50] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255, IEEE, 2009.

[52] T. Atherton and D. Kerbyson, "Size invariant circle detection.," *Image and Vision Computing*, vol. 17, no. 11, pp. 795–803, 1999.

[53] Y. Xie and Q. Ji, "A new efficient ellipse detection method," in *16th International Conference on Pattern Recognition*, vol. 2, pp. 957–960, 2002.

[54] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[55] R. V. Uitert and I. Bitter, "Subvoxel precise skeletons of volumetric data based on fast marching methods," *Medical Physics*, vol. 34, no. 2, pp. 627–638, 2007.

[56] M. Hassouna and A. Farag, "Multistencils fast marching methods: A highly accurate solution to the eikonal equation on cartesian domains," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, pp. 1563–1574, Sept 2007.

[57] J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods*. New York, NY, USA: Wiley-Interscience, 1987.

[58] M. U. Ghani, S. D. Kanik, A. O. Argunsah, T. Tasdizen, D. Unay, and M. Cetin, "Dendritic spine shape classification from two-photon microscopy images," in *IEEE Signal Processing and Communications Applications (SIU)*, 2015.

[59] J. Wallis, T. R. Miller, C. Lerner, and E. Kleerup, "Three-dimensional display in nuclear medicine," *Medical Imaging, IEEE Transactions on*, vol. 8, no. 4, pp. 297–230, 1989.

[60] M. A. Hall, *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.

[61] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 1991.

[62] R. M. Gray, "Entropy and information," in *Entropy and Information Theory*, pp. 21–55, Springer, 1990.

[63] J. Kim, M. Cetin, and A. S. Willsky, "Nonparametric shape priors for active contour-based image segmentation.," *Signal Processing*, vol. 87, no. 12, pp. 3021–3044, 2007.

[64] L. Richard and J. Burden, "Douglas faires, numerical analysis," 1988.

[65] A. Y. Yang, S. S. Sastry, A. Ganesh, and Y. Ma, "Fast l1-minimization algorithms and an application in robust face recognition: a review," tech. rep., 2010.

[66] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale l1-regularized least squares," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, pp. 606–617, Dec 2007.

[67] N. P. Hurley and S. T. Rickard, "Comparing measures of sparsity," *CoRR*, vol. abs/0811.4706, 2008.

[68] P. C. Hansen and D. P. O'Leary, "The use of the l-curve in the regularization of discrete ill-posed problems," *SIAM Journal on Scientific Computing*, vol. 14, no. 6, pp. 1487–1503, 1993.

[69] M. U. Ghani, S. D. Kanik, A. O. Argunsah, I. Israely, D. Unay, and M. Cetin, "Dendritic spine classification based on two-photon microscopic images using sparse representation," in *IEEE Signal Processing and Communications Applications (SIU)*, 2016.

[70] M. U. Ghani, A. O. Argunsah, I. Israely, D. Unay, T. Tasdizen, and M. Cetin, "On comparison of manifold learning techniques for dendritic spine classification," in *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*, IEEE, 2016.

[71] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678, ACM, 2014.

[72] N. Codella, J. Cai, M. Abedini, R. Garnavi, A. Halpern, and J. R. Smith, "Deep learning, sparse coding, and svm for melanoma recognition in dermoscopy images," in *International Workshop on Machine Learning in Medical Imaging*, pp. 118–126, Springer, 2015.

[73] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition.," in *ICML*, pp. 647–655, 2014.

[74] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning*, pp. 161–168, ACM, 2006.

[75] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger, *et al.*, "Comparison of learning algorithms for

handwritten digit recognition," in *International conference on artificial neural networks*, vol. 60, pp. 53–60, 1995.

[76] J. Peacock, "Two-dimensional goodness-of-fit testing in astronomy," *Monthly Notices of the Royal Astronomical Society*, vol. 202, no. 3, pp. 615–627, 1983.

[77] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[78] P. Mitra, C. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 3, pp. 301–312, 2002.

[79] P. A. Devijver and J. Kittler, *Pattern recognition: A statistical approach*. Prentice hall, 1982.

[80] D. Pelleg, A. W. Moore, *et al.*, "X-means: Extending k-means with efficient estimation of the number of clusters.," in *ICML*, vol. 1, 2000.

[81] J. Tonnesen, G. Katona, J. Rózsa, U. Nagerl, *et al.*, "Spine neck plasticity regulates compartmentalization of synapses," *Nature neuroscience*, vol. 17, no. 5, pp. 678–685, 2014.