Statistical Methods for Polyhedral Shape Classification with Incomplete Data

Application to Cryo-electron Tomographic Images

A thesis submitted for the degree of Doctor of Philosophy

Sukantadev Bag

Department of Statistics
College of Science, Engineering and Food Science
National University of Ireland, Cork

Supervisor:	Dr. Kingshuk Roy Choudhury
Co-supervisor:	Prof. Finbarr O'Sullivan
Head of the Department:	Dr. Michael Cronin

May 2015

NATIONAL UNIVERSITY OF IRELAND, CORK

Date: May 2015

Author:	Sukantadev Bag
Title:	Statistical Methods for Polyhedral Shape Classification with Incomplete Data - Application to Cryo-electron Tomographic Images
Department:	Statistics
Degree:	Ph. D.
Convocation:	June 2015

I, Sukantadev Bag, certify that this thesis is my own work and I have not obtained a degree in this university or elsewhere on the basis of the work submitted in this thesis.

Sukantadev Bag.

[Signature of Author]

NATIONAL UNIVERSITY OF IRELAND, CORK

DEPARTMENT OF STATISTICS

The undersigned hereby certify that they have read and recommend to the Faculty of Science, Engineering and Food Science for acceptance a thesis entitled ***Statistical Methods for Polyhedral Shape Classification with Incomplete Data - Application to Cryo-electron Tomographic Images*** by **Sukantadev Bag** in partial fulfilment of the requirements for the degree of **Doctor of Philosophy**.

Dated: May 2015

Supervisor: ....

Readers: ....

....

Contents

List of Figures

List of Tables

List of Acronyms

2D	-	*Two-dimension/ Two-dimensional*
3D	-	*Three-dimension/Three-dimensional*
ART	-	*Algebraic reconstruction technique*
BC	-	*Bayes classifiers*
BP	-	*Boundary points*
CCD	-	*Charge-coupled device*
CE	-	*Complete edge*
CF	-	*Complete face*
Cryo-EM	-	*Cryo-electron microscopy*
eADJ	-	*Edge adjacency (matrix)*
eADJt	-	*Truncated edge adjacency (matrix)*
ECT	-	*Electron cryo-tomography*
EM	-	*Electron microscopy*
EP	-	*Points on edge*
fADJ	-	*Face adjacency (matrix)*
fADJt	-	*Truncated face adjacency (matrix)*
J_n	-	n^{th} *Johnson solid*
LDA	-	*Linear discriminent analysis*
LS	-	*Least squares*
MAP	-	*Maximum a-posterior*
MLEM	-	*Maximum likelihood expectation maximization*
MPS	-	*Metabolosome (complete) profile statistic*
PCA	-	*Principal component analysis*
PE	-	*Partial (missing) edge*
PF	-	*Partial (missing) face*
PPS	-	*Polyhedron profile statistic*
PSDM	-	*Polyhedral structural distance model*
REP	-	*Rotated edge points*
SART	-	*Simultaneous algebraic reconstruction technique*
SD	-	*Standard deviation.*
SIRT	-	*Simultaneous iterative reconstruction technique*
SP	-	*Standard polyhedra*
SPS	-	*Solid profile statistic, equivalent to PPS*
SVM	-	*Support vector machine*
TEM	-	*Transmission electron microscopy*
TMPS	-	*Truncated metabolosome profile statistic*
TPPS	-	*Truncated polyhedron profile statistic*
TSP	-	*Truncated standard polyhedra*
X-ray CT	-	*X-ray computed tomography*

Abstract

A certain type of bacterial inclusion, known as a bacterial microcompartment, was recently identified and imaged through cryo-electron tomography. A reconstructed 3D object from single-axis limited angle tilt-series cryo-electron tomography contains missing regions and this problem is known as the missing wedge problem. Due to missing regions on the reconstructed images, analyzing their 3D structures is a challenging problem.

The existing methods overcome this problem by aligning and averaging several similar shaped objects. These schemes work well if the objects are symmetric and several objects with almost similar shapes and sizes are available. Since the bacterial inclusions studied here are not symmetric, are deformed, and show a wide range of shapes and sizes, the existing approaches are not appropriate.

This research develops new statistical methods for analyzing geometric properties, such as volume, symmetry, aspect ratio, polyhedral structures etc., of these bacterial inclusions in presence of missing data. These methods work with deformed and non-symmetric varied shaped objects and do not necessitate multiple objects for handling the missing wedge problem.

The developed methods and contributions include: (a) an improved method for manual image segmentation, (b) a new approach to 'complete' the segmented and reconstructed incomplete 3D images, (c) a polyhedral structural distance model to predict the polyhedral shapes of these microstructures, (d) a new shape descriptor for polyhedral shapes, named as polyhedron profile statistic, and (e) the Bayes classifier, linear discriminant analysis and support vector machine based classifiers for supervised incomplete polyhedral shape classification.

Finally, the predicted 3D shapes for these bacterial microstructures belong to the Johnson solids family, and these shapes along with their other geometric properties are important for better understanding of their chemical and biological characteristics.

Acknowledgements

There are many people that I must thank, for without their support this thesis would not have to come to fruition. First, I would like to thank my advisor, Dr. Kingshuk Roy Choudhury. In my years of journeying with him, he was not just my advisor; I received his guidance for almost every aspect of my life, research, decision and social activities. Thank you Dr. Kingshuk Roy Choudhury! Without you, my thesis would not be as it is today.

I would also like to thank my co-advisor Prof. Finbarr O'Sullivan, UCC. I have been fortunate to have his help over the years. With respect and gratitude, I acknowledge Prof. Michael Prentice, UCC for his invaluable contributions to my research. I also thank Dr. Mingzhi Liang, UCC for tomographic images and reconstructions. My special gratitude is for Dr. Michael Cronin, UCC for teaching Linear Models courses and for his useful comments on my thesis. I am lucky to have David Hawe, UCC as my friend – a million thanks to him. The list would be incomplete without mentioning all members from Mathematics Research Office at UCC, and our supporting staff. I would also like to thank Janet O'Sullivan, UCC for the partial support to my research from the SFI-PI grant 11/1027.

I would also like to thank all members from the RAI Laboratories, Duke University for providing me with such a wonderful research environment. I specially appreciate Brian Harrawood and Manu Lakshmanan from the RAI Labs for proofreading my thesis. I am also grateful to Dr. Geoffrey D. Rubin, M.D. who gave me opportunities to work on his project.

This is an opportunity to thank all my colleagues and friends in Ireland, USA and India for their tremendous influence on my research. There are too many people to name everyone, but I must make special mention of a few of them including Dr. Dipak Ghosh and Dr. Madhumita Ghosh, Dr. Rima Dey, Nagababu Chinnam, Swati Sahoo and Mayurakshi Roy Choudhury.

Finally, words are inadequate to thank my family members. They prayed for me, blessed me and did everything that was possible to do for me. Thank you Mom and Dad; thank you Dada and Bhabi. I dedicate my thesis to all of my family members.

Chapter 1

Introduction

1.1 Biological Background

1.1.1 Cells and Organelles

In biology, the cell is the smallest functional and structural unit of life. Working as the building blocks of life [1], cells control all living activities. Some living creatures have only a few cells, even as few as one, but most of them are assemblies of millions of different types of cells. For example, a small protozoan *Amoeba Proteus* consists of only one cell, whereas an adult human has trillions of different types of cells [2]. Different types of cells are responsible for different organic activities, having different structures and may contain different cellular inclusions. For instance, the structures, functions and constituent materials of human red blood cells are different from those of human nerve cells.

There are two broad types of living organisms - prokaryotes and eukaryotes. Prokaryotes have cells without a membrane-bound nucleus and the most ancient living creatures, e.g. bacterial cells, belong to this group. Eukaryotes have a well-defined membrane bound nucleus. All modern living creatures, including humans are from this group. The cells in prokaryotes are smaller in size, simpler in structure and contain fewer cellular inclusions than the cells from eukaryotes [3].

1.1.2 Bacterial Microcompartments

An *organelle* or sub-cellular component is a specialized subunit within a cell having some specific functions [4]. For instance, the nucleus, mitochondria, and ribosome are some of these sub-cellular components in eukaryotes. Functionally, mitochondria are responsible for energy conversion and calcium ion storage, while protein biosynthesis is one of the functions of ribosome.

The term ‘organelle’ is mostly used for cellular inclusions in eukaryotes. The organelle-like structures in bacterial cells are often termed *bacterial organelles* or *bacterial microcompartments*. They are about 100-150nm in the cross section and they contain a number of enzymes encapsulated by a solid protein shell [5], [6]. These organelle-like components from several heterotrophic bacteria are thought to take part in at least seven metabolic processes, though many other functions may yet be discovered [7]. Due to their participation in metabolic processes, they are often named *metabolosomes*.

Types of Microcompartments

The microcompartments are classified based on their participation in different metabolic processes, such as carbon fixation or various forms of fermentative metabolism. Until recently only three types of microcompartments were characterized: Carboxysomes, *pdu*-type and *eut*-type [8].

Carboxysomes were the only known bacterial microcompartment for several years [7], [8], [9]. They were observed more than 50 years ago as polyhedral bodies inside cyanobacteria and participate in the carbon fixation processes. Among other microcompartments identified, a recently found one is from heterotrophic bacteria, e.g. Cytrobacter freundii, Escherichia coli (Figure 1.1). These metabolosomes are *pdu*-type and take part in a specific metabolic process called 1, 2 - propanediol utilization [8], [9]. The detailed chemical and biological properties of these *pdu*-type metabolosomes have only been explored recently [9].

The detailed structure of Carboxysomes helped scientists gain a better understanding of its chemical, biological and structural properties and its importance [7], [9]. This work also focuses on the three-dimensional structures of these *pdu*-type metabolosomes described in [9] and is expected to provide more insight into their characteristics. The significance of these structures are described in Section 1.1.4.

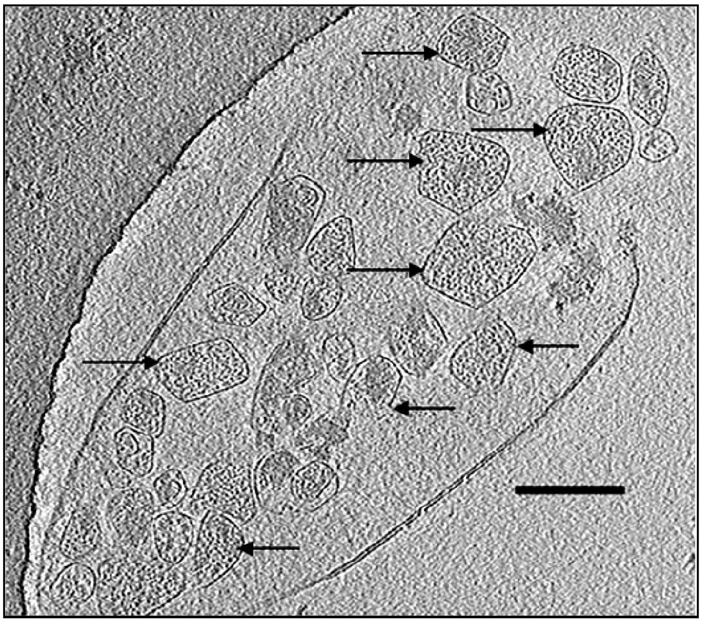
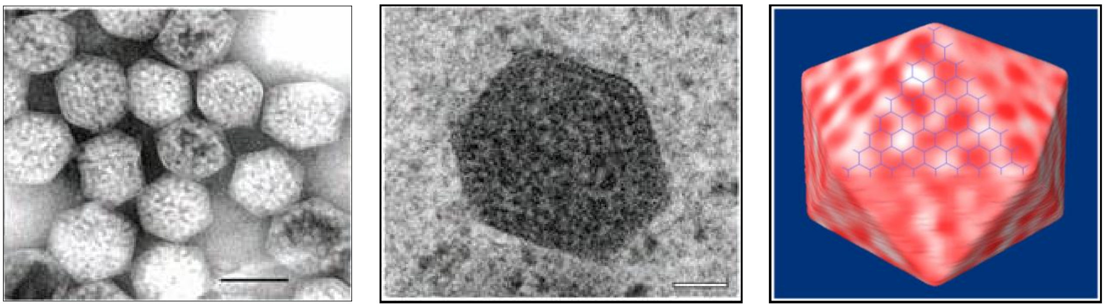


Figure 1.1: The *pdu*-type microcompartments inside a bacterial cell, imaged through electron cryo-tomography. The microcompartments are marked with arrows. The scale bar indicates 100 nm. Image source: Liang et al. [10].

1.1.3 3D Structures of Microcompartments

The 3D structure is one of the important characteristics of microcompartments and it is of current research interest, as the structure plays significant roles in biotechnology, medicine and related areas.



(a) (b) (c)

Figure 1.2: Carboxysome is in two-dimensional and three-dimensional shapes. (a) Purified Carboxysomes from Halothiobacillus neapolitanus cells (scale bar, 100 nm). (b) An enlargement of a single Carboxysome (scale bar, 50 nm). (c) Its icosahedral structure described in [10], [11].

Microcompartments are macro-molecular assemblies, but like many natural objects, these macromolecular structures surprisingly have well-defined geometric shapes. For instance, Carboxysomes have a highly symmetric polyhedral shape, called an icosahedron (Figure 1.2) [12], [10]. Many natural objects, such as poliovirus, rhinovirus, adenovirus, etc. also have similar polyhedral shapes [13]. The amount and arrangement of the proteins and nucleic acid inside viruses determine their sizes and shapes [14].

1.1.4 Biological Significance

There is a growing recognition that living cells are made of multiple interior interconnecting machines, i.e. macromolecular complexes [15]. To appreciate how cells work, it is important to understand the structure of these large protein complexes and how they fit together. A three-dimensional model of these components is essential to understand how they interact with each other and perform their roles within the cell.

Medicine

In medicine, knowing the shape of a biological structure and how it fits together with other structures helps in designing drugs to affect its function. Typically drugs apply their effects by molecular interactions with molecules both inside and outside of cells. The three-dimensional shape of the structures ultimately determines these interactions. For example, extrication of the details of these complex macromolecular structures and the associated self-assembly path leading to their efficient and precise construction, plays an important function in developing medicines to affect their activities [16], [17].

Biotechnology

In biotechnology, knowing the overall shape and inclusions inside these structures permits predictions about their method of assembly, strength, volume, and maximal metabolic capability. For example, the volumes of the microcompartments provide an estimate of the amount of proteins inside the shell. The face area similarly estimates the amount of shell proteins, and the length of edges is an indication of multi-component elasticity shell capsule [18]. The surface area to volume ratio is an estimate of the ratio of area available to exchange metabolites to the amount of enzymes inside available to work on them [16], [19].

Industry

Understanding three-dimensional structures of these bacterial inclusions may also help biologists for on demand production of these macromolecular structures in laboratories.

1.2 The Problem Statement

1.2.1 Problem with Tomographic Imaging

Cryo-electron tomography or electron cryo-tomography (ECT) is a well accepted imaging method for biological microstructures since it allows scientists to visualize biological structures in near-native form by avoiding sample fixation artefacts, such as dehydration [20]. In single-axis ECT, biological samples rotate along a single axis in a limited range of angles (± 60° in general) [21] due to technical restrictions.

Since the sample rotates in a limited angle range only, the reconstructed three-dimensional objects contain unobservable (missing) regions at the top and the bottom (i.e. along z-axis). This problem is commonly termed as the *missing wedge* [21], [22] or *missing cone* problem. A graphical presentation of the missing wedge problem is provided below. A detailed discussion about this problem, underlying reconstruction geometry, proportion of missing data etc. are provided in Section 2.2.5 (Chapter 2).

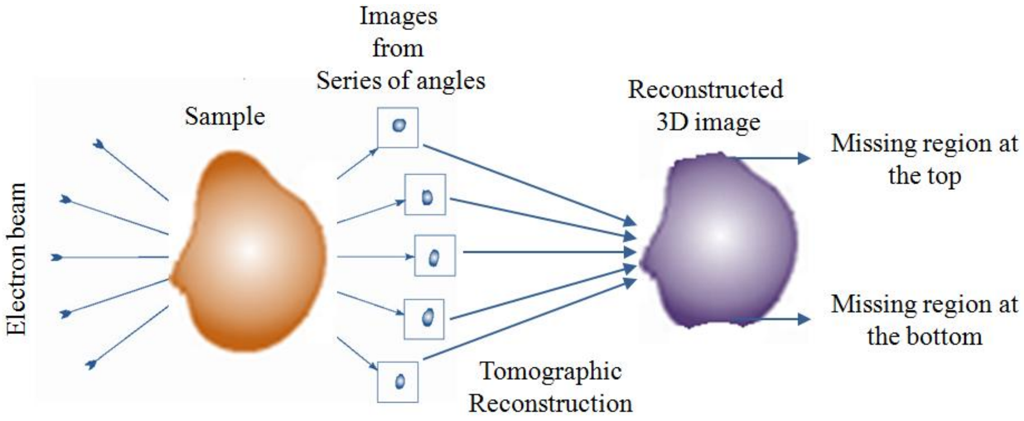


Figure 1.2a: A graphical representation of the missing wedge problem.

These missing regions in reconstructed objects introduce difficulties in three-dimensional structural analysis [21]. In addition, noise, low contrast and low resolution along with the missing wedge problem make structural analysis especially challenging.

1.2.2 Shape Alignment and Averaging

Existing strategies for managing the missing wedge problem from ECT are broadly two types: (1) shape alignment and averaging for single-axis tomography and (2) dual-axis tomography [23], [24]. Since this research is focused on analyzing single-axis tilt-series tomographic images, the dual-axis tomography techniques are not discussed in this dissertation.

For simplicity, we described the shape alignment and averaging method in the context of two-dimensional projection images. For two-dimensional images, this method can enhance signals of the images and reduce noise [25], [26]. Here we have used 2D diagrams to demonstrate how this approach also improves the visibility of the objects in the presence of missing data.

Let us consider a biological specimen containing several copies of the object of interest (e.g. metabolosome). The specimen was imaged through electron microscopy from *only one angle of view*, generating a single two-dimensional projection image of the specimen. Since the specimen contains several copies of the object, this image contains several objects' projections.

Step 1: Alignment

Even when two objects have the same shapes and sizes, their projections may be different due to their varied orientations in 3D [25]. First, from this set of images of the objects, a few 'identical' projections are picked up. These 'identical' projections may only differ with respect to their orientations (Figure 1.3) in the projection plane. To demonstrate, in Figure 1.3 three projection images are identical (hexagonal objects of almost same size) but have different orientations with respect to the reference axes (x- and y- axes).

This step involves finding the orientations and properly aligning the objects based on these orientations such that the aligned objects show the same structural orientation (Figure 1.3). Several methods have been developed for finding the orientation [27], for example, the common lines method [28], polar Fourier transform [29], method of moments [30], model based alignment [31], Procustes method [32] etc. are just a few of them.

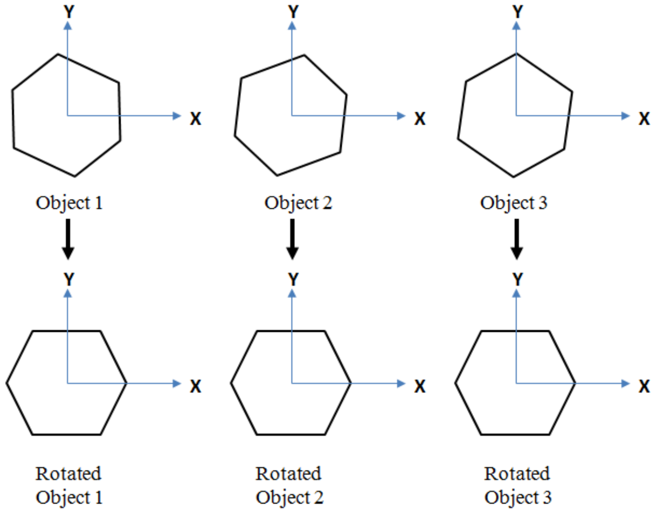


Figure 1.3: Alignment of identical objects so that the objects have same orientation. Here three uniform hexagons are aligned such that the aligned hexagons have same orientation.

Step 2: Averaging

The aligned images now look identical with respect to the orientation as well. Finally, the aligned images are overlapped or 'averaged' to generate a single image expecting that the averaging would improve the signals of the projection images and reduce noise. As shown in Figure 1.4, four aligned objects with missing regions are overlapped. It shows the alignment and averaging technique can also improve the visibility of the missing object boundary.

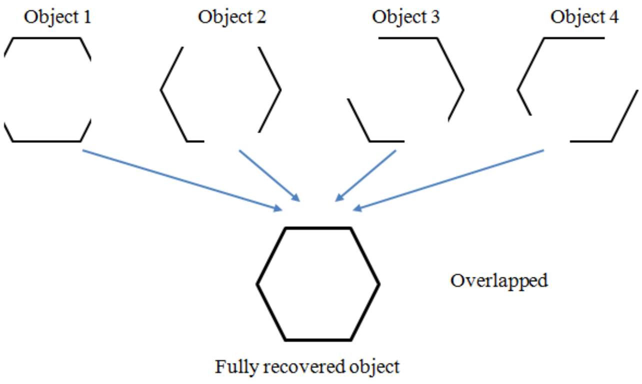


Figure 1.4: The averaging step to manage missing wedges. Four aligned hexagonal objects have missing regions in their different parts. These objects are overlapped to generate a complete hexagon.

The averaging method, followed by a three-dimensional reconstruction was first used by R. A. Crowther [33] to study the three-dimensional structures of a virus (tomato bushy stunt virus). Since this technique can potentially improve the visibility of the missing regions, this concept has also been extended and applied to handle the missing wedge problem in three-dimensional tomographic reconstructions [21], [34].

1.2.3 Averaging in Tomography

Step 1: Tomographic Reconstruction

First, a three-dimensional image is reconstructed from tilt-series tomographic projection images (Chapter 2). The reconstructed three-dimensional image may contain several copies of the object and these three-dimensional objects have missing regions due to the missing wedge problem. The individual objects are isolated from the reconstructed three-dimensional image. This isolation is done by creating a bounding box around the three-dimensional object inside the full tomogram. The process is explained in Section 2.4.1 (Chapter 2).

Step 2: Alignment

The next step is finding orientations of the objects and aligning them in 3D. Several methods have been developed for this purpose [35], [36], for example, alignment with respect to a reference [37], reference free alignment [38], [39], and multiple reference [40] are a few of them. Sometimes, the tomographic images are denoised prior to alignment for better alignment accuracy [41]. As described in [35], "these methods rely in general on the premise that differences observed among projection images of different copies of the same object arise from differences in the orientation of the object relative to the electron beam." Since these methods assume that the different images are just copies of the same object, here scaling is not particularly important.

Step 3: Averaging

Finally, the aligned three-dimensional images are overlapped to generate a single image, expecting that the averaging will improve the visibility of the missing regions of the objects [42], [33]. Since the alignment and averaging require a large number of objects, software (e.g. SPIDER [43]) has been developed for this purpose.

1.2.4 Limitations of Shape Alignment and Averaging

Clearly, these alignment and averaging approaches work well if the objects under study have similar shapes and sizes. If the objects are different in structure, the 'averaged' shape may not be a good representative shape for the objects. However, if the objects have different shapes, then several tomograms are to be identified for each different shape and size class.

In addition, these methods work most efficiently if the objects are *similarly* symmetric [29]. However, recently some methods have been developed without the symmetry assumption [35] but these methods require more samples to align and average, and also involve higher computational costs.

Nonetheless, all these methods rely on the fundamental assumption of single particle analysis that there are several objects available with similar inherent shapes and sizes. Clearly, none of these methods can predict structures based on a single object and hence, a uniquely shaped object is ignored since no other object is available to average with it. In addition, alignment and averaging may eliminate the finer structural distinctions among objects.

1.2.5 Requirements for New Algorithms

As demonstrated in Chapter 4, our objects of interest show a wide range of volumes and they are non-symmetric and non-uniform, i.e. deformed. Since these objects are imaged through single-axis tilt-series ECT [44], they also suffer from the missing wedge problem. In fact, there are no two objects found having the same shape and size (Chapter 6). Hence the fundamental assumptions behind shape averaging are not satisfied here. Therefore, new shape prediction algorithms from ECT images are required having these properties:

1) Can deal with non-symmetric or deformed shapes.
2) Can work with objects having considerably different shapes and sizes.
3) Can predict shapes based on a single object. Hence the uniquely shaped objects can be characterized.

An initial analysis shows (Chapter 4), like Carboxysomes [12], [10], many viruses [13] and crystals [45], the objects under study (metabolosomes) also have convex polyhedral shapes. This observation sets the theme of this research, i.e.

developing statistical methods for predicting three-dimensional convex *polyhedral* structures from single-axis tilt-series ECT images, in the presence of missing data. The statistical methods developed here are also designed to conform to the required properties, i.e. these methods can be applied to the objects with different shapes and sizes, objects having non-symmetric or deformed shapes, and do not require multiple objects to align.

1.3 Polyhedral Shape Classification

1.3.1 Standard Polyhedral Shapes

Standard classes (families) for convex polyhedra were defined long ago [46], [47]. Some common convex polyhedra families are Platonic solids, Archimedean solids, Johnson solids, Catalan solids, prisms and antiprisms [46], [47]. These solids are uniform (Chapter 3), contain some symmetry properties [46] and are few in number. Undoubtedly, infinitely many other convex polyhedral structures can exist in nature or are possible to construct, but they are not ‘standard’, i.e. they do not fit in to these well-defined polyhedral families.

A standard convex polyhedron can generate infinitely many convex polyhedra through affine or non-affine transformations. Though these transformed solids do not carry the symmetry or uniformity characteristics, they still hold identical topological properties like vertex, face, and edge counts from their parent solids. Besides all uniform standard solids, in this work, we also consider this infinitely large set of *deformed standard solids* for shape classification.

1.3.2 Feature Vector for Polyhedra

Statistical shape analysis starts with a shape descriptor. A great number of shape descriptors have been developed and used for different purposes. For example, landmarks [48], [49], dense surface meshes [50], skeleton-based representations [51], [52] are some of them. A review of these methods is provided in [53].

In general, these shape descriptors (or feature vectors) map the image to a higher dimensional space and statistical models are developed to classify the shapes based on these high-dimensional features. However, the choice of the feature vector depends on the intended applications.

Since the training image space for this work consists of standard polyhedral shapes, it needs a special shape descriptor, retaining unique standard polyhedron identification features. The objects under study in this research are deformed (Chapter 4). Clearly, symmetry or uniformity properties based descriptors are not meaningful for these objects. So, we define a set of deformation invariant topological properties as shape descriptors to characterize deformed standard polyhedral shapes (Chapter 3).

1.3.3 Incomplete Polyhedral Shape Classification

Again, analogous to the missing wedge problem, if a standard solid is truncated by two parallel planes from random orientations, it generates a class of different truncated shapes, based on the truncation percentage and orientation (Chapter 6). This observation turns this shape analysis problem into a supervised learning problem [54], where the collections of truncated shapes from standard solids constitute the training classes. This research develops classifiers for incomplete polyhedral shapes, based on the set of *deformation invariant topological properties* (Chapter 6).

1.4 Thesis Outline

As mentioned in Section 1.1.2, Carboxysomes were the only metabolosomes having three-dimensional structures studied so far [12], [10]. This study focuses on developing statistical methods to predict three-dimensional structures of another microcompartment - a *pdu*-type metabolosome (Section 1.1.2). Although previous work [9] highlighted some basic information about the structural features, such as diameter, axial dimensions of the non-spherical polyhedral structures of these metabolosomes, this is the first extensive characterization of their three-dimensional structures.

This research starts with the raw metabolosome images acquired through ECT (Chapter 2). Subsequently, raw projection images were combined using tomographic reconstruction methods, segmented, smoothed and visualized before analyzing their three-dimensional shapes. These essential image data preparation steps are also part of this research and included in Chapter 2. An important part of this chapter is a least squares based method we developed for improved manual image segmentation (Section 2.5.5).

The reconstructed slices and the reconstructed incomplete (due to missing wedge problem) three-dimensional metabolosomes indicate that the metabolosomes have convex polyhedral shapes. In Chapter 3, standard polyhedral families, e.g. Platonic, Archimedean, Johnson and Catalan solids and their properties are described. A notable part of this chapter is the development of topological properties based on the *polyhedron profile statistic*, which is invariant under affine and non-affine transformations.

Chapter 4 is concerned with the fundamental structural properties of the metabolosomes, such as volume, aspect ratio and deformation. It illustrates that the metabolosomes have widely distributed sizes; they are deformed and non-symmetric. Hence we showed the inadequacy of the shape alignment and averaging methods. One significant part of this chapter is a modified INTERPRET program [55] we developed for managing the missing wedge problem in polyhedral structures.

Using the method described in Chapter 4, the incomplete metabolosome shapes are ‘completed’. In Chapter 5, we developed a polyhedral shape matching algorithm (polyhedral structural distance model) to predict the shapes of the metabolosomes based on these ‘completed’ structures.

The goal of the next chapter (Chapter 6) is to develop polyhedral shape classifiers in the presence of missing data. Part I of this chapter describes a simulation scheme to generate classes of truncated standard polyhedral shapes. The second part of this chapter explains a novel Bayes classifier [56] we developed to classify incomplete polyhedral shapes. We also used linear discriminant analysis (LDA) [57] and support vector machine (SVM) [58] to classify incomplete polyhedral structures. This chapter concludes with combining these classifiers into a single classifier and predicted shape for metabolosomes.

Chapter 7 summarizes and concludes from the results obtained from analyses and discusses the future directions of this work. Appendices and references are included at the end of the dissertation.

1.5 Contributions

In this work we present a framework for predicting standard polyhedral shapes from ECT images, in the presence of missing data and applied it to predicting three-dimensional structures of a bacterial inclusion. These algorithms do not need multiple objects to be aligned, work with deformed, non-symmetric, varied sized objects and in the presence of missing data. The contributions include:

1) An algorithm to improve the outcome of manual image segmentation.
2) An algorithm for handling missing wedge problems for polyhedral shapes.
3) Deformation invariant topological properties as a polyhedral shape descriptor.
4) A structural distance model for predicting standard polyhedral shapes of deformed three-dimensional objects, in the presence of segmentation and reconstruction errors and curved surfaces.
5) A Bayes classifier for incomplete polyhedral shape classification with a hierarchical shape matching scheme based on the Bayes classifier.
6) Linear discriminant analysis and support vector machine based classification of incomplete polyhedral shapes.
7) First study of fundamental structural properties, such as volume, aspect ratio, symmetry of the *pdu*-type bacterial microcompartments.

At least two research papers are expected to be published from this work. The manuscripts are submitted and now under review. The details about these papers are included in the 'Publications by the Author' section at the end of this thesis. The author presented this work in two conferences, details about those presentations are also included in the 'Publications by the Author' section.

Chapter 2

Tomographic Reconstruction, Segmentation and Visualization

2.1 Introduction

The metabolosome images from cryo-electron tomography are the basic data for this research. These projection (raw) images are processed using tomographic reconstruction methods and the reconstructed images are used for further analysis. This chapter starts with describing cryo-electron microscopy. Section 2.3 describes imaging metabolosomes and three-dimensional tomographic reconstruction from projection images. The pre-segmentation image processing steps (e.g. trimming, reconstructing images and slicing) are described in Section 2.4. Image segmentation and a proposed least squares based method for improved image segmentation is discussed in Section 2.5. Finally, smoothing and visualizing the segmented metabolosomes are explained in Section 2.6. All projection images are processed through the same image processing steps.

2.2 Tomography and Reconstruction

2.2.1 Transmission Electron Microscopy

An electron microscope differs from a light microscope in that it uses an electron beam instead of light to illuminate a specimen and produce a magnified image [59], [60]. Since electrons have wavelengths much shorter (about 100,000 times) than visible light, electron microscopy (EM) provides much higher magnification and can generate images with resolution as high as 50 picometre (1 picometre = 10^{-12} metre) [61]. The resolution and magnifying power of electron microscopy make it indispensable to investigate the ultrastructure of a wide range of biological and inorganic specimens including micro-organisms, cells, large molecules, biopsy samples, metals, crystals, etc. [62], [63]. For example, EM has been used to describe the structure of a poliovirus having about 15nm radius [64].

Transmission electron microscopy (TEM) is a form of electron microscopy where a beam of electrons is transmitted through an ultra-thin specimen. Depending on the density of the material present, some electrons are scattered out of the beam during transmission. An image is formed by the remaining electrons transmitted through the specimen. The image is magnified and focused onto an imaging device, like a fluorescent screen, photographic film or charge-coupled device (CCD) censor. As a result, TEM generates a two-dimensional projection image of a three-dimensional object [65], [66], [67]. Figure 2.1 shows the structure of a transmission electron microscope.

2.2.2 Cryo-electron Microscopy

Biological specimens are easily damaged during preparation for electron microscopy due to dehydration, adsorption (accumulation of molecules to form a thin film on the surface of a solid) onto the supporting film and other causes [68]. Cryo-electron microscopy (cryo-EM) addresses these problems by using cryogenic temperatures to fix the samples before imaging, preserving the native shape of the specimen [20]. This emerging technology allows thin samples such as macromolecular complexes and small bacterial cells to be imaged in a nearly native state to molecular (~ 4 nm) resolution [69].

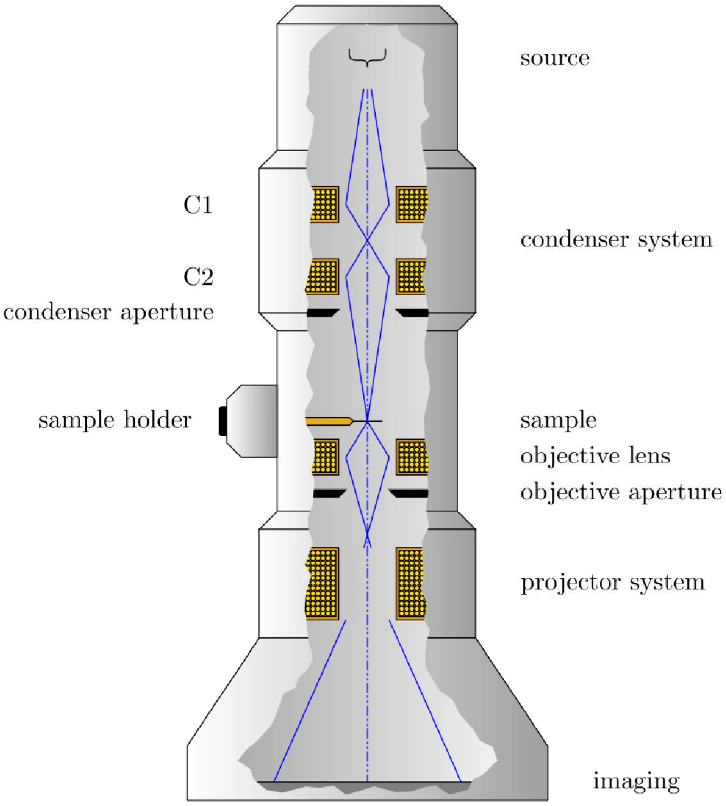


Figure 2.1: The imaging system in a transmission electron microscope. Image source: [70].

The many advantages of cryo-EM over general TEM [69], [71], [72] include:

- It allows the imaging of native and hydrated structural features of the specimen.
- There are no stains or chemical fixatives to distort the sample.
- The contrast between nucleic acids, proteins, and lipids can be distinguished.
- Provides good preservation of biological structure in the microscope vacuum.
- There is no distortion by attaching the sample and flattening against the supporting film.

This imaging method is widely used in biological sciences, studying viruses [73], bacteria [74], cellular inclusions [75], and protein complexes [76]. This project also adopted the cryo-EM technique for imaging metabolosomes.

2.2.3 Electron Tomography

In general, 'electron tomography' refers to any technique that employs the TEM to collect projections of an object that is tilted in multiple directions and uses these projections to reconstruct the three-dimensional object in its entirety [21]. Similar to

TEM, when the penetrating wave is an X-ray, it gives the most familiar application of tomography, known as X-ray computed tomography (X-ray CT) [77]. Tomography has found extensive applications in many scientific areas, including physics, chemistry, biology and medicine.

2.2.4 Tomographic Reconstruction

Electron cryo-tomography (ECT) is a special form of tomographic electron microscopy where cryo-EM is used to generate the projection image. Multiple projection images taken from different directions are required to capture the 3D structure on the specimen using tomographic reconstruction techniques. A tilt-series imaging technique rotates the sample in 1° increments about an axis perpendicular to the electron beam, generating an image at each angle.

A. Klug in his pioneering work in 1970's [33] developed the technique of reconstructing three-dimensional images from tilt-series image sets. Later in 1974, W. Hoppe used this technique to reconstruct 3D images of macromolecules (fatty acid synthetase molecule) [78], [79]. This technique has two steps: first, each projection image from the tilt-series is corrected for unwanted motion artifacts, and second, the projection images are transformed into a single 3-dimensional image.

Motion artifacts are the errors due to the positioning of a sample during imaging. Such errors can occur due to vibration or mechanical glide. Alignment methods are used to correct these errors. Image registration techniques are used to align the various images in the series to each other. There are several methods commonly used for image registration (see [80], [81], [82]).

The set of aligned two-dimensional images are transformed and combined to construct a single three-dimensional image using reconstruction algorithms. The most widely used families of reconstruction algorithms in 3D TEM are direct Fourier inversion, filtered back-projection and algebraic methods. The back-projection reconstruction algorithms are based on inverse Radon’s transform [21], [83], [84]. Additionally, many algebraic methods are also available, for example, the simultaneous iterative reconstruction technique (SIRT), the algebraic reconstruction technique (ART), the simultaneous algebraic reconstruction technique (SART) [84], and the maximum likelihood expectation maximization (MLEM) [85].

Many tomographic reconstruction tools (including IMOD [86]) provide back-projection and SIRT algorithm based reconstruction. For this work, we applied the back-projection method provided in IMOD. Figure 2.2 shows a general scheme for reconstructing a 3D image from tilt series images.

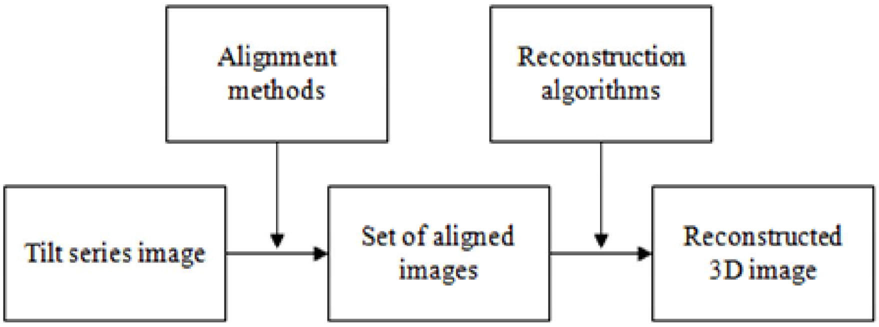


Figure 2.2: Diagram showing the tomographic image reconstruction from tilt series image.

2.2.5 The Missing Wedge Problem

Commercial TEM systems normally have restricted tilt angle ranges and generally, the tilt angle range does not exceed ± 60° [21] due to mechanical support systems for the specimen. This limits the number of projection angles and reduces the volume of the specimen which can be reconstructed in the final image, since a significant portion of the Fourier transform is not measurable [21]. Therefore, the Fourier-based back projection method decreases the range of resolvable frequencies in the 3D reconstruction. This problem is known as 'missing wedge' or 'missing cone' problem. Figure 2.3 shows the Fourier geometry of missing wedge.

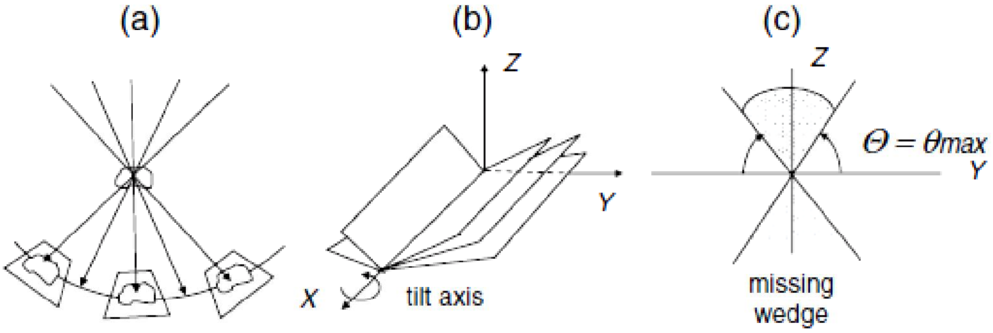


Figure 2.3: Single-axis tilt data collection geometry. (a) Tilting in real space around an axis perpendicular to the plane of this page (which is assumed to be the x-axis). (b) Fourier space representation of the information presented in (a); the different projections provide values in the central sections, through the 3D Fourier Transform of the object with the X-axis in common. (c) Close-up view of (b) from the direction of the positive X-axis, showing the missing wedge. Image and description are from Frank, 2006 [21].

However, mechanical refinements, such as dual-axis tilting [87] or conical tomography [88], [89] can be used to control the impact of the missing data on the observed specimen structures. In addition, numerical techniques exist for improving collected data quality [21]. Figure 2.3a shows the missing regions in a reconstructed and segmented metabolosome.

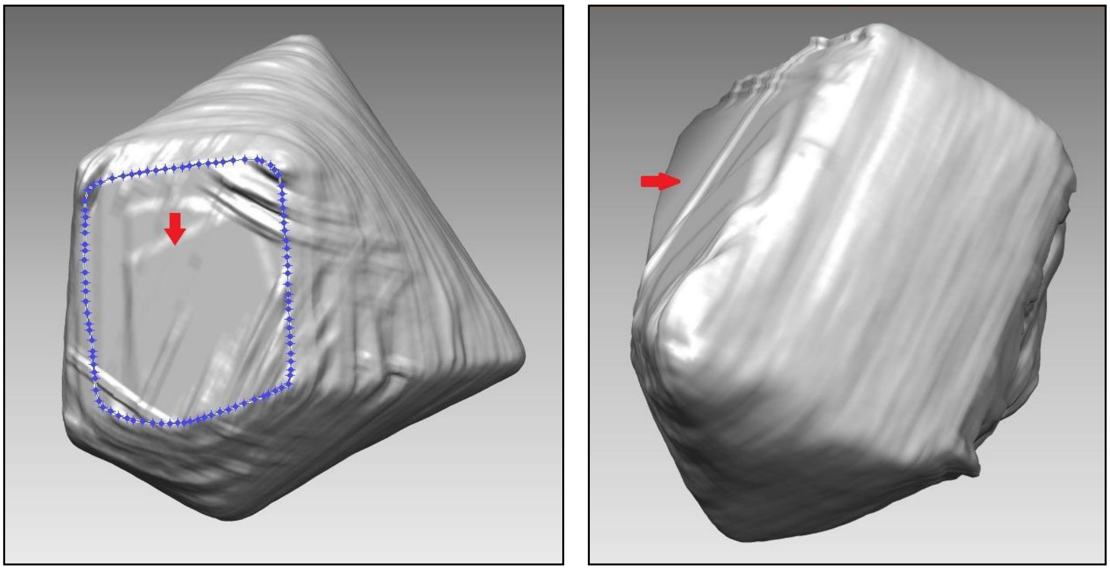


Figure 2.3a: The Missing regions on reconstructed and segmented three-dimensional metabolosomes. The missing regions are marked with red arrows. In left, the missing region is also marked with an approximate boundary; in right, a missing region is shown from a different view.

How much data is truncated?

A very important point to consider: since the tilt angle range does not exceed ± 60°, how much data is truncated? If the object if a perfect sphere, about 33% of its volume are unobservable [90]. Since we are reconstructing the surface only (based on the slice boundaries), the internal structures inside the objects are not considered. So the interest is not in the volume, but in its surface. We need to estimate the proportion of its diameter affected due to the missing wedge problem. Based on the Figure 2.3(c), Figure 2.3b is drawn.

The maximum rotation angle is 60° as marked in the Figure 2.3b. The circle represents a spherical object; so the missing wedge at the top is the 'pie' bounded by the angle *P1CP2*. *CD2* is the radius and due to missing wedge problem the part *D1D2* is missing from the radius. A simple calculation shows, if *CD2* = *r*, *CD1* =

$\frac{\sqrt{3}}{2}\,r$ and hence, *D1D2* = $0.13r$. So, due to ± 60° limited rotation, about 13% (of its height) from top and 13% (of its height) from the bottom of the object are 'truncated',

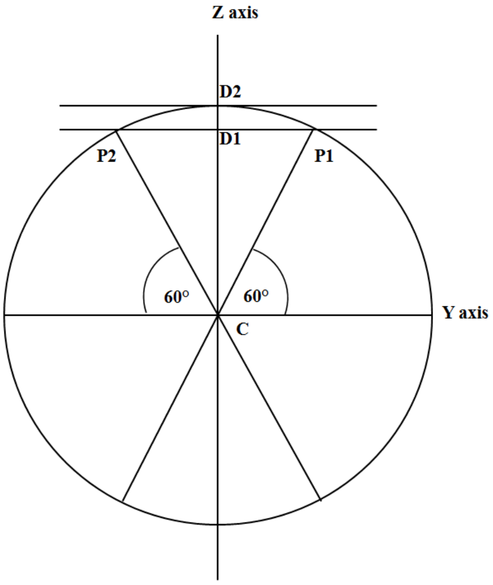


Figure 2.3b: Geometry of missing wedges - extended from Figure 2.3(c).

The metabolosomes are not spherical - they are polyhedral. In addition, they are not uniform and its vertices, faces and edges could be in the missing regions. So the 13% radius truncation rule may not be *strictly* applied to the metabolosomes. In fact, the truncation proportion could also vary for each metabolosome. Here we consider a range of truncation proportions for metabolosomes: 5% to 15% of the vertical height of the object (in place of radius, since the objects are not spherical). The Section I of Chapter 6 discusses more about these truncation proportions.

2.3 Imaging Metabolosomes and Reconstruction

2.3.1 Imaging Metabolosomes

The metabolosomes were imaged through single-axis tilt angle cryo-electron tomography at The Jensen Laboratory [91], California Institute of Technology. During the imaging process, the specimen was tilted around a single axis from -60° to +60° with 1° intervals. Therefore, each specimen was imaged 121 times from consecutive 1° interval angles, generating a sequence of 121 two-dimensional images (projections). The imaging system then combined the tilt series images and generated a single file as output (raw image).

A total of 41 specimens were imaged using Cryo-EM. This research starts with these 41 raw images. The details of biological sample preparation, extracting metabolosomes from bacterial cells and imaging parameters are described in [44].

2.3.2 Tomographic Reconstruction for Metabolosomes

The eTomo module of IMOD [86] was used for the tomographic reconstructions from the raw projection images. As explained below, some of the reconstruction steps in IMOD, e.g. marking gold particles, slice selection for final contrasts were manually executed. During imaging, 121 images (2D) are taken from 121 angles and these 2D images are required to be aligned (known as image registration) properly before 3D reconstruction. To facilitate the alignment process, generally some gold nano-particles are placed near the specimen and these particles generate black 'dots' on these 2D images. As a part of the IMOD reconstruction procedure, these gold particle spots are required to be marked manually.

During the last step of the reconstruction, IMOD requires a range of grayscale values for the final reconstructed image. These values can be input manually, or IMOD can calculate them from a set of selected slices. We selected 5 middle slices for each image for this purpose. This effects the image contrast and brightness only.

IMOD eTomo module provides a program ‘tiltalign’ for image registration to align tilt-series images. eTomo makes available both the SIRT and back-projection methods for reconstruction. The back-projection method was preferred for its efficiency. The final output of eTomo is a reconstructed three-dimensional stacked image.

All of the 41 raw images were reconstructed, but 7 of them do not contain metabolosomes. Also, 18 of the reconstructed images contain a few metabolosomes, but with very low visibility. This very low visibility may occur due to several reasons, such as part of the metabolosome is outside of the image or the image has extremely low contrast. A total of 16 images were selected from the 41 reconstructed images based on visibility of the metabolosomes.

Each reconstructed image has dimension 2048 x 2048 pixels along both x- and y-axis. Along the z-axis the dimensions (thickness) are between 390 - 490 pixels with average thickness of 476 pixels (Table 2.1).

Image Serial Number	Dimension along x-axis (in pixel)	Dimension along y-axis (in pixel)	Dimension along z-axis (in pixel)
05	2048	2048	486
10	2048	2048	390
11	2048	2048	480
12	2048	2048	490
13	2048	2048	484
14	2048	2048	480
15	2048	2048	476
16	2048	2048	484
19	2048	2048	484
20	2048	2048	476
22	2048	2048	490
31	2048	2048	476
32	2048	2048	490
34	2048	2048	468
39	2048	2048	486
41	2048	2048	476

Table 2.1: The dimensions of 16 selected reconstructed images (in pixel).

Based on the dimension of the projection images, IMOD determines the dimension (along x-axis and y-axis) of the reconstructed 3D images. But it calculates dimension along z-axis based on a manually provided parameter called *Sample tomogram thickness* and another manual process called *Create boundary model*. The *Sample tomogram thickness* was set to 500 pixels for all reconstructed images. This step has significant impacts on the final reconstructed images, since it determines the number of slices and physical slice 'thickness'. A smaller value for the sample tomogram thickness (e.g. 250) would 'squeeze' the tomogram and a few objects of interest could be excluded from the reconstructed tomogram. Too large value of the sample tomogram thickness (e.g. say 1000) would generate more slices, but requires longer time for segmentation and also reduces the overall image quality.

2.3.3 Visualizing Raw and Reconstructed Images

It is important to visualize the reconstructed images prior to further processing because we need to determine if a reconstructed image contains metabolosomes and whether the metabolosomes are clearly visible. In addition, the metabolosomes were labeled for identification during visualization. The following image shows a typical slice from a reconstructed image. The labels on the image identify individual metabolosomes.

Several tomographic image visualization software packages, including 3DMOD from IMOD [86], Chimera UCSF [92], AMIRA [93] were used to visualize the reconstructed 3D images and for subsequent analyses. As mentioned before, IMOD and its modules helped to reconstruct the 3D images from the tilt-series images, AMIRA was used to visualize the 2D and 3D images in different steps. Another important purpose of AMIRA is the slicing of the 3D images and stacking the slices back to reconstruct the 3D image. The Chimers UCSF helped to draw the 3D polyhedral structures (Chapter 4). These tools are simple to use and widely accepted, particularly for this type of research problems.

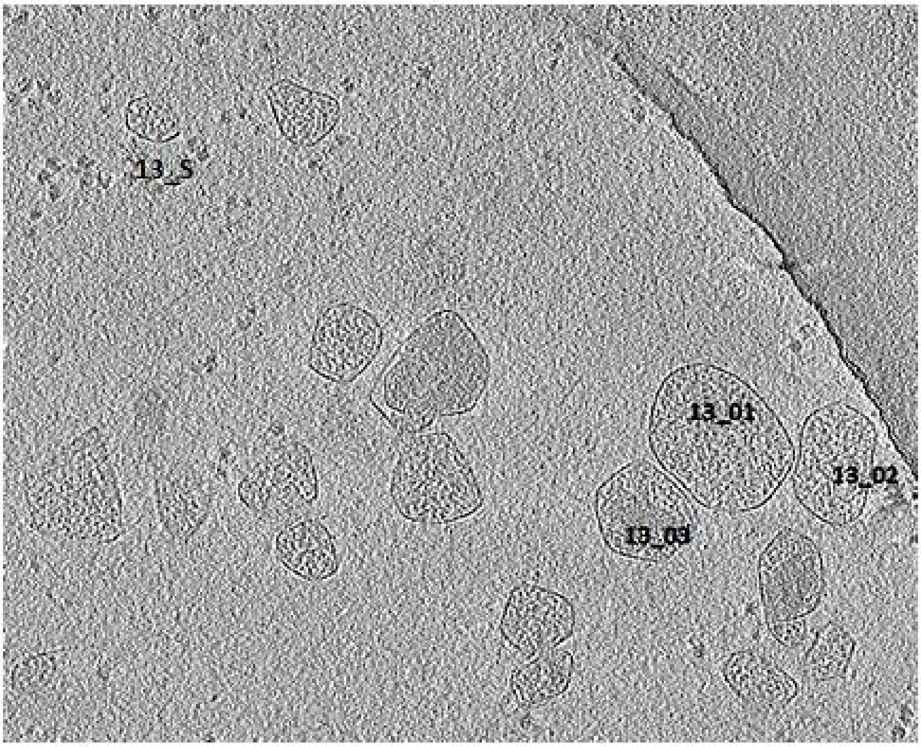


Figure 2.4: A typical slice from a reconstructed 3D image showing few metabolosomes, three of them are labeled.

2.4 Trimming and Slicing 3D Images

2.4.1 Trimming Complete Tomogram

As is visible in Figure 2.4, the reconstructed three-dimensional images contain multiple metabolosomes. A typical metabolosome is much smaller than the full reconstructed image. The individual metabolosomes must be separated from the full reconstructed image for segmentation and subsequent analyses. This process contains three steps:

1) Finding a 'bounding box' for the target metabolosome such that the bounding box contains a complete object and least possible non-contributing exterior.
2) Recording the coordinates from the vertices of this bounding box.
3) Trimming the complete tomogram based on the coordinates from (2).

The 'trimvol' module of IMOD separates the volume selected by the bounding box from a complete tomogram based on the bounding box coordinates.

Selection of metabolosome was subjective. We selected larger objects for segmentation. Larger objects are selected visually, for example, in Figure 2.4, 13_01, 13_02 and 13_03 are larger objects but 13_S is a smaller one. Larger metabolosomes were selected for trimming because larger metabolosomes have larger boundaries in slices and the larger the boundary, the more precise is the segmentation. In other words, larger is better because the relative error of reconstruction is smaller for a larger object, assuming a fixed level of error for boundary drawing. Section 4.5.1 (Chapter 4) describes the volumes of the smallest and largest metabolosomes segmented.

However, we also attempted to segment and reconstruct a smaller object, but the next image processing steps such as structure drawing etc. (Section 4.2, Chapter 4) was difficult. The difficulties were due to problems in identifying edges and vertices on the surface of the segmented small object.

2.4.2 Slicing Trimmed Image

Each 3D trimmed image was sliced into a series of 2D images for manual segmentation and subsequent image processing steps. During slicing, we sliced each object three times: along the x-, y- and z-axes using AMIRA and MATLAB [94]. If the dimension of the trimmed image is $d_1 \times d_2 \times d_3$ in pixels, slicing procedure generates three separate slice sequences with following description:

	Number of Slices	**Dimension of Each slice**
Sliced along x-axis	d_1	$d_2 \times d_3$
Sliced along y-axis	d_2	$d_1 \times d_3$
Sliced along z-axis	d_3	$d_1 \times d_2$

Table 2.2: The number of slices generated from thee-way slicing.

The above slicing method shows, each two-dimensional slice is one pixel 'thick'. Figure 2.5 illustrates the slicing procedure graphically.

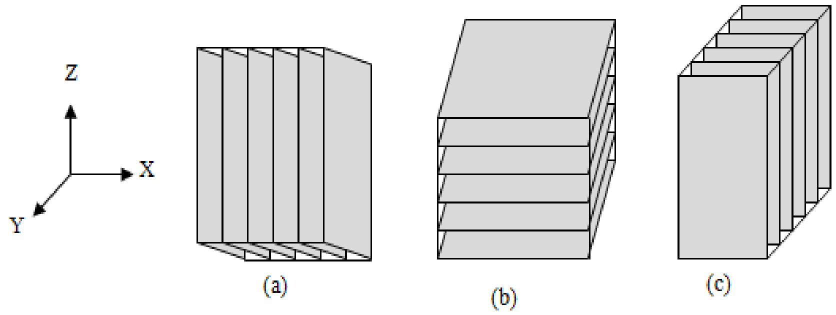


Figure 2.5: Three-way slicing method. (a) Slicing along x-axis generates slices in yz plane, (b) Slicing along z-axis generates slices in xy plane and (c) Slicing along y-axis generates slices in xz plane.

The slices along the z-axis contain the clearest and complete boundaries of the objects and hence are the most informative in general. However, the slices along the x-axis and y-axis also carry useful information about metabolosome shapes and we utilized this information during the image segmentation (Section 2.5.4). The following image (Figure 2.6) shows three slices from three slice sets of a trimmed metabolosome.

2.4.3 Voxel in Metabolosome Images

In digital image analysis, a pixel or picture element is the smallest element or unit of a digital image. Formally, a pixel at (x, y) of a two-dimensional image of size (or dimension) $m \times n$, $1 \leq x \leq m$ and $1 \leq y \leq n$, represents a small square area with center at (x, y) and the image has m and n pixels along axes. The three-dimensional counterpart of pixel is voxel. A voxel with location (*x*, *y*, *z*) represents a small cube (or cuboid) with centroid at (*x*, *y*, *z*).

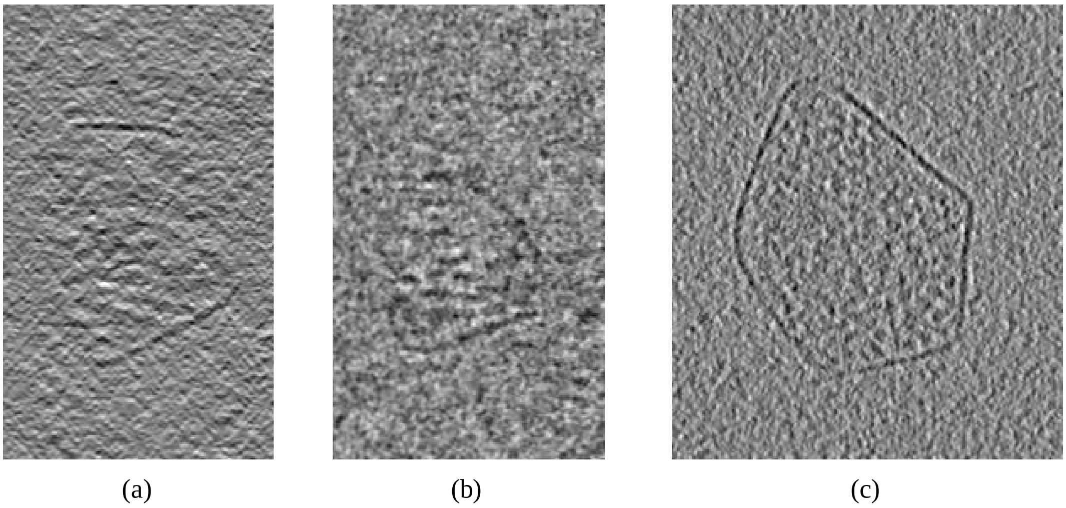


Figure 2.6: Slices from three slice sets of a metabolosome. (a) A slice from slicing along x-axis, (b) A slice from slicing along y-axis and (c) A slice from slicing along z-axis.

The 'trimvol' module of IMOD also displays the voxel size of the three-dimensional trimmed metabolosome images. It shows that the voxels in all reconstructed metabolosomes are uniform in size (i.e. perfect cube) with dimensions 9.6201 × 9.6201 × 9.6201 (in Å unit, 1 Å = 10^{-10} metre). The voxel size is mechanically determined through reconstruction algorithms in IMOD.

2.5 Image Segmentation

2.5.1 General Approaches

The surrounding cellular objects and noise prevent unobstructed visualization of trimmed metabolosomes in 3D. One potential solution to this problem is to extract the 3D metabolosome from its surroundings. This requires image segmentation, an operation where the image is compartmentalized into distinct meaningful regions. Segmenting a 3D object involves segmenting the object based on its boundaries from each of its 2D slices and stacking over the sequence of segmented regions again.

The segmentation procedures are broadly two types - algorithm driven or unsupervised image segmentation and manual. A number of algorithms are established for image segmentation; their review and evaluations are provided in [95] and in particular, Chapter 12 of [21] discussed the segmenting approaches used for cryo-electron tomographic images. Manual segmentation is the most subjective way of segmenting objects; but segmentation by manual contouring is still most popular and usually employed for biological microscopy images due to the extreme complexity and noise in electron tomographic images (see Chapter 12 of [21], [96] and Section 2.5.2).

2.5.2 Limitations of Automated Segmentation

Though algorithm driven segmentation procedures have advantages like reproducibility of the segmentation output, time efficiency and absence of manual segmentation errors, it also has huge disadvantages in metabolosome segmentation as explained below:

Noisy and Low Contrast Image

The efficiency of algorithm driven segmentation procedures greatly depends on image quality. The metabolosome images are noisy and with very low contrast. So detection of metabolosome boundaries may be erroneous. However, denoising procedures and contrast improvement steps may solve this problem to some extent, though the other problems stated below still affect the automated boundary detection efficiency.

Existence of other Cellular Inclusions

It is also seen that some of the slices contain other cellular inclusions (e.g. fibres) which create ambiguity about target metabolosome boundaries (Figure 2.7). An automated segmentation algorithm possibly will consider these inclusions as objects of interest and require manual elimination afterwards.

Existence of Partial Boundary

The bounding box defined for a metabolosome during trimming cannot generally exclusively create a partition for that metabolosome due to overlapping three-dimensional arrangements of the metabolosomes in the complete tomogram. Some of the slices may contain partial boundary of another metabolosome besides the complete boundary of the target metabolosome (Figure 2.7). An automated segmentation algorithm will consider these partial boundaries as objects of interests and require manual inspection for elimination.

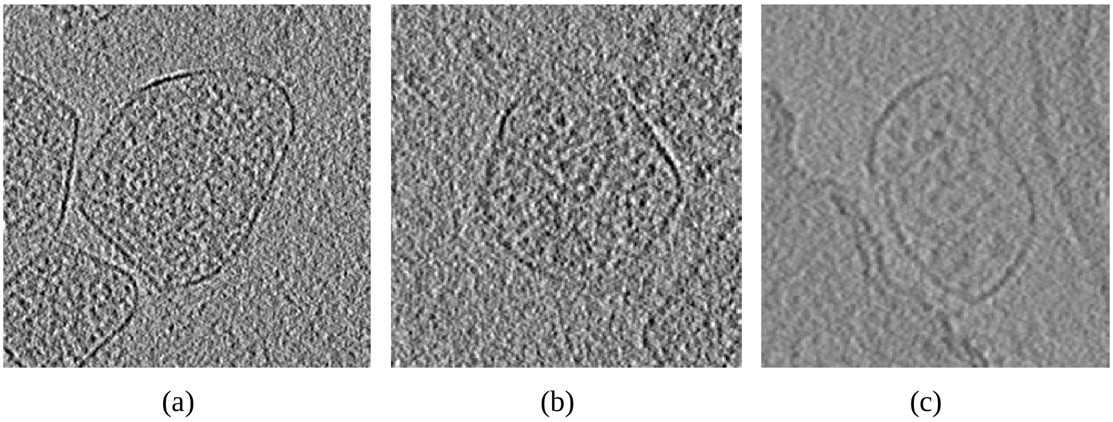


Figure 2.7: Complexities in metabolosome slices. (a) Partial boundaries of other metabolosomes. (b) Missing boundary. (c) Other boundary like inclusions in image.

Missing Wedge Problems

Due to the missing wedge problem (Section 2.2.5) the metabolosome boundaries are often not completely visible. Careful assessment is required to see how an automated segmentation procedure is handling those regions. Automated boundary detection algorithms may lead to poor segmentation for metabolosomes with partial boundaries. Figure 2.7 shows a few examples where these difficulties occurred.

2.5.3 Segmenting Metabolosome

These difficulties with automated segmentation led us to adopt manual segmentation. For similar reasons, Jensen et al. [10] also preferred a manual segmentation process when faced with a similar segmentation task for Carboxysomes. During manual segmentation, we superimpose a hand drawn boundary on each of the original object boundaries and collect the coordinates of those hand drawn boundaries. Figure 2.8 shows the manual segmentation process. The boundaries are drawn using MATLAB.

Since each image was sliced along three different directions, segmentation was carried out for three sets of slices for each metabolosome. For instance, if there is an image with dimensions of 200 × 200 × 200, segmentation is to be carried out for about 550 slices for that image (since few initial and end slices do not contain object boundaries).

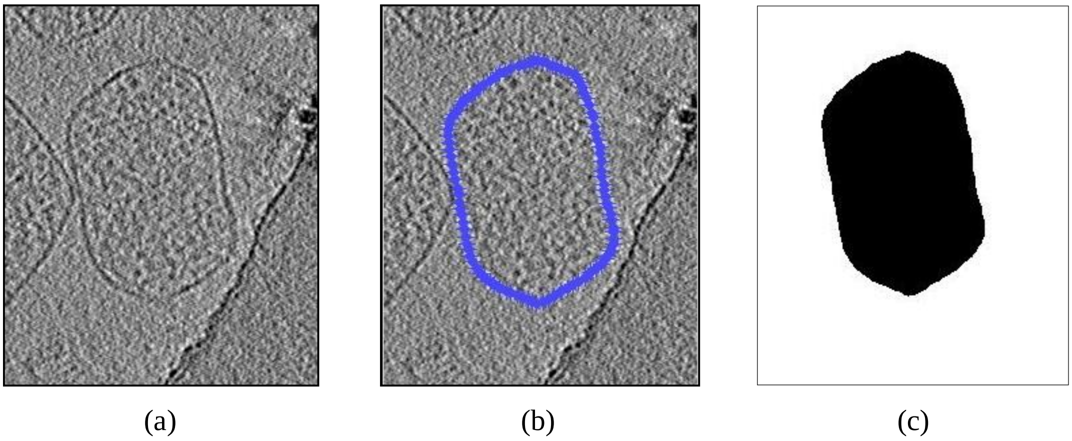


Figure 2.8: Manual segmentation process for metabolosomes. (a) A slice with complete boundary of a metabolosome, partial boundaries of other metabolosomes and a boundary like object. (b) Superimposed hand drawn boundary on the metabolosome boundary. (c) The slice with segmented metabolosome.

2.5.4 Data from Manual Segmentation

Boundary Data

The manual segmentation process creates a hand-drawn boundary superimposed on the image of the metabolosome boundary. This superimposed boundary is an approximately convex polygon defined by a set of cyclical ordered points as vertices of the polygon. We collected the coordinates of these points from *each* slice and used the collection of these boundary points obtained from *all* slices of a metabolosome to represent a collection of points sampled from the surface of that metabolosome.

Interior-exterior Identifier Data

If a slice contains the complete boundary of a target metabolosome, some pixels from this slice will be inside the boundary, some exactly on the boundary and the remaining are outside the boundary. But if a slice contains a partially missing boundary of the target metabolosome, some pixels from this slice will be in the missing regions too. We assume that:

1) The incomplete slice boundary is approximately a subset of a convex polygon.
2) All interior pixels of a metabolosome in a slice belong to that slice; even if there are missing boundaries (i.e. the corresponding bounding box contains the object entirely).

3) The pixels exactly on the boundary of a metabolosome are interior pixels of metabolosome.

We define,

S_z = set of all pixels from z^{th} slice of a metabolosome, z = 1, 2, …, number of slices.

H_z = corresponding hand-drawn boundary i.e. a convex polygon superimposed on the original metabolosome boundary in z^{th} slice during segmentation.

$S_z(x, y)$ = the pixel at (x, y) from S_z

I_z = set of pixels interior of H_z, i.e. (interior pixels)

E_z = set of pixels exterior of H_z, i.e. (exterior pixels)

M_z = set of pixels from missing area (missing pixels) in z^{th} slice

During manual segmentation we assign,

$$P_z(x, y) = \begin{cases} 1 & \text{if } S_z(x, y) \in I_z \\ 0 & \text{if } S_z(x, y) \in E_z \\ \text{NaN (missing value)} & \text{if } S_z(x, y) \in M_z \end{cases}$$

This gives with above assumptions,

$$I_z \cup E_z \cup M_z = S_z \quad \text{and} \quad I_z \cap E_z = E_z \cap M_z = I_z \cap M_z = \phi$$

and

$$P_z = \{P_z(x, y), \forall\, x, y\} \rightarrow S_z$$

Besides boundary data, we also collected $P_z(x, y)$ from each metabolosome slice. The process is repeated for all slices along the x-, y- and z-axes separately.

Segmentation in presence of Partially Missing Boundary

As discussed, some of the slices contained partially missing boundaries of the target metabolosome. This situation frequently occurs near the top and bottom slices of the metabolosomes and in the case of slices obtained from x-axis and y-axis slicing (due to the missing wedge problem). In this case, we superimposed the hand-drawn boundary *only* on the *visible* part of the boundary.

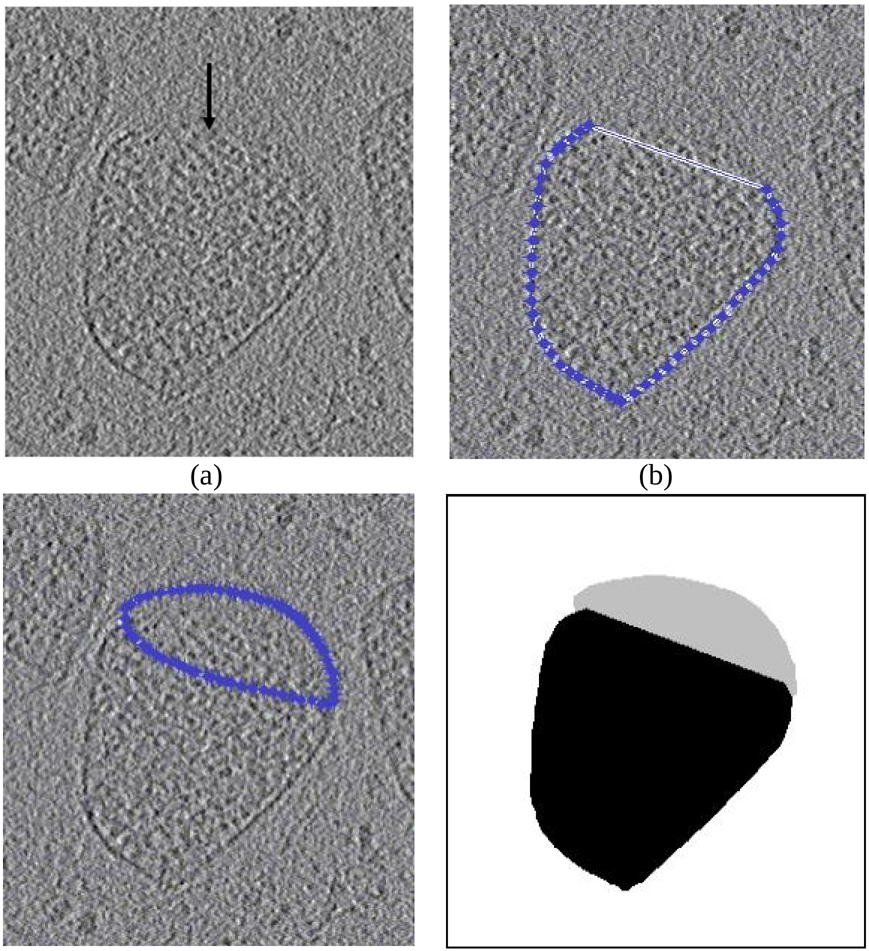


Figure 2.8a: Segmentation process for partially missing boundary. (a) A metabolosome slice with partially missing boundary, marked with an arrow. (b) Hand-drawn boundary superimposed only on visible part of the boundary. (c) Marking the missing region which includes missing part of the metabolosome with high probability. (d) The segmentation outcome: the black region contains interior pixels, white for exterior and gray for missing pixels.

Hence, this procedure generates partial boundary data only from the visible part of the object boundary. However, collecting the interior-exterior identifier data (Section 2.5.4) in these cases is difficult due to the following reason.

A missing region (M_z), by its nature, cannot be exactly marked with a boundary. Since,

$$I_z \cup E_z \cup M_z = S_z$$

and M_z cannot be clearly defined, I_z and E_z cannot be precisely determined as well. To solve this problem heuristically, assign,

$$P_z(x, y) = 1, \text{ if } S_z(x, y) \in C(H_z) \text{ where } C(H_z) = \text{convex hull of } H_z$$

The next step is to visually identify the regions such that these regions *may* include the missing metabolosome boundary with high probability. Define these regions as ${}_1M_z$, ${}_2M_z$, …, ${}_kM_z$, where k is the number of missing regions. We assign,

$$P_z(x, y) = 0 \text{ if } S_z(x, y) \in E'_z$$

where,

$$E'_z = S_z - (C(H_z) \cup {}_1M_z \cup {}_2M_z \cup \ldots \cup {}_kM_z)$$

and finally assign, $P_z(x, y)$ = NaN (= missing value) if:

$$S_z(x, y) \in S_z - (C(H_z) \cup E'_z).$$

Figure 2.8a shows the segmentation of an object with partially missing boundary.

2.5.5 Least Squares Method for Improved Segmentation

The segmentation data from the z-axis slices alone can generate the 3D segmented metabolosome. However, the segmentation data from slices along x-axis and y-axis may consist of exclusive information about metabolosome surface boundary. Hence combining segmentation data from slices along all three axes could result more precise segmentation. In addition, combining three sets of segmented data could also reduce the manual segmentation errors. We propose a least squares (LS) based approach to estimate the interior-exterior identifier value for each voxel utilizing the three-way segmentation data. Figure 2.10 gives a visual demonstration of improvements in segmentation using this approach.

Method

Using the same notation from section 2.5.4, let the pixel $P_z(x, y)$ corresponds the voxel at (x, y, z) in the the three-dimensional image. Let the true (unknown) interior-exterior identifier value for this voxel be $f(x, y, z)$. The purpose is to estimate $f(x, y, z)$ for all x, y, z. Let,

$f_x(x,y,z)$ = the voxel value at (x,y,z) when sliced through x-axis
$f_y(x,y,z)$ = the voxel value at (x,y,z) when sliced through y-axis
$f_z(x,y,z)$ = the voxel value at (x,y,z) when sliced through z-axis

$f_x(x,y,z)$, $f_y(x,y,z)$ and $f_z(x,y,z)$ values are obtained from segmentation as described in Section 2.5.4. We define,

$$\Delta^2 = \left(f(x,y,z) - f_x(x,y,z)\right)^2 + \left(f(x,y,z) - f_y(x,y,z)\right)^2 + \left(f(x,y,z) - f_z(x,y,z)\right)^2.$$

The least squares estimate of $f(x,y,z)$ is obtained by minimizing Δ^2, i.e. by solving

$$\frac{\partial \Delta^2}{\partial f(x,y,z)} = 0$$

which gives the least squares estimate of $f(x,y,z)$ as

$$\hat{f}(x,y,z) = \frac{1}{3}\left(f_x(x,y,z) + f_y(x,y,z) + f_z(x,y,z)\right).$$

Hence, each voxel gets a least squares estimated value utilizing information from all of three sets of slices. To eliminate the influence of the missing voxel values in calculations, we modified this estimate as follows: define,

$$f_i^*(x,y,z) = \begin{cases} f_i(x,y,z) & \text{if } f_i(x,y,z) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$n = |\{f_i(x,y,z) : f_i(x,y,z) = 0 \text{ or } 1\}| \text{ where } i \in \{x, y, z\}.$$

Finally, if $\hat{f}^*(x,y,z)$ is the adjusted LS estimate of $f(x,y,z)$, then

$$\hat{f}^*(x,y,z) = \sum_i \frac{f_i^*(x,y,z)}{n}.$$

We preferred LS method to combine three-way segmented outcomes, because it is simple to apply, easy to interpret and the outcome is satisfactory. However, some other methods (e.g. logistic regression) may be used for the same purpose.

The same voxel is considered three times during the three-way segmentation; hence it is reasonable to consider that $f_x(x,y,z)$, $f_y(x,y,z)$ and $f_z(x,y,z)$ are of equal relevance. Now, since the least squares estimate of $f(x,y,z)$ is the average of three voxel values, a voxel from three different directions receives equal weights and hence the LS approach is appropriate for this problem.

Visualizing Segmented Images

To convert the segmentation data to a 3D image, we define a three-dimensional matrix V such that, $V(x, y, z) = \hat{f}(x,y,z)$ $\forall$ x, y, z. For a particular z, say z_k, the set of voxels $\{V(x, y, z_k)\}$ $\forall$ x, y is the LS estimate of the pixel values of z_kth slice, while slicing along z-axis.

Converting these pixel values to grayscale, we generated the segmented LS estimated slice images. Finally this slice sequence was stacked (using AMIRA) to develop the three-dimensional metabolosome. Figure 2.10 shows a segmented and reconstructed three-dimensional metabolosome. Noticeably, some voxels, mostly near to missing boundary regions may receive fraction values (e.g. 1/2, 1/3, 2/3 etc.) during the LS estimation. These voxels will get a lower grayscale value in visualization.

2.6 Smoothing Metabolosome Surface

The segmentation of the metabolosomes using manual segmentation results in images like those shown in Figure 2.10. The images contain sharp striations due to the imprecise nature of the manual segmentation. This creates difficulties in identifying possible vertices and edges on the surfaces. Statistical smoothing [97], [98], [99] methods are applied to smooth the metabolosome surface, preserving the inherent sharpness of vertices and edges.

Most of the smoothing methods can be extended to work with three-dimensional data as well. We used one of the simplest smoothing methods - the simple moving average (see Chapter 2 of [99]).

2.6.1 Method

For a set of observation $\{x_1, x_2, \ldots, x_n\}$ from a random variable X, sequenced over time or proximity etc., the simple moving average of span k at l^{th} location is:

$$M_l = \frac{1}{k} \sum_{i=l}^{l+k-1} x_i$$

where $1 \le l \le n - k + 1$ and $1 \le k \le n$. We applied the three-dimensional form of this method, also known as box method for smoothing metabolosome surfaces. For the convolution kernel (k_1, k_2, k_3), k_1, k_2, k_3 are odd integers ≥ 3, define:

$$V_N\,(x, y, z) = \{V(n_1, n_2, \mathrm{n}_3)\}$$

such that,

$$x - \frac{k_1-1}{2} \le n_1 \le x + \frac{k_1-1}{2}, \quad y - \frac{k_2-1}{2} \le n_2 \le y + \frac{k_2-1}{2} \text{ and } z - \frac{k_3-1}{2} \le n_3 \le z + \frac{k_3-1}{2}$$

which is a set of all neighborhood voxels of $V(x, y, z)$ forming a cuboid of dimension $k_1 \times k_2 \times k_3$ such that $V(x, y, z)$ locates at the centroid of the cuboid. Then,

$$V_S(x, y, z) = \frac{1}{k_1 k_2 k_3} \sum_{n_1, n_2, n_3} V(n_1, n_2, n_3)$$

is the smoothed voxel value at x, y, z. This is also a simple average like the LS method applied during segmentation, but the LS method considered three values of the *same* voxel, but here averaging takes different adjacent voxels into account.

As input to the smoothing process, we provided the least squares estimate of all voxels and obtained new (smoothed) values of those voxels as output. For a particular value of z, say z_k, $V_S(x, y, z_k)$ is a 2D matrix. Converting this matrix to a grayscale image, we generated the z_k^{th} slice of the smoothed metabolosome. Stacking the slices sequentially generates the 3D smoothed metabolosome. Figure 2.9 shows a slice before and after smoothing.

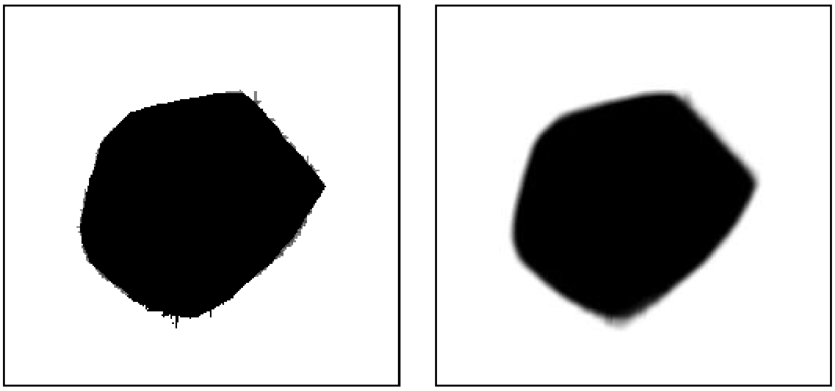
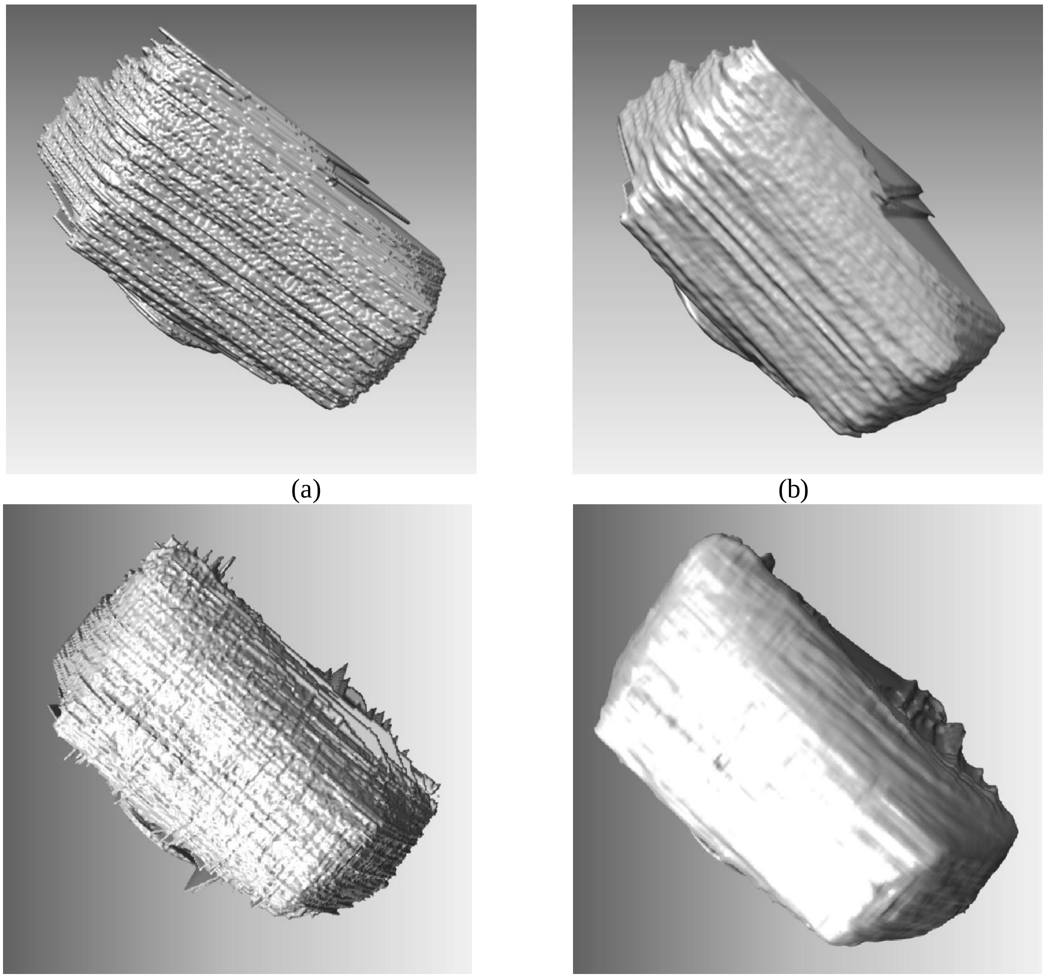


Figure 2.9: Effect of smoothing on individual slice. (a) A typical least squares reconstructed slice. (b) The same slice after applying smoothing.

2.6.2 Smoothing Parameter

For higher values of convolution kernel (k_1, k_2, k_3), the surface becomes smoother, but the vertices and edges lose their sharpness. If a vertex is sharp, it tends to contain less error than a blunt vertex when we manually record its location on the surface. On the other hand, if we do not smooth the surface, locating a vertex on surface may also be difficult. So, the parameter (k_1, k_2, k_3) is to be estimated such that it results in acceptable surface smoothness, preserving the satisfactory sharpness of the vertices and edges.



(a) (b)

(c) (d)

Figure 2.10: 3D segmented metabolosomes in different steps. (a) A 3D metabolosome reconstructed using segmented data only from slices along z-axis. (b) The smoothed form of (a). (c) The same metabolosome is reconstructed using segmentation data obtained from three sets of slices with least squares estimation. (d) The smoothed form of (b). Clearly, the image in (d) is smoother preserving sharpness.

If only one-way segmentation is considered instead of three-way, even the smoothed image contains 'ridges' on surface due to segmentation errors (Figure 2.10-b) which may cause difficulties in identifying edges. However, three-way segmentation and LS reconstruction solves this problem (Figure 2.10-d). Hence our proposed segmentation method works well for these images.

We started with the convolution kernel (k_1, k_2, k_3) = (3, 3, 3) which is the minimum possible value. Then several combinations of values for (k_1, k_2, k_3) are tested. For example, some of the combinations we considered are (3, 3, 5), (3, 5, 5), (3, 5, 3), (5, 3, 3), (5, 5, 7), (5, 3, 7) etc. Finally, we selected (k_1, k_2, k_3) = (3, 3, 5) and determined that this set of values performs satisfactory for all metabolosomes.

The performance was evaluated based on repeated visual inspections. Figure 2.10 illustrates a reconstructed 3D metabolosome in different steps. It also shows the effectiveness of LS approach and smoothing surface. One important observation is: for the best performing combination, k_3 is larger than k_1 and k_2. This is because the transition to a slice from its immediate previous slice (along z-axis) is not as smooth as consecutive points on any boundary from x-y plane. That is why clear 'ridges' on the surface are visible along z-axis (Figure 2.10(a)), but not along x- or y-axis. To achieve approximately same smoothness across three axes, k_3 is chosen larger.

2.7 Summary

The metabolosomes were imaged using cryo-electron tomography. Among 41 raw images reconstructed using IMOD, 16 reconstructed images were considered for future processing. 30 metabolosomes selected from six tomograms were segmented for further analysis. IMOD was used for tomographic reconstruction. Matlab and AMIRA were used for slicing and volume rendering.

A manual segmentation approach was preferred over algorithmic segmentation. We introduced a least squares based method and a three-way segmentation approach for improving manual segmentation. Statistical smoothing was used to reduce the roughness and segmentation errors from the three-dimensional objects. The tomographic reconstruction, trimming and slicing approaches are quite standard, used in many published research, e.g. [10] and here no validation approach was considered for these steps. The segmentation approach is new, and we justified this method (LS methods) theoretically as well as visually examining the final segmented smoothed images. The smoothing method is also very standard, and we justified the method for selecting the kernel for this data. The selections of other tuning parameters (such as tomogram thickness etc.) are justified in corresponding sections, hence no such rigorous validation study is required.

The three-dimensional objects from intermediate image processing steps were visualized through IMOD, Chimera UCSF and AMIRA. The following diagram (Figure 2.11) summarizes the reconstruction, segmentation and visualization approaches applied on the metabolosome raw images to generate final segmented images.

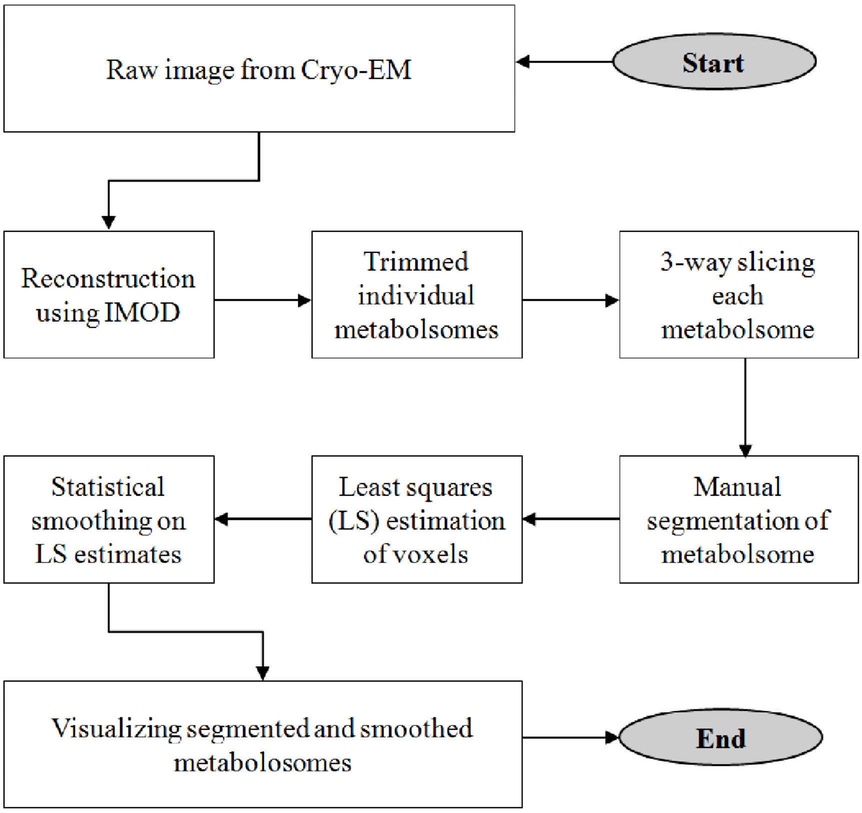

Raw image from Cryo-EM
Start
Reconstruction using IMOD
Trimmed individual metabolsomes
3-way slicing each metabolsome
Manual segmentation of metabolsome
Least squares (LS) estimation of voxels
Statistical smoothing on LS estimates
Visualizing segmented and smoothed metabolosomes
End

Figure 2.11: The summary of steps for three-dimensional tomographic image reconstruction, segmentation, smoothing and visualization.

Chapter 3

Polyhedron Families and their Properties

3.1 Introduction

Polyhedral shapes have been studied by geometers for thousands of years. Long ago, Kepler associated polyhedral shapes with a version of the solar system in his book *Mysterium Cosmographicum* [100], Plato allied them with earth, fire, air, water and ether in his dialogue *Timaeus and Critias* [101], [102], which he believed to be the basic elements of the world. The frequent appearance of polyhedral shapes in nature, architecture, art, cartography, even in philosophy and literature points to their importance in various fields of knowledge. Several books including [46], [103], [47] and [104] have depicted the evolution of polyhedral shapes and their presence in nature, ancient and modern society.

Numerous instances of polygons and polyhedra exist in nature, for instance, benzene molecules have a hexagonal arrangement of carbon atoms, starfish and petals of many flowers form pentagons, diamonds are octahedral, honeybees use the geometry of rhombic dodecahedra to form honeycombs, crystal pyrites may have dodecahedral shapes [45].

Scientists have also found reasons and utilities behind the shapes of these objects. For example, starfish with five arms exhibit the best performance with respect to detection, turning over, autotomy and adherence [105]; five petals actually give a flower optimum growth and stability [106]; the hexagonal structure minimizes the honeycomb wall perimeters and gaps among honeycomb cells [107], the structures of crystals depend on the internal symmetry of the crystal, and the relative growth rates along the various directions in the of the crystal [108], etc.

Modern developments in imaging technology reveal the 3D shapes of microscopic living elements and surprisingly some of them have well defined polyhedral shapes. As described in Chapter 41 of [13]: Structure and Classification of Viruses, many viruses show resemblance to polyhedral shapes, e.g. poliovirus, rhinovirus, and adenovirus have icosahedral shapes.

Bacterial microcompartments may also have polyhedral shapes; for instance, a recently identified microcompartment (named Carboxysome) has an approximately icosahedral shape [10]. One of the main purposes of this work is to identify the possible polyhedral structures of another bacterial microcompartment (called *pdu*-type microcompartment, Section 1.1.2, Chapter 1).

In this chapter we introduce standard polyhedral families, along with their characterizing features. These characterizing features develop a shape descriptor for a polyhedron, named as the *polyhedron profile statistic*. Though, some of these features, for example, vertex types and counts, face types and counts (Section 3.4.1 and 3.4.2) etc. are previously studied in geometry as individual features [109], [110]; a few other features, such as, adjacency matrices for complete and incomplete polyhedra are introduced in this work. In addition, the *polyhedron profile statistic,* which is a combination of different features, is also used for the first time here to characterize complete and incomplete polyhedra.

3.2 Polytope and Polyhedron

A polytope is a geometric figure bounded by portions of lines, planes or hyperplanes. Two-dimensional polytopes are just polygons. To define a polygon formally, a *p-gon* is a circuit of p line-segments A_1A_2, A_2A_3, …, A_pA_1 joining consecutive pairs of p points A_1, A_2, …, A_p. The points are termed as *vertices* and the line segments are *sides* or *edges* of the polygon [47]. For a triangle, p = 3 and for a pentagon, p = 5 etc.

In this chapter and all subsequent chapters, we assume vertices of a polygon are coplanar.

A polyhedron (a polytope in $\mathbb{R}^3$) is a finite, connected set of plane polygons, such that every side of each polygon belongs also to just one another polygon, with the proviso that the polygons surrounding each vertex form a single circuit [47]. The polygons forming a polyhedron are referred to as the *faces* of the polyhedron, provided that all coplanar polygons with common sides or segments of sides are treated as a sole polygon, thus making a single face. The sides and vertices of the faces of a polyhedron are referred to as the *edges* and *vertices* of the polyhedron [102]. Figure 3.1 is a common polyhedron, known as cube.

Even though there are several other definitions of polytopes, polygons and polyhedra in modern algebraic geometry [111], the above classical definition of polyhedra (from Coxeter [47]) is adequate for our purposes.

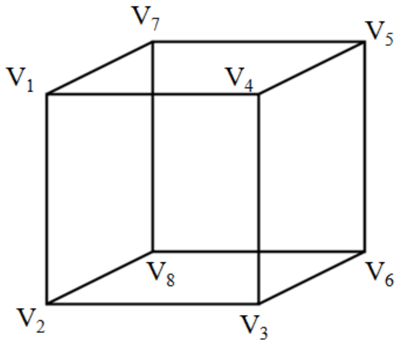


Figure 3.1: A common polyhedron - a hexahedron, also known as cube. The vertices are marked with V_i, i = 1, 2, …, 8. The straight line connecting V_1 and V_2 is an edge and the square with vertices V_1, V_2, V_3 and V_4 is a face.

Polyhedron as Graph and Solid

The three-dimensional objects are often visualized as their projections on a two-dimensional space. For example, an image of a body organ on a X-ray plate, the images taken by a digital camera, the images displayed in general digital displays etc. are the projection of three-dimensional objects on two-dimensional space. Clearly, a projection image contains less information than the original three-dimensional object since the projection image does not contain information along the third direction.

We consider a polyhedron as a geometrical object in the three-dimensional Euclidean space. Frequently, a polyhedron is visually expressed as a *planar graph*, i.e. a two-dimensional projection of its vertices and edges. This has the advantage of allowing all of the vertices, edges, and faces to be viewed at the same time. As in Chapter 1.1 of [102], however, it is sometimes more convenient to regard a polyhedron as a body, for instance when we speak about a point inside a polyhedron or about one polyhedron inside another. In such cases, we usually indicate that the polyhedron under consideration is a *solid*. In this work, these two notions are analogous and we refer to polyhedra as graphs and solids interchangeably.

3.3 Convex Polyhedron

A *convex polyhedron* is a polyhedron composed of finitely many planar polygons so that: (1) it is possible to pass from one polygon to another through polygons having common sides or segments of sides, and (2) the entire figure lies on one side of the plane of each constituent polygon.

It is the second condition that defines *convexity*; the first means that a polyhedron does not split into parts meeting only at vertices or even disjoint from each other [102]. The cube in Figure 3.1 is an example of a convex polyhedron. Many microscopic objects, for example, virus bodies [13] and Carboxysomes [10] are found to be convex.

Concave or non-convex polyhedra are those do not follow any of the above two conditions. There are several types of non-convex polyhedra too (e.g. star polyhedra) and they appear in nature as well, for example, gold nanocrystals [112].

3.3.1 Convex Polyhedron and Slice

Let P be a convex polyhedron and L be a plane in $\mathbb{R}^3$. The intersection, $I = P \cap L$ is a vertex, edge or face of P, or a closed curve (slice), depending on how L intersects P [113], [114]. Extending this notion, suppose L_1, L_2 ,…, L_n are *arbitrary n* parallel planes such that they are distinct and can intersect P simultaneously. Clearly,

$$P \cap \{L_1, L_2, \ldots, L_n\} = \{I_1, I_2, \ldots, I_n\}$$

where I_k is the k^{th} polygon, $k = 1, 2, \ldots, n$ and n is an arbitrary integer. Since in any convex polyhedron, the dihedral angles are less than 180°, I_k's are all convex polygons, for all $k = 1, 2, \ldots, n$, assuming I_k is not a vertex or edge.

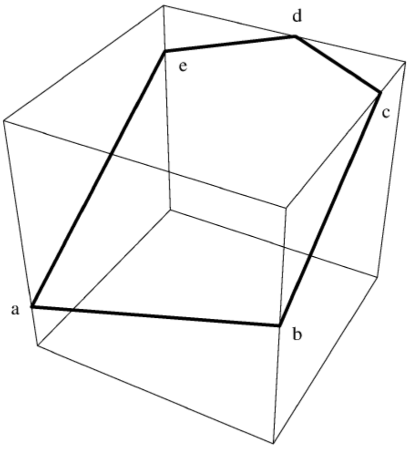


Figure 3.2: A plane intersects a cube and generates a pentagon with vertices at (a, b, c, d, e). Image is from O'Rourke [114].

Since L_1, L_2 ,…, L_n are *arbitrary*, this observation is invariant with respect to angle of intersection. Considering a polyhedron as a solid (Section 3.2), it is equivalent to think that parallel successive slicing of a convex polyhedron produces a sequence of convex polygonal slices. However, the converse statement may not be true always. In other words, if I_k is convex $\forall$ k = 1, 2, …, n from {I_1, I_2, …, I_n}, it *cannot* be said that P is convex. A simple way to determine whether a three-dimensional object is convex when all of its slices are convex could be to reconstruct the three-dimensional object by stacking back its slices and inspect visually. This is how we determined the convexity of the metabolosomes.

3.3.2 Convexity of Metabolosomes

We assume the metabolosomes are randomly oriented inside bacterial cells. While reconstructed 3D metabolosomes are sliced only along the z-axis (Section 2.4.2, Chapter 2), this is equivalent to slicing the metabolosomes along random axes. These slices are parallel to each other and the original three-dimensional object can be reconstructed just by stacking these slices in their original order. The slicing axis (also later mentioned as slicing direction) is the axis orthogonal to the slices.

Considerate and repeated visual inspections show that almost all resulting slices from all metabolosomes contain approximately convex polygonal boundaries. Though some slices contain metabolosomes with partial boundaries (for the missing wedge problem, Section 2.2.5, Chapter 2), the visible parts of those boundaries also suggest the same.

In addition, visual inspections also show that the segmented and smoothed 3D metabolosomes have convex polyhedral shapes (Figure 3.3). An earlier study of a similar bacterial microcompartment (Carboxysome) also revealed convex polyhedral shapes [10]. And further, another work [18] on the shapes formed by multi-component elastic membranes suggests that metabolosomes may have convex polyhedral shapes.

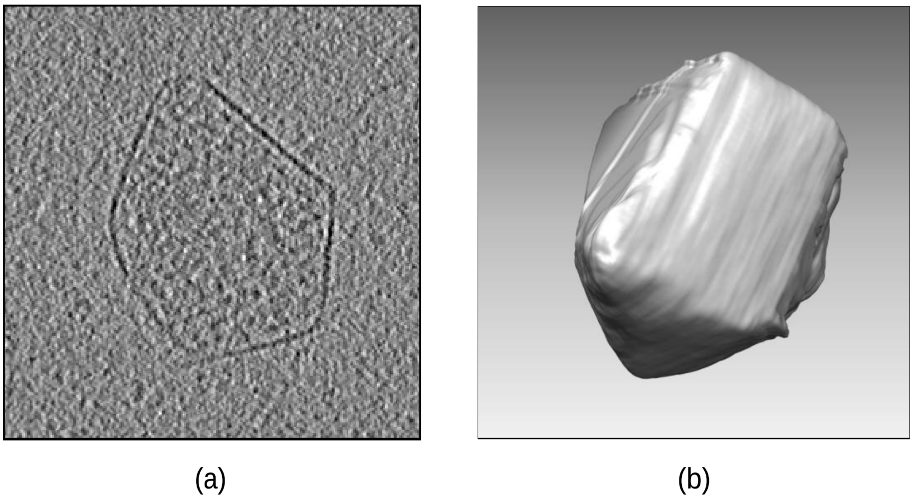


Figure 3.3: Convexity of metabolosomes. (a) A slice from a metabolosome shows convex polygonal shape. (b) A segmented and smoothed metabolosome is also shows convex shape.

Therefore, for the purpose of this analysis, only convex polyhedra were investigated. The words 'polyhedron' and 'solid' in subsequent chapters will imply convex polyhedron only.

3.4 Characterizing Polyhedra

3.4.1 Vertex, Edge and Face Counts

Number of Vertices: The vertex of a polyhedron is explained in Section 3.2. As notation, we define a set of all vertices in a polyhedron P as:

$V = \{V_i,\ i = 1, 2, \ldots, N_V\}$, $V_i = i^{\text{th}}$ vertex, N_V = total number of vertices.

Number of Edges: The edge of a polyhedron is also explained in Section 3.2. If there is an edge in P connecting vertices V_i and V_j, define the edge as:

$$E_{ij} = (V_i, V_j), i, j \in [1, N_V], V_i, V_j \in V$$

E_{ij} exists if and only if there is a graph from V_i to V_j or vice versa, such that there is no node (vertex) between V_i and V_j. Hence E_{ij} does not exist for some i and j. We define the set of edges as,

$$E = \{ E_{ij}, i, j \in [1, N_V] \text{ such that } E_{ij} \text{ exists}\}$$

then, the total number of edges is $N_E = |E|$ = cardinality of E.

Number of Faces: Consider a face F_i (a polygon) with n vertices $V_{i1}, V_{i2}, \ldots ,V_{in} \in V$ from P. We express F_i is a closed walk with no repetitions of vertices or edges are allowed, i.e.

$$F_i = (D_i, A_i), \text{ where } D_i = \{V_{i1}, V_{i2}, \ldots ,V_{in}\} \text{ and}$$

$$A_i = \{(V_{i1}, V_{i2}), (V_{i2}, V_{i3}), \ldots , (V_{in-1}, V_{in}), (V_{in}, V_{i1})\}$$

where (x, y) means a directed path from x to y. For simplicity, when a face F_i is expressed as $F_i = \{ V_{i1}, V_{i2}, \ldots ,V_{in} \}$, it simply means the face has i_n vertices and V_{i1} is connected with V_{i2}, V_{i2} is connected with V_{i3} and so on; lastly V_{in-1} is connected with V_{in}. The set of all faces,

$$F = \{F_i, i = 1, 2, \ldots, N_F\}, \text{ where } N_F \text{ is number of faces in } P.$$

For example, the cube in Figure 3.1 has $N_V = 8$, $N_E = 12$ and $N_F = 6$. N_V, N_E and N_F are the simplest features of a polyhedron. Though these features are insufficient alone to uniquely identify a polyhedron, they are still important in characterizing a polyhedron (Chapter 5 and 6).

3.4.2 Vertex Type and Face Type

All vertices in P may or may not be the same. Some of the vertices may be connected with only three edges whereas remaining vertices are connected with five edges and this is how the edges creates a connected path through vertices and finally build a polyhedron. This connectivity information has importance in distinguishing the standard solids (discussed in subsequent sections). In geometry, a polyhedron feature called *vertex configuration* [109], [115], [110] captures this information. To characterize a vertex V_i, define,

$$V_i^E = |\ \{E_{ij},\ \forall\ j \in [1, N_V] \text{ such that } E_{ij} \in E\}\ |,\ \mathrm{i} = 1, 2, \ldots, N_V$$

i.e. the number of edges connected to V_i or equivalently the number of faces adjacent to V_i. The number of vertices with T edges =

$$N_V^T\ =\ \left|\left\{V_i^E\ :\ V_i^E\ = T, i\ = 1, 2, \ldots N_V\right\}\right|.$$

Besides N_V of P, the number of vertices of different types is also a characterizing feature. Any vertex of a polyhedron is common in at least 3 edges, however, among standard solids, a vertex can be a meeting point of at most 10 edges. Notably, there is no vertex common in exactly seven or nine edges from any standard polyhedron. If any vertex be common in seven or nine edges, the dihedral angle, symmetry properties etc. would not match with those from the standard polyhedral families. For computation, we express this feature as a vector:

$$N_V^* = (\ N_V^3,\ N_V^4,\ N_V^5,\ N_V^6,\ N_V^8,\ N_V^{10}\ )\ .$$

Similarly, all F_i's in P may or may not be same. Some faces are triangular; some are quadrilaterals or pentagons etc. So in addition to the number of faces (N_F), we take number of faces of different types as another feature into account. Define the number of *T-gonal* faces as,

$$N_F^T = |\ \{F_i,\ i = 1, 2, \ldots, N_F : |\ D_i\ | = T\}\ |\ .$$

The standard polyhedra may have triangular to decagonal faces and everything in between except heptagon and nonagon. Similar to N_V^*, we express this feature as,

$$N_F^* = (N_F^3,\ N_F^4,\ N_F^5,\ N_F^6,\ N_F^8, N_F^{10}).$$

For example, N_V^* for a cube (Figure 3.1) is (8, 0, 0, 0, 0, 0) and N_F^* is (0, 6, 0, 0, 0, 0). Importantly, *these features are invariant under deformation* (Section 3.6).

3.4.3 Adjacency Matrix

In geometry and graph theory, an adjacency matrix represents the connection pattern among vertices or nodes (Chapter 8.1 of [116]). Adjacency matrices for polyhedra are symmetric and the entries are values of an indicator function, defined as follows. Let $ADJ(P)$ = adjacency matrix for polyhedron P with N_V vertices.

$$ADJ(P)_{ij} = \begin{cases} 1 & \text{if } E_{ij} \text{ exists}, \; i, j = 1, 2, \dots, N_V \\ 0 & \text{otherwise} \end{cases}$$

This matrix not only carries information on vertex connection patterns, it also provides N_V and N_V^* values. However, this matrix depends on how vertices are numbered. We consider two other types of adjacency matrices for polyhedra, namely *edge adjacency matrix* and *face adjacency matrix*. For the cube in Figure 3.1, the adjacency matrix is:

	V₁	**V₂**	**V₃**	**V₄**	**V₅**	**V₆**	**V₇**	**V₈**
V₁	0	1	0	1	0	0	1	0
V₂	1	0	1	0	0	0	0	1
V₃	0	1	0	1	0	1	0	0
V₄	1	0	1	0	1	0	0	0
V₅	0	0	0	1	0	1	1	0
V₆	0	0	1	0	1	0	0	1
V₇	1	0	0	0	1	0	0	1
V₈	0	1	0	0	0	1	1	0

Edge Adjacency Matrix

In graph theory, an edge adjacency matrix shows which two edges are adjacent. As described in [117], [118] and [119], the edge-adjacency matrix of P is a square and symmetric matrix whose $(i,j)^{th}$ element = 1 if and only if the i^{th} edge is adjacent to j^{th} edge, i.e. if there is a common vertex in two edges and zero elsewhere.

But we define the edge adjacency matrix with a different notion. Each edge in a polyhedron has two terminal points (vertices). Here the edge adjacency matrix captures the vertex type information (Section 3.4.2) of those vertices. The edge adjacency matrix of a polyhedron P, $eADJ(P)$ is defined as follows:

Suppose $E_{mn} \in E$ is an edge connecting two vertices V_m and V_n. Their vertex types are V_m^E (= r) and V_n^E (= s) respectively, i.e. V_m is connected with r edges and V_n with s edges. We consider a matrix M^{mn} for this edge (E_{mn}) such that $M_{r,s}^{mn}$ = 1, and zero elsewhere, i.e. this edge E_{mn} is contributing a unit only at (r, s) location of M^{mn}. Then,

$$eADJ(P)_{r,s} = \sum_m \sum_n M_{r,s}^{mn}$$

However, ignoring the directed path in E_{mn}, it gives $M^{mn}_{r,s} = M^{mn}_{s,r}$. Using this observation, the edge adjacency matrix is converted as an upper triangular matrix. Thus, if:

$$L1_{s,r} = \begin{cases} eADJ(P)_{r,s} & \text{if } r > s \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad L2_{r,s} = \begin{cases} eADJ(P)_{r,s} & \text{if } r \leq s \\ 0 & \text{otherwise} \end{cases}$$

then, the final $eADJ(P)_{r,s} = L1_{r,s} + L2_{r,s}$ for $r \leq s$, zero otherwise. For example, the $(3,4)^{th}$ entry of $eADJ(P)$ is the *total* number of edges in P having vertex pairs at terminals, such that one of the vertices has 3 edges and other has 4 edges connected with it.

Since a cube (Figure 3.1) has 12 edges and all edges have terminal vertex pairs both connected with 3 edges, the $eADJ(P)$ for the cube is as follows:

	V$_1$	**V$_2$**	**V$_3$**	**V$_4$**	**V$_5$**	**V$_6$**	**V$_7$**	**V$_8$**	**V$_9$**	**V$_{10}$**
V$_1$	0	0	0	0	0	0	0	0	0	0
V$_2$	0	0	0	0	0	0	0	0	0	0
V$_3$	0	0	**12**	0	0	0	0	0	0	0
V$_4$	0	0	0	0	0	0	0	0	0	0
V$_5$	0	0	0	0	0	0	0	0	0	0
V$_6$	0	0	0	0	0	0	0	0	0	0
V$_7$	0	0	0	0	0	0	0	0	0	0
V$_8$	0	0	0	0	0	0	0	0	0	0
V$_9$	0	0	0	0	0	0	0	0	0	0
V$_{10}$	0	0	0	0	0	0	0	0	0	0

This matrix is also independent of numbering vertices. As described in subsequent sections, the $eADJ(P)$ is an efficient identifying feature of P.

Face Adjacency Matrix

In graph theory, the face adjacency matrix, $fADJ(P)$, shows which two faces are adjacent [120], [121]. We modified this matrix to include the *face type* data. The modified face adjacency matrix of a polyhedron P, $fADJ(P)$ is defined as follows:

Suppose $E_{mn} \in E$ is an edge common in faces F_m and F_n. F_m and F_n are *r-gonal* and *s-gonal* respectively. We consider a matrix Q^{mn} for this edge (E_{mn}) such that $Q^{mn}_{r,s} = 1$, and zero elsewhere, i.e. this edge E_{mn} is contributing a unit only at (r, s) location of Q^{mn}. Then,

$$fADJ(P)_{r,s} = \sum_m \sum_n Q^{mn}_{r,s}$$

Similar to *eADJ*(*P*), we construct *fADJ*(*P*) also as an upper triangular matrix. The reason for choosing upper triangular form of this matrix is discussed in the Edge Adjacency Matrix section. However, a lower triangular form is also equivalent for analysis. For example, the if there are *exactly* 3 edges in *P* such that the faces adjacent to each of those 3 edges are pentagonal and quadrilateral, then $fADJ(P)_{4,5}$ = 3. As described in subsequent sections, the *fADJ*(*P*) is also an identifying feature of *P*.

Since the maximum number of edges in a face and vertex are 10 in both cases, the dimension of *eADJ*(*P*) and *fADJ*(*P*) are set as 10×10. Sometimes, *eADJ*(*P*) and *fADJ*(*P*) are expressed as vectors, obtained by stacking the columns. So a *fADJ*(*P*) matrix (of size 10 × 10) gives a vector of length 100. Though the vector length is 100, some of the elements are always zero (for example, the first element of the vector, since no edge can have only one other edge connected with its terminals; it needs at least two more edges). Later, during calculations, these elements are not important.

3.5 Standard Regular Polyhedron Families

Convex polyhedra are classified into different families based on their characteristics, for example, with respect to the symmetry group [122], the rotation group [47], the face type and vertex type (Section 3.4.2), (Chapter 8 of [102]), [47] etc. Since the metabolosomes have deformed shapes (Chapter 4), we consider only the classifying features that are invariant under deformation (Section 3.6). Face type and vertex type are such two features invariant under deformation and these two features in combination generate three convex polyhedra families - Platonic solids, Archimedean solids and Johnson solids. In addition, we consider a group of dual solids.

3.5.1 Platonic Solids

As defined in [123], "A Platonic, or regular, solid is a polyhedron whose faces are identical regular polygons with all vertex angles being equal. There are precisely 5 Platonic solids, the tetrahedron, octahedron, cube or hexahedron, icosahedron and dodecahedron". The regular polyhedra are known as the 'Platonic solids' because the Greek philosopher Plato (427–347 B. C. E.) immortalized them in his dialogue *Timaeus* [101], [103]. The origin, appearance in nature, etc. of Platonic solids are depicted in Chapter 2 of [46], Chapter 1 of [47], and Chapter 3 of [120]. Figure 3.4 shows a Platonic solid - Icosahedron.

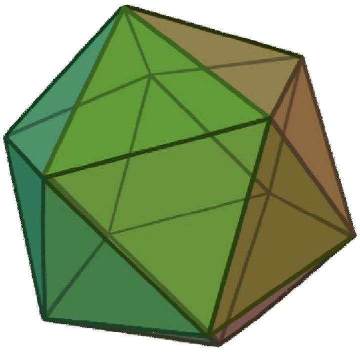


Figure 3.4: An Icosahedron - an example of Platonic solids. All faces are the same (triangles) and all vertices are identical (adjacent to five triangles). Image is from Wikipedia [124].

The above definition shows a platonic solid has the same number of faces that meet at each vertex and all faces are regular and congruent. In other words:

$$|\{N_V^T : N_V^T \neq 0 \ \forall \ T\}| = 1 \text{ and } |\{N_F^T : N_F^T \neq 0 \ \forall \ T\}| = 1$$

where T indicates edges. For example, for an icosahedron (Figure 3.4),

$$N_V^T = \begin{cases} 12 & \text{for } T = 5 \\ 0 & \text{Otherwise} \end{cases} \quad \text{and} \quad N_F^T = \begin{cases} 20 & \text{for } T = 3 \\ 0 & \text{Otherwise} \end{cases}$$

If *P* is a Platonic solid, important observations from *eADJ*(*P*) and *fADJ*(*P*) are:

$$eADJ(P)_{i,i} = \begin{cases} > 0 & \text{for only one } i \\ = 0 & \text{otherwise} \end{cases} \quad \text{and} \quad eADJ(P)_{i,j} = 0 \ \forall \ i \neq j$$

$$fADJ(P)_{i,i} = \begin{cases} > 0 & \text{for only one } i \\ = 0 & \text{otherwise} \end{cases} \quad \text{and} \quad fADJ(P)_{i,j} = 0 \ \forall \ i \neq j$$

where *i* = 3, 4, 5, 6, 8, 10.

Appendix 3.1 provides the N_V, N_E, N_F, N_V^* and N_F^* values for all Platonic solids. The following tables (Table 3.1a and Table 3.1b) show the face type and vertex type data for Platonic solids. As discussed above, in all cases the distribution is degenerated in a single point per solid.

Solid Name	**Number of Edges per Face (Face Type Data)**					
	3	**4**	**5**	**6**	**8**	**10**
Tetrahedron	4	0	0	0	0	0
Cube	0	6	0	0	0	0
Octahedron	8	0	0	0	0	0
Dodecahedron	0	0	12	0	0	0
Icosahedron	20	0	0	0	0	0

Table 3.1a: The face type data from the five Platonic solids.

Solid Name	**Number of Edges per Vertex (Vertex Type Data)**					
	3	**4**	**5**	**6**	**8**	**10**
Tetrahedron	4	0	0	0	0	0
Cube	8	0	0	0	0	0
Octahedron	0	6	0	0	0	0
Dodecahedron	20	0	0	0	0	0
Icosahedron	0	0	12	0	0	0

Table 3.1b: The vertex type data from the five Platonic solids.

3.5.2 Archimedean Solids

An Archimedean solid is a highly symmetric, semi-regular convex polyhedron consisting of more than one type of regular polygonal face, but meeting in identical vertices. The vertices are identical indicating that the same number of faces occurs in the same order about each vertex. Archimedean solids are distinct from the Platonic solids, as Platonic solids are composed of only one type of polygonal face (Chapter 3 of [120]).

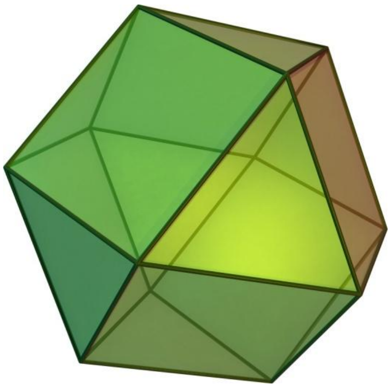


Figure 3.5: A cubeoctahedron - an example of Archimedean solids. All vertices are identical but there are two types of faces. Image is from Wikipedia [124].

Figure 3.5 is an example of an Archimedean solid, called a Cubeoctahedron, having N_V = 12, N_E = 24 and N_F = 14. It has two types of faces - 8 triangles and 6 quadrilaterals, though all vertices have same configuration (4 edges in each vertex with alternating triangle and quadrilateral). According to this definition of Archimedean solids,

$$|\{N_V^T : N_V^T \neq 0 \ \forall\ T\}| = 1 \quad \text{and} \quad |\{N_F^T : N_F^T \neq 0 \ \forall\ T\}| > 1$$

For example, a cubeoctahedron gives:

$$N_V^T = \begin{cases} 12 & \text{for } T = 4 \\ 0 & \text{Otherwise} \end{cases} \quad \text{and} \quad N_F^T = \begin{cases} 8 & \text{for } T = 3 \\ 6 & \text{for } T = 4 \\ 0 & \text{Otherwise} \end{cases}$$

If *P* is an Archimedean solid, *eADJ*(*P*) has same property as Platonic solids, i.e.

$$eADJ(P)_{i,i} = \begin{cases} > 0 & \text{for only one } i \\ = 0 & \text{otherwise} \end{cases} \quad \text{and} \quad eADJ(P)_{i,j} = 0 \ \forall\ i \neq j.$$

Solid Name	**Number of Edges per Vertex (Vertex Type Data)**					
	3	**4**	**5**	**6**	**8**	**10**
Truncated tetrahedron	12	0	0	0	0	0
Cuboctahedron	0	12	0	0	0	0
Truncated cube	24	0	0	0	0	0
Truncated octahedron	24	0	0	0	0	0
Rhombicuboctahedron	0	24	0	0	0	0
Truncated cuboctahedron	48	0	0	0	0	0
Snub cube	0	0	24	0	0	0
Icosidodecahedron	0	30	0	0	0	0
Truncated dodecahedron	60	0	0	0	0	0
Truncated icosahedron	60	0	0	0	0	0
Rhombicosidodecahedron	0	60	0	0	0	0
Truncated icosidodecahedron	120	0	0	0	0	0
Snub dodecahedron	0	0	60	0	0	0

Table 3.2a: The vertex type data from the 13 Archimedean solids.

However, the *fADJ*(*P*) from an Archimedean solid does not have the same property as for Platonic solids. There are exactly 13 Archimedean solids. Appendix 3.2 gives their names and N_V, N_E, N_F, N_V^* and N_F^* values. The above tables (Table 3.2a and Table 3.2b) show the face type and vertex type data for all Archimedean solids. As discussed above, in all cases the distribution vertex type data degenerate at a single point for each solid but this is not true for face type.

	Number of Edges per Face (Face Type Data)					
Solid Name	**3**	**4**	**5**	**6**	**8**	**10**
Truncated tetrahedron	4	0	0	4	0	0
Cuboctahedron	8	6	0	0	0	0
Truncated cube	8	0	0	0	6	0
Truncated octahedron	0	6	0	8	0	0
Rhombicuboctahedron	8	18	0	0	0	0
Truncated cuboctahedron	0	12	0	8	6	0
Snub cube	32	6	0	0	0	0
Icosidodecahedron	20	0	12	0	0	0
Truncated dodecahedron	20	0	0	0	0	12
Truncated icosahedron	0	0	12	20	0	0
Rhombicosidodecahedron	20	30	12	0	0	0
Truncated icosidodecahedron	0	30	0	20	0	12
Snub dodecahedron	80	0	12	0	0	0

Table 3.2b: The face type data from the 13Archimedean solids.

3.5.3 Johnson Solids

A Johnson solid is also a strictly convex polyhedron, each face of which is a regular polygon, but the solid is not uniform i.e. there is no requirement that each face must be the same polygon, or that the same number of polygons join around each vertex. In 1966, Norman Johnson published a list which included all 92 solids, and gave them their names and numbers [125] and later it was proved that the list is exhaustive [126].

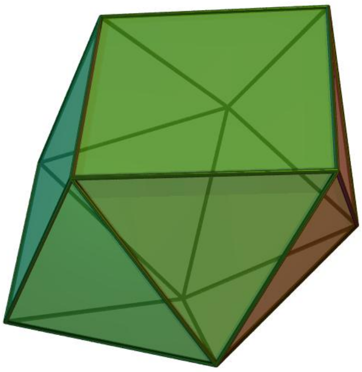


Figure 3.6: A Sphenocorona - example of Johnson solids. Image is from Wikipedia [124].

The difference between the Archimedean solid and the Johnson solid is in their vertex configurations. An Archimedean solid has all vertices identical but this is not so in a Johnson solid. The following image (Figure 3.6) shows a sphenocorona, the 86th Johnson Solid with N_V = 10, N_E = 22 and N_F = 14. It also has two types of faces: 12 triangles and 2 quadrilaterals. But the vertex configurations differ across vertices. The following inequalities follow from the definition of Johnson solids,

$$|\{N_V^T : N_V^T \neq 0 \ \forall \ T\}| > 1 \text{ and } |\{N_F^T : N_F^T \neq 0 \ \forall \ T\}| > 1$$

So a sphenocorona gives,

$$N_V^T = \begin{cases} 6 & \text{for } T = 4 \\ 4 & \text{for } T = 5 \\ 0 & \text{Otherwise} \end{cases} \quad \text{and} \quad N_F^T = \begin{cases} 12 & \text{for } T = 3 \\ 2 & \text{for } T = 4 \\ 0 & \text{Otherwise} \end{cases}$$

Appendix 3.3 gives the N_V, N_E, N_F, N_V^* and N_F^* values for all Johnson solids. The following tables (Table 3.3a and Table 3.3b) give the distributions of the face type and the vertex type data for 6 selected Johnson solids. These solids are especially selected, because an initial analysis (Chapter 5) predicted these shapes as the shapes of the metabolosomes. These shapes have maximum three types of vertices and two types of faces. As mentioned before, unlike Platonic solids, each solid has at least two types of faces and unlike Archimedean solids, each solid has at least two types of vertices.

Solid Name	**Number of Edges per Vertex (Vertex Type Data)**					
	3	**4**	**5**	**6**	**8**	**10**
Elongated pentagonal bipyramid	0	10	2	0	0	0
Biaugmented triangular prism	0	6	2	0	0	0
Metabidiminished icosahedron	2	6	2	0	0	0
Sphenocorona	0	6	4	0	0	0
Augmented sphenocorona	0	3	8	0	0	0
Sphenomegacorona	0	4	8	0	0	0

Table 3.3a: The vertex type data from 6 Johnson solids.

Solid Name	**Number of Edges per Face (Face Type Data)**					
	3	**4**	**5**	**6**	**8**	**10**
Elongated pentagonal bipyramid	10	5	0	0	0	0
Biaugmented triangular prism	10	1	0	0	0	0
Metabidiminished icosahedron	10	0	2	0	0	0
Sphenocorona	12	2	0	0	0	0
Augmented sphenocorona	16	1	0	0	0	0
Sphenomegacorona	16	2	0	0	0	0

Table 3.3b: The face type data from 6 Johnson solids.

3.5.4 Catalan Solids

In geometry, when the vertices of one polyhedron correspond to the faces of another polyhedron, the polyhedron pair is called dual (Chapter 3 of [120]). The dual of Platonic solids are again Platonic solids. Eugène Catalan [127] described first the dual of Archimedean solids, thus known as Catalan solids. So there are exactly 13 Catalan solids. The following figure is a Catalan solid - Tetrakis Hexahedron.

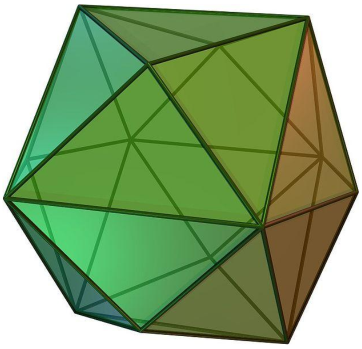
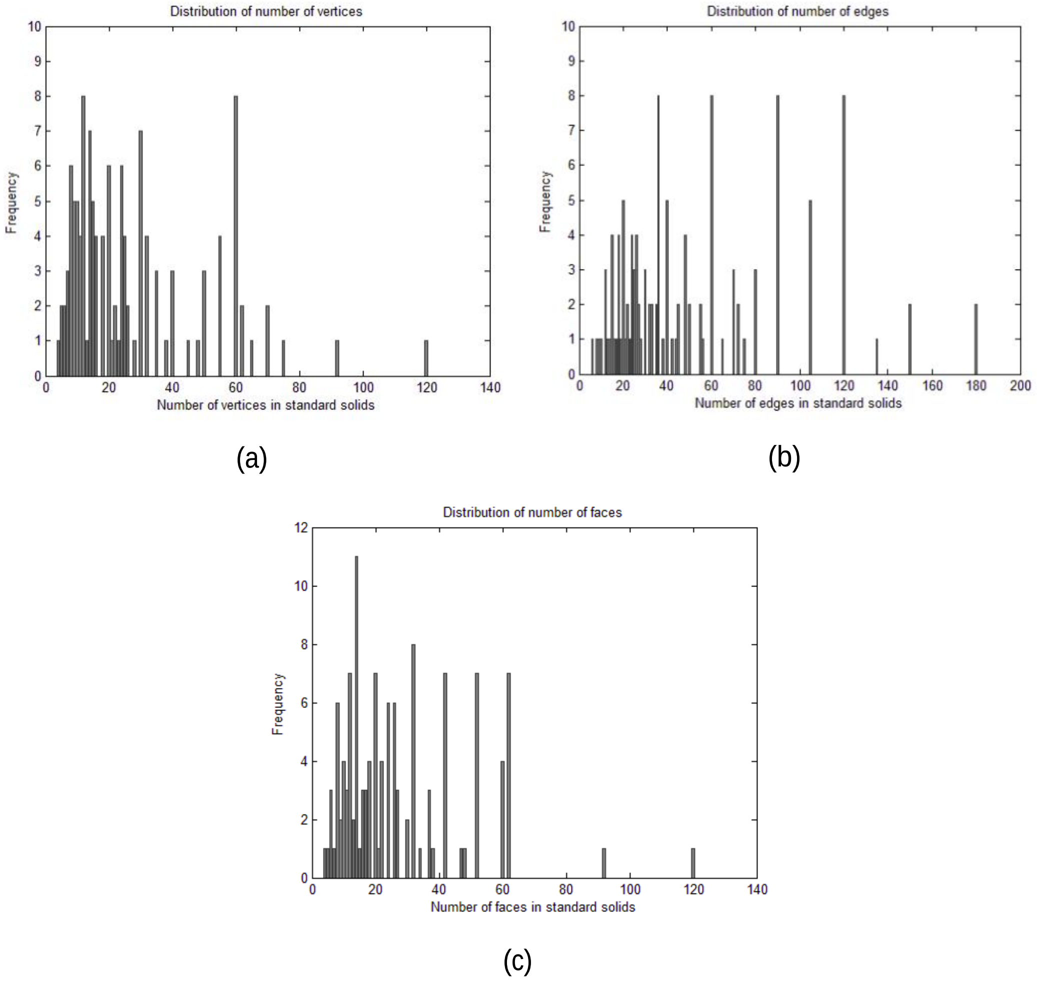


Figure 3.7: A Tetrakis hexahedron - an example of a Catalan solid. Image: Wikipedia [124].

There are a few other classes of polyhedra, for example, duals of Johnson solids, prisms and antiprism. However, visual inspections on the metabolosomes show they are very unlikely to belong to these remaining classes. So these 123 solids only from the four classes (5 Platonic, 13 Archimedean, 92 Johnson and 13 Catalan solids) are considered for further analyses. The following plots (Figure 3.8) show the distribution of three basic features (N_V, N_F and N_E) for these 123 solids. They show $N_V \in [4, 120]$, $N_F \in [4, 120]$ and $N_E \in [6, 180]$.

Two polyhedra can be considered closer to each other when they are from one family than they are from two different families when their face types and vertex types are considered only. But if the other characteristics are also considered, this family-based classification is not sufficient (for example, solids from each of these classes may have say, 12 vertices). So to quantify 'distance' between two polyhedra, a more sophisticated model has been developed in Chapter 5.



(a)

(b)

(c)

Figure 3.8: Distribution of vertex, face and edge counts from standard solids. (a) The number of vertices from the standard solids. (b) The number of edges from the standard solids. (c) The number of faces from the standard solids.

3.6 Polyhedron Profile Statistic

Section 3.4.1 and Section 3.4.2 describe that each standard polyhedron has N_V, N_F, N_E counts, 6 numbers for N_V^* and 6 numbers for N_F^*. For example, Table 3.4 shows these values for a Sphenocorona (Figure 3.6), the 86th Johnson solid.

Let us construct a vector of length 15 using these feature values in the following order:

$$(N_V, N_F, N_E, N_F^3, N_F^4, N_F^5, N_F^6, N_F^8, N_F^{10}, N_V^3, N_V^4, N_V^5, N_V^6, N_V^8, N_V^{10}).$$

This feature vector describes a polyhedral shape and we term it as the ‘*polyhedron profile statisti*c’. This statistic is used as a shape descriptor for all polyhedral shapes in subsequent analyses. For example, the polyhedron profile statistic for a Sphenocorona is (10, 14, 22, 12, 2, 0, 0, 0, 0, 0, 6, 4, 0, 0, 0).

Feature Class	**Features**	**Values**
Vertex, Face and Edge Counts	N_V	10
	N_F	14
	N_E	22
N_F^* (Face Type Data)	N_F^3	12
	N_F^4	2
	N_F^5	0
	N_F^6	0
	N_F^8	0
	N_F^{10}	0
N_V^* (Vertex Type Data)	N_V^3	0
	N_V^4	6
	N_V^5	4
	N_V^6	0
	N_V^8	0
	N_V^{10}	0

Table 3.4: The values corresponding to the characterizing features of Sphenocorona.

This profile statistic uniquely identifies a standard polyhedron from all 123 polyhedra; with the exception of 16 cases involving 38 solids (Appendix 3.5). The solids in each of these cases have the same profile statistics. However, this profile statistic combining with edge adjacency and face adjacency matrices (Chapter 6) can uniquely identify 123 solids except for only 8 pairs (16 solids) (Appendix 3.5). Among these 8 pairs, solids from 4 pairs can be differentiated based on their symmetry groups [122] and polyhedron net [128]. Due to deformation (Chapter 4), metabolosomes lose their symmetry properties, so the symmetry groups are not considered here.

However, these almost identical solids are very unlikely to appear as shapes for metabolosomes, since N_V for these solids is much larger than the maximum possible vertex counts from metabolosomes. The minimum number of vertices among these 8 pairs (16 solids) is 18 and maximum is 70 (Appendix 3.5), whereas we found that the metabolosomes have up to 12 vertices (Chapter 5 and 6).

3.6.1 Inter-feature Relationships

The parameters contributing to the polyhedron profile statistic are not all independent; rather multiple relationships exist among them. The most notable one is the relation among number of vertices, faces and edges, known as Euler's formula [129]. It states that,

$$N_V - N_E + N_F = 2$$

Additionally, two other trivial relationships of interest are:

$$N_F = \sum_T N_F^T$$

and

$$N_V = \sum_T N_V^T$$

Although some of the features are functions of other features, they are still important in the polyhedron profile statistic. These are discussed in Chapter 6.

3.6.2 Profile Statistic and Transformation

Transformations applied on a standard polyhedron generate another polyhedral shape that may or may not be another standard polyhedron. When the transformation does not generate any (1) new edge, nor (2) new vertex and (3) maintains convexity, we term the transformed polyhedron a deformed polyhedron. These transformations may be affine transformations [130] or non-affine also.

Undoubtedly, every polyhedron may generate infinitely many distinct deformed polyhedra based on different transformation functions. For example, a slightly flattened icosahedron (flattening deforms the icosahedron) also has $N_V = 12$, $N_F = 20$ and $N_E = 30$ and it is still convex, but unlike an ideal icosahedron (Figure 3.4), it has unequal edge lengths and unequal face areas. Also, infinitely many such deformed icosahedrons are possible. In nature, objects with elastic surfaces belonging to an environment with uneven distributed energy may be deformed [18].

Deformation Invariance

Many polyhedron characteristics, like symmetry, planar angle, or solid angle may be altered due to deformation. However, the *polyhedron profile statistic described above are invariant with respect to deformation* when the above three conditions are satisfied. The proof is given here:

Let P is a polyhedron and P* is its deformed form. The deformation satisfies the above three conditions. So, from the conditions:

$$N_V(P) = N_V(P^*) \text{ and } N_E(P) = N_E(P^*),$$

i.e. the number of vertices and edges remain unchanged. From Euler's formula, $N_F(P) = N_F(P^*)$, i.e. the number of faces also remains unchanged.

Let us consider the i^{th} edge in P is $E_i(P)$, $i = 1, 2, ..., N_E(P)$ and its corresponding edge in P* is $E_i(P^*)$. Assume $E_i(P)$ connects two vertices $V_m(P)$ and $V_n(P)$, and $V_m(P)$ is connected with *r* edges and $V_n(P)$ is connected with *s* edges, m, n = 1, 2, ..., $N_V(P)$. By condition (1), since no new edges are generated due to deformation, $V_m(P^*)$ is also connected with *r* edges and $V_n(P^*)$ is also connected with *s* edges.

Since the above is true for all m, n, the (r, s)th element of the edge adjacency matrix $eADJ(P)_{r,s} = eADJ(P^*)_{r,s}$. For the same reasons, the face adjacency also shows: $fADJ(P)_{r,s} = fADJ(P^*)_{r,s}$. Since r, s are arbitrary, it finally gives,

$$eADJ(P) = eADJ(P^*) \text{ and } fADJ(P) = fADJ(P^*).$$

Now, the number of triangles in P is:

$$\frac{1}{3}\,[2 \times fADJ(P)_{3,3} + fADJ(P)_{3,4} + \ldots + fADJ(P)_{3,10}]$$

Similarly, the number of quadrilaterals in P is:

$$\frac{1}{4}\,[2 \times fADJ(P)_{4,4} + fADJ(P)_{4,3} + fADJ(P)_{4,5} + \ldots + fADJ(P)_{4,10}]$$

and similar for other face types. Since $fADJ(P) = fADJ(P^*)$, the above two equations gives, $N_F^*(P) = N_F^*(P^*)$, i.e. face type data also remains unchanged due to deformation. Similarly it can be shown that, the vertex type data also remains unchanged. Hence the polyhedron profile statistic is invariant under deformation. So we conclude that, this polyhedral shape descriptor is suitable for analyzing *deformed polyhedra* as well.

However, the invariant property is *only* valid when the above three conditions are satisfied. However, for incomplete polyhedral shape classification (Chapter 6), we assume that these conditions are strictly followed, i.e. the metabolosomes shapes are the deformed shapes of the standard polyhedra and the deformation satisfied the above three conditions. However, a little relaxation on these conditions is allowed for the *polyhedral structural distance model* (Chapter 5) for shape prediction.

3.7 Data Collection

Drawing a polyhedron in $\mathbb{R}^3$ requires the coordinates of the vertices and the information about the directed path (Section 3.4.1) constructing each of the facets (face data). These two attributes are sufficient to generate all other required features of that polyhedron. For example, face data for the cube (Figure 3.1) is as follows:

$F = \{\{V_1, V_2, V_4, V_3\}, \{V_3, V_4, V_8, V_7\}, \{V_5, V_6, V_8, V_7\}, \{V_5, V_6, V_2, V_1\}, \{V_1, V_3, V_7, V_5\}, \{V_2, V_4, V_8, V_6\}\} = \{F_i, i = 1, 2, \ldots, 8\}$.

This information provides:

$$N_V = | F_1 \cup F_2 \cup \ldots \cup F_8 | \text{ and } N_F = | F |.$$

Consequently, following the notation from section 3.4.1,

$$F_1 = \{V_1, V_2, V_4, V_3\} \Leftrightarrow F_1 = (D_1, A_1) \text{ where}$$

$D_1 = \{V_1, V_2, V_3, V_4\}$ and

$A_1 = \{(V_1, V_2), (V_2, V_4), (V_4, V_3), (V_3, V_1)\} = \{E_{12}, E_{24}, E_{43}, E_{31}\}$ etc.

Since $E_{ij} = E_{ji}\ \forall\ i, j \in 1, 2, \ldots, 8$, $N_E = | A_1 \cup A_2 \cup \ldots \cup A_8 |$.

Similarly, the number of faces with 4 edges, for example:

$$N_F^4 = | \{F_i, \text{i} = 1, 2, \ldots, 8 \text{ such that } | D_i | = 4\} | = 6$$

and the number of vertices with 3 edges:

$$N_V^3 = \left| \{V_i^E : V_i^E = 3, \text{ i} = 1, 2, \ldots, 8\} \right| = 8$$

and so on. The vertex coordinates are useful for drawing polyhedra and with face data it facilitates calculating edge lengths, and face areas. The Mathematica [131] package ‘JohnsonSolids.m’ [132] provides the face data for all Platonic solids, Archimedean solids, Johnson solids and Catalan solids. Appendices 3.1 to 3.4 give the profile statistics for all solids considered.

3.8 Conclusions

There are 5 Platonic solids, 13 Archimedean solids, 92 Johnson solids and 13 Catalan solids considered as standard polyhedra for all subsequent analysis. Among many features of polyhedra, the features like vertex, face and edge counts, vertex type, face type and two adjacency matrices were recorded. Based on these features, a polyhedron profile statistic is proposed and data collected on it from all of these solids. This profile statistic is suitable for characterizing deformed shapes as well. An algorithm based on the polyhedron profile statistic is developed in Chapter 5 for predicting shapes for metabolosomes.

The polyhedron profile statistic is developed for convex polyhedra and in subsequent chapters we establish that this statistic is a good shape descriptor and also performs well for *incomplete* convex standard polyhedral shapes. This consideration is useful for this research problem particularly; but if the objects of interest were more complicated or belong to other polyhedron classes (for example concave polyhedra), this statistic may not perform well. In those cases, a few more characteristics might be required. This aspect has not been explored in this work and could be a potential direction for future research.

Chapter 4

Statistical Analysis of Fundamental Structural Properties

4.1 Introduction

Chapter 2 of this thesis describes the procedures for generating segmented and smoothed 3D objects (metabolosomes) from the raw images. This chapter starts with those segmented and smoothed metabolosomes. As discussed in Section 2.2.5 (Chapter 2), due to limited-angle single-axis tomography, the reconstructed metabolosomes suffer from the missing wedge problem. This chapter first explains the method we developed to solve this problem for polyhedral shapes.

The subsequent sections describe some fundamental structural properties of the metabolosomes, such as aspect ratio, volume, and deformation. The properties like aspect ratio, distribution of edge lengths and face area are *not* invariant with respect to deformation. So the analyses of these characteristics show whether the metabolosome shapes are deformed.

Another purpose of these analyses is to test if the metabolosomes show substantial discrepancies based on these structural parameters. The choice of shape prediction algorithms largely depends on these results (Chapter 5 and 6). These outcomes also have impacts from biological perspectives - they are discussed subsequently. However, this is the first ever study of these properties for these bacterial microcompartments.

4.2 Polyhedral Approximation of Metabolosome Shapes

4.2.1 Polyhedral Structure for Metabolosomes

The segmented and smoothed metabolosome were visualized using UCSF Chimera [92]. As displayed in Figure 4.1, the 3D views show each metabolosome has:

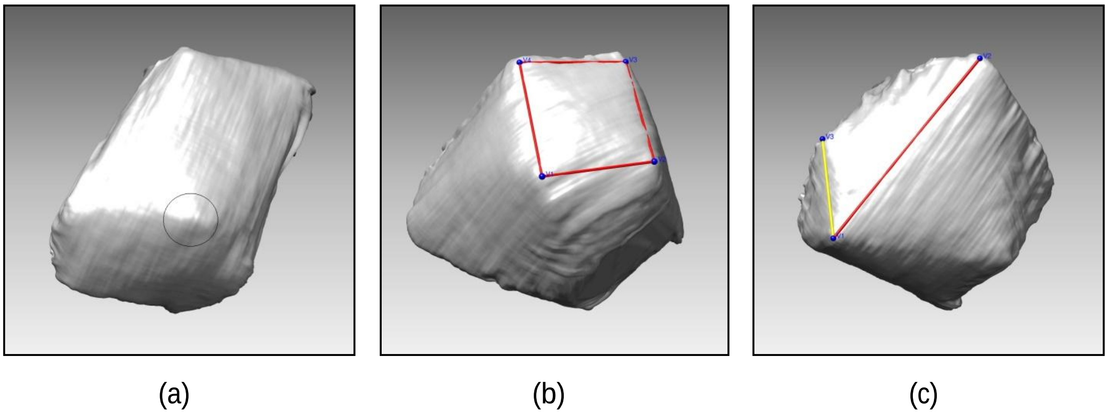
i) Some vertices.

ii) Few clearly visible complete and incomplete edges.

iii) Complete and incomplete facets; some of the facets are slightly curved.

iv) Some abrupt peaks (mostly at missing regions, due to segmentation errors).

Section 3.3.2 (Chapter 3) explained some reasons supporting the assumption that the metabolosomes have (convex) polyhedral shapes. In addition, the above observations (i) - (iii) also support some polyhedral models for the metabolosomes. A similar bacterial microcompartment called Carboxysome also has a convex polyhedral shape [10].



(a) (b) (c)

Figure 4.1: Vertices, faces and complete and incomplete edges in metabolosomes. (a) A clearly visible vertex, marked with a circle (b) A complete face (marked with vertices in blue, edges in red) (c) a complete (red) and an incomplete (yellow) edge.

However, not all microscopic biological objects fit to a polyhedral model. For example, the Rous sarcoma virus cores fit well with ellipsoid or cylindrical models [133], Rhabdoviruses has bullet shaped morphology [134]. Some recent studies proposed a few fullerene models [135] for viruses as well.

4.2.2 Drawing Incomplete Structures

Since the metabolosomes have missing regions due to the missing wedge problem (Section 2.2.5, Chapter 2), fitting a closed (or complete) polyhedron to a metabolosome is not possible at this step. So we superimposed an *incomplete* polyhedral structure on these *incomplete* metabolosomes (Figure 4.2). This has been accomplished in three steps, using Chimera UCSF [92].

Step 1: Identifying vertices visually and labeling them.

Step 2: Connecting identified vertices - gives *complete* edges.

Step 3: Identifying *incomplete* edges visually and labeling them.

For a metabolosome (M), an incomplete structure (IS_M) is defined as:

$IS_M = (V, IV, E, IE)$, where,

$V = \{V_i, i = 1, \ldots, N_V\}$ = set of all *visible* vertices, N_V = number of visible vertices.

$E = \{E_{ij}, i \neq j = 1, \ldots, N_V\}$ = set of all *complete* edges, E_{ij} connects V_i and V_j.

$N_E = | E |$ is the number of *complete* edges,

$IE = \{IE_i, i = 1, \ldots, N_{IE}\}$ = set of all *incomplete* edges where,

N_{IE} = number of *incomplete* edges.

An incomplete edge may have one or both terminals missing (only one edge is found from 30 metabolosomes where both of the terminals of one edge are missing). We termed the endpoints of an incomplete edge adjacent to missing regions as intermediate vertices (IV). If the edge is complete, the IVs would be just points on the edges.

Let, $IV = \{IV_i, i = 1, \ldots, N_{IV}\}$ = set of all intermediate vertices where N_{IV} is the number of intermediate vertices. Since the truncation of a polyhedron from top and bottom needs at least two vertices, and each vertex is connected with at least three edges, it immediately follows that $N_{IV} \geq 6$ and $N_{IE} \geq 6$. Here the V, IV, E and IE on IS_M were identified through repeated and careful visual inspections.

Reproducibility

Since the vertices, edges and hence faces are identified manually, reproducibility of the results are required to be considered. As mentioned in this section, the first step of drawing an incomplete structure for a metabolosome is to identify vertices. Some vertices are very sharp and clearly visible - these vertices are unlikely to be mistaken.

However, some vertices are not so sharp and due to segmentation errors, they may be surrounded by a few small 'peaks'. The existence of a vertex in such a case could be realized by visually analyzing the gradients of the adjacent faces, possibility of a meeting point of the adjacent visible edges etc. Though the vertices in these cases are very likely to be identified by any examiner, the coordinates of the marked vertex locations may differ to some extent across examiners. The coordinates are important indeed, but a small variation in the coordinate would not affect the overall shape of the object.

Once the vertices are identified, their connected edges are also identified. There may be some ambiguities about the existence of an edge on a curved face - this problem is solved by the principal component analysis (Section 4.3.3); so there is no issue with the reproducibility in these cases.

But one more reproducibility problem still exists - the problem with incomplete edge identification. An incomplete edge has only one visible vertex. To draw an incomplete edge, another random point must be chosen on the visible part of that edge and this point selection is very much subjective. However, this subjective selection may not have much effect on the overall structure as long as the incomplete edges are identified.

In conclusion, the manual structure drawing approach is reproducible to a great extent, thought the process may contain some operator effects. Since an algorithm driven approach is disadvantageous (Section 4.6) for this problem, we preferred manual vertex and edge identification.

Data from Incomplete Structures

From these superimposed incomplete polyhedral structures we collected the coordinates of V and IV along with information on their connectivity, completeness and the terminal nodes of E and IE. Chimera stores all of these data (file type: Python

[136], file extension: .py) for each structure. We collected these data from 30 segmented incomplete metabolosomes.

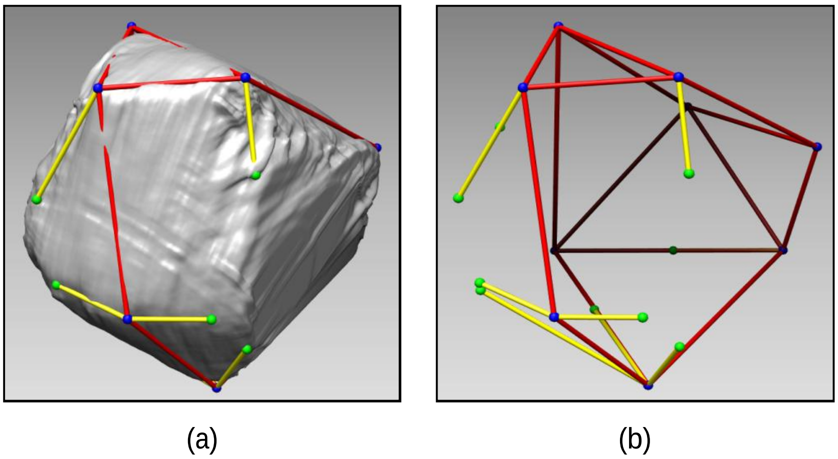


Figure 4.2: Superimposing an incomplete structure on a metabolosome. (a) A superimposed structure on a metabolosome. (b) The same structure without metabolosome. The complete edges are marked in red and incomplete edges are marked in yellow color.

4.3 Solving the Missing Wedge Problem

Section 2.2.5 (Chapter 2) describes the missing wedge problem in electron cryo-tomography. The existing method attempts to solve the missing wedge problem by the shape alignment and averaging (Section 1.2.2, Chapter 1) method. In this chapter, we demonstrate that the reconstructed metabolosomes do not satisfy the assumptions for this method. So we developed a new method to solve this problem. Section 4.3.1 and 4.3.2 describe this method.

4.3.1 The Problem in Computer Vision

The partially missing edges in a metabolosome are straight lines, but one or both ends of those straight lines are missing due to the missing wedge problem. Clearly, if the metabolosomes had no missing regions, the superimposed structures would not have any incomplete straight lines. So the goal is now to predict the complete polyhedron from the superimposed incomplete polyhedral structure corresponding to a metabolosome. This prediction requires 'completing' the incomplete structures, i.e.

finding all missing vertices, all completely missing edges and completing the partially missing edges.

Let us consider a similar problem in computer vision - a photograph of a three-dimensional object having straight edges (e.g. a cubic box). Suppose, due to noise or obstructions during imaging, some of the edges cannot be completely identified, i.e. the 2D image consists of imperfect (incomplete) 2D line data corresponding to the edges. Now the problem of interpreting the 3D object from the scene image from its imperfect 2D line data is a classical problem in computer vision. If we consider IS_M = (V, IV, E, IE) as a planar graph instead of a solid (Section 3.2, Chapter 3), our problem is now similar to this computer vision problem.

This problem in computer vision is approached through several algorithms over the years [137], [138]. One of the simplest methods is the INTERPRET program, where the problem is resolved by "fitting straight line segments to the edge points, extending these lines to form corners, identifying closed regions, and determining which closed regions constitute the background" [55]. However, even a modified simpler form of INTERPRET is adequate to solve our problem.

4.3.2 The Proposed Algorithm

The basic idea of our method is to extend the incomplete edges towards the missing regions and check if the extended edges *might* meet together, at least approximately.

Step 1*: *Incomplete Edge Identification

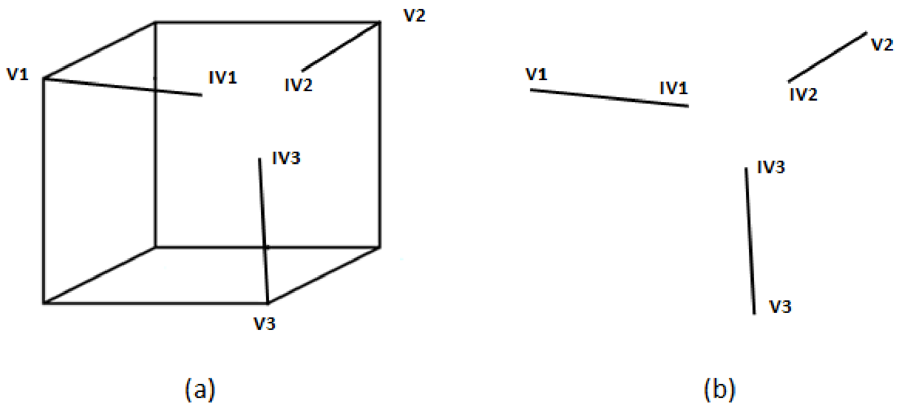


Figure 4.3: First step of the proposed algorithm for completing structures demonstrated in a cube. (a) Three incomplete edges which if extended, may meet with each other. (b) Three incomplete edges, separated from (a).

First, we visually identified three incomplete edges from IS_M which if extended toward the missing region, *might* meet together. Certainly, the meeting point would be in a missing region. We considered three incomplete edges, because a vertex in a polyhedron needs at least three edges to meet in 3D. A simulated scenario is provided in Figure 4.3.

Step 2: ***The extrapolation method for straight lines in 3D***

In this step, the incomplete edges were sufficiently extended towards the missing regions. This extension was carried out though extrapolation method. For simplicity, let us denote IE_1, IE_2 and IE_3 as 3 such incomplete edges and their terminal vertices are (V_1, IV_1), (V_2, IV_2) and (V_3, IV_3) respectively, $V_i \in V$ and $IV_\mathrm{i} \in IV$, i = 1, 2, 3 (as in Figure 4.3). Then we sampled a set of *ordered* points $\{p_{m1}, p_{m2}, \ldots, p_{mn}\}$ on the m^{th} incomplete edge [Figure 4.4 (b)], such that:

$$\mathrm{d}(p_{m1}, p_{m2}) = \mathrm{d}(p_{m2}, p_{m3}) = \ldots = \mathrm{d}(p_{m(n-1)}, p_{mn})$$

where, $\mathrm{d}(x, y)$ = Euclidean distance between x and y in $\mathbb{R}^3$. The value of n is chosen such that $\mathrm{d}(p_{m1}, p_{m2})$ is very small (e.g. we sampled 1000 points per nanometer). Based on these points on the m^{th} incomplete edge, another set of *ordered* points $\{q_{m1}, q_{m2}, \ldots, q_{mk}\}$ on the m^{th} incomplete edge was extrapolated toward the missing region [Figure 4.4 (c)], satisfying the conditions:

$$\mathrm{d}(q_{m1}, q_{m2}) = \mathrm{d}(q_{m2}, q_{m3}) = \ldots = \mathrm{d}(q_{m-1,k}, q_{mk}) \text{ and } \frac{k}{n} \geq 10.$$

Hence, the set $\{q_{m1}, q_{m2}, \ldots, q_{mk}\}$ are also equidistant points on the extended part of the m^{th} incomplete edge. The equidistance condition ensures the new path is 10 times longer than the existing (visible) path. At least 10 times length confirms the new path is long enough to go beyond the possible missing vertices locations.

Step 3: Proximity Measurement

After extension, the m^{th} incomplete edge now consists of two parts - the previously visible part and the extended part. In other words,

$$IV_m = \{p_{m1}, p_{m2}, \ldots, p_{mn}\} \cup \{q_{m1}, q_{m2}, \ldots, q_{mk}\}.$$

In this step we checked if these extended paths meet with each other. This test relies on the fact that if three straight lines in 3D intersect; they intersect at *exactly* one point. First we considered three points, one from each of the three incomplete edges, say q_{1i}, q_{2j}, and q_{3l} and then defined a distance among these three points:

$$D_{ijl} = \mathrm{d}(q_{1i}, C_{ijl}) + \mathrm{d}(q_{2j}, C_{ijl}) + \mathrm{d}(q_{3l}, C_{ijl}) \text{ where } C_{ijl} = (q_{1i} + q_{2j} + q_{3l}) / 3$$

where d(x, y) = Euclidean distance between x and y in $\mathbb{R}^3$. So D_{ijl} is the sum of the distances from the vertices to the centroid of the triangle [Figure 4.4 (d)] formed by $\{q_{1i}, q_{2j}, q_{3l}\}$.

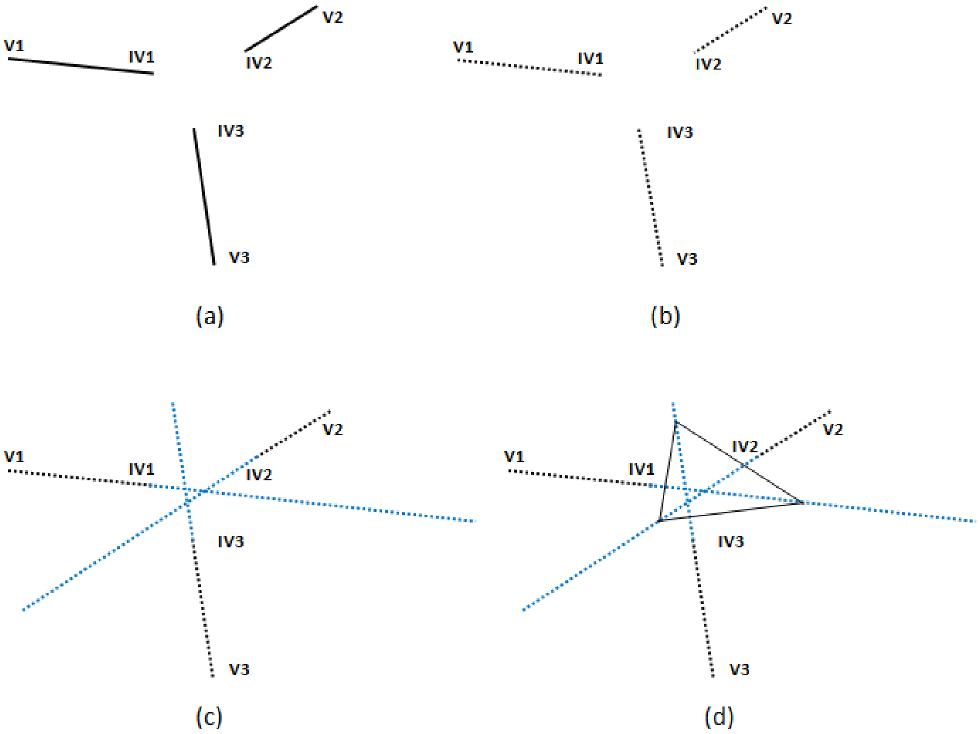


Figure 4.4: The edge extrapolation method. (a) The three incomplete edges from Figure 4.3. (b) Equidistant points are sampled on the incomplete edges. (c) The extrapolation method is applied to generate points on the extended edge (in blue) and (d) a triangle considered for calculating *D*.

Let $D = \{D_{ijl}, \forall\ i, j, l\}$, then $min(D) \approx 0$ if and only if the three extended edges *exactly* meet at one point. Since neither the vertices nor the edges are very sharp; there are segmentation errors and vertices were selected manually i.e. the measurements contain subjective errors, three extended edges of a metabolosome are unlikely to meet. So we set a small threshold value t, such that:

$$C_{ijl} \text{ is a possible vertex if } D_{ijl} \leq t.$$

For metabolosomes, this threshold value is set as 30 Å based on visual inspections.

Step 4: Justification

If any C_{ijl} is qualified as a vertex by Step 3, this step further justifies if it is a real vertex. Let C_{ijl} is a new vertex 'recovered' through extending incomplete edges. It would give three new 'complete' edges with terminal vertices (V_1, C_{ijl}), (V_2, C_{ijl}) and (V_3, C_{ijl}). C_{ijl} is finally considered as a vertex if:

$$\mathrm{d}(V_m, C_{ijl}) \leq \mu_l + 3\sigma_l \ , \forall m = 1, 2, 3$$

where, μ_l and σ_l are respectively the mean and the standard deviation of the *complete* edge lengths from the corresponding incomplete structure.

If X is a continuous random variable representing the edge lengths of a metabolosome, then by Chebyshev's inequality [139],

$$P(|X - \mu_l| \geq 3\sigma_l) \leq \frac{1}{9}$$

In other words, this limit ensures that at least about 90% of the extended edges will satisfy the condition in Step 4. During calculation, we found that none of the extended edges violated this limit. However, after the missing vertices are recovered and incomplete edges are completed, μ_l and σ_l are calculated based on *all* edges. We found that 12 edges from 12 metabolosomes (out of 30 metabolosomes) exceed the $\mu_l + 2\sigma_l$ limit. Hence, the choice of the $\mu_l + 3\sigma_l$ limit is not 'too liberal'.

Step 5: Completing the Incomplete Edges

We repeated Step 1 - 5 for all possible incomplete edge triplets and checked if this algorithm gives any new vertex.

Let the recovered vertices be ${}_RV = \{{}_RV_i, i = 1, \ldots, N_{RV}\}$, N_{RV} is the number of recovered vertices. Incorporating ${}_RV$ in previous .py file for IS_M (from Chimera UCSF) displays ${}_RV$ alongside M and IS_M. Finally, connecting ${}_RV$ with relevant visible vertices generates the completed edges corresponding to each IE_i, $i = 1, \ldots, N_{IE}$.

However, ${}_RV_i$ may be connected with more than 3 edges. To find other edges connected to it, first we considered all edges (${}_RV_i$, V_j), $i = 1, \ldots, N_{RV}$ and $j = 1, \ldots, N_V$. (${}_RV_i$, V_j) is considered an edge if it maintains the convexity of the structure and the coplanarity of the vertices of a non-triangular face.

4.3.3 Curved Facets Approximation

Section 4.2.1 explains that some metabolosomes have curved facets (Figure 4.5). This phenomenon raises ambiguity about the existence of possible edges on those curved facets. For example, a curved quadrilateral facet actually may consist of two adjacent triangles, or a curved pentagonal facet on a metabolosome may be actually a triangle adjacent with a quadrilateral. We used the principal component analysis to handle this problem.

Principal Component Analysis

The principal component analysis (PCA) [140], [141] converts a set of observations into another set of values of uncorrelated variables through an orthogonal transformation. Suppose that, X is a vector of p random variables x_1, x_2, …, x_p. The principal component of X is a set of another p variables (z_1, z_2, …, z_p) where,

$$z_i = a_i^{'}X = \sum_{j=1}^{p} a_{ij}\, x_j \text{ , } i = 1, 2, \ldots, p$$

where, a_i's are constants such that,

$$\mathrm{Var}(Z_1) \geq \mathrm{Var}(Z_2) \geq \ldots \geq \mathrm{Var}(Z_p) \text{ and } \mathrm{Cov}(Z_{i,} Z_j) = 0 \;\forall\; i \neq j,$$

Var(.) and Cov(.) are the variance and covariance of corresponding random variables respectively. The variances of the principal components measure different dimensions of the data along orthogonal directions. We used this notion to test coplanarity of a set of points in 3D.

Coplanarity and PCA

In geometry, a set of points are coplanar if they are on the same plane. Vertices of a triangle in 3D are always coplanar, but this is not necessarily the case for sets of four or more points. Also, due to segmentation and detection errors, it is very unlikely that all vertices of the n-gonal facets ($n \geq 4$) in a metabolosome are exactly coplanar.

For example, let a curved facet have a quadrilateral boundary defined with vertices $Q = (V_1, V_2, V_3, V_4)$, where, V_i's are the coordinates of the vertices in 3D and the variables here are x, y and z in the coordinate system (Figure 4.5). PCA on Q gives 3 principal components, say PC_1, PC_2 and PC_3 such that,

$$\mathrm{Var}(PC_1) \geq \mathrm{Var}(PC_2) \geq \mathrm{Var}(PC_3).$$

Var(PC_3) is null if (V_1, V_2, V_3, V_4) are *exactly* coplanar. For slightly curved facets,

$$\text{Var}\ (PC_1) \geq \text{Var}(PC_2) \gg \text{Var}(PC_3) \neq 0$$

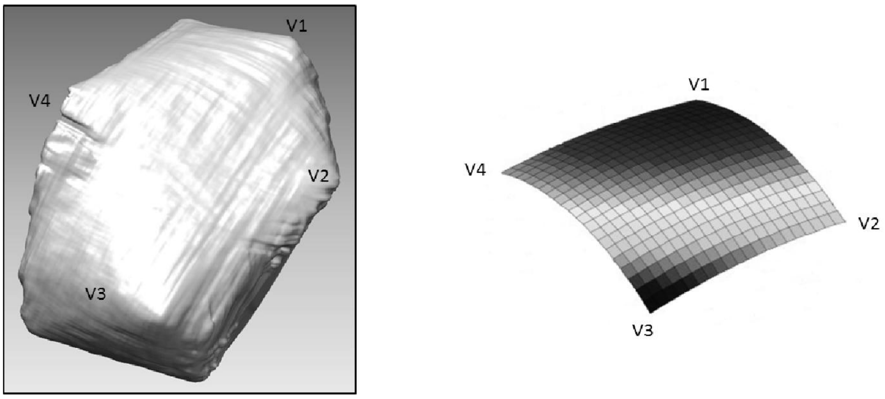


Figure 4.5: A curved facet in a metabolosome, marked with vertices (V_1, V_2, V_3, V_4) and its graphical presentation.

Since Var(PC_3) increases with curvature, we considered a coplanarity statistic (T_{cop}) as:

$$T_{cop} = \sqrt{\frac{Var(PC_3)}{Var(PC_1)}}$$

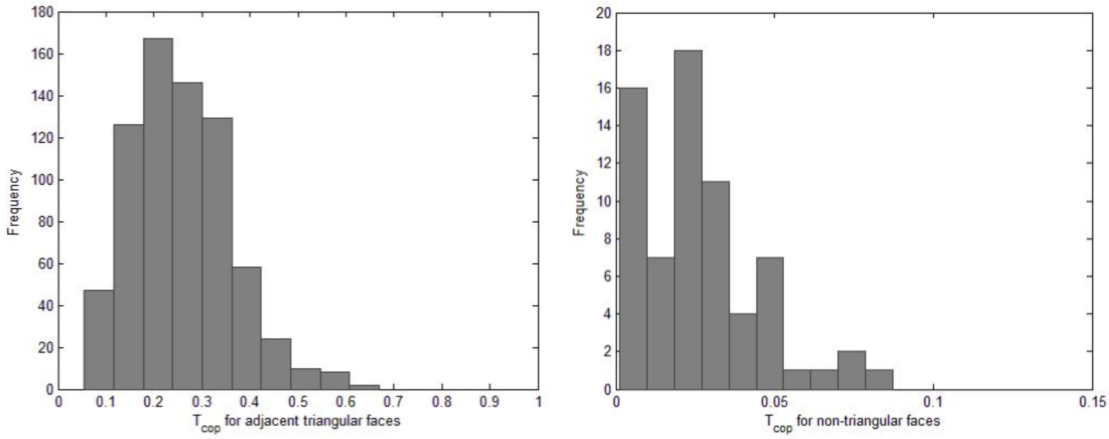
A rectangular curved facet is considered as two adjacent triangles if $T_{cop} \geq t$ where t is a threshold value, determined heuristically. First, for the few rectangular facets that were visually indentified, T_{cop} was calculated. Let $t_1 = max(T_{cop})$ for these cases. Next, the same procedure was repeated for several adjacent clear triangular faces; let $t_2 = min(T_{cop})$ for these cases. We found that $t_1 = 0.0491$ and $t_2 = 0.0548$ and selected t such that $t_1 \leq t \leq t_2$.

Results and Discussion

The threshold value (t) was set to 0.05. Since vertices were selected manually, and t largely depends on these coordinates, an insignificant relaxation on t was considered. Repeated and careful visual inspections identified only 6 cases where $T_{cop} \geq 0.05$ but two adjacent triangular faces constituted a quadrilateral. These values are 0.072, 0.052, 0.076, 0.087, 0.060 and 0.069. However, these are not too far from the considered threshold. The remaining 58 quadrilaterals from 30 metabolosomes satisfied $t_1 \leq t \leq t_2$.

Figure 4.6 (a) displays the histogram of T_{cop} where two triangles adjacent to an edge are not coplanar and Figure 4.6 (b) shows the same for T_{cop} calculated from the vertices of the non-triangular faces. These two plots justify that the threshold (t) well separates the adjacent triangle pairs from non-triangular facets.

The coplanarity statistic (T_{cop}) depends on the Var(PC_1) as well, i.e. the performance of this statistic depends on the span of the surface along the direction of its first principal component. In other words, this statistic is more sensitive for detecting an edge on a smaller surface than a larger one. However, no simulation study was carried out for comparing performance of this statistic for different types of faces.



(a) (b)

Figure 4.6: Distribution of the coplanarity statistics. (a) The distribution of T_{cop} from the cases where two adjacent triangular faces are not coplanar and (b) T_{cop} calculated for the non-triangular faces.

4.4 Data from Fitted Polyhedra

Section 4.2.2 describes the data from an incomplete polyhedron. The algorithm in Section 4.3.2 generates vertices in missing regions. The procedure in Section 4.3.3 decides if edges are possible on curved facets. Combining these together, we fitted a *complete* polyhedron for each metabolosome.

As described in Section 4.2.2 and Section 4.3.2, Chimera UCSF records information about the fitted polyhedron. Section 3.6 (Chapter 3) describes the *polyhedron profile statistic*. We extracted (1) the polyhedron profile statistic values, (2) the coordinates of vertices and (3) data on the directed path for each face from the completed polyhedron fitted to each of the metabolosomes from Chimera output.

Distribution of Fitted Polyhedron Features

The following plots (Figure 4.7 and Figure 4.8) displays the distribution of two important features - number of edges connected to each vertex and types of faces from metabolosomes. There may be some errors in structure drawing, manual segmentation etc. as discussed in Section 4.2.2; however, these plots accurately displays the structures we finalized.

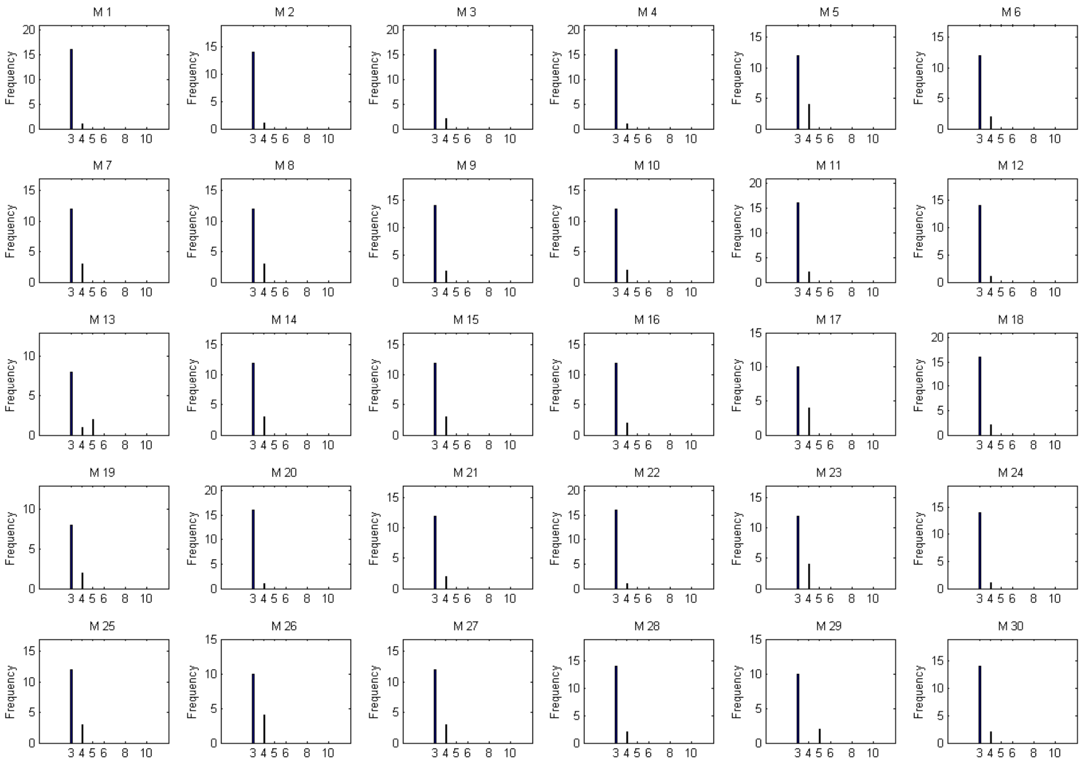


Figure 4.7: The distribution of face type data in metabolosomes. The 'M' stands for metabolosome in the plot.

Figure 3.6 (in Chapter 3) describes that the distribution of face type data for the Platonic solids degenerates at a *single* value. Figure 4.7 shows non-zero frequencies for more than one value which is clearly opposing the fact from the Platonic solids. Therefore, the metabolosomes do not have Platonic solids shapes.

Similar observations from the following plot (Figure 4.8) imply that the metabolosomes cannot be Archimedean solids. Figure 3.8 (from Chapter 3) shows the distribution of the vertex type data degenerates at a single point for the Archimedean solids. The vertex type data for the metabolosomes do not agree.

The Catalan solids differ from the Archimedean solids in terms of face and vertex type - a Catalan solid has only one type of face, but its vertex types may be different. So, metabolosomes cannot be Catalan solids as well, since a Catalan solid can have only one type of face.

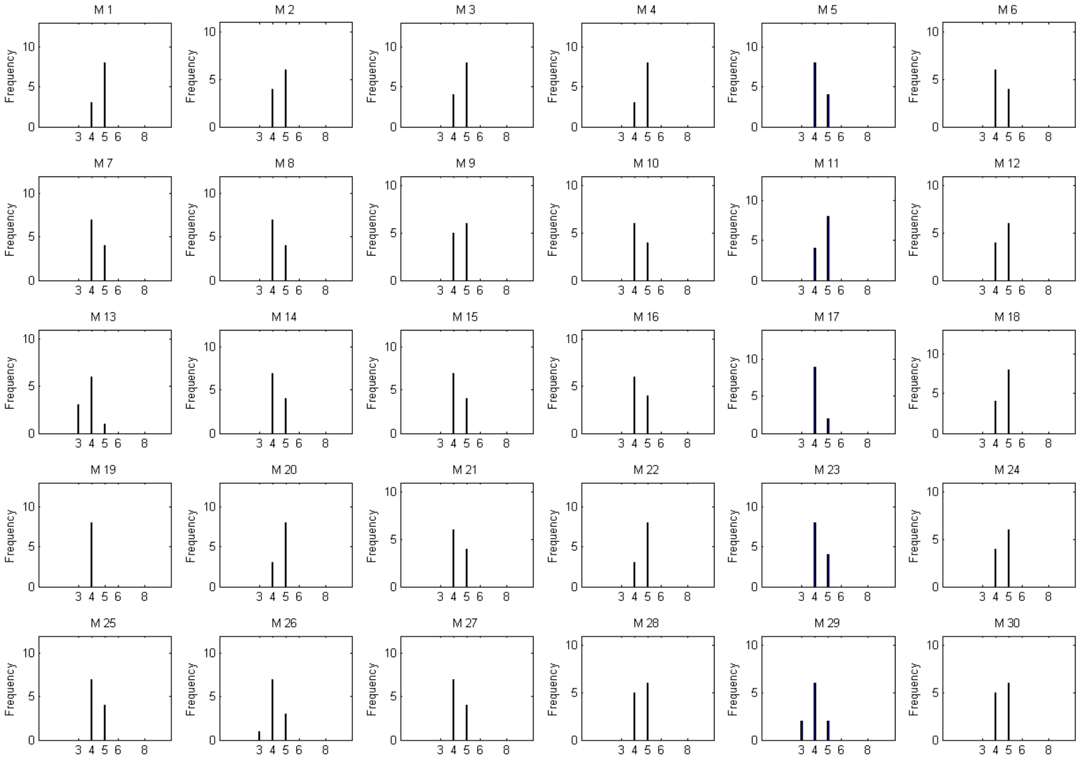


Figure 4.8: The distribution of number of edges connected to each vertex in metabolosomes. The letter 'M' stands for metabolosome in the plot.

4.5 Structural Properties of Metabolosomes

4.5.1 Volume

The metabolosome volumes were estimated using voxel-count and adjusted for missing wedges. When slicing a metabolosome (M) along the z-axis (Section 2.4.2, Chapter 2), we define slices as $S = \{S_i, i = 1, 2, \dots , N_s\}$, N_s = number of slices along the z-axis. Since top and bottom slices of the IMOD reconstructed trimmed images

(Section 2.4, Chapter 2) do not contain the object (metabolosome), let S_{start} = first segmented slice and S_{end} = last segmented slice, $1 \leq S_{start} \leq S_{end} \leq S_{Ns}$.

Recalling the *interior pixel* from Section 2.5.4 (Chapter 2), we denote the number of interior pixels in S_i as P_i, $S_{start} \leq i \leq S_{end}$. Since each slice is one pixel 'thick', and the voxels have uniform dimensions (9.62 Å × 9.62 Å × 9.62 Å), the number of voxels M_{vox} and the physical volume (M_{vol}) of M are estimated as:

$$M_{vox} = \sum_i P_i \quad \text{and} \quad M_{vol} = M_{vox} \times 9.62^3 \text{ Å}^3$$

Adjustment for Missing Wedge

Due to missing wedge problem (Section 2.2.5, Chapter 2), M_{vol} is slightly underestimated. To consider this downward bias, the sequence (P_i, $S_{start} \leq i \leq S_{end}$) has been extrapolated for ± 10 slices at both ends. Let (P_i*, $S_{start} - 10 \leq i \leq S_{end} + 10$) is the new sequence. The adjusted volume is:

$$M_{vol}(adj) = \sum_i P_i^* \times 9.62^3 \text{ Å}^3 .$$

Results

Figure 4.9 shows the distribution of $M_{vol}(adj)$ from the 30 metabolosomes. The minimum volume is 0.04 attoliter (1 attoliter = 10^9 Å^3), whereas the maximum is 3.35 attoliter with average volume of 1.15 attoliter, i.e. the metabolosomes show about 80-fold variability in their volumes.

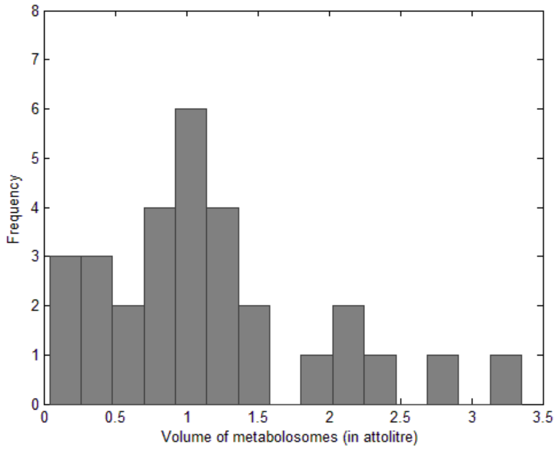


Figure 4.9: Distribution of volumes of the 30 metabolosomes considered for analysis.

The volume distribution is left-truncated since the metabolosomes smaller than 0.04 attoliter are excluded from segmentation and hence excluded from estimating volume distribution. The smaller objects are excluded because they are not good for segmentation. Larger objects have larger boundaries in slices and the larger the boundary, the more precise is the segmentation. In other words, the relative error of reconstruction is smaller for a larger object, assuming a fixed level of subjective errors for boundary drawing.

Cluster Analysis of Volumes

Cluster analysis is a method of grouping a set of observations in such a way that the observations within a group (cluster) are more alike than the observations from other groups [142]. There are several methods for clustering; we used the hierarchical clustering with Ward's minimum variance method [143] (package 'hclust' in R) [144] for metabolosome volumes. The cluster analysis on $M_{vol}(adj)$ shows (Figure 4.10) there may be three volume groups, though there are at least 2-fold within-group variations. The results have biological significance as discussed below.

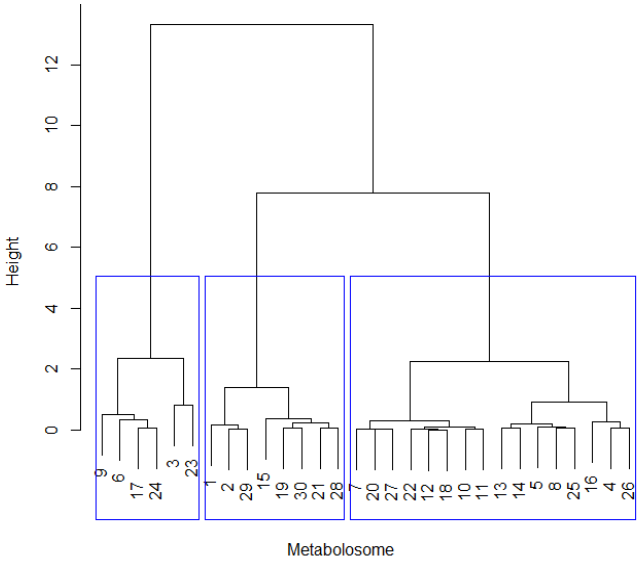


Figure 4.10: A dendrogram from cluster analysis of the volumes of the metabolosomes. The rectangles (in blue) show the volume groups.

Biological Significance

The volume analysis helps biologists to estimate the amount of proteins or inclusions inside a compartment. Biologists may also be interested to know the variability in relative proportions of building proteins across different volume groups.

The metabolosomes may have different shapes as well (Chapter 5). It is also of interest to analyze if particular shaped metabolosomes are more likely to be in a certain volume group than others and vice-versa which may help illuminating the assembly mechanism of proteins inside the compartments. We also found that, there is no clear relation between volume groups and predicted shapes i.e. all types of predicted polyhedral shapes (Chapter 5 and 6) appear in all volume groups.

4.5.2 Regularity

Edge Length for Deformed Polyhedron

Regular polyhedra like the Platonic solids, Archimedean solids or Johnson solids, all have regular faces, i.e. all faces have edges with equal lengths. Hence, one way of identifying deformation is to analyze the edge lengths distribution of metabolosomes. Recalling the notations from Section 3.4 (Chapter 3):

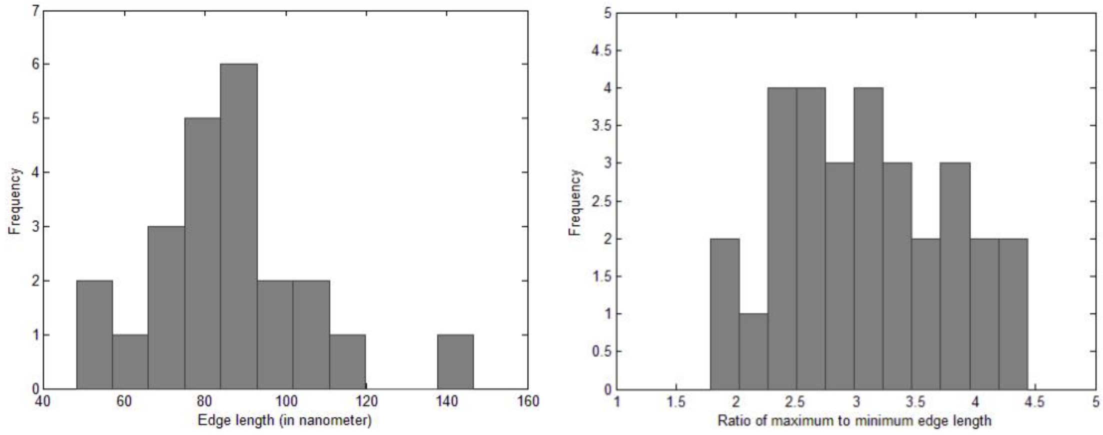
$V = \{V_i,\ i = 1, 2, \ldots, N_V\}$, $V_i = i^{\text{th}}$ vertex, N_V is the total number of vertices,
$E = \{E_{ij},\ i, j \in [1, N_V]\}$ = set of all edges and N_E = number of edges.
Let, EL_r = length of r^{th} edge, $r = 1, 2, \ldots, N_E$ and $L = Max_r\,(EL_r)$ and $l = Min_r\,(EL_r)$.

We considered the ratio $\frac{L}{l}$ to check for deformation. Clearly $\frac{L}{l} \geq 1$ with equality holding for regular polyhedral shapes. The edge length was calculated based on the data collected from fitted complete polyhedron to the metabolosomes (Section 4.4). While comparing the edge lengths within a single metabolosome, the ratio $\frac{L}{l}$ is useful. Since the same $\frac{L}{l}$ ratio could arise from very different distributions of edge lengths, however, it does not provide information about edge length distributions across metabolosomes. Appendix 4.1 provides some basic statistical measures on the edge lengths from individual metabolosomes.

Results

These measurements were calculated from *completed* metabolosome structures. The minimum of $\frac{L}{l}$ ratio is 1.7855 and maximum is 4.4306 with mean 3.0866, i.e. the longest edge of any metabolosome is about twice of the smallest one. *This observation suggests that the metabolosome structures are deformed.* Figure 4.11

shows the distribution of edge lengths for a single metabolosome and the distribution of $\frac{L}{l}$ for the 30 metabolosomes.

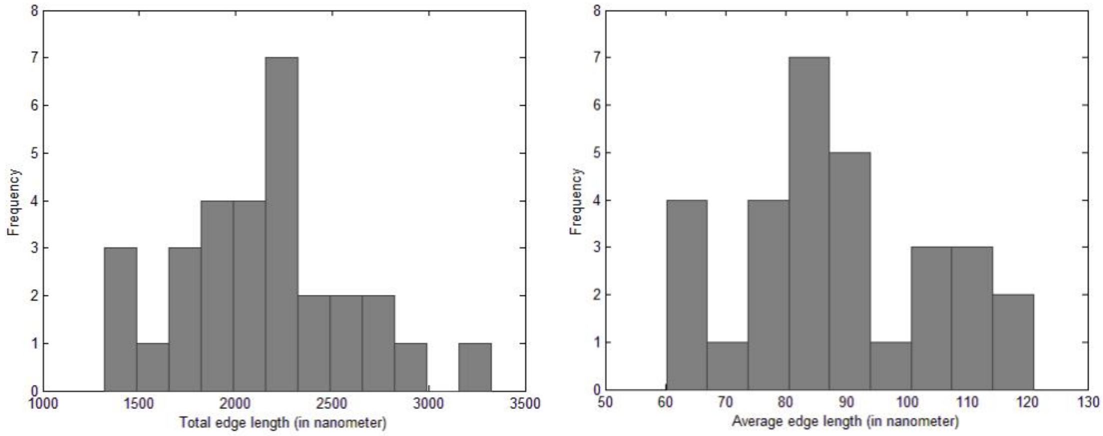


(a)   (b)

Figure 4.11: Checking deformation through edge length. (a) Distribution of edge lengths from a single metabolosome. (b) Distribution of edge length ratio from 30 metabolosomes.

Total and Average Edge Length

The above calculations also give total edge length and average edge length for each metabolosome. Let, TL_r = total edge length and μ_r = average edge length of r^{th} metabolosome, r = 1, 2, …, 30.



(a)   (b)

Figure 4.12: (a) Distribution of total edge lengths and (b) average edge lengths from 30 metabolosomes.

As provided in the table in Appendix 4.1, the Min(μ_r) = 62.53 (in pixel) and the Max(μ_r) = 125.76 (in pixel), which is about two times of the minimum average edge lengths. An ANOVA for comparing equality of μ_r is not appropriate here since some of the edge length distributions are not normal and the equality of variance assumption is not satisfied. The similar observation is found from TL_r as well. The Min(TL_r) = 1371 (in pixel) and Max(TL_r) = 3457 (in pixel) which is about 2.5 times of the minimum. Hence we conclude that the average and total edge length vary significantly (2- folds and 2.5-folds respectively) across metabolosomes. These results confirm that the metabolosomes are not regular polyhedra; they are rather deformed. Figure 4.12 shows the distribution of total and average edge length from 30 metabolosomes.

Biological Significance

Recent studies show that the multicomponent elastic membranes of bacterial inclusions can be shaped to the standard polyhedral structures [18], [145]. According to these models, the shell proteins contributing to the edges are stiffer than faces. Hence, total edge length may work as an estimated amount of those less elastic proteins. The distribution of average edge lengths across metabolosomes helps to understand the average amount of those proteins per edge.

Face area for Deformed Polyhedron

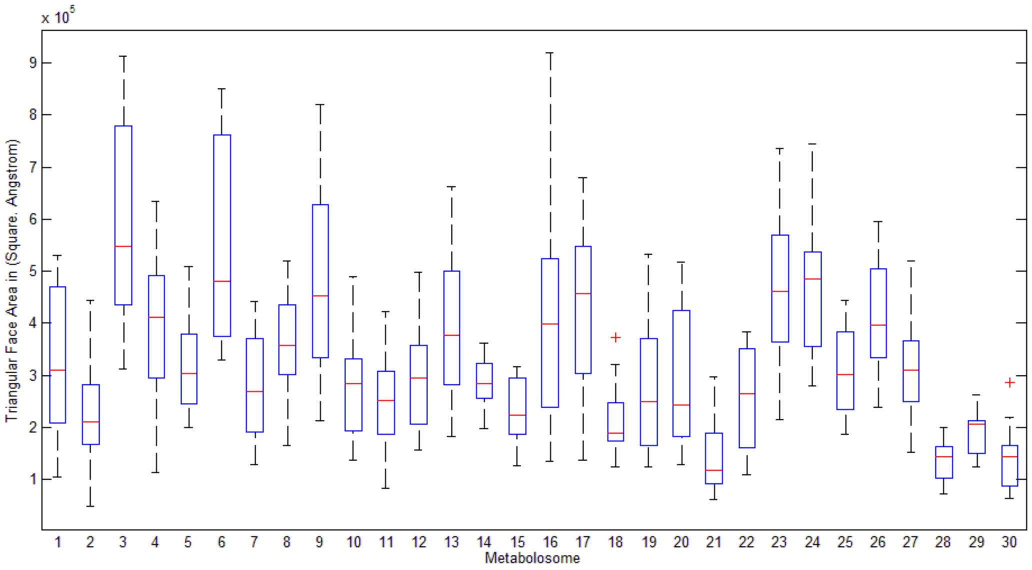


Figure 4.13: A box plot showing the distribution of the areas of all triangular faces from 30 metabolosomes.

Along with edge length, face area is also an indicator of deformation. Similar type faces in any non-deformed standard polyhedron have exactly equal area. For example, a sphenocorona (86th Johnson solid) has all (12) triangles with equal area and all squares (2) with equal area. Thus, similar shaped faces with different areas indicate the object is deformed, though the reverse may not be true.

Similar to edge length, the ratio of minimum to maximum (triangular) face area show it too varies within metabolosome. Figure 4.13 shows the distribution of triangular face area. Appendix 4.2 gives some basic statistical measures on the triangular face areas from 30 metabolosomes.

Biological Significance

Face area also has some impacts on the function of these microcompartments. The surface area to volume ratio is the area available to exchange metabolites vs. the amount of enzymes inside available to work on them. The total face area provides an estimate for the amount of shell proteins enveloping the compartments.

4.5.3 Aspect Ratio and Sphericity

The aspect ratio of a geometric shape is the ratio between its lengths in different dimensions, most commonly the ratio of its longest length to shortest length. For example, a square and a circle have aspect ratios 1:1; in landscape a rectangle has aspect ratio as width: height, for an ellipse it is the ratio of lengths of major to minor axis etc. The definition for multidimensional object is provided in [146].

An ellipsoid is defined with three orthogonal axes: one major and two minor axes. Let r_1, r_2 and r_3 are the lengths of these three axes such that $r_1 \geq r_2 \geq r_3$. If $r_1 = r_2 = r_3$, the ellipsoid turns out to be a sphere. Thus, a sphere has aspect ratio 1:1 and an ellipsoid has aspect ratio $\frac{r_1}{r3} > 1$. We estimate the r_1, r_2 and r_3 of the best fit ellipsoid to a metabolosome through PCA and hence test their sphericity.

Method

Recalling the Section 2.5.4 (Chapter 2), we collected the coordinates from the hand-drawn boundaries, sliced along the z-axis. For a metabolosome, let the set of all boundary points from the t^{th} slice (S_t) is $SBP_t = \{(SX_{ti}, SY_{ti}, SZ_{ti})\}$, i = 1, 2, …, number of points on t^{th} slice boundary, $S_{start} \leq t \leq S_{end}$. So, the set of boundary points (BP) from all slices, i.e. the collected surface points from visible metabolosome regions,

$$BP = \bigcup_{t = S_{start}}^{S_{end}} SBP_t$$

Suppose, BP_{PC1}, BP_{PC2}, and BP_{PC3} are the three principal components from *BP*, satisfying SD(BP_{PC1}) ≥ SD(BP_{PC2}) ≥ SD(BP_{PC3}) where SD(.) is the standard deviation. We use these standard deviations to estimate the aspect ratio of the fitted ellipsoid to the metabolosome. If the fitted ellipsoid has axes lengths r_1, r_2 and r_3 ($r_1 \geq r_2 \geq r_3$), then the aspect ratio:

$$\frac{r_1}{r_3} \approx \frac{SD(BP_{PC1})}{SD(BP_{PC3})}.$$

For convenience, we considered the inverse aspect ratio: $R_1 = \frac{r_3}{r_1}$.

The minimum and maximum of this ratio for the metabolosomes are 0.4251 and 0.8017 respectively with the mean of 0.6207. Notably, about 97% of the values are less than 0.75. In addition, the ratio:

$$R_2 = \frac{SD(BP_{PC2})}{SD(BP_{PC1})}.$$

is also calculated for these metabolosomes. The minimum, maximum and average of R_2 are 0.4989, 0.9343 and 0.8108 respectively. Figure 4.14 gives the distributions of these ratios from all metabolosomes.

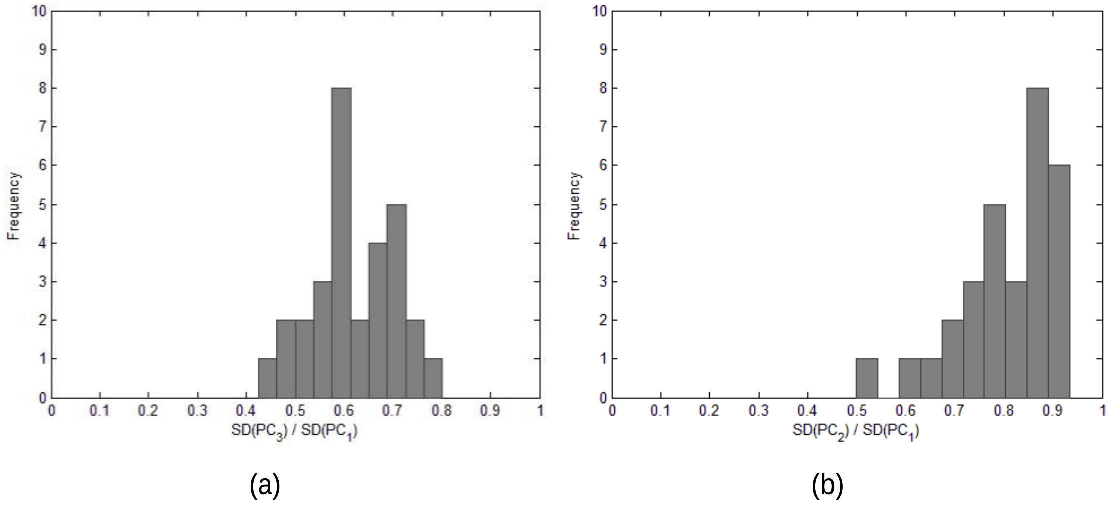


Figure 4.14: The distribution of the aspect ratios. (a) The distribution of R_1 and (b) the distribution of R_2, calculated from 30 metabolosomes.

However, the metabolosomes have missing regions and these ratios are from visible data only. The data from missing regions can affect this ratio but it would need a large increment in the ratio for the metabolosomes to be spherical. For example, the ratios below 0.70 must be increased by ≥ 42.86% to be approximately unity, which is unlikely. This result indicates most of the metabolosomes are non-spherical.

Biological Significance

The estimated length along the major axis is also an estimate of the maximum possible diameter of a metabolosome cross-section. The cross-section length may provide an estimate of the number of discrete molecules of specific sizes that can fit side-by-side. Cross-section and aspect ratio together may provide the 'thickness' of the compartment. This estimate along with other structural information provides assembly mechanism for molecules inside metabolosomes. More about these aspects are discussed in [19].

4.5.4 Polyhedron Profile Statistic based Clustering

In this section, we carried out a cluster analysis on the metabolosomes based on the polyhedron profile statistic to have an initial understanding on how many types of structures are there. We used the hierarchical clustering with Ward's minimum variance method [143] (package 'hclust' in R) [144].

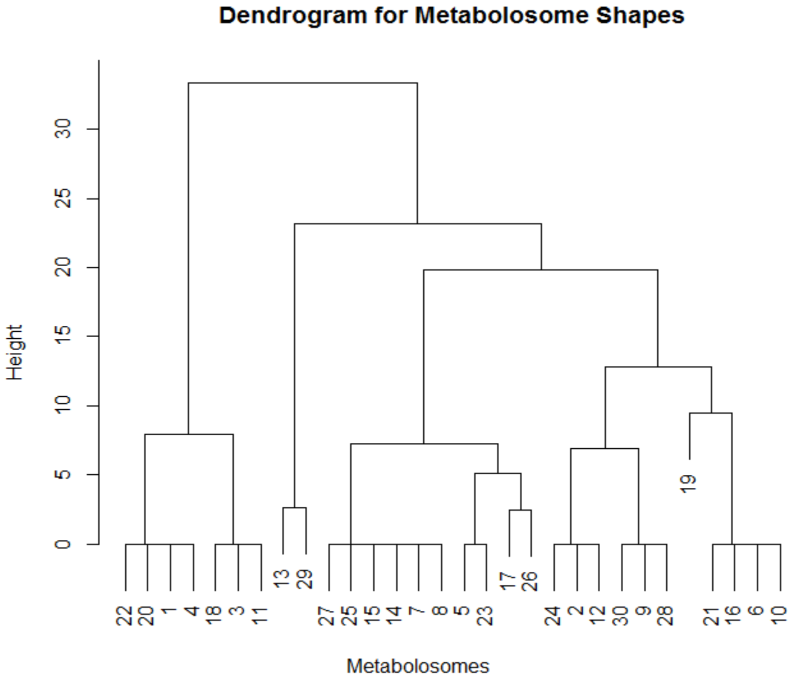


Figure 4.15: A dendrogram from the cluster analysis of the metabolosomes based on the polyhedron profile statistic.

The above dendrogram shows, based on the polyhedron profile statistic there may be approximately 5-6 types of metabolosomes. An important observation is: in spite of considerable variability among any two metabolosomes with respect to their aspect ratios, volumes, edge lengths, face areas etc., there are only a few types of polyhedral structures possible. However, more sophisticated methods for classifying metabolosomes are discussed in Chapter 5 and 6.

4.6 Conclusions and Discussion

In this chapter, we found that the *metabolosomes have convex polyhedral shapes*. Most importantly, the *metabolosomes largely vary in size*. In addition, *they are non-symmetric*, *non-uniform (deformed)* and *have varied aspect ratios*. We also developed an algorithm for handling the missing wedge problem in the polyhedral shaped cryo-EM tomographic reconstructed images. This chapter also provides an indication that the metabolosomes may have structures resembling the Johnson solids, as opposed to symmetric icosahedral shape of the Carboxysome.

The features extraction from the incomplete metabolosome reconstructions were accomplished manually and may be subject to little inconsistency. Here an automated algorithm to identify vertices, edges and faces could be more appropriate. Though the algorithm driven identification results are reproducible, but the manual feature identification has advantages in the case of metabolosomes.

The manual feature extraction is advantageous due to several reasons. First, as described in Section 4.2.1, the reconstructed metabolosomes have some abrupt peaks which may be considered as a part of the object in automated algorithms. Next, the reconstructed metabolosomes have missing regions. The missing regions boundaries will be treated as metabolosome boundaries in automated algorithms and would require manual intervention. Again, the non-missing areas also contain errors from manual segmentation and there are curved edges. It is very difficult to set a global threshold for fitting planes to the metabolosome surfaces due to these errors. These difficulties led us to visually identify the polyhedral features.

Chapter 5

Polyhedral Structural Distance Model

5.1 Introduction

The analyses of the fundamental geometric properties establish that the metabolosomes vary largely in shapes and sizes, they are non-uniform and non-symmetric. As discussed earlier, the traditional shape alignment and averaging methods cannot be applied here to manage the missing wedge problem.

This chapter is concerned with characterizing polyhedral shapes of the metabolosomes and it describes a method we developed for predicting polyhedral shapes in the presence of missing data. This is the first ever shape characterization of these bacterial inclusions.

Two different approaches were undertaken to solve this shape prediction problem. The first approach is discussed in this chapter and uses the ‘completed’ metabolosome structures (Section 4.3, Chapter 4) to match with the standard polyhedra (Section 3.5, Chapter 3). The second approach is discussed in Part II of Chapter 6.

The approach discussed here is a method we developed using the polyhedron profile statistic (Section 3.6, Chapter 3) and a distance function for polyhedral shape matching. Importantly, this method is also equally applicable to the deformed objects with varying sizes.

5.2 Shape Prediction using Completed Shapes

The initial metabolosome structures are incomplete but have been subsequently 'completed' using an algorithm described in Chapter 4. This section is concerned with a polyhedral structural distance model we developed to evaluate the similarities among the *completed* metabolosome structures and the solids from the standard polyhedral families.

5.2.1 Shape Descriptor for Polyhedron

In general, the statistical shape matching algorithms have two main components - a shape descriptor and a proximity measure function [147]. Since the shapes considered here are standard polyhedra, the shape descriptor should contain the discriminating features of the standard polyhedra. The *polyhedron profile statistic* (Section 3.6, Chapter 3) is such a shape descriptor we used, since:

1) It characterizes the features of standard polyhedra.
2) This statistic can differentiate almost all standard polyhedra except very few, as discussed in Section 3.6 (Chapter 3).
3) It contains topological properties invariant under deformation.

As discussed in Chapter 1, several shape descriptors are developed for use in a different contexts. But very little attention had been directed towards translation invariant standard polyhedral shape descriptors to classify standard polyhedral shapes families.

We also developed a specialized distance function for polyhedral shape matching, particularly suitable for these biological microstructures from their ECT images, and it is discussed in the next section.

5.2.2 The Structural Distance Model

Let us recall the polyhedron profile statistic from Section 3.6 (Chapter 3) and the key notations. MPS_i is the profile statistic for the i^{th} *completed* metabolosome and SPS_j is the same for the j^{th} standard polyhedron, i = 1, 2,…, 30 and j = 1, 2, …, 123. Let MPS_{ik} and SPS_{jk} be the observations for the k^{th} feature, k = 1, 2, …, 15, since the polyhedron profile statistic has 15 entries. N_V and N_E are the number of vertices and edges respectively and N_F^r be the number of faces with r edges, $r \in \{3, 4, 5, 6, 8, 10\}$.

We define the structural distance (D_{ij}) between the i^{th} *complete* metabolosome and the j^{th} standard solid as:

$$D_{ij} = \sum_{k=1}^{15} w_k \, d_k$$

where,

$$d_k = \begin{cases} l_{1k} \times |MPS_{ik} - SPS_{jk}| & \text{if } (MPS_{ik} - SPS_{jk}) \leq 0 \\ l_{2k} \times |MPS_{ik} - SPS_{jk}| & \text{if } (MPS_{ik} - SPS_{jk}) \geq 0 \end{cases}$$

and the parameter w_k represents the weight of the k^{th} feature, l_{1k} and l_{2k} indicate the robustness (Section 5.2.3) of the k^{th} feature, k =1, 2, …, 15. The i^{th} metabolosome is predicted to have t^{th} standard solid's shape if $\textit{argmin}_j \, D_{ij} = t$, j = 1, 2, …, 123 and i = 1, 2,…,30. However, the parameters l_{1k} and l_{2k} also put some weights to the k^{th} feature, but these are separately considered from w_k since l_{1k} and l_{2k} are used for different purposes as described in next section.

5.2.3 Parameters Selection

Weights

It is possible to convert any non-triangular face to a number of coplanar triangular faces. A quadrilateral can be considered as two adjacent coplanar triangles or a pentagon can be partitioned as three adjacent coplanar triangles. Notably, this 'triangularization' does not change the shape of the object, but changes its polyhedron profile statistic, since converting a non-triangular face to triangles needs extra edges.

But in 3D, four or five vertices from a metabolosome are very unlikely to be coplanar (at least approximately) and when they are, it may indicate a meaningful characteristic. In other words, a clearly visible quadrilateral or a pentagonal facet in a metabolosome is especially an important feature. These observations are incorporated in the structural distance model through weights (w).

Naturally, the more important characteristics are given greater weights. In notation: $w_m \geq w_n$ if the m^{th} feature is more important than the n^{th} one, m, n = 1, 2, …, 15. However, the weights corresponding to the features are assigned heuristically and a possible set of values we used for w_k is provided in Appendix 5.1.

Robustness

Since the metabolosome data may contain errors arising from manual segmentation or visual feature identification, the polyhedron profile statistic from a metabolosome may not exactly be equal to that of any standard polyhedron. So, if MPS_i is such a metabolosome profile, $MPS_{ik} - SPS_{jk} \neq 0$ for some k = 1, 2, …, 15 and for *all* j = 1, 2, …, 123. However, $|\{k : (MPS_{ik} - SPS_{jk}) \neq 0\}|$ varies, i.e. mismatches may occur for a different number of features.

Now, if the profile statistic from a metabolosome does not match with any of the standard polyhedron, we find the 'nearest' standard polyhedron for this metabolosome. In other words, we find that standard polyhedron whose structure can be achieved by *minimal meaningful changes* on the metabolosome structure. This is explained as follows.

The structural changes of a polyhedron can be done through several ways, e.g.

1) Adding one or more vertices to the structure, and connecting these new vertices with the existing vertices. Note that, one new vertex in 3D needs at least three new edges to be connected. It automatically generates new faces as well. However, this approach changes the original shape of the structure.
2) Adding diagonals to convert quadrilateral facets to coplanar triangles or pentagonal facets to coplanar triangles and quadrilaterals etc. This procedure does not change the shape, but changes the edge counts and different types of face counts.

Now, for example, adding a vertex and the required (at least 3) edges to a metabolosome structure generates a new structure which exactly resembles the SPS_m. Also, adding one diagonal of just one quadrilateral face to the same metabolosome structure generates another structure which exactly resembles the SPS_n. We prefer the SPS_n since it involves *minimal* changes (requires just one edge) rather than the SPS_m (which requires one vertices and at least three edges).

In addition, since the incomplete metabolosome structures are 'completed', we assume that the probability of existence of a new vertex beyond the observed 'completed' structure is very small. Rather, due to curved facets and segmentation errors, it is highly probable that edges may exist at the diagonals of curved quadrilateral or pentagonal facets. So the number of vertices is less preferable to alter, hence less robust than the number of faces.

Again, adding diagonals changes the number of edges, the number of faces, the types of faces and the types of vertices. If e is an additional diagonal added to a curved quadrilateral face, the triangle count (N_F^3) increases and the quadrilateral count (N_F^4) decreases. But adding a diagonal to a curved pentagonal face results increment in N_F^3 and N_F^4 simultaneously. So, some features tend to increase (e.g. N_F^3), some to decrease (e.g. N_F^5) and some features (e.g. N_F^4) may show both (pentagons are highest edged faces in the metabolosomes). So an important consideration is:

$$sign(MPS_{ik} - SPS_{jk}) \text{ where, } (MPS_{ik} - SPS_{jk}) \neq 0.$$

The parameter l_{1k} manages the increments and l_{2k} manages the decrement of the k^{th} feature. The higher value of l_{1k} indicates that the k^{th} feature is more probable to decrease during structural changes and it is just opposite for l_{2k}. All these observations are included in the structural distance model (Section 5.2.2). However, the values of these parameters are assigned intuitively (with the following justification). Appendix 5.1 gives a possible set of values we used for l_{1k} and l_{2k}.

Selection of l_1, l_2 and w

First, l_1 and l_2 for all features are set to 1. Then, based on the importance of the features and effects of these parameters on individual features (discussed in Section 5.2.3), the values are changed. The rationale behind the amount of changes made on these two parameters (from its initial value =1) are also discussed in Section 5.2.3.

For the parameter w (weight), initially all values are set to 1/15 = 0.0667, so that total weight becomes unit. Then based on the importance of a feature (discussed in Section 5.2.3) the corresponding weight is increased and the increment is subtracted equally from other features' weights to keep the total as unit. This process is repeated until all features get appropriate weights based on the discussion in Section 5.2.3. Finally, the l_1, l_2 and w are fine tuned through the following process:

As discussed in Section 5.2.4, there are 11 metabolosomes shows exact match with the standard polyhedra. For these 11 metabolosomes, these parameters are irrelevant, because in the structural distance model, d_k is zero for all k. We first record these shapes; they are J_{62}, J_{86}, J_{87} and J_{88} (J = Johnson solid). Next, some errors are introduced (example below) to these structures and applied the *structural distance model* to check if the prediction is correct. The parameters are 'tuned' until the model predicts the correct initial shapes. This is an iterative process and *many possible choices for the parameters are possible.*

Example: The polyhedron profile statistic of a Sphenocorona (J_{86}) is:

$$T = (10, 14, 22, 12, 2, 0, 0, 0, 0, 0, 6, 4, 0, 0\ , 0).$$

Due to curved surface, segmentation error etc., say one edge adjacent to two triangles was not recorded from the structure. So the recorded polyhedron profile statistic is:

$$T^* = (10, 13, 21, 10, 3, 0, 0, 0, 0, 0, 8, 2, 0, 0, 0).$$

(The vertex count remains same, face count is reduced by 1 as two adjacent triangles now form one quadrilateral, edge count is reduced by 1, triangle count is reduced by 2, quadrilateral count is increased by 1. Due to removal of an edge, two vertices are affected; the number of vertices with 4 edges is increased by 2 and the number of vertices with 5 edges is decreased by 2).

Finally the structural distance model is applied to T^* and the parameters are 'tuned' until J_{86} gets the lowest structural distance. The same process is repeated for J_{62}, J_{87} and J_{88} and for different types of errors as well (e.g. non-recorded vertex or non-recorded edge and vertex both etc.). The final set of parameters are selected such that the set provides the best possible prediction (may not be correct prediction in a few cases) for this controlled experiment.

5.2.4 Results: Predicted Shapes

The distance matrix $D = ((D_{ij}))$ is displayed as a heatmap [148] using R [149], [150]. However, a transformation has been applied to D_{ij} for a vivid display:

$$d_{ij} = \log_e((c \times D_{ij}) + 1)\ \forall\ i, j \text{ and } c = 5.$$

The distribution and the following heatmap (Figure 5.1) of d_{ij} shows the metabolosomes may have about 6-8 standard polyhedral shapes. As described in Section 4.4 (Chapter 4), *almost all of these predicted shapes are Johnson solids*. This result also supports the initial observation from another study regarding the non-symmetric structure of the metabolosomes [151].

It may also be useful to check if all predicted shapes are equally likely to appear as shapes of the metabolosomes. The most frequent predicted shapes are the 86^{th}, 87^{th} and 88^{th} Johnson solids [125] with names Sphenocorona, Augmented Sphenocorona and Sphenomegacorona respectively. Table 5.1 gives the names of all predicted solids and their frequencies.

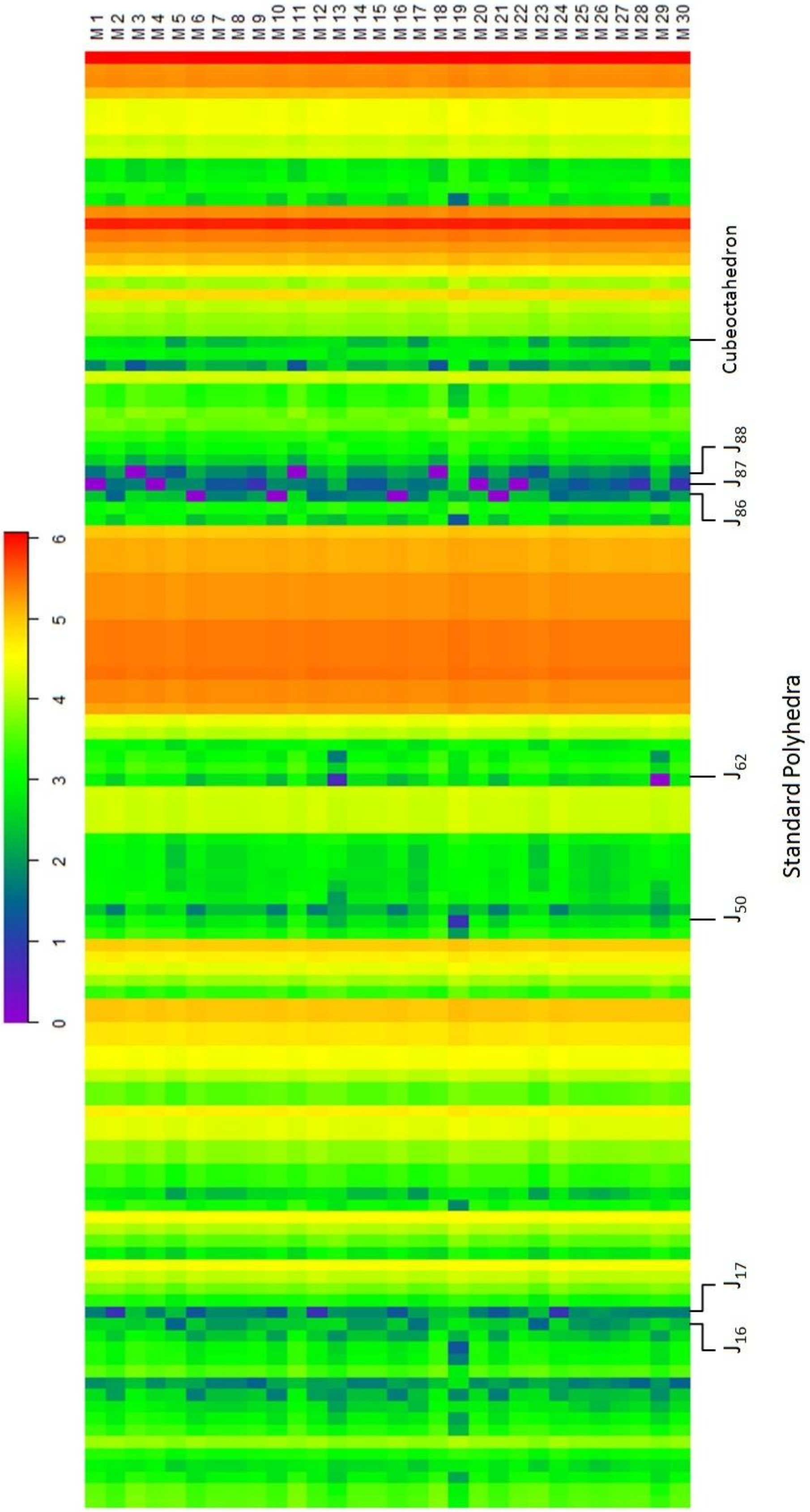


Figure 5.1: A heatmap displaying the distances among the metabolosome and the standard solids obtained through the structural distance model. 'M' stands for metabolosome and 'J' for Johnson solids. The locations corresponding to the predicted solids are marked.

Polyhedron Name	Frequency
Augmented sphenocorona (J_{87})	15
Sphenocorona (J_{86})	6
Sphenomegacorona (J_{88})	3
Elongated pentagonal bipyramid (J_{16})	1
Metabidiminished icosahedron (J_{62})	2
Biaugmented triangular prism (J_{50})	1
Cubeoctahedron (Archimedean solid)	1
Gyro-elongated square bipyramid (J_{17})	1

Table 5.1: The predicted shapes and their frequencies obtained through the structural distance model. 'J' in table stands for Johnson solids, followed by solid number.

Figure 5.2a shows these most frequent shapes. There are 11 metabolosomes from 30 metabolosomes show D_{ij} = 0. The polyhedron profile statistic for these solids is provided in Appendix 3.3.

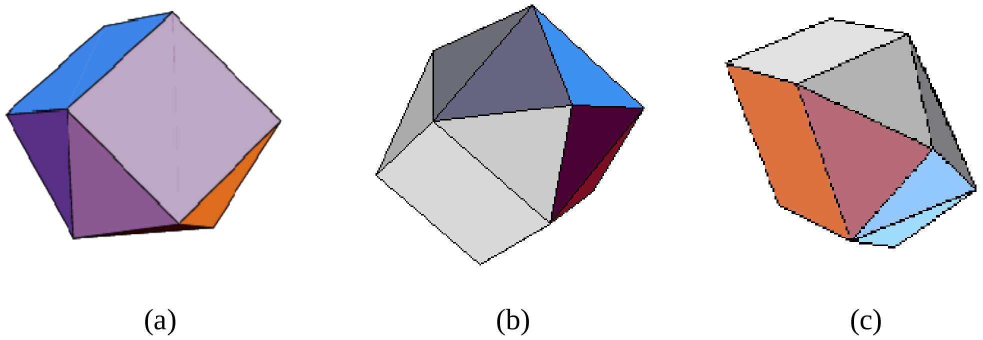


Figure 5.2a: Identified metabolosome shapes using the structural distance model. (a) Sphenocorona (N_V = 10, N_E = 22, N_F = 14 and it has two adjacent quadrilaterals), (b) Augmented Sphenocorona (N_V = 11, N_E = 26, N_F = 17 and it has only one quadrilateral) and (c) Sphenomegacorona (N_V = 12, N_E = 28, N_F = 18 and it has two adjacent quadrilaterals).

In this context, it may also be relevant to see if there is any pattern in the distances among the metabolosomes. So, D_{ij} is calculated i = 1, 2, ..., 30 and j = 1, 2, ..., 30 among metabolosomes and the distances are plotted as a heatmap (Figure 5.2b). The plot shows there are clear similarities among metabolosomes and a few different types of metabolosomes are there.

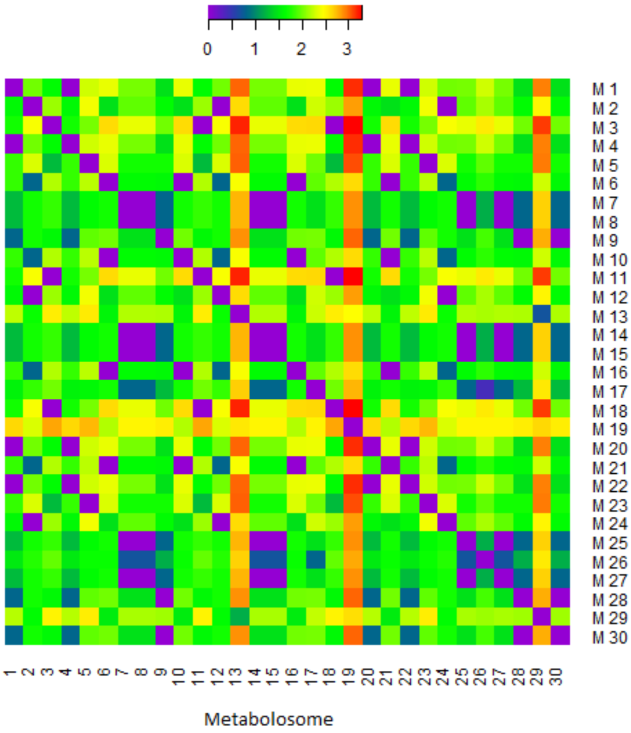


Figure 5.2b: A heatmap displaying the distances among metabolosome structures obtained through the structural distance model. 'M' stands for metabolosome.

However, it is also useful to check how this model affects the standard polyhedral structures, i.e. to check if this model predicts a standard polyhedra as another polyhedra, at least for the predicted shapes. So, like Figure 5.2b, distances are calculated among the standard solids and plotted as a heatmap (Figure 5.2c). The plot shows that, this model discriminates most of the standard solids and in particular, the predicted shapes for the metabolosomes would *not* be wrongly predicted as other solids. This experiment validates the usefulness of this distance measure.

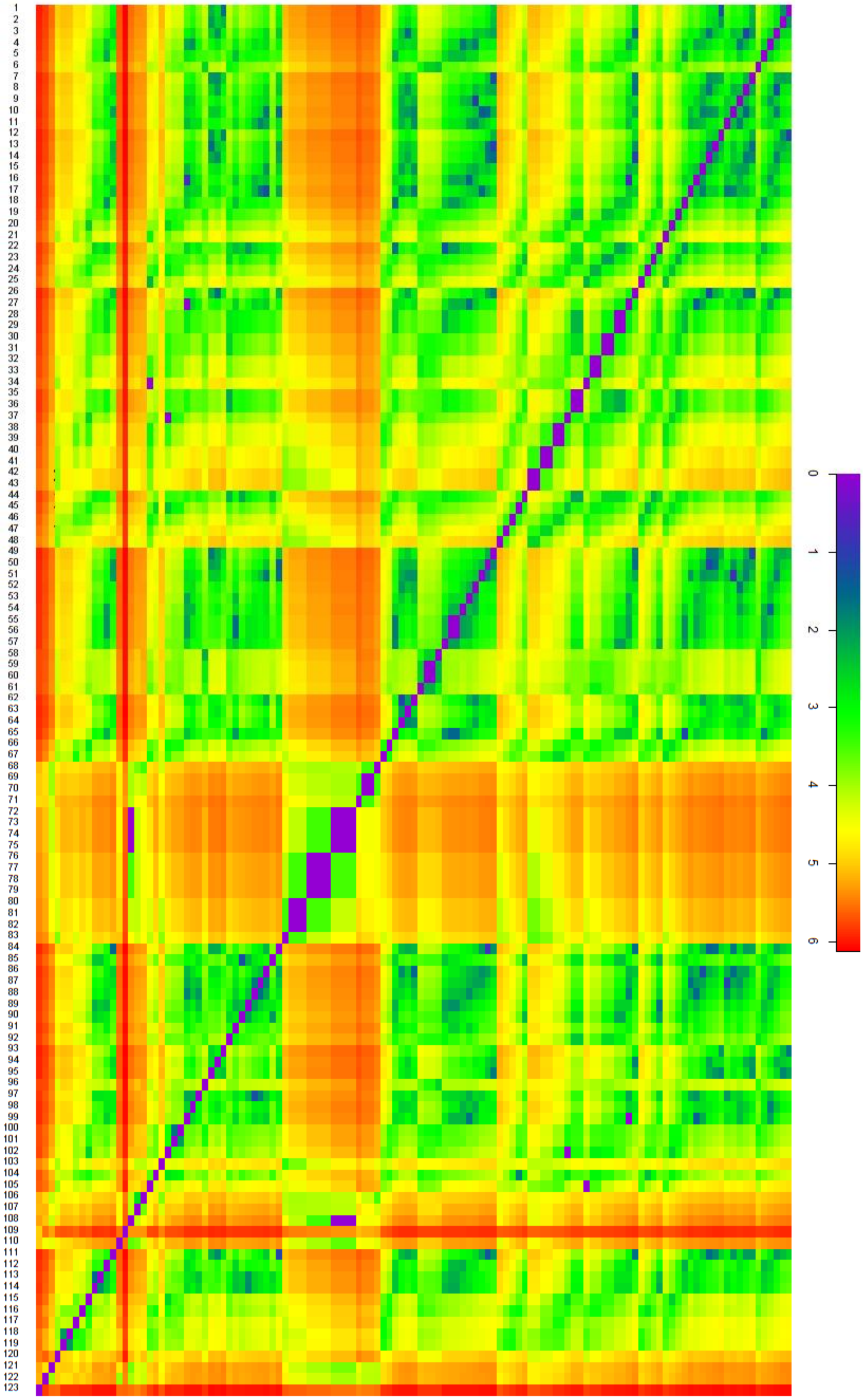


Figure 5.2c: A heatmap displaying the distances among standard polyhedra obtained through the structural distance model.

5.2.5 Visual Validation of Predicted Shapes

Data from the metabolosomes may contain errors due to manual image segmentation and visual feature extraction. So we visually checked if the predicted shapes are relevant to the objects. For example, if the predicted shape for a metabolosome has two adjacent pentagons (e.g. Metabidiminished icosahedron) we checked if that metabolosome really may have two adjacent pentagonal facets. We found that all of these predicted shapes are relevant to the metabolosome structures. However, a few other shapes are also possible for each metabolosome. These possibilities are based on the second or third lowest D_{ij} values for each i, but also utilize some visual characterization, such as the possibility of an *unidentified* vertices, edges or possible missing proportions.

Appendix 5.2 shows the results. In summary, about 70% of the second 'nearest' solids are also Johnson Solids. Noticeably, any polyhedron with N_V = 12 infers an obvious choice as an Icosahedron which is a Platonic solid, since additional diagonals can convert all non-triangular faces to adjacent triangles and an all-triangular faced convex polyhedron with N_V = 12 is eventually an Icosahedron.

5.3 A Simulation Study

The purpose of this simulation study is to estimate the probability that the structural distance method along with shape completion (Section 4.3.2, Chapter 4) can predict the correct parent shapes from their incomplete structures, for varying degrees of missing percentages.

Since about 80% of the proposed shapes of the metabolosomes were classified as the Sphenocorona, Augmented Sphenocorona and Sphenomegacorona, only these three solids and Icosahedron are considered for this simulation study. The Icosahedron is also included since a previously identified microcompartment called Carboxysome [12], [10] has this shape.

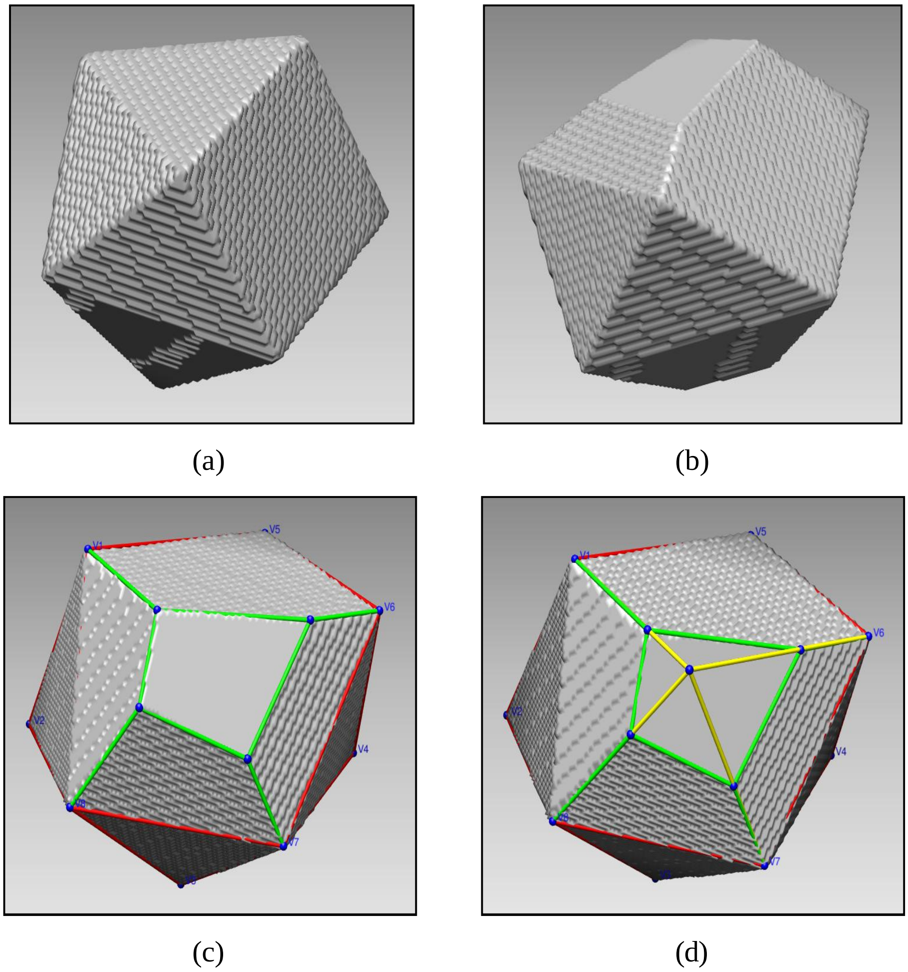
5.3.1 Simulation Algorithm

Step 1: Generating 3D Solids

Simulate a standard polyhedron using vertex coordinates and the information on directed path (Section 3.4.1, Chapter 3) constructing each of the facets (face data). Figure 5.3(a) shows a simulated Sphenocorona.

Step 2: Truncation

Randomly rotate the simulated solid. Analogous to the missing wedge problem in metabolosomes, truncate the simulated solid from top and bottom by certain proportion (e.g. 10%) of its original length. Figure 5.3(b) shows a truncated Sphenocorona.



(a) (b)

(c) (d)

Figure 5.3: A simulation study on the structural distance model on sphenocorona. (a) A simulated sphenocorona, (b) the sphenocorona is truncated from top, (c) the identified vertices, completed and incomplete edges from truncated sphenocorona and (d) completed structure from (c).

Step 3: Completing Structure and Data Collection

Identify the complete and incomplete polyhedral features (vertices, edges, faces etc.) and complete those using the modified INTERPRET algorithm (Section 4.3.2, Chapter 4). Collect the *polyhedron profile statistic* from these *completed* solid. Figure 5.3(d) shows a completed structure from the truncated one.

Step 4: Structural Distance Model

Apply the structural distance model to predict the parent solid from the *polyhedron profile statistic*. We repeated Step 1- Step 4 for the four solids, for several random rotations (here 1000 times) and for different truncation percentages (here, 1% to 80% with 1% interval). If this model can correctly predict *n* times for a particular solid and for a fixed truncation proportion, the correct prediction probability = $\frac{n}{1000}$.

5.3.2 Results from Simulation Study

Figure 5.4 and Table 5.2 summarize the results from this simulation study. It shows that if the solid is truncated by ≤ 30%, the probability that this algorithm will predict the correct parent shape is ≥ 0.90 and in cases of ≤ 20% truncation, this probability is ≈ 1. So if the metabolosomes have missing regions of ≤ 30%, this algorithm predicts the correct shapes of the metabolosomes with probability ≥ 0.90.

	100% Correct Prediction	90% Correct Prediction	80% Correct Prediction
	Truncate Proportion (at most)	Truncate Proportion (at most)	Truncate Proportion (at most)
Sphenocorona	18%	38%	42%
Augmented Sphenocorona	26%	40%	44%
Sphenomegacorona	21%	37%	40%
Icosahedron	41%	48%	51%

Table 5.2: The correct prediction probabilities with varying truncation percentage for four standard solids.

As discussed in Section 2.2.5 (Chapter 2), the truncation proportion at the top and the bottom of a spherical object could be around 13% in case of ± 60° limited angle tilt-series imaging. Considering this proportion (26%, top and bottom together) as a reference, we selected 30% in this context (since the metabolosomes are not spherical and may have varied truncation proportion as well). The above table shows, for 90% correct prediction, the truncation proportion may be up to 37%.

This result also shows, for higher truncation proportions, among these four solids, the Sphenomegacorona is least predictive and the Icosahedron is most predictive. This simulation is conducted for deformed solids as well, where the deformation has been controlled based on the average aspect ratios of the metabolosomes and it shows exactly the same results as non-deformed objects.

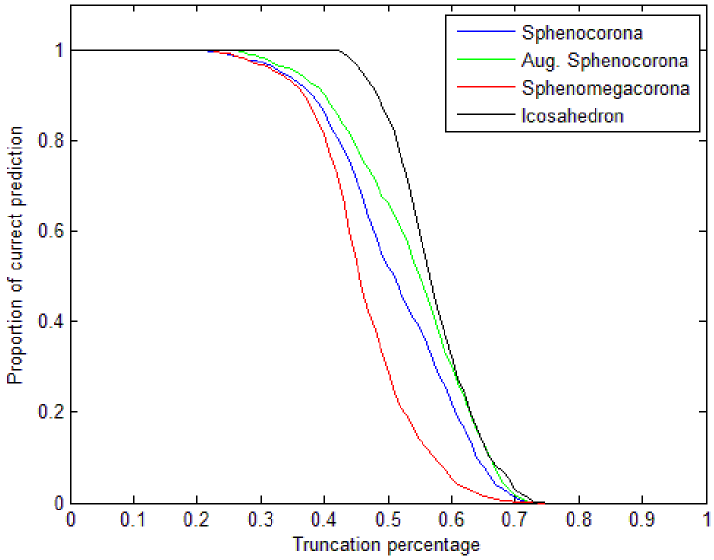


Figure 5.4: The plot shows the gradual decrease of correct prediction probabilities across four solids for varying truncation percentage.

In this simulation, the predicted shapes do not match with the original shapes (misclassified) only when some of the vertices are not recovered from the truncated solids by this algorithm. Now, for example, a vertex could not be recovered and the original solid was a Sphenocorona (number of vertices = 12). The polyhedral structural distance model would not predict a (nearest) solid which has 12 vertices, rather it would predict a solid with 11 vertices due to the higher weights assigned on the vertices. The Section 5.2.3 describes the reasons behind the higher weight assignments. This is the main aspect of the misclassified solids.

5.4 Strengths and Limitations

5.4.1 Strengths of the Structural Distance Model

1) The structural distance (D_{ij}) is calculated based on the polyhedron profile statistic. Since this statistic is invariant under deformation, one of the strengths of this method is it can compare non-symmetric, non-uniform shapes with varying sizes as well. For instance, it can identify the inherent structure of a deformed cube as a cube.
2) An additional strength is feature prioritization and feature robustness, which is particularly important for the biological microstructures. The polyhedron profile statistic contains 15 features, but not all features carry equal importance nor are they equally sensitive to image segmentation and visual identification errors. The model parameters here capture these observations.
3) This model has another potential use - it can predict the 'nearest' standard solid when the polyhedron profile statistic from a metabolosome does not exactly match with any of the standard solids. This model predicts the standard solid as 'nearest' which requires *minimum* changes in its polyhedral structure.
4) Finally, this model is free from missing constraints, since it works with completed structures.

5.4.2 Limitations of the Structural Distance Model

Though the polyhedral structural distance model has elegance in predicting correct polyhedral shapes for lesser truncation, it has some limitations too.

1) This algorithm relies on the *completed* metabolosome structures. The shape completion depends on 'recovering' vertices in the missing regions. But, there is a chance that some vertices are not recovered. This situation may occur due to unknown missing proportion in the metabolosomes, errors in manual segmentation and visual feature extraction.

2) For similar reasons, the recovered vertices may not be 'real'. In addition, a wrong vertex completely changes the polyhedron profile statistic, since the features in the polyhedron profile statistic are correlated on the vertex counts.
3) No statistical method is developed to estimate the model parameters ($w_{k,}$ l_{1k} and l_{2k}). Hence these values are heuristically selected and preference based. Since they are not unique, it may predict different sets of solids for different choices of parameters.
4) This method does not provide the misclassification probabilities, i.e. given an unknown polyhedral structure, this method only predicts the 'nearest' standard solid, but without any probability attached with it.
5) Finally, this method is not useful to compare the incomplete polyhedral shapes, since after truncation, two different polyhedra can result in the same polyhedron profile statistics.

5.5 Conclusions

A polyhedral structural distance is described in this chapter and the metabolosome shapes are predicted by minimizing this distance. The predicted shapes are Johnson solids and about 80% of them are just three Johnson solids - the Sphenocorona, Augmented sphenocorona and Sphenomegacorona.

The results from a simulation study support the potential of this model for polyhedral shape prediction. This model is designed for a particular biological microstructure and works well for it, but suffers from limitations, as described in Section 5.4.2. To address these limitations, we developed a novel incomplete polyhedral shape classification model, described in the next chapter.

Chapter 6

Incomplete Polyhedral Shape Classification

Section I: Simulation

6.1 Introduction

The results from Chapter 4 confirm that the metabolosomes have largely varied shapes and sizes. They are also non-uniform and non-symmetric, i.e. deformed. Thus the traditional shape averaging method is not applicable here. In Chapter 4 and 5, we developed an algorithm for handling the missing wedge problem and a structural distance model to predict the most probable polyhedral structures for the metabolosomes. However, as discussed in Section 5.4.2 (Chapter 5), the structural distance model has a few limitations.

In this chapter the metabolosome shape analysis problem is approached in a different way. Instead of 'completing' the incomplete metabolosome structures, the standard solids are rather truncated and then the *incomplete* metabolosome structures are compared with the *truncated* standard solids.

Later this chapter (Section II), we discuss that the polyhedral shape prediction for the incomplete metabolosomes can be considered as a classification problem through supervised learning [152]. We developed a novel Bayes classifier [54] for incomplete polyhedral shapes classification. We also developed classifiers using the linear discriminant analysis [57] and the support vector machine [58] for the same purpose and the metabolosome shapes are predicted using each of these classifiers. The first section of this chapter deals with truncating standard polyhedron.

6.2 Truncating Standard Polyhedron

Chapter 3 describes the standard polyhedra and their families. A polyhedral structure is characterized by its *polyhedron profile statistic* and this statistic also serves as a polyhedral shape descriptor. This section describes the notion of truncated polyhedron and the polyhedron profile statistic for truncated polyhedron.

6.2.1 Truncation by a Single Plane

Let SP_i is the i^{th} standard polyhedron, $i = 1, 2, \ldots, 123$. If there exists a plane (P) in $\mathbb{R}^3$ such that $SP_\mathrm{i} \cap P$ is a *p-gonal* ($p \geq 3$) polygon, then P 'divides' SP_i into two disjoint but adjacent polyhedra, say ${}_1SP_\mathrm{i}$ and ${}_2SP_i$. Considering SP_i is a solid instead of a graph (Section 3.2, Chapter 3), $SP_i = {}_1SP_i \cup {}_2SP_i$ and each of ${}_1SP_i$ and ${}_2SP_i$ is truncated form of SP_i. Evidently, infinitely many truncated polyhedra pairs (${}_1SP_i$, ${}_2SP_i$) are possible from SP_i based on the choices of P.

Now, if there are two distinct planes (say, P_1 and P_2) in $\mathbb{R}^3$ such that $SP_i \cap P_1$ and $SP_i \cap P_2$ are both *p-gonal* ($p \geq 3$) polygons, then P_1 and P_2 divide SP_i into a set of polyhedra say, $\{{}_1SP_i, {}_2SP_i, \ldots, {}_\mathrm{k}SP_i\}$, $k = 3, 4$, such that $SP_i = {}_1SP_i \cup {}_2SP_i \cup \ldots \cup {}_kSP_i$. Here too, depending on the selection of P_1 and P_2, $\{{}_1SP_i, {}_2SP_i, \ldots, {}_kSP_i\}$ and k vary.

6.2.2 Truncation by Two Parallel Planes

If $P_1 \parallel P_2$ (the symbol '$\parallel$' indicates parallel), then P_1 and P_2 separate SP_i into exactly three polyhedra when both of $SP_i \cap P_1$ and $SP_i \cap P_2$ are *p-gonal* ($p \geq 3$). Consider P_1 and P_2 ($P_1 \parallel P_2$) divide SP_i into ${}_1SP_i$, TSP_i and ${}_3SP_i$, $TSP_i \in P_1 \oplus P_2$ where $\oplus$ denotes the intermediate space bounded by two planes, then TSP_i is the most important part for our future analyses and we define it as the truncated standard polyhedra (TSP) from the i^{th} solid (TSP_i). The following figure (Figure 6.1) illustrates this.

Now, a set of equidistant parallel planes 'divides' a 3D metabolosome into a set of 2D slices and the original 3D metabolosome may be formed by stacking the slices back in proper order (Section 2.4.2 provides the method behind slicing through a graphical presentation). Hence any two slices are parallel to each other. Recalling the missing wedge problem in a 3D reconstructed ECT images (Section 2.2.5, Chapter 2), a few slices from top and bottom of the metabolosomes could not be segmented. Since the last segmented slices at the top and the last segmented slices at the bottom of the segmented metabolosomes are also parallel, it is equivalent to assume that the metabolosomes are truncated from top and from bottom by two approximately parallel planes and therefore, the condition $P_1 \parallel P_2$ is imposed here.

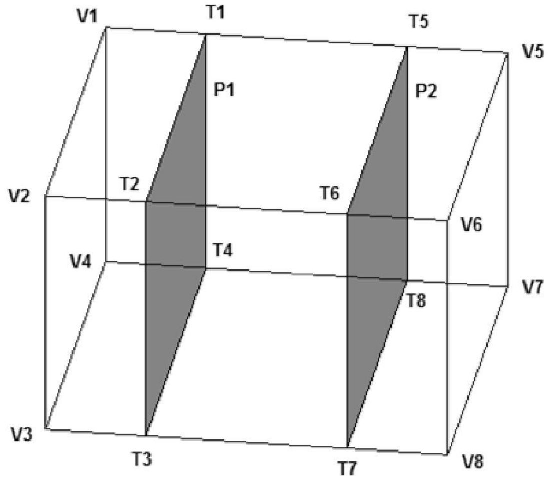


Figure 6.1: Truncating standard polyhedra - a cube. Two parallel planes *P*1 and *P*2 intersects the cube {*V*1, *V*2, …, *V*8}. Among three new polyhedra generated, the polyhedra with vertices {*T*1, *T*2, …, *T*8} is the *TSP*.

6.2.3 Truncation Parameters

But, as mentioned in last section, SP_i can generate infinitely many TSP_i based on the selection of P_1 and P_2, even though the condition $P_1 \parallel P_2$ is satisfied. However, when $P_1 \parallel P_2$, TSP_i depends on only two other characteristics of P_1 and P_2: $f_1(P_1, P_2)$ and $f_2(P_1, P_2)$ as described below.

To define these functions, let A = the largest diagonal of SP_i obtained by connecting two farthest vertices (say, V_1, V_2) from SP_i and $A \perp P_1, P_2$ ($\perp$ means perpendicular). When P_1 and P_2 intersect SP_i, suppose $A \cap P_1 = a_1$ and $A \cap P_2 = a_2$ are the two points on A such that the paths (V_1, a_1), (a_1, a_2) and (a_2, V_2) are disjoint.

Then, $f_1(P_1, P_2) = (\mathrm{d}(V_1, a_1), \mathrm{d}(a_2, V_2))$ and $f_2(P_1, P_2) = (\theta_x, \theta_y, \theta_z)$ where d(.) is the Euclidean distance and θ_x, θ_y, θ_z are the angles of A with the x-, y- and z-axis respectively. Since $\mathrm{d}(V_1, V_2)$ depends on V_1 and V_2, $f_1(P_1, P_2)$ is rather expressed as:

$$f_1(P_1, P_2) = \left(\frac{\mathrm{d}(V_1, a_1)}{\mathrm{d}(V_1, V_2)}, \frac{\mathrm{d}(a_2, V_2)}{\mathrm{d}(V_1, V_2)}\right).$$

The function $f_1(P_1, P_2)$ is termed as truncation proportion and $f_2(P_1, P_2)$ as truncation angle.

6.2.4 Example: Effect of the Truncation Parameters

For simplicity, the effects of these two parameters $f_1(P_1, P_2)$ and $f_2(P_1, P_2)$ are described for a two-dimensional image. The three-dimensional cases are similar. The truncation planes here are just two parallel straight lines.

Figure 6.2 (b) and (d) show that, for a fixed $f_1(P_1, P_2)$, truncation angles determine whether the vertices will be missing or complete edges as well. Figure 6.2 (b) and (c) compares the truncation proportion for a fixed $f_2(P_1, P_2)$.

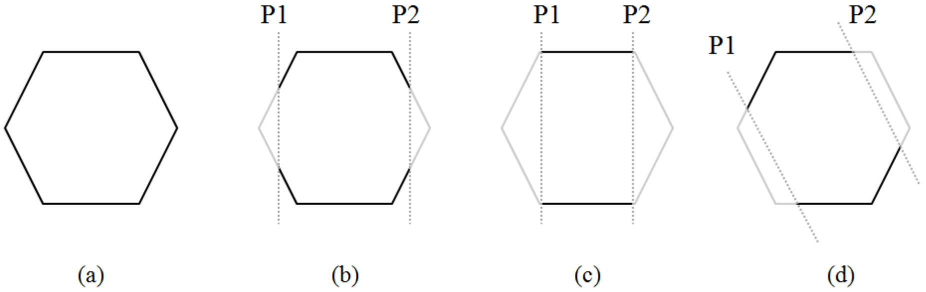


Figure 6.2: Effects of truncation parameters. (a) A complete hexagon. (b) Two parallel planes $P1$ and $P2$ truncate the shape. The truncated shape has 4 vertices and 6 edges. (c) Increased truncation proportion keeping the truncation angle fixed as (b) - the truncated shape has only 2 edges, no vertex. (d) Changed truncation angle, keeping truncation proportion fixed as (b) - the truncated shape has 2 vertices and 4 edges.

6.3 Characterizing Truncated Polyhedron

In Section 3.6 (Chapter 3), we defined the *polyhedron profile statistic* and utilized it as a polyhedron shape descriptor in Chapter 5. The components of this statistic are vertex, face, edge counts, different types of vertices and faces and two adjacency matrices. However, they have been developed in the context of complete polyhedra thus far.

But this statistic from TSP_i would naturally differ from the same from SP_i since some of the vertices, edges and faces have been removed due to truncation. In addition, truncation generates extra features as well, such as *incomplete* edges and *incomplete* faces which were not considered before. A modified polyhedron profile statistic is thus needed to addresses these differences.

6.3.1 Vertex, Edge and Face Counts

Section 3.4.1 (Chapter 3) describes the notion of vertex, face and edges in standard complete polyhedron. This section describes them in the context of the truncated (incomplete) polyhedron.

Number of Vertices

We define, set of all vertices in a *SP* as: $V = \{V_i, i = 1, 2, \ldots, N_V\}$, $V_i = i^{\text{th}}$ vertex, N_V is the total number of vertices. However, due to truncation, some vertices are removed. So the set of remaining vertices from *TSP* is defined as: $TV = \{TV_i, i = 1, 2, \ldots, N_{TV}\}$, where $TV_i = i^{\text{th}}$ vertex from *TSP*, N_{TV} = total number of visible vertices. Clearly, $TV \subset V$ and $N_{TV} \leq N_V - 2$. As an example, Figure 6.3 is a cube truncated by two parallel planes; this *TSP* has $N_{TV} = 6$.

Number of Edges

The edge of a polyhedron is explained in Section 3.2 (Chapter 3). The set of edges from *SP* is defined as, $E = \{E_{ij}, i, j \in [1, N_V]$ such that E_{ij} exists$\}$ and the total number of edges is $N_E = |E|$. However, truncation leaves the following 3 types of edges in a polyhedron:

1. Some edges may completely disappear (missing edges).
2. Some edges may partially disappear (partially missing edges).
3. Remaining edges remain undamaged (complete edges).

So, a *TSP* may have two types of edges - a set of partially missing edges (*PE*) and another set of complete edges (*CE*). Since *PE* are edges indeed, the number of edges in *TSP*, denoted as $N_{TE} = |PE| + |CE|$, $PE \cup CE \subseteq E$ and $N_{TE} \leq N_E$. For example, Figure

6.3 shows N_{TE} = 12, |*PE*| = 6 and |*CE*| = 6. Notably, none of the edges in Figure 6.3 are completely missing here due to truncation.

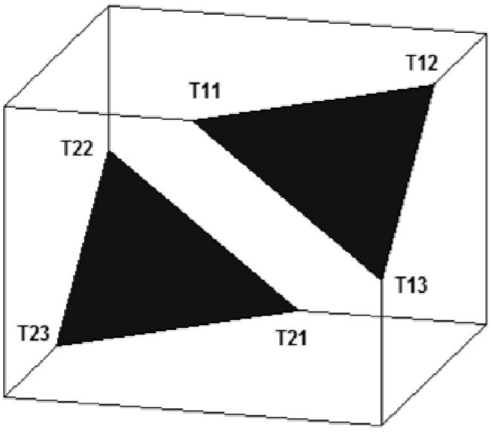


Figure 6.3: A cube truncated by two parallel planes (*T11*, *T12*, *T13*) and (*T21*, *T22*, *T23*).

Number of Faces

The notion of faces is also explained in Section 3.2 (Chapter 3). We define the set of all faces from *SP* as $F = \{F_i,\ i = 1, 2, \ldots, N_F\}$, where N_F is number of faces in *P*. However, similar to edges, truncation also generates the following 3 types of faces:

1. Some faces may completely disappear (missing faces)
2. Some faces may be truncated partially (partially missing faces)
3. Remaining faces stay unaltered (complete faces)

Here too, a *TSP* may have two types of faces - partially missing faces (*PF*) and complete faces (*CF*). Since *PF* are also distinct faces, the number of faces of *TSP*, denoted as N_{TF} = |*PF*| + |*CF*|. Similar to N_E, $PF \cup CF \subseteq F$ and $N_{TF} \leq N_F$. For example, the truncated cube in Figure 6.3 has N_{TF} = 6, |*PF*| = 6 and |*CF*| = 0, i.e. all faces are affected due to truncation.

6.3.2 Vertex Type and Face Type

Recalling from Section 3.4.2 (Chapter 3), the vertex type is defined as the number of edges connected to a vertex or equivalently the number of faces adjacent to a vertex. As defined there, for a *SP*, the number of vertices with *k* edges,

$$N_V^k = \left|\left\{V_i^E : V_i^E = k, i = 1,2,..N_V\right\}\right|$$

where,

$$V_i^E = |\ \{E_{ij}, \forall\ j \in [1, N_V] : E_{ij} \in E\}\ |, i = 1, 2, \ldots, N_V.$$

But truncation may remove some vertices altogether and even when a vertex is visible, all of its attached edges may or may not remain visible. To capture this phenomenon we introduce here the notion of censoring.

First, we consider only vertices having all of its attached edges at least partially visible. Define, $TN_V^k =$ number of vertices from *TSP* with *k* edges attached to it *and* none of its attached edges are missing, *k* = 3, 4, 5, 6, 8, 10.

Next, we consider only those visible vertices having one or more of its attached edges completely missing and define, tN_V^k = number of vertices from *TSP* with ***exactly k visible edges***, *k* = 3, 4, …, 10. Combining TN_V^k and tN_V^k, it gives the number of vertices from *TSP* with ***at least k visible edges*** (ATN_V^k). Hence,

$$ATN_V^k = \left|\left\{TV_i^E : TV_i^E \leq k, i = 1,2,..N_{TV}\right\}\right|$$

where, TV_i^E = number of visible edges from *TSP* connected with TV_i, *i* = 1, 2, …, N_{TV}. However, both of TN_V^k and ATN_V^k are considered separately for characterizing *TSP*. For example, in Figure 6.3 the numbers of vertices with exactly 3 and 4 vertices are 6 and zero respectively. The number of vertices with at least 3 vertices is also 6 in this example.

The similar notion is applied to face type data as well. Since, some of the faces are partially truncated; these faces are no longer triangles or quadrilaterals. So the *CF*'s are characterized by $TN_F^k =$ Number of faces from *TSP* with *exactly k* visible edges *and* none of its edges were incomplete, *k* = 3, 4, 5, 6, 8, 10.

The PF's are characterized by tN_F^k = number of partially missing faces with ***exactly k visible edges***, *k* = 3, 4, …, 10. Combining TN_F^k with tN_F^k results in the number of faces with ***at least k visible edges*** $\left(ATN_F^k\right)$, *k* = 3, 4, …, 10. For example, for the truncated solid in Figure 6.3, number of complete faces with *exactly* 4 edges is zero, but the number of faces with *at least* 4 faces is 6. Clearly, the *'at least type data'* also takes the incomplete faces into account and hence is more informative.

6.3.3 Adjacency Matrices

The notion of edge adjacency matrix, face adjacency matrix and their construction methods are described in Section 3.4.3 (Chapter 3). Here for *TSP*, exactly the same concept is used, *but only the complete edges, complete faces and complete vertices* are considered.

For simplicity, let us consider Figure 6.4, where a cube is truncated by a single plane only (*T1, T2, T3*). There are only 3 edges, (*V*1, *V*6), (*V*5, *V*6) and (*V*6, *V*7) where both of the adjacent faces are intact after truncation and there are 9 edges having their terminal vertices unaffected due to truncation. So the face adjacency matrix (*fADJt*) for *TSP* considers only 3 edges and the edge adjacency matrix (*eADJt*) for *TSP* reflects 9 edges. Hence,

$$fADJt_{ij} = \begin{cases} 3 & \text{if } i = j = 4 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad eADJt_{ij} = \begin{cases} 9 & \text{if } i = j = 3 \\ 0 & \text{otherwise} \end{cases}$$

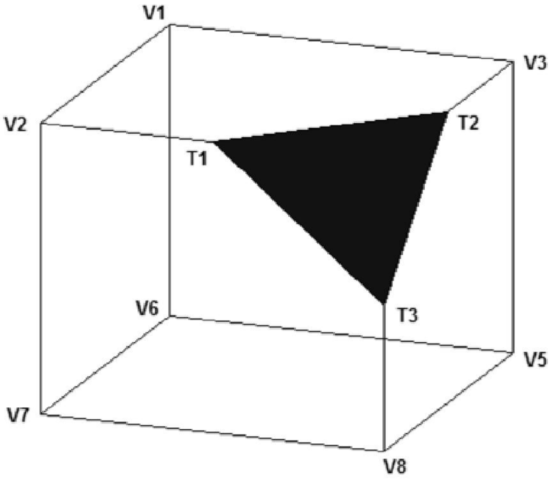


Figure 6.4: A cube truncated by a single plane (*T*1, *T*2, *T*3).

6.3.4 Polyhedron Profile Statistic for Truncated Polyhedron

As discussed, the truncated standard polyhedra have important characterizing features which are expressed through the *at least type* data. These extra features are important, because these numbers also capture the polyhedra features affected due to truncation. Here is the summary of them:

1) Vertex, face and edge counts: (N_{TV}, N_{TF}, N_{TE})
2) Complete vertex type: $\mathrm{TN}_{\mathrm{V}}^{*}$ = (TN_V^3, TN_V^4, TN_V^5, TN_V^6, TN_V^8, TN_V^{10})
3) Complete face type: $\mathrm{TN}_{\mathrm{F}}^{*}$ = (TN_F^3,TN_F^4, TN_F^5, TN_F^6, TN_F^8, TN_F^{10})

4) At least vertex type: $\mathrm{ATN}_{\mathrm{V}}^{*}$ = (ATN_V^3, ATN_V^4, ATN_V^5, ATN_V^6, ATN_V^7, ATN_V^8, ATN_V^9, ATN_V^{10})
5) At least face type: $\mathrm{ATN}_{\mathrm{F}}^{*}$ = (ATN_F^3, ATN_F^4, ATN_F^5, ATN_F^6, ATN_F^7, ATN_F^8, ATN_F^9, ATN_F^{10})
6) Edge adjacency matrix (*eADJt*) = 10 × 10 matrix, transformed as a vector of length 100
7) Face adjacency matrix (*fADJt*) = 10 × 10 matrix, transformed as a vector of length 100.

Combining all these features in the above order into a single vector generates the new *polyhedron profile statistic* for *truncated standard polyhedra* (*TPPS*, truncated polyhedron profile statistic). It should be noted that a profile statistic for *TSP*$_i$ is not just *SP*$_i$'s profile statistic with some missing entries. The *TPPS* serves as the truncated polyhedral shape descriptor for further analysis.

Similar to *TPPS*, the *complete* solids also have these properties as discussed in Section 3.4 (Chapter 3). Combining these features in the same order as *TPPS*, we get the polyhedron profile statistic for complete solids. A single standard polyhedron can generate several *TPPS*, but can have only *one* complete polyhedron profile statistic.

The polyhedron profile statistic for a complete polyhedron is discussed and applied in Chapter 3 and Chapter 5. We found this statistics is a good polyhedral shape descriptor and it can capture useful structural properties to distinguish a polyhedron from polyhedra families. It is also utilized in the *polyhedral structural distance model* to predict the shapes of the metabolosomes.

Since the *TPPS* is based on the same notion of the polyhedron profile statistic, *TPPS* should also be a meaningful and useful statistic for the truncated polyhedra. In fact, the second part of this chapter shows that various classifiers which are developed based on the *TPPS* are found to be considerably powerful classifiers for classifying incomplete polyhedra. Hence, we considered *TPPS* is a relevant statistics for this purpose.

6.4 Truncated Standard Polyhedron Simulation

6.4.1 Purpose of Simulation

Since the incomplete metabolosomes have truncated polyhedral shapes, the purpose of this simulation is to generate truncated standard polyhedra analogous to the truncated metabolosomes.

123 standard polyhedra from four polyhedron families are considered for this analysis. These standard polyhedral structures are truncated from varying truncation proportions and angles in $\mathbb{R}^3$. Each standard polyhedron generates a class of truncated polyhedra based on these truncation parameters. The ultimate goal is to classify an incomplete metabolosome structure into any of these classes.

First, a complete standard polyhedron is generated based on the polyhedra data discussed in Section 3.7 (Chapter 3). Then the polyhedron is rotated and truncated based on pre-defined truncation parameters. Finally, the polyhedron profile statistic is collected from the truncated standard polyhedron through an automated algorithm. This process is repeated based on pre-determined truncation parameters.

6.4.2 Simulation Parameters

While truncating a polyhedron, the truncation characteristics should be analogous to the missing wedges so that truncated metabolosomes are comparable with the truncated polyhedron. Section 6.2 describes one such characteristic incorporated by imposing the constraint $P_1 \parallel P_2$. In this section, the other truncation parameters, i.e. truncation proportion and truncation angles are explained.

Truncation Proportion

The value of the function $f_1(P_1, P_2)$ also has to be chosen to reflect the truncation proportions of the metabolosomes. But the actual values of $f_1(P_1, P_2)$ for the metabolosomes are unknown and are expected to vary slightly across objects due to some segmentation and reconstruction errors. However, repeated visual inspection suggests that the reconstructed metabolosomes may have about 25% missing.

But, a missing proportion (e.g. 0.20) in standard polyhedra can be achieved through numerous selections of $f_1(P_1, P_2)$, e.g. (0.15, 0.05) or (0.12, 0.08). To restrict this choice of $f_1(P_1, P_2)$ to a finite class, we impose the condition:

$$D = \frac{\mathrm{d}(V_1, a_1)}{\mathrm{d}(V_1, V_2)} = \frac{\mathrm{d}(a_2, V_2)}{\mathrm{d}(V_1, V_2)}$$

and allow D to taking values from {0.050, 0.075, 0.100, 0.125, 0.150}. So, during simulation, each of the 123 standard polyhedra will be truncated from top and bottom with equal proportion for each value of D.

In general a *TSP* with $D = 0.050$ contains more information than $D = 0.150$ for a fixed $f_2(P_1, P_2)$. If the parent solid can be predicted from *TSP* with $D = 0.150$, it is very likely that the same can be achieved when $D = 0.050$. Thus $max(D) = 0.150$ which truncates total 30% of a polyhedron is chosen to exceed the maximum possible truncation proportion in the metabolosomes.

Truncation Angle and Number of Rotations

As mentioned before, the structure of a *TSP* not only depends on the $f_1(P_1, P_2)$, but also on the angle of truncation, i.e. $f_2(P_1, P_2) = (\theta_x, \theta_y, \theta_z)$. Similarly to truncation proportion, the truncation angles $(\theta_x, \theta_y, \theta_z)$ for metabolosomes are also unknown. However, in simulation, the *SP* is truncated by $P_1 \parallel P_2$ from several angles in $\mathbb{R}^3$, expecting that it includes the actual truncation angles occur in the metabolosomes. The rotation angles $(\theta_x, \theta_y, \theta_z)$ are independently selected such that:

$$\theta_t \sim \text{uniform}(-2\pi, 2\pi), \qquad t = x, y, z.$$

The parameters $(\theta_x, \theta_y, \theta_z)$ are sampled a sufficiently large number of times (N_{sim}) so that an exhaustive set of *TSP* for each of 123 *SP* is generated. For this simulation study $N_{sim} = 5000$ and in subsequent sections we show that N_{sim} is sufficiently large to produce an exhaustive set of *TSP*.

6.4.3 Rotating Plane vs. Rotating Object

Let D be fixed, $P_1 \parallel P_2$ and $(\theta_x, \theta_y, \theta_z)$ are set to some values. P_1 and P_2 truncate *SP* from angles $(\theta_x, \theta_y, \theta_z)$ to generate a *TSP*. This is equivalent to the following:

Suppose, A is a straight line in $\mathbb{R}^3$ such that $A \perp P_1 \parallel P_2$ and $(\theta_{x^*}, \theta_{y^*}, \theta_{z^*})$ are the rotation angles required to set A parallel to z-axis. Now the *SP* is rotated using

(θ_{x*}, θ_{y*}, θ_{z*}) and two new planes P_{1Z} and P_{2Z} are defined such that $P_{1Z} \parallel P_{2Z}$ and P_{1Z}, $P_{2Z} \perp z$-axis and D is the distance between P_{1Z} and P_{2Z}. If P_{1Z} and P_{2Z} truncates SP simultaneously, it generates same TSP as by P_1 and P_2 above.

This observation is implemented in the simulation for computational simplicity. The SP is rotated using independently distributed $\theta_t \sim uniform(-2\pi, 2\pi)$, $t = x, y, z$ and each rotated SP is truncated by $P_1 \parallel P_2$ with distance D and $P_1 \perp$ z-axis. The truncation proportion (D) is calculated based on the 'height' of the rotated SP along the z-axis. The truncation proportion = 0.100 means it truncates 10% from top and 10% from bottom of the rotated SP. The following algorithm gives details about the truncation process.

6.4.4 The Truncation Algorithm

Notation

SP_i is the i^{th} complete standard polyhedron, $i = 1, 2, \ldots, 123$.
N_V = total number of vertices in SP_i.
N_E = total number of edges in SP_i.
$E = \{E_{jk}, j, k \in [1, N_V]$ such that E_{jk} exists$\}$ is the set all edges from SP_i and
E_{jk} = the edge connecting vertices V_j and V_k, $E_{jk} \in E$.

Step 1: Simulating a Complete Standard Polyhedron

Section 3.7 (Chapter 3) shows that, from the face data of SP_i, the terminal vertices of each edge can be extracted. Let E_{jk} from SP_i connects V_j and V_k. Instead of considering E_{jk} as a straight line in $\mathbb{R}^3$, let us define $E_{jk} = \{q_{jk1}, q_{jk2}, \ldots, q_{jkm}\}$ where q_{jk}'s points are sampled on the path $V_j \to V_k$ such that m is very large (here 1000 per unit edge length) and $d(q_{jk1}, q_{jk2}) = d(q_{jk1}, q_{jk3}) = \ldots = d(q_{jk(m-1)}, q_{jkm})$. Here d(.) is the Euclidean distance in $\mathbb{R}^3$. Clearly, the collection of the points on edges (EP_i) = $\bigcup_{j,k} E_{jk}$ is the set of all sampled points from all edges of SP_i. Certainly, plotting all points from EP_i would generate the graph of SP_i. Hence the simulated SP_i is expressed as the set of points EP_i.

Step 2: Rotating the Simulated Polyhedron

Since rotating SP_i is equivalent to rotating EP_i, a set of rotational transformations are applied to EP_i to rotate the object in 3D. This transformation is for elemental rotation, i.e. rotation about Cartesian axes. If rotation of EP_i by θ_x, θ_y and θ_z angles along the x-, y- and z-axes respectively generates the rotated object REP_i, then $REP_i = R_x(\theta_x) \times R_y(\theta_y) \times R_z(\theta_z) \times EP_i$ where $R_x(\theta_x)$, $R_y(\theta_y)$ and $R_z(\theta_z)$ are rotation matrices defined as follows [153]:

$$R_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta_x & -sin\theta_x \\ 0 & sin\theta_x & cos\theta_x \end{bmatrix}, \quad R_y(\theta_y) = \begin{bmatrix} cos\theta_y & 0 & sin\theta_y \\ 0 & 1 & 0 \\ -sin\theta_y & 0 & cos\theta_y \end{bmatrix} \text{ and}$$

$$R_z(\theta_z) = \begin{bmatrix} cos\theta_z & -sin\theta_z & 0 \\ sin\theta_z & cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Step 3: Truncating Rotated Polyhedron

In this step the rotated solid (REP_i) is truncated by two planes perpendicular to the z-axis, where the distance between two planes (D) is predetermined. The algorithm is:

```
SET D
REP_i(z) = set of z-coordinates of all points in REP_i.
MAX_Z = Maximum(REP_i(z))
MIN_Z = Minimum(REP_i(z))
LENGTH_ALONG_Z = MAX_Z - MIN_Z
TOP_TRUNCATION_LIMIT = MAX_Z - D
BOTTOM_TRUNCATION_LIMIT = MIN_Z + D
FOR K = 1 TO NUMBER OF POINTS IN REP_i
      IF Z_COORDINATE OF K^th POINT ≤ BOTTOM_TRUNCATION_LIMIT
      OR
      IF Z_COORDINATE OF K^th POINT ≥ TOP_TRUNCATION_LIMIT
            SET COORDINATES OF K^th POINT of REP_i=(NaN,NaN,NaN)
            (NaN = Missing value)
END
```

This step finally gives the coordinates of the points on the edges *only* belonging in the region bounded by two parallel D-distant planes P_1 and P_2. In other words, this algorithm provides the coordinates of the points *only* on the edges from TSP_i. This set of points is the representation of the truncated solid.

Step 4: Polyhedron Profile Statistic from Truncated Polyhedra

An extension of the above algorithm can keep track of each edge from SP_i, and record whether an edge is affected due to truncation. This edge information eventually provides the status of each face of TSP_i, i.e. whether truncation affected a face. A face is partially visible if at least two of its edges remain visible after truncation. Finally, as discussed in Section 3.7 (Chapter 3), this face data can generate all other entities in the polyhedron profile statistic.

A relevant fact in this context is differentiating a completely missing edge from a partially truncated edge. Consider $E_{jk} = \{q_{jk1}, q_{jk2}, \ldots, q_{jkm}\}$ as an edge affected due to truncation and let $\{q_{jkr}, \ldots, q_{jkm}\}$, $1 \leq r \leq m$ be the points truncated. The problem is what the value of r should be such that E_{jk} would be a partially missing edge instead of completely missing. Theoretically, for any $1 < r < m$, E_{jk} can be considered as a partially missing edge.

But in metabolosomes, due to curved facets, reconstruction and segmentation errors, an incomplete edge is 'identifiable' only if at least 'a considerable proportion' of that edge is visible. Based on a visual inspection of the metabolosomes, we set this proportion to 0.25, i.e. a partially truncated edge from TSP_i is 'completely missing' only if less than 25% of that edge remains after truncation. In other words, E_{jk} is completely missing if:

$$d(q_{jkr}, q_{jkm}) \geq 0.75 \times d(q_{jk1}, q_{jkm}\}.$$

Step 5: Repeating the Simulation

```
FOR Polyhedra = 1 TO 123
      FOR d IN {0.050, 0.075, 0.100, 0.125, 0.150}
            FOR Rotation = 1 TO 5000
                  GENERATE RANDOM $\theta_x$, $\theta_y$, $\theta_z$
                  REPEAT Step 2 TO Step 4
            END
      END
END
```

The final output of this algorithm is a set of N_{TSP} = (5000 × 123) polyhedron profile statistics, 231 entries in each, creating a numerical matrix with dimension 615000 × 231.

Class Labels for Truncated Standard Polyhedra

Each of the *TPPS* is 'labeled' with its parent solid's name so that for any particular *TPPS*, its original complete solid can be identified. Clearly, the set $\{TPPS_{ij}\ \forall\ j\}$ forms the i^{th} class of truncated standard polyhedra, $i = 1, 2, \ldots, 123$. One of the purposes of this labeling is to facilitate this data for use in supervised learning.

6.4.5 Unique Polyhedron Profile Statistic

For a fixed *D*, let TSP_{ij} be the j^{th} *TSP* generated from SP_i using the above algorithm, $i = 1, 2, \ldots, 123$ and $j = 1, 2, \ldots, 5000$. The j^{th} truncated standard polyhedra is the outcome of the j^{th} random truncation. As mentioned earlier, for fixed *D*, TSP_{ij} varies based on $(\theta_x, \theta_y, \theta_z)$.

Let TSP_{ij} correspond to $TPPS_{ij}$, $\forall\ i = 1, 2, \ldots, 123$ and $j = 1, 2, \ldots, 5000$. Since the *TPPS* is the only shape descriptor for *TSP*, $TSP_{mn} = TSP_{rs} \Leftrightarrow TPPS_{mn} = TPPS_{rs}\ \forall\ m, r \in 1, 2, \ldots, 123$ and $n, s \in 1, 2, \ldots, 5000$. Interestingly, the situation $TPPS_{mn} = TPPS_{rs}$ occurred for several *m*, *n*, *r*, and *s* in the simulated dataset.

Now it is of interest to see how many 'unique' *TPPS* are generated from an individual solid and from the complete simulated datasets since it justifies whether N_{sim} (= 5000) simulations were adequate. Let $\Lambda(TPPS_{\ i})$ be the number of unique *TPPS* from SP_i, i = 1, 2, …, 123 and $\Lambda(TPPS)$ the total number of unique *TPPS* from all N_{TSP} profiles together. We found that, for $D = 0.100$ the proportions of unique $TPPS = \Lambda(TPPS) / N_{TSP}$ with 231 features is 0.0448. Figure 6.5 shows the distribution of $\Lambda(TPPS_{\text{i}}) / N_{TSP.}$

We found that only 7 solids among 123 solids have this ratio larger than 20% and only 13 solids have this ratio 10% or more. All of these 13 solids are Johnson solids (J_{47}, J_{48}, J_{68}, J_{70}, J_{71}, J_{72}, J_{74}, J_{75}, J_{78}, $J_{79,}$ J_{81}, J_{82}, J_{83}). A possible reason for the higher proportions of unique *TPPS* is the complicated structure of these solids. Among these solids, the lowest number of vertices is 35 (J_{47}) and the highest is 75 (J_{71}). The higher number of vertices associate with a higher number of edges and hence with a higher number of different face types. In addition, the adjacency matrices also tend to have more non-zero elements. Thus, when truncated, these structures are very likely to introduce more heterogeneity in *TPPS* than the simpler structures.

Since these ratios are small, the profiles are expected to repeat a considerable number of times and hence we assume the simulation generates almost all possible *TSP*. However, more samples lead to better estimation of the distribution of *TPPS*, but also introduce overfitting in learning [154]. The distribution of *TPPS* is described in Section II of this chapter.

One important observation is, if the $TPPS_{ij}$ would contain fewer features, e.g. only three features - vertex, face and edge counts, the $\Lambda(TPPS)$ is expected to reduce considerably. With D = 0.100, $\Lambda(TPPS)$ / N_{TSP} = 0.0075 when only these three features are considered.

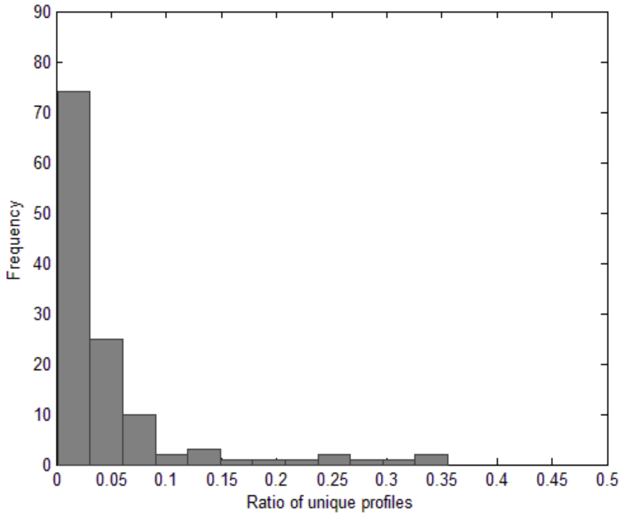


Figure 6.5: The distribution of $\Lambda(TPPS_i)$/ N_{TSP} calculated from 123 solids with truncation proportion 0.100 and all of 231 features are considered.

A relevant point is discussed in this context - where there may be infinitely many ways to truncate a polyhedron, whether there are surely only a finite set of distinct truncated polyhedra possible. Since a truncated polyhedra is defined by its *TPPS*, we plotted the number of unique *TPPS* with respect to the number of random truncations for verification. The result is provided in Figure 6.5a.

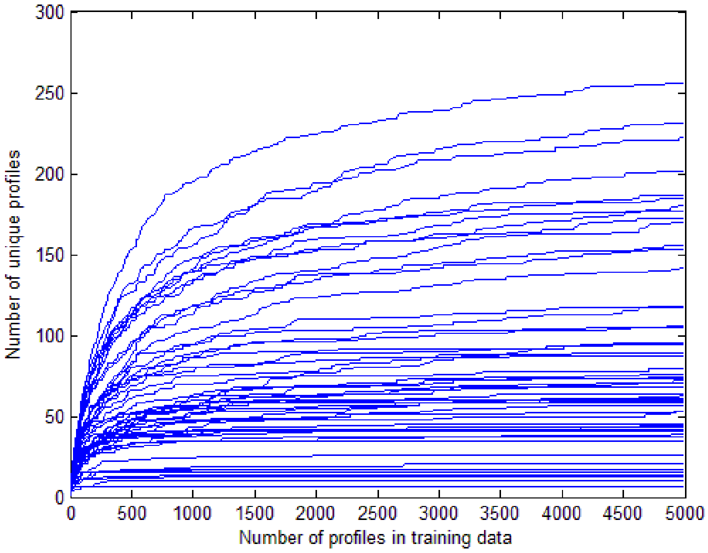


Figure 6.5a: The increments of unique polyhedron profile statistic with increasing number of random truncations.

The plot shows, the slope of the lines are almost zero toward the end, i.e. a new profile is very unlikely to appear beyond 5000 simulations. Though theoretically it is possible that infinitely many truncated polyhedron may be generated from a standard polyhedron, this observation demonstrates that 5000 simulations are sufficient to include *almost* all of them. Thus we have a good sample representing the truncated standard polyhedron population.

6.5 Incomplete Metabolosomes Data

6.5.1 Data Collection

Section 4.2.2 (Chapter 4) described the detailed procedure of superimposing an *incomplete* polyhedron on an *incomplete* reconstructed metabolosome. As mentioned, Chimera UCSF records the fitted *incomplete* polyhedron in a .py file. From this associated file, the data on directed paths for all complete and incomplete faces are extracted manually.

This extracted face data can generate all other features (Section 3.7, Chapter 3) required to develop the truncated polyhedron profile statistic for the metabolosomes. This truncated metabolosome profile statistic (*TMPS*) is computed for all 30 metabolosomes. Analogous to *TTPS*, *TMPS* also has a length of 231. The vertex, face, edge counts, face and vertex type data for all incomplete metabolosomes are provided in Appendix 6.1.

6.5.2 Distribution of Metabolosome Features

The distribution of the first feature of *TMPS*, i.e. the number of vertices in the *incomplete* metabolosomes (IMN_V) is displayed in following figure. As mentioned in Section 6.3.1, since $N_{TV} \leq \mathrm{N}_\mathrm{V} - 2$, $min(IMN_V) = 6$ and $max(IMN_V) = 10$, almost surely the distribution of IMN_V is $IMN_V \in [8, 20]$ in the complete metabolosomes. The results obtained from the *polyhedral structural distance model* (Chapter 5) also support this observation.

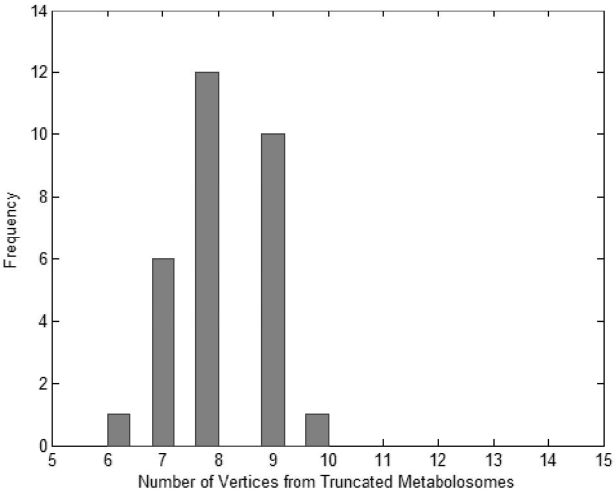


Figure 6.6: Distribution of number of vertices from 30 incomplete metabolosomes.

6.5.3 Reduction in Computational Cost

The distribution of IMN_V shows that instead of 123 standard solids, considering only *complete* solids having $N_V \in [8, 20]$ would be sufficient for shape prediction. Exactly 55 solids from 123 standard solids have $N_V \in [8, 20]$. So for further analysis, only these 55 solids were considered.

The support vector machine, used for subsequent analysis is computationally intensive [155] and requires significantly longer time and larger memory when 123 solids instead of 55 solids are considered. The setup with 123 solids, 2500 TPPS from each solid as training data and remaining 2500 *TPPS* as test data and *TPPS* length as 231 took approximately three days for SVM with one-vs-one voting method (Section 6.10.1). For computing purpose, the Duke Compute Cluster [156] with 128 GB of memory was used. With the same setup and computing resources, it took about only 3−5 hours (depending on the truncation proportion) when considering 55 solids with $N_V \in [8, 20]$. Since the simulation studies were required to repeat more than hundred times (for parameters tuning, different truncation proportions, different feature sets, etc.), the execution time was considered as an important factor for these calculations.

Thus reducing the number of solids saves computing time and resources. However, considering a fewer number of solids does not impact the individual misclassification probabilities (Section 6.8.3). To support this, misclassification probabilities from two cases (with 123 and 55 solids) were computed for the Bayes

Classifier, compared for the common predicted solids and found no difference in their misclassification probabilities.

6.6 Summary

An algorithm is developed to truncate 123 standard solids from random directions and in varying truncation proportions. The truncation is performed a sufficiently large number of times and for each truncation the corresponding truncated polyhedron profile statistic is collected through an algorithm. The similar profile statistic is also collected from 30 incomplete metabolosomes.

The collection of these truncated polyhedra constitutes a truncated polyhedra library. Each truncated polyhedron is characterized by its truncated polyhedron profile statistic and labeled by its parent shape's name. However, due to computing time and resource constraints, instead of 123 solids only 55 solids were considered for further analyses - these 55 solids have vertices fewer than 20 but larger than 8.

Chapter 6

Section II: Incomplete Polyhedral Shape Classification

6.7 Introduction

Section I of this chapter describes the classes of truncated polyhedral shapes. Since the metabolosomes also have truncated polyhedral shapes, the objective is now to classify these incomplete metabolosomes to any of the standard polyhedra classes. Thus the problem of predicting metabolosome shapes turns out to be a statistical classification problem.

Developing classifiers involves 'learning' the observed data. The simulated data from Section I is utilized for 'learning' about truncated polyhedral shapes. These methods also require a qualitative or quantitative description on the characterizing features of the classifying objects. The truncated polyhedron profile statistic (Section 6.3.4) serves this purpose.

Since the class labels for simulated truncated shapes are known and there are more than two classes, this is a multiclass classification problem with supervised learning. In this section of the chapter, we developed some supervised learning methods for classifying incomplete polyhedral structures and used these classifiers to predict the metabolosome shapes.

6.7.1 Training Data and Test Data

Notations

Here are some key notations used in this subsection and also in the remaining part of this chapter. The remaining notations are explained in subsequent sections as required.

$SP = \{SP_1, SP_2, \ldots, SP_{55}\}$ = the class of 55 standard solids.
$TPPS_{ij}$ = j^{th} *truncated* polyhedron profile statistic (*TPPS*) from the i^{th} standard solid, $i = 1, 2, \ldots, 55, j = 1, 2, \ldots, 5000$.
$TMPS_k$ = k^{th} *truncated* metabolosome profile statistic (*TMPS*), $k = 1, 2, \ldots, 30$.
$TPPS_{ij}^{m}$ = m^{th} entry in $TPPS_{ij}$ and $TMPS_{k}^{m}$ is the same for $TMPS_k$, $m = 1, 2, \ldots, 231$.
PPS_i = polyhedron profile statistic (*PPS*) for the i^{th} *complete* standard polyhedra, $i = 1, 2, \ldots, 55$.
PPS_{i}^{m} = m^{th} entry in PPS_i, $m = 1, 2, \ldots, 231$, $i = 1, 2, \ldots, 55$.
As described in Section 6.3, each of the *TPPS* and *TMPS* contains the feature set:

$$\{N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*, ATN_V^*, ATN_F^*, eADJt, fADJt\}$$

which form the *polyhedron profile statistic* for a truncated polyhedron. Finally, as described in Section 3.6 (Chapter 3) and Section 6.3.4, each of the *PPS* contains the feature set (in the same order as *TPPS* and *TMPS*):

$$\{N_V, N_F, N_E, N_V^*, N_F^*, AN_V^*, AN_F^*, eADJ, fADJ\}$$

Training Data and Test Data Separation

The set of all $TPPS_{ij}$ with even j are considered as training data and remaining $TPPS_{ij}$ are kept aside as test data. Since the truncation was carried out from random orientations, this partition for training data eliminates the selection bias [157]. There are 5000 *TPPS* from each standard polyhedron, hence test data and training data both has 2500 profiles from each standard polyhedron.

This training set was utilized to develop the classification rules for all subsequent classification approaches. The test data was used to calculate the misclassification probabilities (Section 6.8.3) to evaluate the performance of these classifiers. The following diagram (Figure 6.7) shows the training data and test data selection scheme.

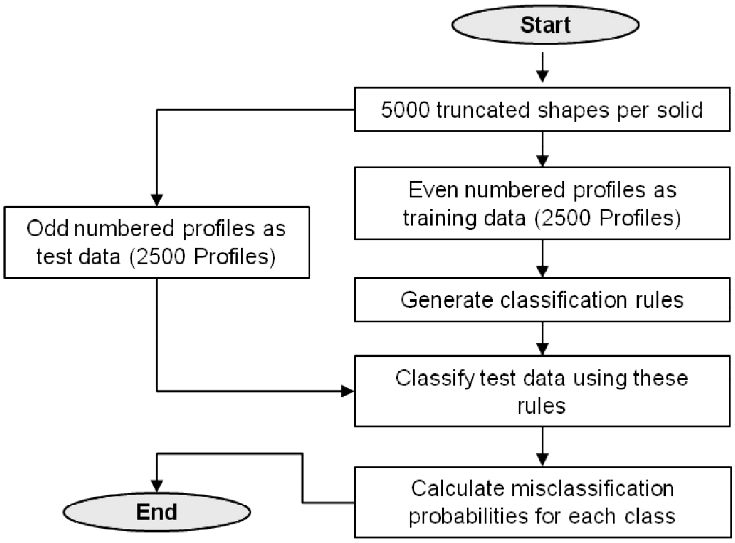

Start
5000 truncated shapes per solid
Odd numbered profiles as test data (2500 Profiles)
Even numbered profiles as training data (2500 Profiles)
Generate classification rules
Classify test data using these rules
Calculate misclassification probabilities for each class
End

Figure 6.7: The training and test data selection and classification scheme.

6.7.2 Selection of Truncation Proportion

As discussed in Section 6.2, the characteristics of a simulated truncated polyhedron (*polyhedron profile statistic*) may depend on truncation proportion, i.e. $f_1(P_1, P_2)$. This scenario was demonstrated in Figure 6.2. Five different truncation proportions were considered for simulation.

The exact values of D (equivalently $f_1(P_1, P_2)$) for different metabolosomes are difficult to find and since the metabolosomes have different shapes and sizes, there is no reason that the values of D will be exactly same for all metabolosomes. However, as discussed in Section 2.2 (Chapter 2), for a spherical object with ± 60° tilt angle ECT, D is approximately 0.13.

A few end slices of any metabolosome cannot be segmented (as discussed before), but those slices contain some sort of 'signatures' (dark regions, slightly different texture than neighborhood regions etc.) from where it could be visualized that a part of the metabolosome exists there. We counted the number of such slices starting from the last segmented slice at the both ends and estimated D from them. Thus through visual inspections we determine that the truncation proportion (*D*, as discussed in Section 6.4.2) for metabolosomes is approximately 0.100. So the results presented in this chapter are based on $D = 0.100$

The another approach for determining D could be comparing 'completed' polyhedral structure (Section 4.3) with the corresponding incomplete metabolosome, if there is not much segmentation errors (abrupt peaks) near the last segmented slices. Since D can be slightly different for different metabolosomes, the results for other *D*, i.e. 0.050, 0.075, 0.125 and 0.150 are also provided (in the Appendix section).

6.8 The Bayes Classifier

The Bayes classifier [158] has long been applied in pattern recognition, shape classification [56] and many other classification problems. The construction of these classifiers is based on the Bayes' theorem [159] and it is an optimal classifier [160] with respect to minimization of probability of classification errors (Section 6.8.3).

The Bayes classifier requires prior knowledge of probability distributions of the classified objects. Since these are rarely known in advance, the probabilities are to be estimated from training data [161]. If the number of features in the training data is very large, more samples are needed to estimate these probabilities and in the absence of a sufficiently large number of samples, this estimation may be erroneous. This phenomenon is also known as the 'curse of dimensionality' [162]. However, our data do not suffer from this problem (Section 6.4.5). In this section, we develop the Bayes classifiers to classify incomplete polyhedral shapes.

One similar classifier is the Naive Bayes classifier [56], [163], [164], which assumes that the effect of a variable on a given class is independent of the other variables. This assumption is known as the class-conditional independence [164], [165]. However, since the variables in *TPPS* have strong relationships among themselves, the Naive Bayes classifier is not appropriate here.

6.8.1 Probability Distribution for Truncated Polyhedra

As discussed in Section 6.5.3, we considered 55 classes of polyhedra for *all* subsequent classification approaches. Here the observations are the simulated truncated polyhedra and the polyhedron profile statistic is the feature vector for these observations.

Section 6.4.6 explains that $TPPS_{mj}$ = $TPPS_{nj}$ for many choices of (*m*, *n*) and $TPPS_{ij}$ = $TPPS_{ik}$ for several (*j*, *k*) pairs. So we expect that, $TPPS_{ij}$ has non-null frequencies across classes. Here we calculated the probability distributions of $TPPS_{ij}$. The probability distribution for the $TPPS$ is calculated based on the training data consisting of N_T = 137500 (= 55 × 2500) observations. Let the m^{th} $TPPS$ in the training data ($TPPS_m$) have frequency $NTPPS_m$, $m = 1, 2, \ldots, N_T$. Now, if

$SP = \{SP_1, SP_2, \ldots, SP_{55}\}$ is the class of 55 polyhedra and

$nTPPS_{im}$ = frequency of the m^{th} $TPPS$ from SP_i, $nTPPS_{im} \in [0, NTPPS_m]$,

the probability of $TPPS_m$ appearing in SP_i is:

P ($TPPS = TPPS_m \mid SP = SP_i$) = $\frac{nTPPS_{im}}{NTPPS_m}$, $m = 1, 2, \ldots, N_T$ and $i = 1, 2, \ldots, 55$.

Also, P ($TPPS = TPPS_m$) = $\frac{NTPPS_m}{N_T}$, $m = 1, 2, \ldots, N_T$ and P ($SP = SP_i$) = 1/55 $\forall$ i.

The posterior probability, i.e. P ($SP = SP_i \mid TPPS = TPPS_m$) is calculated using the Bayes theorem:

$$\mathsf{P}(SP = SP_i \,|\, TPPS = TPPS_m) = \frac{\mathsf{P}(TPPS = TPPS_m \,|\, SP = SP_i)\,\mathsf{P}(SP = SP_i)}{\sum_k \mathsf{P}(TPPS = TPPS_m \,|\, SP = SP_k)\,\mathsf{P}(SP = SP_k)}$$

where, $m = 1, 2, \ldots, N_T$ and $i = 1, 2, \ldots, 55$. P ($SP = SP_i \mid TPPS = TPPS_m$) describes: given the m^{th} $TPPS$, the probability that SP_i will contain this profile. These probabilities are used for constructing Bayes classifiers for truncated polyhedral shapes. An example in the next section (Section 6.8.2) illustrates this computation.

6.8.2 Bayes Classifiers for Truncated Polyhedra

Let $TMPS_k$ be the k^{th} truncated *metabolosome* profile statistic, described in Section 6.5, $k = 1, 2, \ldots, 30$. The Bayes classifier states that, select that solid which maximizes the posterior probabilities for this particular profile, i.e. assign $TMPS_k$ to t^{th} standard polyhedra (SP_t), t = 1, 2, …, 55 if:

1) $TMPS_k = TPPS_m$ for some m, $m \in \{1, 2, \ldots, N_T\}$ and
2) $argmax_{\ r}\ \mathsf{P}(SP = SP_r \,|\, TPPS = TPPS_m) = t$

Example

The following example illustrates the construction of the Bayes classifiers for truncated polyhedral shapes. For simplicity, consider 3 standard polyhedra {SP_1, SP_2,

SP_3} only and the test data contains just 4 profiles from each class. Also consider that the *TPPS* and *TMPS* both consist of just 5 features (vertex, face and edge counts, triangle and quadrilateral counts). The following table gives a sample data for *TPPS*.

Solid Number	**Vertex**	**Face**	**Edge**	**Triangle**	**Quadrilateral**
SP_1	7	14	21	4	1
SP_1	7	13	20	4	1
SP_1	7	14	21	4	1
SP_1	7	14	21	4	1
SP_2	7	12	20	4	0
SP_2	7	12	20	4	0
SP_2	7	14	21	4	1
SP_2	7	14	20	4	0
SP_3	7	14	21	4	2
SP_3	6	14	21	3	0
SP_3	6	14	21	3	0
SP_3	8	14	21	4	1

Table 6.1: A sample dataset consisting of fewer features from truncated polyhedron profile statistic.

This data generates the following frequency distribution:

Vertex	**Face**	**Edge**	**Triangle**	**Quadrilateral**	**Frequency**		
					$\mathbf{SP_1}$	$\mathbf{SP_2}$	$\mathbf{SP_3}$
7	14	21	4	1	3	1	0
7	13	20	4	1	1	0	0
7	12	20	4	0	0	2	0
7	14	20	4	0	0	1	0
7	14	21	3	2	0	0	1
6	14	21	3	0	0	0	2
8	14	21	4	1	0	0	1

Table 6.2: The frequency distribution of the *TPPS* from Table 6.1.

Now, let a *TMPS* = (7, 14, 21, 4, 1). Since *TMPS* = $TPPS_1$ (first row of the frequency table), we calculate the probabilities: P($TPPS_1$ | SP_1) = 0.75, P($TPPS_1$ | SP_2) = 0.25 and P($TPPS_1$ | SP_3) = 0.00. Since, P(SP_1) = P(SP_2) = P(SP_3) = 1/3, the posterior probabilities are: P(SP_1 | $TPPS_1$) = 0.75, P(SP_2 | $TPPS_1$) = 0.25 and P(SP_3 | $TPPS_1$) = 0.00. The Bayes classifier predicts this TMPS to have SP_1 with probability 0.75, since $argmax_k$ P(SP_k| $TPPS_1$) = 1.

6.8.3 Misclassification Probability

The error in classification occurs when $TPPS_{ij} \rightarrow SP_k$, i.e., a test $TPPS_{ij}$ is classified to SP_k for some $k \neq i$ through the classifier (here the Bayes classifier). The misclassification probability for a *particular profile*, say $TPPS_{ij}$ is

$$1 - \sum_{k \neq i} \mathrm{P}(TPPS_{ij} \mid SP = SP_k)).$$

However, we are interested in $L^*(i)$ = the probability that a *TPPS* from SP_i is misclassified to SP_k for some $k \neq i$. The $L^*(i)$, also termed as the overall misclassification probability [56] for SP_i, is estimated as follows.

All test $TPPS_{ij}$ for a fixed i have same parent class labels = SP_i for all j. Each of these $TPPS_{ij}$ for a fixed i is classified based on the Bayes classifiers. We define an identity function for $TPPS_{ij}$ as:

$$I_{\{i,j\}} = \begin{cases} 1 & TPPS_{ij} \rightarrow SP_i \\ 0 & \text{Otherwise} \end{cases}$$

and this function gives an estimate of $L^*(i)$ as:

$$L^*(i) = 1 - \frac{1}{n} \sum_{j} I_{\{i,j\}}$$

where $n = 2500$. The following plot shows the distribution of $L^*(i)$ for $i = 1, 2, \ldots, 55$ when we consider all features (i.e. the profile statistic has length 231). The minimum, maximum and mean ± SD of $L^*(.)$ are 0.0000, 0.3560 and 0.0323 ± 0.0577 respectively. Also, 90.91% of the solids have $L^*(.) \leq 0.10$. These results demonstrate:

1) The Bayes classifier can be considered as a strong classifier for this problem
2) The TPPS is a 'good' descriptor for the truncated standard polyhedral shapes

Appendix 6.2 provides the distributions and summary of overall misclassification probabilities from the Bayes classifiers for different truncation proportions.

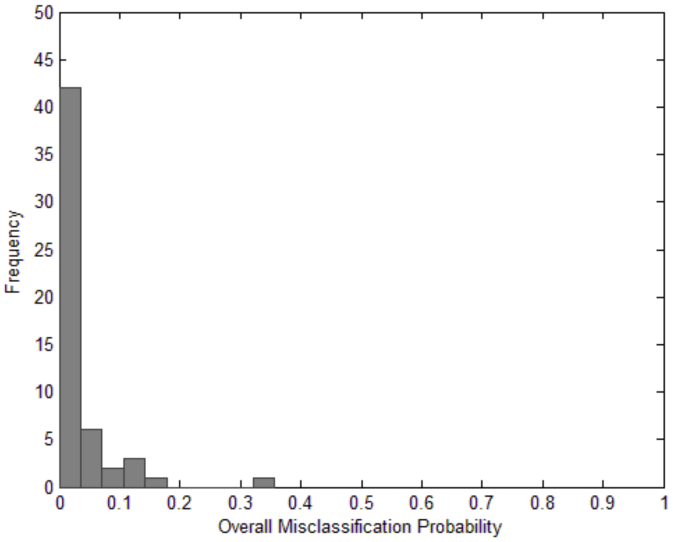


Figure 6.8: The distribution of overall misclassification probabilities from 55 solids, calculated for Bayes classifiers when truncation proportion is 0.100. The outlier proportion is from J_{35}.

6.8.4 Bayes Classifiers for Subset Features

Section 6.8.2 states that for a *TMPS*, the Bayes classifier requires $TMPS_k = TPPS_m$ for at least one m, $m = 1, 2, \ldots, N_T$ (Section 6.8.2, condition 1). In case of metabolosomes, due to manual segmentation and feature extraction errors this condition may not be satisfied. In fact, there is no *TMPS* (with all 231 features) which satisfies this condition. The simulation generates a sufficiently large number of observations and this problem does not occur frequently for test *TPPS*. This is rather due to errors in the *TMPS* data. So, this is not due to the 'curse of dimensionality' [162]. Two modified forms of Bayes classifier are considered for this problem - distance based *profile* selection and hierarchical *feature* selection.

Distance Based Profile Selection

The distance based profile selection first computes distance between two polyhedron profile statistics, say, $TPPS_m$ and $TPPS_n$ as

$$\mathrm{d}(m,n) = \sum_{i=1}^{231} w_i |\, TPPS_m^i \; - \; TPPS_n^i \,|^k$$

where $TPPS_m^i$ is the value of the i^{th} feature from $TPPS_m$, w_i is the weight for the i^{th} entry in *TPPS*, $i = 1, 2, \ldots, 231$. When $w_i = 1 \; \forall \; i$, $k = 1$ gives the 1- norm distance and $k = 2$ gives the squared Euclidean distance etc. Clearly, $\mathrm{d}(m, n) = 0$ gives exact match. For a *TMPS*, say $TMPS_k$, this distance is calculated with all training *TPPS* and the best matched *TPPS* is the one with minimum distance. Let the selected *TPPS* is $TPPS_m$. Similar to Section 6.8.2, the Bayes classifier classifies $TMPS_k$ as SP_t if:

$$argmax_{\ r}\ \mathrm{P}(SP = SP_r \,|\, TPPS = TPPS_m) \ = \ t.$$

But this approach has a flaw. As discussed in Section 5.2.2 and 5.2.3 (Chapter 5) in the context of the *polyhedral structural distance model*, all features in a *TPPS* may not carry the same importance. So it is necessary to include the feature based weights (w_i) in the model. Since the predicted shapes largely depend on w_i's and choosing the w_i's is a genuine problem, this approach is not used for further analysis.

6.8.5 Hierarchical Feature Selection

Since the distance based profile selection is not feasible here, we developed the Bayes classifiers based on hierarchically selected features. The central notion of hierarchical feature selection is to eliminate the 'weakest' features step-by-step from the *TPPS* and develop the Bayes classifiers based on reduced length *TPPS*.

Step 1: Feature Selection

First, we need to decide which feature is to be excluded. For simplicity, let *fADJt* be excluded first. We can express,

$TPPS$ = {N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*, ATN_V^*, ATN_F^*, *eADJt*} ⊕ {*fADJt*} = ${}_1TTPS$ ⊕ ${}_2TTPS$, where ⊕ denotes augmentation and ${}_2TTPS$ = {*fADJt*}. Similar separations are made for *PPS* and *TMPS* too.

Step 2: Filtering Solids

This step utilizes the fact that $TPPS_{ij}^m \leq PPS_i^m$, ∀ i, j, m, i.e., any feature from the truncated solid should have a lower value than that from the corresponding complete solid. Using this observation we create the following rule:

$${}_2TPPS_{ij}^m \ \geq \ {}_2PPS_k^m \text{ for any } m \ \Rightarrow \text{ exclude } SP_k \text{ from calculation for } TPPS_{ij}$$

where m = 1, 2, …, 100 (since *fADJt* is of length 100), k = 1, 2, …, 55. In other words, if any value of the face adjacency matrix from the truncated polyhedron is larger than the corresponding entry from a complete solid, the truncated profile cannot be from that complete solid - so we exclude that solid from further analysis. Let the filtered results $\{SP_k, k \in [1, 55], max(k) \leq 55\}$ for a particular $TPPS_{ij}$, while filtering is done based on *fADJt*.

Step 3: Probability Distribution and Bayes Classifiers re-construction

Recalling Section 6.8.1 for the probability distribution of $TPPS_{ij}$, the same procedure was applied for calculating the probability distribution of ${}_1TTPS_{ij}$. ${}_1TTPS_{ij}$ has length 131 and $i = 1, 2, \ldots, n$ (≤ 55). Based on the probability distribution of ${}_1TTPS_{\mathrm{ij}}$, we re-calculated the Bayes classifiers using the same way as in Section 6.8.2. In this example, the Bayes classifiers will be constructed based on 131 features of *TPPS* instead of 231 features and the number of target classes is possibly less than 55.

Stopping Criterion

If a ${}_1TMPS_k$ does not match with any of the ${}_1TTPS_{ij}$, then some more features from ${}_1TTPS_{ij}$ are to be excluded. If the next discarded feature is *eADJt*, then $TPPS = {}_1TTPS \oplus {}_2TTPS$ where ${}_1TTPS = \{N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*, ATN_V^*, ATN_F^*\}$ and ${}_2TTPS = \{eADJt, fADJt\}$. Here ${}_1TTPS$ has 31 features only and filtering will be based on ${}_2TTPS$, which has 200 entries. We repeat the process until ${}_1TMPS_k = {}_1TTPS_{ij}$ for at least one i, j, where $i \in$ remaining solids after filtering at that stage. The following diagram describes this algorithm.

Order of Features for Exclusion

This feature selection approach needs to determine which feature to exclude first, which second, and so on. The order of exclusion is determined as follows:

The *TPPS* includes two types of features - some features are based on only complete data and some are based on both complete and incomplete data. For example, *eADJt* and *fADJt* relies on only *completely visible* edges and vertices, but N_{TE} includes both of *complete and incomplete* edges. The features incorporating both of complete and incomplete data convey more information and hence are more 'contributive'.

The exclusion criterion is set up in such a way that, the remaining features take account of *maximum* possible information *at that step*. The Table 6.3 describes the stepwise feature selection scheme adopted for *TPPS*.

Feature Name	**Selection Matrix**										
	S1	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**	**S11**
N_{TV}, N_{TF}, N_{TE}	x	x	x	x	x	x	x	x	x		
ATN_F^*	x	x	x	x	x	x				x	x
ATN_V^*	x	x	x	x	x					x	
TN_F^*	x	x	x	x			x	x			
TN_V^*	x	x	x				x				
eADJt	x	x									
fADJt	x										

Table 6.3: The order of exclusion of the features - 'S' in columns indicates the steps and the 'x' marked features are included for constructing Bayes classifier at that step. The features in blank cells participate in filtering.

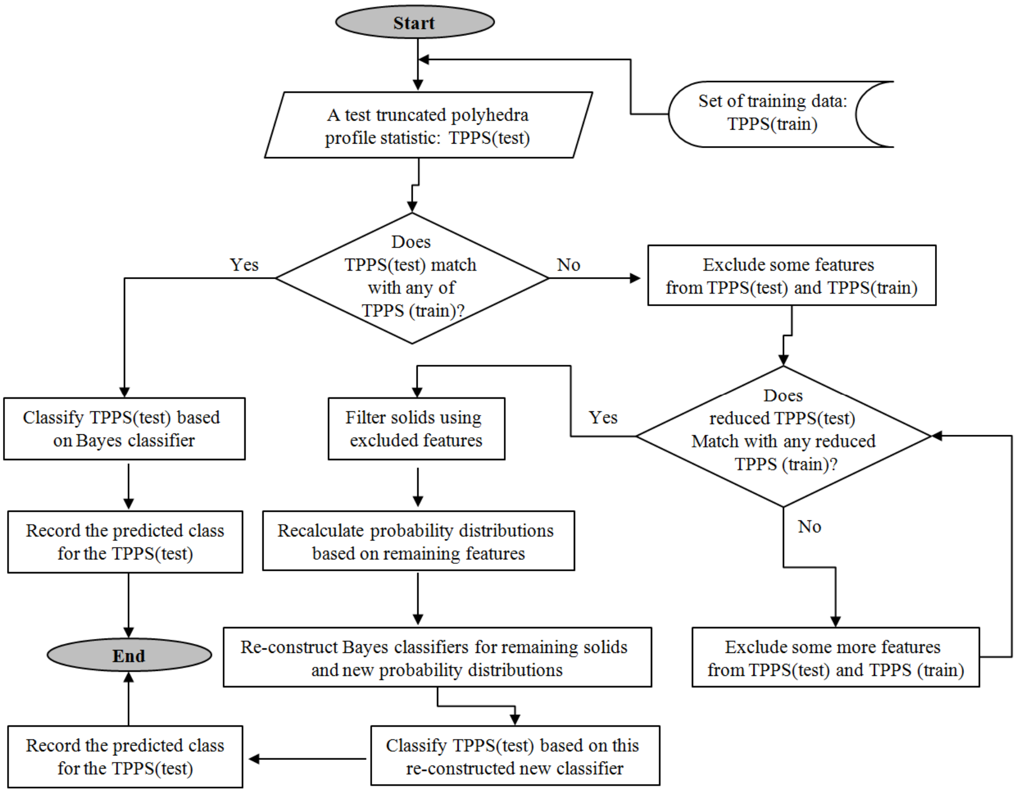

Start
Set of training data: TPPS(train)
A test truncated polyhedra profile statistic: TPPS(test)
Yes
Does TPPS(test) match with any of TPPS (train)?
No
Exclude some features from TPPS(test) and TPPS(train)
Classify TPPS(test) based on Bayes classifier
Filter solids using excluded features
Yes
Does reduced TPPS(test) Match with any reduced TPPS (train)?
Record the predicted class for the TPPS(test)
Recalculate probability distributions based on remaining features
No
End
Re-construct Bayes classifiers for remaining solids and new probability distributions
Exclude some more features from TPPS(test) and TPPS (train)
Record the predicted class for the TPPS(test)
Classify TPPS(test) based on this re-constructed new classifier

Figure 6.9: The hierarchical approach for selecting features and Bayes classifiers construction.

Feature Combination	**Misclassification Probability**
N_{TV}, N_{TF}, N_{TE}	0.7108
N_{TV}, N_{TF}, N_{TE}, ATN_F^*, ATN_V^*	0.1716
N_{TV}, N_{TF}, N_{TE}, ATN_F^*, ATN_V^*, *eADJt*	0.0236
All features combined	0.0192

Table 6.4: The changes in misclassification probabilities of Sphenocorona when some features are excluded.

However, the $L^*(.)$ increases significantly with the exclusion of features. For example, the above table shows the effect of exclusion on $L^*(.)$ for the 86th Johnson solid, the Sphenocorona. The distributions of $L^*(.)$ from 55 solids for different feature sets are provided in Appendix 6.3.

Effects of Filtering on Classification

Another question to assess is: if the filtering has any effect on classification. As discussed in Section 3.5 (Chapter 3), the *eADJ* and *fADJ* have capabilities to discriminate the Platonic and Archimedean solids from the Johnson solids. So *eADJ* and *fADJ* may help to separate the Johnson solids from others.

To evaluate this effect, $L^*(.)$ is calculated for 55 solids before and after filtering when *TPPS* excludes *eADJt* and *fADJt*. Let ${}_{D}L^*(i) = {}_{F}L^*(i) - {}_{NF}L^*(i)$, where ${}_{F}L^*(i)$ and ${}_{NF}L^*(i)$ are $L^*(i)$ with and without filtering respectively. We expect that ${}_{D}L^*(.)$ should not have a degenerate distribution at zero and the following plot confirms this.

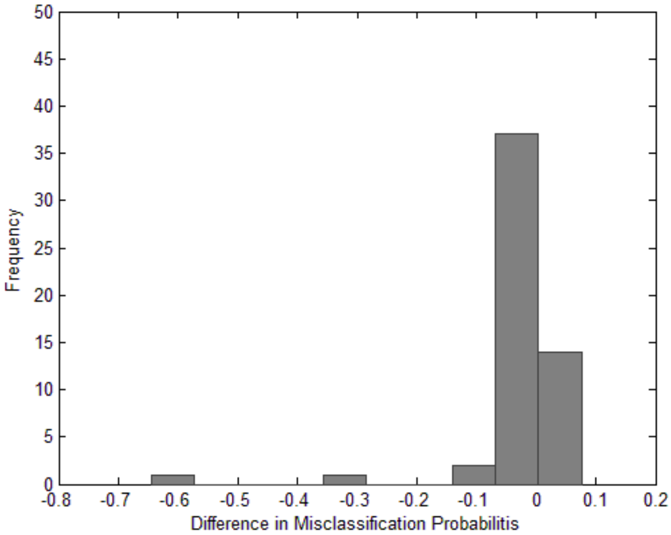


Figure 6.10: The plot showing the effects of filtering on overall misclassification probabilities.

6.8.6 Predicted Metabolosome Shapes

As mentioned before, a *TMPS* may not exactly match with any of the *TPPS*. So the metabolosomes are treated with the hierarchical feature selection approach. The following table gives the predicted shapes and corresponding frequencies for metabolosomes.

The most frequent predicted shapes are Sphenocorona (J_{86}), Sphenomegacorona (J_{88}), Augmented Sphenocorona (J_{87}) and Elongated pentagonal bipyramid (J_{16}). The other solids are Metabidiminished icosahedron (J_{62}), Gyroelongated pentagonal pyramid (J_{11}), Gyroelongated triangular cupola (J_{22}), Augmented Elongated square bipyramid (J_{15}) and Augmented and biaugmented pentagonal prisms (J_{52}, J_{53}) where 'J' stands for Johnson solid.

Predicted Solids	**Solid Type**	**Frequency**
Sphenocorona	J_{86}	8
Sphenomegacorona	J_{88}	6
Augmented Sphenocorona	J_{87}	5
Elongated pentagonal bipyramid	J_{16}	3
Other Solids	$J_{11,}$ J_{15}, J_{22}, J_{52}, J_{53}, J_{62}	8

Table 6.5: Predicted shapes for metabolosomes and their frequencies, using Bayes classifiers. 'J' stands for Johnson solid followed by Johnson solid numbers.

6.8.7 Conclusions

The predicted metabolosome shapes form a small number of Johnson solids [125]. Notably, these predicted shapes agree with the results from polyhedral structural distance model (Chapter 5).

However, excluding features increases $L^*(.)$, that means, if a *TMPS* needs too many features excluded, the misclassification probability may not be negligible. Also the Bayes classifiers are very sensitive with respect to errors in data, i.e., during manual feature identification in metabolosomes, if one or more features are wrongly recorded, the impact may be large. An analysis is carried out to identify the effects of data errors on the Bayes classifiers in Section 6.11.2.

6.9 Linear Discriminant Analysis

The Bayes classifiers are very sensitive to data errors, hierarchical feature selection may increase $L^*(.)$ and distance based profile selection could not be used due to weight selection problems. So some classifiers are developed where all features are utilized and expected to be less sensitive with respect to data errors. One of them is the linear discriminant analysis (LDA).

The LDA [57] is used extensively for classification problems in many varied fields; for example speech analysis [166], facial recognition [167], [168], medical image processing [169], bioinformatics [170], and data mining [171] are a few of them. Here we develop classifiers for incomplete polyhedral structures using LDA.

The standard algorithm behind LDA is well known and available in many statistics books, such as [172], [173]. Briefly, suppose there are only two standard polyhedra classes, say SP_1 and SP_2, x is the feature vector (*TPPS*), μ_1 and μ_2 are the class means (of *TPPS*) and Σ_1 and Σ_2 are their covariance matrices respectively. The Fisher's LDA defines a linear combination $w \cdot x$ and calculates the ratio:

$$S = \frac{(w \cdot (\mu_1 - \mu_2))^2}{w' \left(\Sigma_1 + \Sigma_2 \right) w}$$

which is actually the ratio of the variance between classes to the variance within classes. The maximum separation between two classes occurs when S is maximum, i.e. $w \propto (\Sigma_1 + \Sigma_2)^{-1}(\mu_1 - \mu_2)$. The Fisher's LDA assigns a new x to SP_2 if $w \cdot x > c$ for some threshold constant c.

However, Fisher's LDA is nonparametric; but the solution of LDA is optimal only when the distributions of samples from different classes satisfy the homoscedastic Gaussian model, i.e., distinct mean vectors but with the same covariance matrix for all the classes [174]. Also, Fisher's LDA was initially developed as a binary classifier, and is generalized by [175] for multiclass classification. In the following section, we discuss applying LDA to our problem.

6.9.1 LDA on Truncated Polyhedra

The training data selection for the LDA uses the scheme explained in section 6.7.1 and the training *TPPS* contain 231 features. The constant features (e.g., the first entry from *eADJt*) are removed to avoid singularity problems [176] prior to calculation. The training data with reduced dimension is used to develop the LDA classifiers using the library 'MASS' [177] in R [144].

The constant features from the test data are also removed to match the dimension of the training data and classified using LDA classifiers. The LDA predicted classes for test data are further analyzed to estimate the misclassification probabilities as discussed in Section 6.8.3. Finally, these rules are applied to predict the metabolosome shapes.

The MASS library in R uses moments as standard and unbiased estimators of the mean and variances [177] and calculates the prior probabilities of class membership from data, which is in fact, 1/55 for all cases. It also transforms observations to discriminant functions, normalized such that within groups, the covariance matrix is spherical [177]. Classifying new data results in its *maximum a posterior* (MAP) class as well as the posterior class probabilities.

6.9.2 LDA Predicted Metabolosome Shapes

The most frequent predicted shapes are Gyroelongated pentagonal pyramid (J_{11}), Elongated pentagonal bipyramid (J_{16}), Sphenocorona (J_{86}) and Augmented Sphenocorona (J_{87}). The other solids are Metabidiminished icosahedron (J_{62}), Sphenomegacorona (J_{88}), Elongated square bipyramid (J_{15}), Augmented pentagonal prisms (J_{52}) and Gyroelongated square bipyramid (J_{17}). Clearly all of them are Johnson solids [125]. The following table gives the frequencies of the predicted solids:

Predicted Solids	**Solid Type**	**Frequency**
Gyroelongated pentagonal pyramid	J_{11}	7
Elongated pentagonal bipyramid	J_{16}	10
Sphenocorona	J_{86}	5
Augmented Sphenocorona	J_{87}	2
Other Solids	$J_{62,}$ J_{88}, J_{15}, J_{52}, J_{17}	6

Table 6.6: The frequency of LDA predicted metabolosome shapes.

Since the LDA also calculates posterior probabilities for each class for each *TMPS*, it may be useful to see the posterior probabilities for MAP and other nearest classes. These probabilities are provided in Appendix 6.4.

The posterior probabilities for the first solid and differences in probabilities between first and second solids are substantially large in most of the cases. This indicates the classes for the *TMPS* are predicted with high certainty. However, as a classifier, the LDA would be a strong one if overall misclassification probabilities, i.e., $L^*(.)$ are small. These probabilities are analyzed in the next section.

6.9.3 LDA Misclassification Probabilities

The misclassification probabilities for each class are calculated based on the test data, as described in Section 6.8.3. The distribution of $L^*(i)$, $i = 1, 2, \ldots, 55$ is shown in Figure 6.11. It shows, there are quite a few classes (65.45%) with $L^*(.) \geq 0.10$. The minimum, maximum, mean and SD of $L^*(.)$ are 0.0000, 0.6336, 0.1791 and 0.1521 respectively. These summary statistics indicate that the LDA is not a very strong classifier for this problem. Appendix 6.2 provides the distributions and summary of overall misclassification probabilities from the LDA for different truncation proportions.

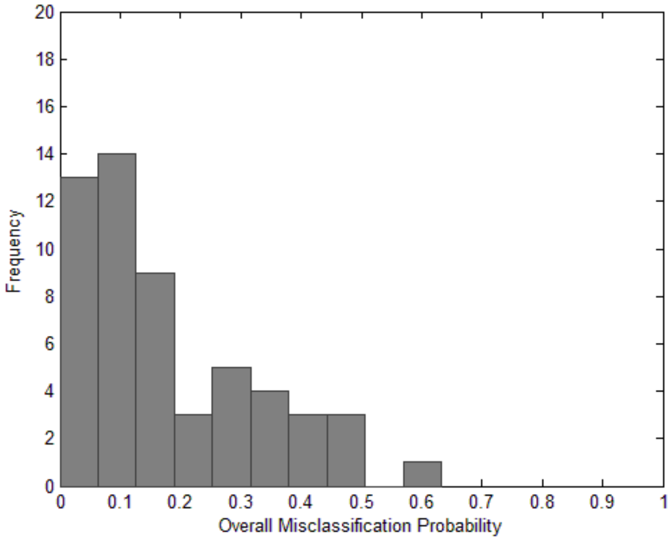


Figure 6.11: The distribution of overall misclassification probabilities for 55 solids, obtained from LDA. The largest misclassification probability is from J_{36}.

One expected reason could be that some of the features from some of the solids are overlapped. Figure 6.12 confirms this fact for J_{16}, whose 21.48% profiles are classified as J_{27} ('J' stands for Johnson solids). The plot displays two features - N_{TV}, and N_{TE} from J_{16} and J_{27}. These two features have strong impacts too on LDA, because the coefficients of the linear discriminants corresponding to these two variables are considerably large (for first linear discriminant, these coefficients are –0.1791 and 1.440 respectively, where the largest coefficient is –2.047 and the smallest coefficient is –0.0048, in absolute terms).

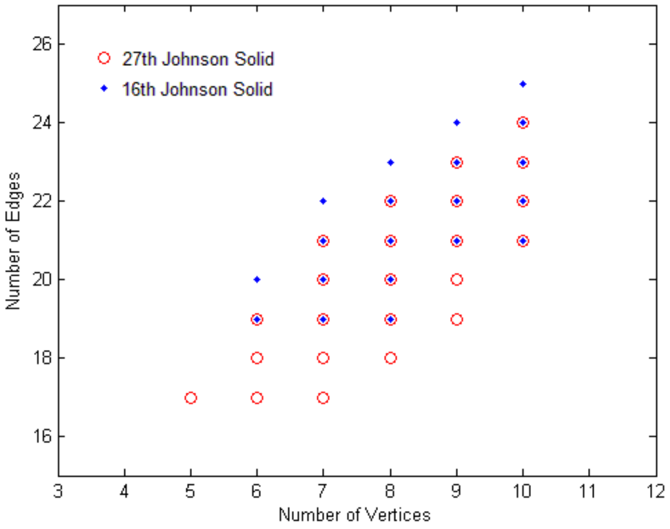


Figure 6.12: Overlapped distributions of vertex and edge counts from two Johnson solids: J_{16} and J_{27}.

6.9.4 Conclusion

There are some good points for the LDA on truncated polyhedral shapes. The LDA predicted shapes are in line with Bayes classifiers as well as the polyhedral structural distance model. It is simple to implement, efficient in terms of computation and memory usage. It is less sensitive with respect to data error (Section 6.11.2) than the Bayes classifiers and uses all the features from the *TPPS*. Also, the posterior class probabilities of the predicted classes of the metabolosomes are very high.

However, the LDA does not give good overall misclassification probabilities. The reasons may be the overlapping in the *TPPS* distributions, the validity of the assumptions, etc. So we need to consider a classifier which can handle overlapped distributions and can address the problems of the Bayes classifiers and the LDA. The support vector machine is one of them used here.

6.10 Support Vector Machine

The Support vector machines (SVM) or support vector networks (SVN) are supervised learning methods, frequently used for classification and regression analysis. The SVM was proposed by [178], [58] as a new learning machine for binary classification problems [179] and after that, it has been extensively applied to several different areas with the name Support Vector Machine [180]. For example, face detection [181], [182], image classification [183], object recognition [180], [184], [185], and hand writing and digital recognition [186], [187] are a few of them. We developed classifiers using SVM to classify truncated polyhedral structures. The metabolosome shapes are then predicted using these classifiers. In the subsequent sections we explain the procedure.

One of the strengths of SVM over LDA is: SVM classifies with better accuracy in many cases where linear classifiers end with high error rates [188]. The SVM applies kernel transformations to the training data so that the classes become well separable in the kernel space (often with higher dimension than the feature vector's dimension) by the decision plane [189]. The standard algorithm behind SVM learning and their properties are explained in some detail in [54].

6.10.1 Binary and Multi-class SVM

The initial SVM algorithms are developed as binary classifiers [58], [178], and later extended for multi-class classification problems [190]. Many of these extensions divide a multi-class problem into a set of several binary classification problems and finally combine those binary classifiers [191], and a few of them consider it as a single optimization problem, e.g. [192].

Here we adopted the first approach, i.e. combination of binary classifiers. In this method, the predicted class is determined using "one-vs-all" or "one-vs-one" voting methods [193], [194]. Though posterior class probabilities for a test object can also be calculated [195], [196], [197], the voting scheme is more appropriate for class selection [193], [198].

Since there are 55 classes of truncated polyhedra, the one-vs-one voting scheme considers all possible pairs from 55 classes, i.e., total ${}^{55}C_2$ = 1485 pair of classes and assigns a test object to a class from each pair using the SVM classifiers. Let $SP_i\,(j)$ =1 if the test object is classified to SP_i for j^{th} pair, j = 1, 2, …, 55 and zero otherwise. Then the total 'vote' for SP_i is $VSP_i = \sum_j SP_i\,(j)$. If $argmax_{\,k}\, VSP_k = c$, the test object is classified to SP_c.

Among the two voting schemes, the one-vs-one method is generally preferred over one-vs-all [193], though some studies defend the power of the later [199]. We use 'one-vs-one' voting scheme for classifying our incomplete polyhedral shapes.

6.10.2 Parameters for SVM Learning

The development of SVM classifiers needs to specify the kernel function. The Gaussian Radial Basis kernel is one of the most popular kernels used for SVM [189] and for our problem we also preferred this kernel. However, the data must be scaled to zero means and unit variances prior to calculation. This kernel is defined as:

$$k(x_1, x_2) = exp\ (-\frac{1}{2\sigma^2}\ ||x_1\ -\ x_2||^2)$$

where x is the observed data and the hyper-parameter σ is the Gaussian kernel width [200]. The calculation is carried out using the 'kernlab' [201] library in R [144]. This library uses a heuristic method [202] to estimate σ.

Among other parameters the cost of constraints violation [201] is set to 5. A 3-fold cross validation [201] on the training data is performed to assess the quality of the model, i.e., the accuracy rate for the SVM classification. As discussed in the previous section, the prediction for test data and metabolosomes use the 'one-vs-one' voting scheme.

We started with default parameters provided in the 'kernlab' library. The kernel is selected and other parameters are tuned by running the SVM program several times with different combination of parameters. The combination with best performance was selected for further calculations. The σ is estimated by the 'kernlab' library itself, so we did not work on estimating this parameter.

6.10.3 SVM for Truncated Polyhedra

Section 6.7.1 described the training and test data selection. The same training data is used to develop the SVM classifiers. The feature vector for each data point has length 231. Each test *TPPS* is then classified using these developed rules.

Finally the overall misclassification probability $L^*(.)$ for SVM for each class is calculated using the method described in 6.8.3. The minimum, maximum, mean and SD of these errors are 0.0000, 0.3360, 0.0394 and 0.0621 respectively. Since these values are considerably small, we conclude that, similar to the Bayes classifiers, the SVM also provides good classification for this problem. In fact, 87.27% of the solids have $L^*(.) \leq 0.10$. Figure 6.13 shows the distribution of $L^*(.)$, calculated from the SVM. Appendix 6.2 provides the distributions and summary of overall misclassification probabilities from SVM for different truncation proportions.

One observation is: the solids with largest misclassification probabilities from these three methods. For LDA, the largest misclassification probability came from J_{36} and for other two methods, it is J_{35}. The distribution of the misclassification probabilities is bimodal, irrespective of the classification method used. We found that, the more symmetric and simpler objects (Archimedean and Platonic solids) tend to have lower misclassification probabilities from all three methods. This is expected, because, the Archimedean and Platonic solids have some strong distinguishing features e.g. vertex type and face type. However, the Johnson solids appear in either groups.

6.10.4 SVM Predicted Metabolosome Shapes

Each of the *TMPS* is classified using the SVM developed rules using the one-vs-one voting scheme. Unlike the Bayes classifier, filtering is not required and each *TMPS* has a length of 231. The following table (Table 6.7) gives the predicted shapes for metabolosomes and corresponding frequencies.

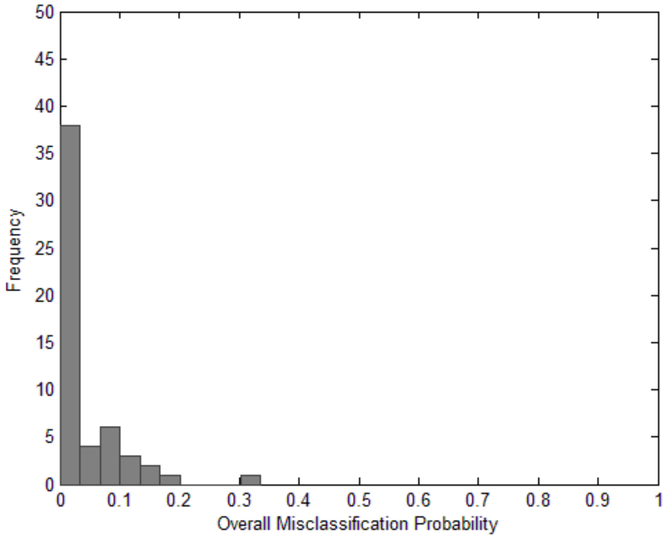


Figure 6.13: The distribution of overall misclassification probabilities for 55 solids, resulted from SVM. The largest misclassification probability is from J_{35}.

The most frequent predicted shapes (Table 6.7) are Sphenocorona (J_{86}), Sphenomegacorona (J_{88}), Augmented Sphenocorona (J_{87}), Elongated pentagonal bipyramid (J_{16}) and Gyroelongated pentagonal pyramid (J_{11}). The other predicted solids are Metabidiminished icosahedron (J_{62}), Gyroelongated square bipyramid (J_{17}), Biaugmented triangular prism (J_{50}) and Augmented hexagonal prism (J_{54}).

Predicted Solids	Solid Type	Frequency
Sphenocorona	J_{86}	8
Augmented Sphenocorona	J_{87}	3
Sphenomegacorona	J_{88}	2
Elongated pentagonal bipyramid	J_{16}	9
Gyroelongated pentagonal pyramid	J_{11}	4
Other Solids	J_{17}, J_{50}, J_{54}, J_{62}	4

Table 6.7: Predicted shapes for metabolosomes and their frequencies, using SVM. 'J' stands for Johnson solid followed by Johnson solid number.

6.11 Classification Summary

6.11.1 Summary of the Metabolosome Shapes

The SVM predicted solids are similar to those are predicted using the Bayes classifiers, LDA and the polyhedral structural distance model (Chapter 5). The following table shows the predicted solids across all three methods.

Predicted Solid Name	Frequency		
	Bayes	LDA	SVM
Gyroelongated pentagonal pyramid	2	7	4
Elongated square bipyramid	1	1	0
Elongated pentagonal bipyramid	3	10	9
Gyroelongated square bipyramid	0	2	1
Gyroelongated triangular cupola	2	0	0
Elongated square bipyramid	0	0	1
Augmented pentagonal prism	1	1	0
Biaugmented pentagonal prism	1	0	0
Augmented hexagonal prism	0	0	1
Metabidiminished icosahedron	1	1	1
Sphenocorona	8	5	8
Augmented Sphenocorona	5	2	3
Sphenomegacorona	6	1	2

Table 6.8: The frequency of the predicted solids from three classification methods. The solids having non-zero frequencies from all three methods are highlighted.

Interestingly, there are just 6 Johnson solids predicted through all classifiers. These 6 solids constitute about 86.66%, 86.66% and 90% of the metabolosomes respectively. It suggests that, no matter what methods are used, the metabolosomes have very particular shapes. The results in the above table also suggest an entire new class of shapes for bacterial microcompartments as opposed to previous studies on the Carboxysomes.

6.11.2 Prediction Agreement Matrix

The metabolosome shapes are predicted by four methods: polyhedral structural distance model (*PSDM*), Bayes classifiers (*BC*), LDA and SVM. Let the i^{th} metabolosome be predicted as ${}_iS_{PSDM}$, ${}_iS_{BC}$, ${}_iS_{LDA}$ and ${}_iS_{SVM}$ by these four methods respectively, i = 1, 2, …, 30.

Let *PSDM* and *BC* 'agree' for i^{th} metabolosome if ${}_iS_{PSDM} = {}_iS_{BC}$. Hence, the total number of agreement between ${}_iS_{PSDM}$ and ${}_iS_{BC}$ is

$$AG(PSDM, BC) = \sum_i I_{\left({}_iS_{PSDM} = {}_iS_{BC} \right)}$$

where, $I(.)$ is the identity function. For example, let us consider the following matrix ($AM_{PSDM,BC}$) for *PSDM* and *BC*. The entry in (1, 2) location shows there is one metabolosome which is predicted as J_{10} by *PSDM* and J_{11} by *BC*. It is not an agreement. Clearly, *AG(PSDM, BC)* = *trace*($AM_{PSDM,BC}$) and here *AG(PSDM, BC)* = 12.

PSDM vs BC

	J_{10}	J_{11}	J_{15}	J_{16}	J_{17}	J_{22}	J_{50}	J_{52}	J_{53}	J_{54}	J_{62}	J_{86}	J_{87}	J_{88}	J_{99}
J_{10}	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{11}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{15}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{16}	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
J_{17}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{22}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{50}	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
J_{52}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{53}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{54}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J_{62}	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
J_{86}	0	0	0	0	0	0	0	0	0	0	0	4	2	0	0
J_{87}	0	1	0	2	0	2	0	0	1	0	0	4	3	2	0
J_{88}	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0
J_{99}	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

The 'J' stands for Johnson solids, followed by their identification number. J_{99} is actually icosahedron which is a Platonic solid. The Table 6.9 gives the agreements among all 4 methods.

	PSDM	Bayes	LDA	SVM
PSDM	30	12	5	13
Bayes		30	13	15
LDA			30	15
SVM				30

Table 6.9: Prediction agreement among 4 classification methods.

This table shows that the agreement between polyhedral structural distance model and the LDA is least and the SVM matches with other methods in maximum cases. Interestingly, still all 4 methods eventually suggest that there are only about 6 classes.

6.12 Comparing Classifiers

Comparing classifiers first requires a measurement of their performance. The performance of a classifier can be evaluated using many factors such as time and space complexity, or misclassification errors [203]. However, since no universally 'best' classifier exists and performance of a classifier widely varies across applications [171], our performance evaluation and comparison of classifiers must address our particular problem, i.e., incomplete polyhedral shape classification.

There are several methods to compare classifiers [203], [204], [205] and we adopted a common evaluation method - comparing misclassification probabilities across methods. We compare these probabilities corresponding to each standard polyhedral class, calculated based on the Bayes, LDA and SVM based classifiers.

6.12.1 Comparing Misclassification Probabilities

Let the misclassification probabilities for SP_i be ${}_1L^*(i)$, ${}_2\mathrm{L}^*(i)$ and ${}_3L^*(i)$ for the Bayes classifier, LDA and SVM respectively, i = 1, 2, …, 55. The following box plots give distributions of ${}_1L^*(i)$, ${}_2L^*(i)$ and ${}_3L^*(i)$, calculated based on 231 features.

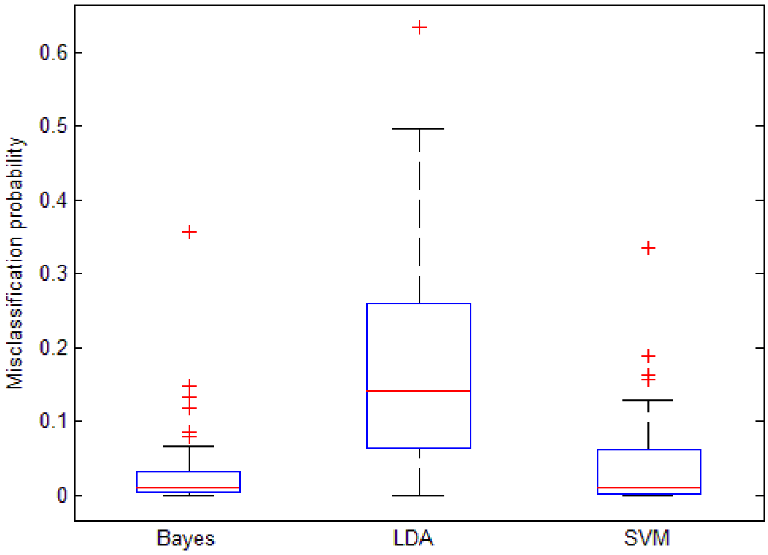


Figure 6.14: The box plots showing distributions of L*(.), calculated from three methods.

The plot shows that the Bayes classifiers are the best predictor followed by SVM. As we found in previous sections, the LDA does not provide good classification for this problem. If m_1, m_2 and m_3 are the medians of ${}_1L^*(.) - {}_2L^*(.)$, ${}_2L^*(.) - {}_3L^*(.)$, and ${}_1L^*(.) - {}_3L^*(.)$ respectively, the null hypothesis H_0: $m_1 = 0$ against not H_0 is rejected by the Wilcoxon signed rank test [206] with p-value 0.0000. A similar result is also obtained for H_0: $m_2 = 0$ against not H_0. But the same test cannot reject the H_0: $m_3 = 0$ against not H_0 (p-value = 0.0917).

However, it is also important to examine ${}_1$L*(.), ${}_2$L*(.) and ${}_3$L*(.) for individual solids, at least for the leading predicted solids for the metabolosomes. Table 6.10 gives these probabilities.

Predicted Solid Name	**Misclassification Probability**		
	Bayes	**LDA**	**SVM**
Gyroelongated pentagonal pyramid (J_{11})	0.0052	0.0328	0.0204
Elongated pentagonal bipyramid (J_{16})	0.0004	0.3380	0.0000
Metabidiminished icosahedron (J_{62})	0.0092	0.1224	0.0588
Sphenocorona (J_{86})	0.0192	0.1740	0.0196
Augmented Sphenocorona (J_{87})	0.1180	0.2952	0.1064
Sphenomegacorona (J_{88})	0.0368	0.2404	0.1064

Table 6.10: The misclassification probabilities for the leading predicted solids and common in three methods.

While considering the Bayes classifiers, the maximum misclassification probability is 0.1180. The other values are even smaller than 0.05. For the SVM, the maximum is 0.1064 and the LDA shows comparatively high misclassification probabilities. Also, Bayes classifiers are only slightly outperforming the SVM in these classes. We conclude that the leading shapes for the metabolosome correspond to good prediction accuracies.

6.12.2 Effects of Data Error

As discussed in Chapter 4 and Chapter 5, the metabolosomes features are collected manually. In addition, the truncation planes for metabolosomes may not be exactly parallel as in the simulation study. They may have varying unknown truncation proportions at the ends, contain segmentation errors and polyhedral approximation leaves out some structural information. So it is possible that some features from metabolosomes are missed or wrongly recorded. Since the features have inter-dependencies, one error in one feature affects the whole statistic. We term this as *data error*.

We developed another classification scheme based on these three classifiers individually to test the effects of data error on $L^*(i)$, $i = 1, 2, \ldots, 55$. The algorithm is same for all three classifiers - the following steps describe the scheme in general.

Step 1: Training Step

As with previous classifiers, 2500 *TPPS* are considered from each of the standard solids as training data. This training dataset develops the classification rule, based on the classification method (Bayes, LDA or SVM) applied. This step is exactly the same as previous classifiers developments.

Step 2: Introducing Errors in Test Data

Recalling the simulation section of this chapter, the rotated solid (*REP*) was truncated by two planes $P_1 \parallel P_2$, perpendicular to the z-axis (Step 3, Section 6.4.4). This truncation generated *TSP* and *TPPS* was recorded from this *TSP*.

Here, *only* for test data, we randomly remove *one more vertex* from *TSP*, which is equivalent that a vertex from metabolosome is missed during data collection. Let this dropped vertex *TSP* be denoted by *TSP** and corresponding *TPPS* is denoted by *TPPS**. An important point is, *TPPS** cannot be obtained just by adding some random integer noise to *TPPS*, because *TPPS** should also contain the information about inter-dependencies among features.

Step 3: Classifying TPPS using the rules from TPPS*

The *TPPS** is classified using the rules developed in Step 1. Similar to Section 6.8.3, the overall misclassification probabilities are calculated for each class based on predicted shapes of *TPPS**. The following diagram displays this new scheme.

Results

The misclassification probabilities are calculated for the Bayes, LDA, SVM based classifiers and the Bayes with filtering. The effects of introducing data error to the test data is assessed through comparing these error rates. The analysis is carried out based on truncation proportion = 0.10. Figure 6.16 compares the distribution of these probabilities calculated from 4 methods.

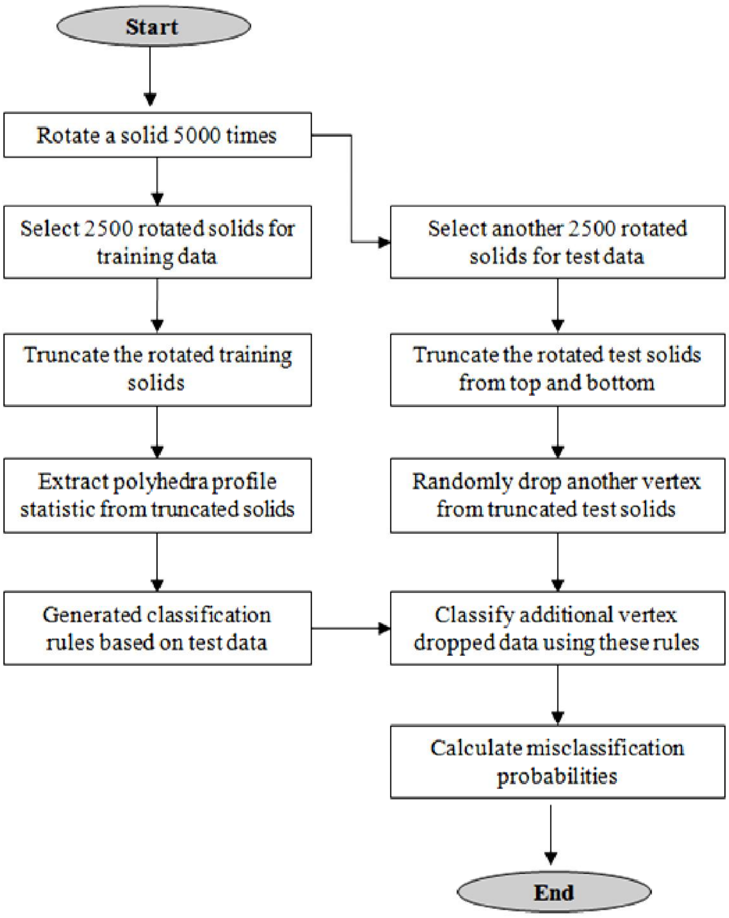

Start
Rotate a solid 5000 times
Select 2500 rotated solids for training data
Select another 2500 rotated solids for test data
Truncate the rotated training solids
Truncate the rotated test solids from top and bottom
Extract polyhedra profile statistic from truncated solids
Randomly drop another vertex from truncated test solids
Generated classification rules based on test data
Classify additional vertex dropped data using these rules
Calculate misclassification probabilities
End

Figure 6.15: The flowchart describing the classification method for measuring data error.

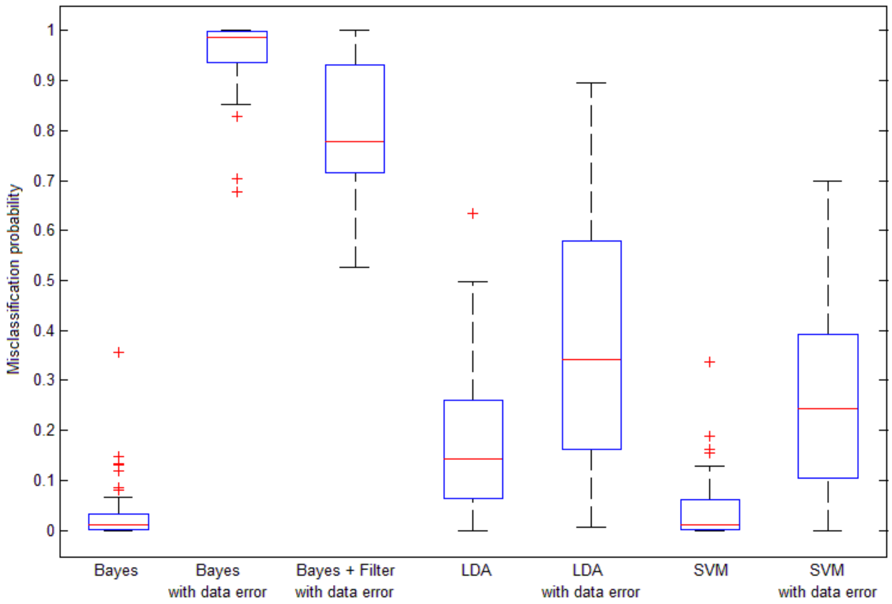


Figure 6.16: The distributions of the misclassification probabilities across three classifiers in comparison with the same when there are data errors in test data.

The results show that the Bayes classifiers are most affected by introducing test data errors and the SVM is least effected here. These probabilities from the LDA are also increased significantly. Table 6.11 summarizes the misclassification probabilities in the presence of data errors.

	LDA	**SVM**	**Bayes**	**Bayes + Filter**
Minimum	0.0072	0.0000	0.6768	0.5276
Maximum	0.8956	0.6984	1.0000	1.0000
Mean	0.3832	0.2606	0.9598	0.7968
Standard Deviation	0.2484	0.1853	0.0661	0.1347

Table 6.11: Summary statistics of the misclassification probabilities in presence of data error across classifiers.

The very high misclassification probabilities in Bayes classifiers are not surprising, because the test data did not appear in the training data library in most of the cases, and they are treated as misclassified. In the presence of filtering, the classifier is naturally based on less than 231 features which can lead to higher error rates. The SVM gives best error rates among all methods in presence of data error.

6.13 Summary and Conclusion

This section summarizes the incomplete polyhedral shape classification methods and the results. We developed the Bayes classifiers and a modified form of it for this supervised learning problem. We also applied the LDA and the SVM for this problem. The predicted shapes for the metabolosomes were very consistent across classifiers. In addition, these shapes are also predicted when the polyhedral structural distance model was used. There are just 6 leading shapes and all of them are the Johnson solids, as opposed to a previous study on Carboxysome which was found to have a Platonic solid shape.

The full feature based misclassification errors are minimum in the Bayes classifiers, followed by the SVM and LDA. When the Bayes and SVM classifiers are considered, the predicted solid classes have considerably smaller overall misclassification probabilities. However, Bayes classifiers with filtering approach may have higher misclassification probabilities. We also found that errors in test data or in a classifying object have a large impact if the Bayes classifiers are used. This impact is least for the SVM, followed by the LDA. Hence, considering all results, the SVM is a 'strongest' and the most 'reliable' classifier for this problem.

Chapter 7

Conclusions and Future Research

The first section of this chapter presents concise notes on the research problem, the solutions we developed and the results we found. In Section 7.2, the advantages and disadvantages of this work are discussed. Some scope of future improvements related to these methods are discussed in Section 7.3. Finally, we focus on how these methodologies could be rolled out for more routine use, e.g. applying these methods to similar other problems (Section 7.4).

7.1 Methods and Results Summary

The goal of this thesis is a methodological development for classifying incomplete polyhedral structures and to apply these methods to predict 3D polyhedral structures for a recently identified bacterial inclusion, called metabolosome.

7.1.1 Statistical Problem, Sampling and Inference

This is a statistical classification problem through supervised learning; but unlike general classification problems, here the training data does not consist of the measurements of the representative objects from each class. The training data does not contain any measurements of the standard polyhedra, rather we 'measured' their transformed forms (truncated polyhedra). Hence this particular classification problem comes with an additional mapping problem: relations among the standard solids and their truncated forms.

Sampling

This mapping problem here, in fact, brings the notion of sampling. A standard polyhedron may produce a set of infinitely large number of truncated shapes; hence an efficient sampling scheme is essential to collect a finite set of truncated shapes such that this finite set can well represent the original complete shapes. In Section I of Chapter 6, the simulation procedure we developed is actually such a sampling scheme. We also showed that, the set of samples is a well representative of the population (Section 6.4.5).

Inference

A statistical classification problem (here through supervised learning) also can be formulated as a standard inference problem, but instead of significance testing, here the misclassification probabilities are to be considered. Also, unlike usual hypothesis testing, for a classification problem the test statistic is unknown and the main purpose is to 'learn' this statistic (also known as classification rule). In general terms, here the hypothesis to test is:

$$H_0: M \in \mathcal{C}(P_i) \text{ against not } H_0 \text{ for a particular } i,$$

where i = 1, 2,..., number of the standard polyhedron, M is an incomplete metabolosome and $\mathcal{C}$ (P_i) is the class of *truncated* polyhedral shapes from the i^{th} standard polyhedron. In other words, the central statistical problems we addressed here are: what is the best possible polyhedral class for a given incomplete metabolosome and what is the probability that the incomplete metabolosome will belong to this predicted class. Hence both of sampling and inference are accounted for this work to a great extent.

7.1.2 Imaging, Reconstruction and Segmentation

The research starts with the raw images from cryo-electron tomography. These raw images were reconstructed through IMOD [86] software to generate 3D stacked images. The reconstructed three-dimensional images were then trimmed, sliced, manually segmented and smoothed to generate final metabolosome images. We processed 30 metabolosomes for further analysis. In this step, we developed a least-squares based method to combine 3-way segmented image boundaries. Visual inspections showed the metabolosomes have convex polyhedral structures.

7.1.3 Fundamental Structural Properties

Statistical analyses describe the fundamental structural properties, such as volume, symmetry, and aspect ratio of the reconstructed and segmented metabolosomes. We found that the metabolosomes widely vary in size. The largest metabolosome is about 80 times larger than the smallest one. The minimum and maximum of the aspect ratio of the metabolosomes are 0.4251 and 0.8017 respectively with a mean of 0.6207. This shows the metabolosomes are non-spherical.

The analyses also show the ratio of the longest to shortest edge lengths ranges between 1.7855 and 4.4306 with a mean of 3.0866. So, the metabolosomes are non-uniform or deformed in structures.

In summary, the metabolosomes have widely varied sizes, with non-spherical and deformed polyhedral structures. This is the first investigation to produce these findings for these properties of the metabolosomes. These findings differ significantly from the studies on another bacterial microcompartment named as Carboxysomes which show that those objects have uniform, highly symmetric convex polyhedral shapes [12], [10].

7.1.4 Polyhedral Structural Distance Model

Since the reconstructed objects suffer from missing wedges and they vary widely in shapes and sizes, the existing shape averaging methods are not useful here. So we developed an algorithm to logically 'complete' these incomplete structures and a statistical method to find the best fit standard polyhedral shapes to these completed structures (Chapter 5).

This analysis demonstrates that 29 of 30 metabolosomes have shapes from the Johnson solids. The leading predicted solids are Sphenocorona, Augmented Sphenocorona and Sphenomegacorona which are 86th, 87th and 88th Johnson solids respectively. Previous work found that a similar microcompartment (Carboxysomes) to have an icosahedral shape [12], [10] which is a Platonic solid.

7.1.5 Polyhedron Profile Statistic

Analyzing shapes requires a shape descriptor. Here we developed a shape descriptor for complete and incomplete polyhedral structures and named it the *Polyhedron Profile Statistic* (Section 3.6, Chapter 3 and Section 6.3.4, Chapter 6). This feature vector not only contains vertex, edge or face counts of a polyhedron, it also captures more complex topological properties of a polyhedron. Importantly, this feature vector is invariant under some conditional deformation and hence can describe the deformed polyhedral objects as well.

7.1.6 Simulation for Shape Library

Predicting shapes of the incomplete metabolosomes is fundamentally a supervised learning problem. The standard polyhedral shapes constitutes the classes for this problem and the objects themselves are described through a feature vector - polyhedron profile statistic. But we lacked sufficient samples to train this learning model. So we developed a simulation algorithm to generate samples of truncated polyhedral shapes. The standard polyhedra were randomly truncated followed by an automated extraction of their polyhedron profile statistic. This develops a library of truncated standard polyhedral shapes (Chapter 6, Part I).

7.1.7 Incomplete Polyhedral Shape Classification

Finally, we developed novel Bayes classifiers for the incomplete polyhedral shapes classification. In addition, we also applied LDA and SVM for the same classification problem. The misclassification probabilities evaluate the performance of these classifiers. Finally we compare and combine the classifiers and predicted the shapes of metabolosomes using these classifiers (Chapter 6, Part II).

The leading shapes from these classifiers are Gyroelongated Pentagonal Pyramid (J_{11}), Elongated Pentagonal Bipyramid (J_{16}), Metabidiminished Icosahedron (J_{62}), Sphenocorona (J_{86}), Augmented Sphenocorona (J_{87}) and Sphenomegacorona (J_{88}). Notably, these shapes are all Johnson solids ('J' stands for Johnson solids).

7.2 Discussion

7.2.1 Some Observations

Consistency in Results across Methods

The polyhedral structural distance model (Chapter 5) and three classifiers (Chapter 6, Part II) are applied to predict the shapes of the metabolosomes. There is a clear consistency: each method predicts about same set of shapes. We found that, though the metabolosomes widely vary in size, average edge length, aspect ratio, and other respects, they have only a very limited number of shapes and all of them are the Johnson solids (Chapter 3). The consistency of results across methods strengthens the claims about the validity of the prediction.

Performance of Classifiers

The polyhedral structural distance model cannot provide the misclassification probabilities, but other methods do. The analysis on the misclassification probabilities shows the Bayes classifiers have the lowest error rates and LDA has the highest error rates in general. However, SVM is almost equally as good as the Bayes classifier for this problem (Chapter 6, Part II).

But we also see that the Bayes classifiers can be affected largely in presence of data errors while SVM is least affected. Hence SVM is most 'reliable' for this classification problem. The highest error rate is about 10% in SVM and in the Bayes classifiers for leading metabolosome shapes, which we accepted (Chapter 6, Part II).

7.2.2 Strengths and Limitations

The methods have several unique strengths. The biggest overall strength is: these methods are able to work with non-symmetric or deformed objects and can operate on a single metabolosome at a time; thus these methods solve the problem of predicting shapes from incomplete structures even when no two objects are alike. Thus the main goal of this research is achieved.

These methods also have the additional advantage that they are not computationally intensive. However, these methods are developed specifically for convex polyhedral structures, so are limited to investigating convex polyhedral shapes only. They may not be applied to non-convex polyhedral shapes, for example, concave polyhedron, structures of proteins, cylindrical objects etc. The advantages and disadvantages, strengths and limitations of the individual methods are discussed here.

Claim 1: Three-way segmentation and LS method results better segmentation output.

Manual segmentation for these images is an accepted method. Now, three-way segmentation definitely gathers more information than the traditional one-way segmentation; but also accumulates more segmentation errors. The least squares (LS) method actually 'averages' the values of the same voxel obtained from three different segmentations and thus the errors are expected to be 'normalized' largely. The visual inspection also shows the superiority of the outputs from the three-way segmentation and the LS method than a one way segmentation. So the Claim 1 is justified conceptually and also visually.

However, we believe that three values of a voxel carry same reliability while segmented from three different directions; so there is no conflict that during LS estimation, these three values got the same weight (=1). However, in any other case, if it is realized that three values of a voxel are not equally reliable, then some other methods (e.g. logistic regression) may be more appropriate.

Claim 2: The approach to 'complete' an incomplete structure is efficient.

The approach is straightforward and a simulation study (Section 5.3) shows that if the solid is truncated up to by ≤ 30%, this algorithm will predict the correct parent shape for ≥ 90% cases. This is checked for the most common predicted solids. In addition, we established theoretically and visually that the metabolosomes are very likely to be truncated by ≤ 30%. Thus for this particular problem the algorithm is strong enough. Hence the Claim 2 is justified by the results from a simulation.

But this algorithm suffers with generalization issues. As discussed in Chapter 5, at least three incomplete edges are required to be visible for this algorithm to work, otherwise it may fail. It also depends on the coordinates of the terminal vertices of the incomplete edges and a cut-off proximity measure (we set it as 30 Å) - these two are subjective and situation dependent. Finally, this algorithm is not tested with more complicated solids (i.e. the solids with higher number of vertices and with more verities of faces).

Claim 3: The predicted shapes by the polyhedral structural distance model are relevant.

This claim is justified in two different ways. The predicted shapes by the polyhedral structural distance model are also in line with the shapes predicted by other three classification algorithms (Chapter 6). Each predicted shape is also visually verified whether the predicted shape can really fit to the metabolosome. For example, if the predicted shape contains a vertex with 5 edges, we verified whether the metabolosome can actually have such a vertex. If not, then we considered the second nearest shape. So the results in this particular problem are reliable. Hence the Claim 3 is justified.

However, this model has two serious drawbacks. This model does not provide any probability associated with its prediction; as if the results (predicted shapes) are perfect, but that is not the case. The second one is the parameter selection. The weights and the loss functions are assigned heuristically. Though we provided a justification on the parameter selection in Section 5.2.3, that is specific to this problem only. In future, a more sophisticated and generalized method for parameter estimation are required to be developed.

Claim 4: Polyhedron profile statistic is a good shape descriptor and deformation invariant.

This claim is justified indirectly. Except for only 8 pairs out of 123 standard polyhedra, this shape descriptor can uniquely identify the remaining standard solids (non-truncated). Secondly, while using this statistic for the Bayes classifier or SVM, the misclassification probabilities are in acceptable range in almost all cases. Third, this statistic consists of some very powerful topological features (such as vertex type and face type) of a standard polyhedron which even alone can separate the solids into different polyhedral families. In addition, we also explained in Section 3.6.2 that it is invariant under some restricted deformation. Hence, we believe that the polyhedron profile statistic is a good shape descriptor for this problem.

However, this statistic has some limitations. This shape descriptor is quite appropriate for convex standard polyhedra and (with some justifications) we assume that the metabolosomes are also convex standard polyhedra. However, for more complicated shapes (for example concave polyhedra) and for convex solids outside of these standard families, this shape descriptor may not be equally useful. However, extensive research is required to establish usefulness of this shape descriptor for other solids.

Finally, this statistic will definitely lose its deformation invariant property if any one of the three conditions (described in Section 3.6.2) are violated. Since there are infinitely many types of deformation possible, we need to justify the validity of these conditions before using this statistic for deformed polyhedral structures.

Claim 5: The approach for simulation is efficient to generate a good sample from truncated polyhedron population, resembling truncated metabolosomes.

The simulation study relies on three assumptions - the truncation proportions at both side are equal, the truncation is carried out by two *parallel* planes and the truncation proportions do not exceed by 15% at each side. All of these assumptions are justified to be approximately valid for the metabolosomes (in Section 6.2.2 and Section 2.2.5).

The quality of the sample is justified through the question: is there any new truncated polyhedra possible outside of this sample? This is justified in Section 6.4.5 and we found that increasing sample size (beyond 5000 per solid) does not result much increments in the new profiles. Hence the Claim 5 is justified - the simulation efficiently generated a 'good' sample, resembling truncated metabolosomes.

However, scenarios may occur when the truncation proportions are not the same at both sides or the truncation planes are not parallel. That may not be for the metabolosomes or similar ECT images, but may be for other applications. There this simulation is to be re-parameterized based on the problem. Hence this simulation here *does not* generate a general library of the truncated standard polyhedra.

Claim 6: The classifiers we developed are well capable to classify incomplete polyhedral shapes.

This is justified by the misclassification probabilities. There is nothing new approach we developed for classification, as the Bayes, LDA and SVM are standard methods. But we used these methods to developed new classifiers for the truncated polyhedra. Since there is no 'benchmark' for the acceptable misclassification probabilities for this problem, we depended on experts' judgments and found that the Bayes and SVM based classifiers show acceptable misclassification probabilities.

However, LDA did not work well for this problem, though the predicted shapes are in line with other three methods. This misclassification probabilities are much higher than the Bayes and SVM. The exact reasons are not investigated; we just showed, some features are overlapped (Section 6.9.3). In future it may need more investigations and based on that a new classifier might be developed.

Claim 7: The metabolosomes have predicted standard polyhedral shapes.

It is difficult to justify why the metabolosomes must have standard polyhedral shapes. Few research articles (e.g. [18]) claim that the standard polyhedral shapes are formed due to some thermodynamic processes inside multi-component elastic membranes, as like metabolosomes. Also, another similar microcompartment called Carboxysome has standard polyhedral shape; some natural objects like viruses etc. are polyhedral. Hence, this assumption is thought to be valid based on the literatures and from the instances in nature. However, the polyhedral structural distance model resulted in 11 out of 30 metabolosomes are showing an *exact* match to the standard solids. This result, to some extent, validates the assumption.

On the other hand, for the shape classification purpose, we must need a standard reference class (class of solids). It is true that, beyond of these standard solids there are infinitely many convex solids possible, but they are not 'standard', i.e. do not carry geometric properties like a standard solid. This is another reason that we assumed metabolosomes have standard polyhedral shapes. If any metabolosome actually does not have a standard polyhedral shape, these methods in fact find a most probable standard polyhedral shape for it with a strong logical explanation.

7.2.3 Reproducibility

This shape classification task is a workflow consisting of a number of steps. Some of these steps are executed manually as described in corresponding chapters. There are no known automated algorithm driven methods to work satisfactorily for these steps. Naturally there are errors due to subjectivity and hence some reproducibility issues arise.

As we explained in corresponding sections, the tasks like structure drawing, manual segmentation etc. would definitely depend on the subjects, but if the tasks are carefully executed and examined repeatedly, the overall shapes of the objects should not be affected much. However, these tasks can be repeated by different subjects and the outcomes from individuals could be assessed and then combined to eradicate some of the subjective errors.

Some parameters are also determined heuristically (e.g. parameters in the polyhedral structural distance model, smoothing parameters for metabolosome surface etc.), but they are justified in the corresponding sections. These parameters are good for this problem only, and may need to change for another similar problem.

7.3 Future Research

Methodological Improvements

Segmentation

This research starts with 3D image reconstruction and segmentation. The segmentation was carried out manually. Several algorithm-driven, automated segmentation methods exist, but biologists generally prefer manual segmentation for this type of image. Since manual segmentation contains subjective errors, it is worthwhile to develop an automated or semi-automated segmentation algorithms for this type of image. In addition, since 3D smoothing is followed by segmentation, a joint 3D segmentation-smoothing algorithm could be useful. However, this topic is beyond the scope of this thesis.

Features Extraction

The polyhedra profile statistic for metabolosomes requires identification of vertices, faces and edges from the segmented images. These features are identified through visual inspections which may again introduce subjectivity errors. The metabolosomes contain curved facets, segmentation errors and missing regions and some of the vertices, faces and edges are not readily apparent. Hence, the possibility for identification errors is not negligible. An algorithm driven approach to identify these features from the segmented objects could reduce the uncertainty of the predicted shapes. Again, this was not the prime focus of this work.

Retrospective Analysis

The structures of these metabolosomes were studied for the first time using these new methods. Since the objects and the methods both are new, we need some retrospective studies to review the strengths of these methods. For example, these methods can be applied to other similar image data (such as Carboxysomes), whose shapes are already established earlier. These methods however, do show very promising outcomes when applied to simulated data.

Classifiers

The Bayes classifiers and SVM have good misclassification probabilities, while LDA is weak for this problem. Like our modified form of the Bayes Classifiers, the standard LDA may be customized to work better for this problem. Also a logistic discriminant analysis may be developed for this problem. Another common approach is a tree-based model for classification, which might also be worthy to explore in future for this type of data. However, since there is no 'best' classifier for any classification problem, our research in fact opens the opportunities to search for better classifiers for this problem. It may be through developing a better shape descriptor, a new classifier or a strong ensemble method.

Fullerene Models

Recently, three-dimensional structures of many viruses are expressed using fullerene models [207], [135]. However, the topological properties, such as number of edges per vertices and number of edges per face do not support a possible fullerene model for the metabolosomes, since the fullerenes have only pentagonal or hexagonal facets [208], [209]. However, the problem of using fullerene models for structures with missing regions is an entirely different research area and certainly needs attention in the future.

7.4 Future Applications

Incompleteness in shapes may be due to various reasons and the limited angle single axis cryo-tomography is just one of them. Irrespective of the reasons behind incompleteness, if the underlying structures are convex polyhedra, these methods could be appropriate to analyze their shapes. The objects may be other bacterial inclusions or any other incomplete, deformed shaped polyhedral structures.

For instance, computer vision scientists may need to identify objects on the basis of incomplete scene images. The research presented here may be of benefit in solving such problems.

Automation

The complete research, starting from image reconstruction to shape classification, is a time consuming, laborious and diligent process. This is mostly due to the manual execution of some steps. We believe that, the complete workflow could be rolled out to more routine use in future with some automation introduced to the process.

A software pipeline could be thought to develop for this purpose, but a big challenge is to develop algorithm driven ways to execute current manual process. For example, algorithms for segmentation and polyhedral feature extraction from a segmented metabolosome are the initial requirements. Another challenge is to integrate the existing software driven tasks (for example, image reconstruction, etc.) to the new software pipeline. We hope, in future these developments will be accomplished and these methods will be used for many similar other problems.

Applications

Finally, this polyhedral shape prediction approach definitely can be applied to similar shape prediction problems. This approach has two main parts: getting the incomplete structures (data collection) and shape prediction (analysis). We followed a fairly standard data preparation approach (used by other published researches as well). The data collection (information about incomplete structures) approach is manual here, though may be automated as well (e.g. computer vision applications). As discussed before, the shape library is almost complete but may need to use different parameters depending on the problems. The polyhedron profile statistic is a strong one for standard polyhedral shapes, but before using for any other problem, it is recommended to check if the objects of interest are convex polyhedral and the deformation assumptions are appropriate.

We believe that this research will significantly contribute to future developments in statistics, machine learning, structural biology, medicine and other related fields.

Bibliography

1. Alberts, B., et al., *Molecular Biology of the Cell. Chapter 21: Development of Multicellular Organisms*. 4th ed. 2002, New York: Garland Science.
2. Bianconi, E., et al., *An estimation of the number of cells in the human body.* Annals of human biology 2013. **40**(6): p. 463–471.
3. Bolsover, S.R., et al., *Cell Biology: A Short Course*. 3rd ed. 2011: Wiley-Blackwell.
4. Campbell, N.A., L.G. Mitchell, and J.B. Reece, *Biology*. 1999, Menlo Park, CA: Benjamin Cummings Publishing Company, Inc.
5. Jorda, J., et al., *Using comparative genomics to uncover new kinds of protein-based metabolic organelles in bacteria.* Protein Sc., 2013. **22**(2): p.179–195.
6. Kerfeld, C.A., S. Heinhorst, and G.C. Cannon, *Bacterial Microcompartments.* Annual Review of Microbiology, 2010. **64**: p. 391–408.
7. Cheng, S., et al., *Bacterial Microcompartments: their properties and paradoxes.* BioEssays, 2008. **30**(11-12): p. 1084–1095.
8. Bobik, T.A., *Bacterial microcompartments.* Microbe 2007. **2**(1): p. 25–31.
9. Parsons, J.B., et al., *Biochemical and structural insights into bacterial organelle form and biogenesis.* Journal of biological chemistry, 2008. **283,** (21): p. 14366–14375.
10. Iancu, C.V., et al., *The Structure of Isolated Synechococcus Strain WH8102 Carboxysomes as Revealed by Electron Cryotomography.* Journal of molecular biology, 2007. **372**(3): p. 764–773.
11. Tanaka, S., et al., *Atomic-Level Models of the Bacterial Carboxysome Shell.* Science, 2008. **319** (5866): p. 1083–1086.
12. Schmid, M.F., et al., *Structure of Halothiobacillus neapolitanus Carboxysomes by Cryo-electron Tomography.* Journal of molecular biology, 2006. **364**(3): p. 526–535.

13. *Medical Microbiology*. 4th ed, ed. S. Baron. 1996: Univ. of Texas Medical Branch, Galveston.
14. Britannica, E. *virus*. 2015. [Cited on April 05, 2015]; Available from: http://www.britannica.com/EBchecked/topic/630244/virus/32742/Size-and-shape.
15. Pawson, T. and P. Nash, *Assembly of cell regulatory systems through protein interaction domains.* Science, 2003. **300**(5618): p. 445–452.
16. Kuriyan, J., B. Konforti, and D. Wemmer, *The Molecules of Life: Physical and Chemical Principles*. 1st ed. 2012: Garland Science, Taylor & Francis Group.
17. *Single-cell-based models in biology and medicine.* 2007: Basel: Birkhäuser.
18. Graziano, V., R. Sknepnek, and M. Olvera de la Cruz, *Platonic and Archimedean geometries in multicomponent elastic membranes.* Proceedings of the National Academy of Sciences, 2011. **108**(11): p. 4292–4296.
19. Phillips, R., et al., *Physical Biology of the Cell*. 2nd ed. 2012, New York: Garland Science, Taylor and Francis Group.
20. Li, Z. and G.J. Jensen, *Electron cryotomography - a new view into microbial ultrastructure.* Current Opinion in Microbiology, 2009. **12**(3): p. 333–340.
21. Frank, J., *Electron tomography: methods for three-dimensional visualization of structures in the cell*. 2006, New York: Springer.
22. Hoppe, W. and R. Hegerl, *Three-dimensional structure determination by electron microscopy*, in *Computer Processing of Electron Microscope Images*, Ed. P.W. Hawkes, 1980. Springer-Verlag: Heidelberg. p. 127–186.
23. Penczek, P., et al., *Double-tilt electron tomography.* Ultramicroscopy, 1995. **60**(3): p. 393–410.
24. Taylor, K.A., et al., *Three-dimensional reconstruction of rigor insect flight muscle from tilted thin sections.* Nature, 1983. **310**(5975): p. 285–291.
25. McIntosh, R., D. Nicastro, and D. Mastronarde, *New views of cells in 3D: an introduction to electron tomography.* Trends in Cell Biology, 2005. **15**(1): p. 43–51.
26. Frank, J., A. Verschoor, and M. Boublik, *Computer averaging of electron micrographs of 40S ribosomal subunits.* Science, 1981. **214**(4527): p. 1353–1355.

27. Heel, M.v., et al., *Single-particle electron cryo-microscopy: towards atomic resolution.* Quarterly Reviews of Biophysics, 2000. **33**(4): p. 307–369.
28. Crowther, R.A., *Procedures for three-dimensional reconstruction of spherical viruses by Fourier synthesis from electron micrographs.* Philosophical Transactions of the Royal Society of London, 1971. B, Biological Sciences, **261**(837): p. 221–230.
29. Baker, T.S. and R.H. Cheng, *A model-based approach for determining orientations of biological macromolecules imaged by cryoelectron microscopy.* Journal of Structural Biology, 1996. **116**(1): p. 120–130.
30. Goncharov, A.B. and M.S. Gelfand, *Determination of mutual orientation of identical particles from their projections by the moments method.* Ultramicroscopy 1988. **25**(4): p. 317–327.
31. Cheng, H., et al., *Functional implications of quasi-equivalence in a T = 3 icosahedral animal virus established by cryo-electron microscopy and X-ray crystallography.* Structure 1994. **2**(4): p. 271–282.
32. Goodall, C., *Procrustes methods in the statistical analysis of shape.* Journal of the Royal Statistical Society, 1991. Series B(Methodological): p. 285–339.
33. Crowther, R.A., D.J. DeRosier, and A. Klug, *The Reconstruction of a Three-Dimensional Structure from Projections and its Application to Electron Microscopy.* Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, 1970. **317**(1530): p. 319–340.
34. Bartesaghi, A. and S. Subramaniam, *Membrane protein structure determination using cryo-electron tomography and 3D image averaging.* Current Opinion in Structural Biology 2009. **19**(4): p. 402–407.
35. Bartesaghi, A., et al., *Classification and 3D averaging with missing wedge correction in biological electron tomography.* Journal of Structural Biology, 2008. **162**(3): p. 436–450.
36. Ruprecht, J. and J. Nield, *Determining the structure of biological macromolecules by transmission electron microscopy, single particle analysis and 3D reconstruction.* Progress in Biophysics and Molecular Biology, 2001. **75**(3): p. 121–164.
37. Walz, J., et al., *Electron tomography of single ice-embedded macromolecules: three-dimensional alignment and classification.* Journal of Structural Biology, 1997. **120**(3): p. 387–395.

38. Penczek, P., M. Radermacher, and J. Frank, *Three-dimensional reconstruction of single particles embedded in ice.* Ultramicroscopy, 1992. **40**(1): p. 33–53.

39. Penczek, P., R.A. Grassucci, and J. Frank, *The ribosome at improved resolution: new techniques for merging and orientation refinement in 3D cryo-electron microscopy of biological particles.* Ultramicroscopy, 1994. **53**(3): p. 251–270.

40. Frank, J., *Three-dimensional electron microscopy of macromolecular assemblies*. 1996: Academic Press.

41. Frangakis, A. and R. Hegerl, *Noise reduction in electron tomographic reconstructions using nonlinear anisotropic diffusion.* Journal of Structural Biology, 2001. **135**(3): p. 239–250.

42. Asai, K., et al., *Clustering and averaging of images in single-particle analysis.* Genome Informatics Series, 2000: p. 151–160.

43. Frank, J., et al., *SPIDER and WEB: processing and visualization of images in 3D electron microscopy and related fields.* Journal of Structural Biology, 1996. **116**(1): p. 190–199.

44. Liang, M., et al., *Bacterial propanediol utilisation microcompartments are non-uniform polyhedra resembling Johnson solids.* Submitted to Journal of Molecular Biology, April 2014.

45. Dana, J.D., G.J. Brush, and E.S. Dana, *The System of Mineralogy*. Vol. I : Elements, Sulfides, Sulfosalts, Oxides 1944, New York: Wiley.

46. Cromwell, P.R., *Polyhedra*. 1997, Cambridge: Cambridge University Press.

47. Coxeter, H.S.M., *Regular Polytopes*. 1973, New York: Dover Publications Inc.

48. Bookstein, F.L., *Landmark methods for forms without landmarks: Morphometrics of group differences in outline shape.* Medical Image Analysis, 1997. **1**(3): p. 225–243.

49. Cootes, T.F., et al., *Training Models of Shape from Sets of Examples.* Proceedings of British Machine Vision Conference, Springer, London, 1992: p. 9–18.

50. Brechbühler, C., G. Gerig, and O. Kübler, *Parameterization of Closed Surfaces for 3-D Shape Description.* CVGIP: Image Understanding, 1995. **61**(2): p. 154–170.

51. Fritsch, D.S., et al., *The Multiscale Medial Axis and Its Applications in Image Registration.* Patter Recognition Letters, 1994. **15**(5): p. 445–452.

52. Golland, P., W.E.L. Grimson, and R. Kikinis, *Statistical Shape Analysis Using Fixed Topology Skeletons: Corpus Callosum Study*. Information Processing in Medical Imaging. Vol. 1613. 1999, Berlin Heidelberg: Springer 382–387.

53. Zhang, D. and G. Lu, *Review of shape representation and description techniques.* Pattern recognition, 2004. **37**(1): p. 01–19.

54. Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2009, New York: Springer.

55. Falk, G., *Interpretation of Imperfect Line Data as a Three-Dimensional Scene.* Artificial Intelligence, 1972. **3**: p. 101–144.

56. Murty, M.N. and V.S. Devi, *Pattern Recognition - An Algorithmic Approach*. 2011, New york: Springer.

57. Fisher, R.A., *The Use of Multiple Measurements in Taxonomic Problems.* Annals of Eugenics 1936. **7**(2): p. 179–188.

58. Cortes, C. and V.N. Vapnik, *Support-Vector Networks.* Machine Learning, 1995. **20**(3): p. 273–297.

59. Bozzola, J.J. and L.D. Russell, *Electron Microscopy: Principles and Techniques for Biologists*. 1999: Jones & Bartlett Learning.

60. Goodhew, P.J., J. Humphreys, and R. Beanland, *Electron Microscopy and Analysis*. 3rd ed. 2006: Taylor and Francis.

61. Erni, R., et al., *Atomic-Resolution Imaging with a Sub-50-pm Electron Probe.* Physical review letters, 2009. **102**(9:096101).

62. Watt, I.M., *The Principles and Practice of Electron Microscopy*. 2nd ed. 1997: Cambridge University Press.

63. Hayat, M.A., *Principles and Techniques of Electron Microscopy: Biological Applications*. 4th ed. 2000: Cambridge University Press.

64. Doryen, B., et al., *The structure of the poliovirus 135S cell entry intermediate at 10-angstrom resolution reveals the location of an externalized polypeptide that binds to membranes.* Journal of virology, 2005. **79**(12): p. 7745–7755.

65. Williams, D.B. and C.B. Carter, *Transmission Electron Microscopy - A Textbook for Materials Science*. 2nd ed. 2009, New York: Springer.

66. Graef, M. D., *Introduction to Conventional Transmission Electron Microscopy*. Cambridge Solid State Science Series. 2003: Cambridge University Press.

67. Reimer, L. and H. Kohl, *Transmission Electron Microscopy: Physics of Image Formation*. 5th ed. Springer Series in Optical Sciences. 2008, New York: Springer.

68. Adrian, M., et al., *Cryo-electron microscopy of viruses.* Nature 1984. **308**(5954): p. 32 – 36.

69. Tocheva, E.I., Z. Li, and G.J. Jensen, *Electron Cryotomography.* Cold Spring Harbor Perspectives in Biology, 2010. **2**(6:a003442).

70. Jensen, E. *Example: Transmission Electron Microscope System*. 2012
[Cited 2014 April 10]; Available from:
http://www.texample.net/tikz/examples/transmission-electron-microscope/.

71. Jensen, G.J. and A. Briegel, *How electron cryotomography is opening a new window into prokaryotic ultrastructure.* Current Opinion in Microbiology, 2007. **17**(2): p. 260–267.

72. Cavalier, A., D. Spehner, and B.M. Humbel, *Handbook of Cryo-Preparation Methods for Electron Microscopy*. Methods in Visualization. 2009: Taylor & Francis Group.

73. Zanetti, G., et al., *Cryo-electron tomographic structure of an immunodeficiency virus envelope complex in situ.* PLoS Pathog, 2006. **2**(8).

74. Kürner, J., A.S. Frangakis, and W. Baumeister, *Cryo-electron tomography reveals the cytoskeletal structure of Spiroplasma melliferum.* Science 2005. **307**(5708): p. 436–438.

75. Akey, C.W., *Interactions and structure of the nuclear pore complex revealed by cryo-electron microscopy.* The Journal of Cell Biology, 1989. **109**(3): p. 955–970.

76. Shaoxia, C., et al., *Location of a folding protein and shape changes in GroEL–GroES complexes imaged by cryo-electron microscopy.* Nature, 1994. **371**: p. 261 – 264

77. Cierniak, R., *X-Ray Computed Tomography in Biomedical Engineering*. 2011, New York: Springer.

78. Hoppe, W., *Drei-dimensional abbildende Elektronenmikroskope.* Z. Naturforsch, 1972 **27**(a): p. 919–929.

79. Hoppe, W., et al., *Three-dimensional reconstruction of individual negatively stained fatty-acid synthetase molecules from tilt series in the electron microscope.* Hoppe Seylers Z Physiol Chem, 1974. **355**: p. 1483–1487.

80. Brown, L.G., *A survey of image registration techniques.* ACM Computing Surveys, 1992. **24**(4): p. 325–376.

81. Zitova, B. and J. Flusser, *Image registration methods: a survey.* Image and Vision Computing, 2003. **21**(11): p. 977–1000.

82. Maintz, J.B. and V.M. A., *A survey of medical image registration.* Medical Image Analysis, 1998. **2**(1): p. 01–36.

83. Radon, J., *On the determination of functions from their integrals along certain manifolds [in German].* Math Phys Klass, 1917. **69**: p. 262–277.

84. Kak, A.C. and M. Slaney, *Principles of Computerized Tomographic Imaging*. Classics in Applied Mathematics. 2001: Society for Industrial and Applied Mathematics.

85. Rockmore, A.J. and A. Macovski, *A maximum likelihood approach to emission image reconstruction from projections.* IEEE Transactions on Nuclear Science, 1976. **23**(4): p. 1428–1432.

86. Kremer, J.R., D.N. Mastronarde, and J.R. McIntosh, *Computer visualization of three-dimensional image data using IMOD.* Journal of Structural Biology, 1996. **116**(1): p. 71–76.

87. Pawel, P., et al., *Double-tilt electron tomography.* Ultramicroscopy, 1995. **60**(3): p. 393–410.

88. Lanzavecchia, S., et al., *Conical tomography of freeze-fracture replicas: a method for the study of integral membrane proteins inserted in phospholipid bilayers.* Journal of Structural Biology, 2005. **1**(87–98).

89. Radermacher, M. and W. Hoppe, *Properties of three-dimensionally reconstructed objects from projections by conical tilting compared to single axis tilting.* Proc. 7th Eur. Congr. Electron Microsc., 1980. **1**: p. 132–133.

90. *Electron Tomography: Three-dimensional Imaging with the Transmission Electron Microscope*. Language of Science. 1992: Plenum Press. 399.

91. Jensen, G.J. *The Jensen Laboratory*. 2014 [Cited 2014 January]; Available from: http://www.jensenlab.caltech.edu/.

92. Pettersen, E.F., et al., *UCSF Chimera - A Visualization System for Exploratory Research and Analysis.* Journal of Computational Chemistry, 2004. **25**(13): p. 1605–1612.

93. Amira. *Amira | FEI Visualization Sciences Group*. 2014
[Cited 2014 January]; Available from: www.vsg3d.com/amira/overview.

94. Mathworks. *MATLAB - The Language of Technical Computing - MathWorks*. [Cited 2014 January]; 2012b:
Available from: www.mathworks.com/products/matlab/.

95. Zhang, H., J.E. Fritts, and S.A. Goldman, *Image segmentation evaluation: A survey of unsupervised methods.* Computer Vision and Image Understanding, 2008. **110**(2): p. 260–280.

96. Pruggnaller, S., M. Mayr, and A.S. Frangakis, *A visualization and segmentation toolbox for electron microscopy.* Journal of Structural Biology, 2008. **164**(1): p. 161–165.

97. Simonoff, J.S., *Smoothing Methods in Statistics*. 2nd ed. Springer Series in Statistics. 1998, New York: Springer.

98. Härdle, W., *Smoothing Techniques: With Implementation in S*. 1991 Edition. Springer Series in Statistics. 1991: Springer.

99. Montgomery, D.C., C.L. Jennings, and M. Kulahci, *Introduction to Time Series Analysis and Forecasting*. 1st ed. 2008: John Wiley & Sons.

100. Kepler, J., *Mysterium Cosmographicum*. 1596: Tübingen.

101. Plato, *Timaeus and Critias*. 1965: Penguin Books.

102. Alexandrov, A.D., *Convex Polyhedra*. 2005 ed. Springer Monographs in Mathematics. 2005: Springer.

103. *Shaping Space: Exploring Polyhedra in Nature, Art, and the Geometrical Imagination*. 2013 ed, ed. M. Senechal. Springer.

104. Ghyka, M., *The geometry of art and life*. 1977, New York: Dover Publications Inc.

105. Wu, L., et al., *The advantages of the pentameral symmetry of the starfish.* arXiv, 2012 (preprint arXiv: 1202.2219).

106. Nishiyama, Y., *Five Petals: The Mysterious Number "5" Hidden in Nature.* Int. J. Pure Appl. Math, 2012. **78(3)**(349362).

107. Lightman, A., *The Accidental Universe: The World You Thought You Knew*. 2014: Vintage.

108. *Frequently Asked Questions about Crystals for Students*. 1998 [Cited 2015 April 06, 2015]; Available from: http://dwb4.unl.edu/Chem/CHEM869V/CHEM869VLinks/laue.chem.ncsu.edu/student_faq_xtal.html.

109. Maeder, R. E., *Uniform Polyhedra.* The Mathematica Journal, 1993. **3**(4): p. 48–57.

110. Marzal, J., H. Xie, and C.C. Fung, *Vertex Configurations and Their Relationship on Orthogonal Pseudo-Polyhedra.* World Academy of Science, Engineering and Technology, 2011. **77**: p. 1–8.

111. Ziegler, G.M., *Lectures on Polytopes*. Graduate Texts in Mathematics. 1995, New York: Springer-Verlag

112. Burt, J.L., et al., *Beyond Archimedean solids: Star polyhedral gold nanocrystals.* Journal of Crystal Growth, 2005. **285**(4): p. 681–691.

113. O'Rourke, J. and C. Schevon, *On the development of closed convex curves on 3-polytopes.* Journal of Geometry, 1989. **35**(1): p. 152–157.

114. O'Rourke, J., *On the development of the intersection of a plane with a polytope.* Computational Geometry, 2003. **24**(1): p. 03–10.

115. V. A. Blatov, M.O.K., and D. M. Proserpio, *Vertex-, face-, point-, Schläfli-, and Delaney-symbols in nets, polyhedra and tilings: recommended terminology.* Cryst. Eng. Comm., 2010. **12**(1): p. 44–48.

116. Godsil, C. and G. Royle, *Algebraic Graph Theory*. 2001 ed. Graduate Texts in Mathematics. Vol. 207. New York: Springer.

117. Rouvray, D.H., *The topological matrix in quantum chemistry.* Chemical Applications of Graph Theory, 1976: p. 175–222.

118. Trinajstic, N., *Chemical Graph Theory*. 2nd ed. 1992: CRC Press.

119. Estrada, E., *Spectral Moments of the Edge Adjacency Matrix in Molecular Graphs - Definition and Applications to the Prediction of Physical Properties of Alkanes.* Journal of Chemical Information and Computer Sciences, 1996. **36**(4): p. 844–849.

120. Cundy, H.M. and A.P. Rollet, *Mathematical Models*. 2nd ed. 1961, Oxford: Oxford University Press.

121. Prabhakar, S. and M.R. Henderson, *Automatic form-feature recognition using neural-network based techniques on boundary representations of solid models.* Computer-Aided Design, 1992. **24**(7): p. 381–393.

122. Burns, G. and A.M. Glazer, *Space Groups for Scientists and Engineers*. 2nd ed. 1990, Boston: Academic Press, Inc.

123. Atiyah, M. and P. Sutcliffe, *Polyhedra in Physics, Chemistry and Geometry*. Milan Journal of Mathematics, 2003. **71**(1): p. 33–58.

124. *Wikipedia*. [Cited 2014 January]; Available from: www.wikipedia.org.

125. Johnson, N.W., *Convex polyhedra with regular faces*. Canadian Journal of Mathematics, 1966. **18**(1): p. 169–200.

126. Zalgaller, V.A., *Convex Polyhedra with Regular Faces*. 3rd ed. Vol. 2. 1967, Steklov Math Institute, Leningrad Sem. in Math. (in Russian)

127. Catalan, E., *Mémoire sur la Théorie des Polyèdres*. J. l'École Polytechnique (Paris) 1865. **41**: p. 1–71.

128. Wenninger, M.J., *Polyhedron models*. 1974: Cambridge University Press.

129. Richeson, D.S., *Euler's Gem: The Polyhedron Formula and the Birth of Topology*. 2008: Princeton University Press.

130. Berger, M., *Geometry I*. 1987, Berlin: Springer.

131. *Mathematica*. [Cited 2014 January];
Available from: www.wolfram.com/mathematica.

132. Weisstein, E.W. *Mathematica*. 2003 [Cited 2014 January]; Available from: http://mathworld.wolfram.com/packages/JohnsonSolids.m.

133. Heymann, J.B., et al., *Irregular and Semi-Regular Polyhedral Models for Rous Sarcoma Virus Cores*. Computational and Mathematical Methods in Medicine, 2008. **9**(3-4): p. 197–210.

134. Rupprecht, C.E., *Chapter 61. Rhabdoviruses: Rabies Virus*, in *Medical Microbiology*, S. Baron, Editor. 1996, University of Texas Medical Branch at Galveston: Texas.

135. Butan, C., et al., *RSV capsid polymorphism correlates with polymerization efficiency and envelope glycoprotein content: implications that nucleation controls morphogenesis*. Journal of Molecular Biology, 2008. **376**(4): p. 1168–1181.

136. *Python: Python Software Foundation*. 1991. [Cited 2014 January]; Available from: https://www.python.org/.

137. Sugihara, K., *Machine Interpretation of Line Drawings*. Vol. 1. 1986, Cambridge: MIT Press.

138. Chung, R. and K. Leung, *3-D Interpretation of Imperfect Line Drawings*, in *British Machine Vision Conference*. 1995.

139. Fink, A.M. and M. Jodeit Jr, *On Chebyshev's other inequality*. Lecture Notes-Monograph Series, 1984: p. 115–120.

140. Pearson, K., *On Lines and Planes of Closest Fit to Systems of Points in Space*. Philosophical Magazine, 1901. **2**(11): p. 559–572.

141. Hotelling, H., *Analysis of a complex of statistical variables into principal components*. Journal of Educational Psychology, 1933. **24**(6): p. 417–441, 498-520.

142. Everitt, B.S., S. Landau, and M. Leese, *Cluster Analysis*. 4th ed. 2009, New York: Wiley.

143. Ward, J.H., Jr., *Hierarchical Grouping to Optimize an Objective Function*. Journal of the American Statistical Association, 1998. **58**(301): p. 236–244.

144. R Core Team., *R: A language and environment for statistical computing* R Foundation for Statistical Computing, Vienna, Austria, 2013.
[Cited 2014 January] (www.R-project.org).

145. Sknepnek, R., V. Graziano, and M.O.d.l. Cruz, *Buckling of multicomponent elastic shells with line tension*. Soft Matter, 2012. **8**(3): p. 636–644.

146. Smith, W.D. and N.C. Wormald, *Geometric separator theorems and applications*. Proceedings 39th Annual Symposium on Foundations of Computer Science, IEEE, 1998: p. 232–243.

147. Dryden, I. L. and K. V. Mardia, *Statistical Shape Analysis*. 1998, New York: Wiley.

148. Wilkinson, L. and M. Friendly, *The history of the cluster heat map*. The American Statistician, 2009. **63**(2).

149. Furrer, R., D. Nychka, and S. Sain, *fields: Tools for spatial data. R package version 6.8*. 2013. (http://CRAN.R-project.org/package=fields).

150. Warnes, G.R., et al., *gplots: Various R programming tools for plotting data. R package version 2.11.3*. 2013. [Cited 2014 January]
(http://CRAN.R-project.org/package=gplots).

151. Havemann, G.D. and T.A. Bobik, *Protein content of polyhedral organelles involved in coenzyme B12-dependent degradation of 1, 2-propanediol in Salmonella enterica serovar Typhimurium LT2*. Journal of Bacteriology, 2003. **185**(17): p. 5086–5095.

152. Mohri, M., A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. 2012: MIT Press.

153. Goldstein, H., *Classical Mechanics*. 1980, MA Addison-Wesley.

154. Hawkins, D.M., *The problem of overfitting.* Journal of Chemical Information and Computer Sciences, 2004. **44**(1): p. 1–12.

155. Fehr, J., K. Z. Arreola, and H. Burkhardt, *Fast support vector machine classification of very large datasets*, In *Data Analysis, Machine Learning and Applications*. C. Preisach, et al., Ed. 2008, Springer Berlin, Heidelberg. p.11–18.

156. Pormann, J. *DSCR - Scalable Computing Support Center - DukeWiki*. 2015 [Cited 2015 April 10, 2015];
Available from: https://wiki.duke.edu/display/SCSC/DSCR.

157. Heckman, J.J., *Sample selection bias as a specification error.* Econometrica: Journal of the Econometric Society, 1979: p. 153–161.

158. Duda, R.O. and P.E. Hart, *Pattern classification and scene analysis*. Vol. 3. 1973, New York: John Wiley & Sons.

159. Bolstad, W.M., *Introduction to Bayesian statistics*. 2nd ed. 2007. New York: John Wiley & Sons.

160. Gyorfi, L. D. L., G. Lugosi, and L. Devroye, *A Probabilistic Theory of Pattern Recognition*. Stochastic Modelling and Applied Probability. Vol. 31. 1996: Springer.

161. Coultrip, R.L. and R.H. Granger, *Sparse random networks with LTP learning rules approximate Bayes classifiers via Parzen's method.* Neural Networks, 1994. **7**(3): p. 463–476.

162. Bellman, R.E., *Dynamic Programming*. 1st ed.: Courier Dover Publications.

163. Zhang, H., *The optimality of naive Bayes.* American Association for Artificial Intelligence, 2004. **1**(2): p. 3.

164. Rish, I., *An empirical study of the naive Bayes classifier.* IJCAI 2001 workshop on empirical methods in artificial intelligence,2001.**3**(22): p.41–46.

165. Domingos, P. and M. Pazzani, *On the optimality of the simple Bayesian classifier under zero-one loss.* Machine Learning, 1997. **29**(2-3): p. 103–130.

166. Haeb-Umbach, R. and H. Ney. *Linear discriminant analysis for improved large vocabulary continuous speech recognition.* In *ICASSP-92 (IEEE*

International Conference on Acoustics, Speech, and Signal Processing). 1992. San Francisco, CA: IEEE.

167. Zhao, W., R. Chellappa, and P.J. Phillips, *Subspace linear discriminant analysis for face recognition.* 1999, Computer Vision Laboratory, Center for Automation Research, University of Maryland, Maryland.

168. Abate, A.F., et al., *2D and 3D face recognition: A survey.* Pattern Recognition Letters, 2007. **28**(14): p. 1885–1906.

169. Chan, H.P., et al., *Computer-aided classification of mammographic masses and normal tissue: linear discriminant analysis in texture feature space.* Physics in Medicine and Biology, 1995. **40**(5).

170. Guo, Y., T. Hastie, and R. Tibshirani, *Regularized linear discriminant analysis and its application in microarrays.* Biostatistics, 2007. **8**(1): p.86–100.

171. Clarke, B.S., E. Fokoué, and H.H. Zhang, *Principles and theory for data mining and machine learning,* 2009. New York: Springer.

172. Mardia, K.V., J.T. Kent, and J.M. Bibby, *Multivariate analysis*. Probability and Mathematical Statistics, 1979. London: Academic Press.

173. Izenman, A.J., *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*. 1st ed. Springer Texts in Statistics. 2008, New York: Springer.

174. Petridis, S. and S.J. Perantonis, *On the relation between discriminant analysis and mutual information for supervised linear feature extraction.* Pattern Recognition, 2004. **37**(5): p. 857–874.

175. Rao, C.R., *The utilization of multiple measurements in problems of biological classification.* Journal of the Royal Statistical Society. Series B (Statistical Methodology), 1948: p. 159–203.

176. Krzanowski, W.J., et al., *Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data.* Applied Statistics, 1995. **44**: p. 101–115.

177. Venables, W.N. and B.D. Ripley, *Modern Applied Statistics with S*. 4th ed. 2007, New York: Springer.

178. Vapnik, V.N., *The nature of statistical learning theory*. 2nd ed. Information Science and Statistics, 2000, New York: Springer - Verlag.

179. Cristianini, N. and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*, Cambridge University Press, 2000.

180. Byun, H. and S.W. Lee, *A survey on pattern recognition applications of support vector machines.* International Journal of Pattern Recognition and Artificial Intelligence, 2003. **17**(3): p. 459–486.

181. Guodong, G., S. Li, and C. Kapluk, *Face recognition by support vector machines*, in *Proceedings of IEEE international conference on automatic face and gesture recognition*, 2000, IEEE. p. 196–201.

182. Ai, H., L. Liang, and G. Xu, *Face detection based on template matching and support vector machines.* Proceedings of International Conference on Image Processing, 2001: p. 1006–1009.

183. Dhasal, P., et al., *An optimized feature selection for image classification based on SVM-ACO.* International Journal of Advanced Computer Research, 2012. **2**(5): p. 123 – 128.

184. Zhang, J., et al., *Local features and kernels for classification of texture and object categories: A comprehensive study.* International Journal of Computer Vision, 2007. **73**(2): p. 213–238.

185. Pontil, M. and A. Verri, *Properties of support vector machines.* Neural Computation, 1998. **10**(4): p. 955–974.

186. Kumar, P., N. Sharma, and A. Rana, *Hand written character recognition using different kernel based SVM classifier and MLP neural network - a comparison.* Int. Journal of Computer Application, 2012. **53**(11): p. 25–31.

187. Oliveira, L.S. and R. Sabourin, *Support vector machines for handwritten numerical string recognition.* International Workshop on Frontiers in Handwriting Recognition, 2004. IWFHR-9, 2004. IEEE: p. 39–44.

188. Gokcen, I. and J. Peng. *Comparing linear discriminant analysis and support vector machines*. In *Advance in Information Systems*. 2002. Turkey: Springer.

189. Muller, K.R. and S. Mika, *An introduction to kernel-based learning algorithms.* IEEE Transactions on Neural Networks, 2001. **12**(2): p.181– 201.

190. Weston, J. and C. Watkins, *Multi-class support vector machines.* Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London, 1998.

191. Franc, V. and V. Hlavác. *Multi-class support vector machine*. in *16th IEEE International Conference on Pattern Recognition*. 2002. IEEE.

192. Crammer, K. and Y. Singer, *On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines.* Journal of Machine Learning Research, 2001. **2**: p. 265–292.

193. Hsu, C. W. and C. J. Lin, *A comparison of methods for multiclass support vector machines.* IEEE Transactions on Neural Networks, 2002. **13**(2): p. 415–425.

194. Fürnkranz, J., *Round robin classification.* The Journal of Machine Learning Research, 2002. **2**: p. 721–747.

195. Wu, T.F., C.J. Lin, and R.C. Weng, *Probability estimates for multi-class classification by pairwise coupling.* Journal of Machine Learning Research, 2004. **5**: p. 975–1005.

196. Platt, J., *Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods.* Advances in Large Margin Classifiers, 1999. **10**(3): p. 61–74.

197. Hastie, T. and R. Tibshirani, *Classification by pairwise coupling.* The Annals of Statistics, 1998. **26**(1): p. 451–471.

198. Milgram, J., M. Cheriet, and R. Sabourin. *Estimating accurate multi-class probabilities with support vector machines*. in *IEEE International Joint Conference on Neural Networks (IJCNN'05)*, 2005. IEEE.

199. Rifkin, R. and A. Klautau, *In defense of one-vs-all classification.* The Journal of Machine Learning Research, 2004. **5**: p. 101–141.

200. Lee, S. and K. Daniels. *Gaussian kernel width generator for support vector clustering. International Conference on Bioinformatics and its Applications*, 2004.

201. Karatzoglou, A., et al., *kernlab - An S4 Package for Kernel Methods in R.* Journal of Statistical Software, 2004. **11**(9): p. 1–20.

202. Caputo, B., et al. *Appearance-based object recognition using SVMs: which kernel should I use?* In *Proc. of NIPS workshop on Statitsical methods for computational experiments in visual processing and computer vision*, 2002.

203. Dietterich, T. G., *Approximate statistical tests for comparing supervised classification learning algorithms.* Neural Computation, 1998. **10**(7): p. 1895–1923.

204. Demšar, J., *Statistical comparisons of classifiers over multiple data sets.* The Journal of Machine Learning Research, 2006. **1**: p. 1–30.

205. Salzberg,S.L., *On comparing classifiers:Pitfalls to avoid and a recommended approach.* Data Mining and Knowledge Discovery, 1997. **1**(3): p. 317–328.

206. Gibbons, J.D. and S. Chakraborti, *Nonparametric Statistical Inference*. 5th ed. Statistics: Textbooks & Monographs. 2010, Boca Raton, FL: Chapman & Hall/CRC Press, Taylor & Francis Group.

207. Dresselhaus, M.S., G. Dresselhaus, and P.C. Eklund, *Science of fullerenes and carbon nanotubes: their properties and applications*. Academic Press, 1996.

208. Roger, T. and D.R. Walton, *The chemistry of fullerenes.* Nature, 1993. **363**: p. 685–693.

209. *Fullerenes and Related Structures*. Topics in Current Chemistry, Ed. A. Hirsch. 1999, Springer-Verlag, Berlin - Heidelberg.

Appendix

Appendix for Chapter 3

Appendix 3.1

As described in Chapter 3, the polyhedron profile statistic for standard polyhedra contains 15 features. The following table gives the polyhedron profile statistic for the 5 Platonic solids. However, this statistic was extended later to contain more features as described in Chapter 6, Part I. Throughout the Appendix, the following symbols are used:

N_V = Number of vertices, N_E = Number of edges, N_F = Number of faces,
N_F^i = Number of faces with *i* edges, *i* = 3, 4, 5, 6, 8, 10 and
N_V^i = Number of vertices with *i* edges, *i* = 3, 4, 5, 6, 8, 10.

Feature Name	**Tetrahedron**	**Cube**	**Octahedron**	**Dodecahedron**	**Icosahedron**
N_V	4	8	6	20	12
N_F	4	6	8	12	20
N_E	6	12	12	30	30
N_F^3	4	0	8	0	20
N_F^4	0	6	0	0	0
N_F^5	0	0	0	12	0
N_F^6	0	0	0	0	0
N_F^8	0	0	0	0	0
N_F^{10}	0	0	0	0	0
N_V^3	4	8	0	20	0
N_V^4	0	0	6	0	0
N_V^5	0	0	0	0	12
N_V^6	0	0	0	0	0
N_V^8	0	0	0	0	0
N_V^{10}	0	0	0	0	0

Table A3.1: The polyhedron profile statistic (with 15 features) for the 5 Platonic solids.

Appendix 3.2

Similar to Appendix 3.1, the following table gives the polyhedron profile statistic (with 15 features) for the 13 Archimedean solids.

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Truncated tetrahedron	12	8	18	4	0	0	4	0	0	12	0	0	0	0	0
Cuboctahedron	12	14	24	8	6	0	0	0	0	0	12	0	0	0	0
Truncated cube	24	14	36	8	0	0	0	6	0	24	0	0	0	0	0
Truncated octahedron	24	14	36	0	6	0	8	0	0	24	0	0	0	0	0
Rhombicuboctahedron	24	26	48	8	18	0	0	0	0	0	24	0	0	0	0
Truncated cuboctahedron	48	26	72	0	12	0	8	6	0	48	0	0	0	0	0
Snub cube	24	38	60	32	6	0	0	0	0	0	0	24	0	0	0
Icosidodecahedron	30	32	60	20	0	12	0	0	0	0	30	0	0	0	0
Truncated dodecahedron	60	32	90	20	0	0	0	0	12	60	0	0	0	0	0
Truncated icosahedron	60	32	90	0	0	12	20	0	0	60	0	0	0	0	0
Rhombicosidodecahedron	60	62	120	20	30	12	0	0	0	0	60	0	0	0	0
Truncated icosidodecahedron	120	62	180	0	30	0	20	0	12	120	0	0	0	0	0
Snub dodecahedron	60	92	150	80	0	12	0	0	0	0	0	60	0	0	0

Table A3.2: The polyhedron profile statistic (with 15 features) for the 13 Archimedean solids.

Appendix 3.3

Similar to Appendices 3.1 and 3.2, the following table gives the polyhedron profile statistic (with 15 features) for the 92 Johnson solids.

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Square pyramid	5	5	8	4	1	0	0	0	0	4	1	0	0	0	0
Pentagonal pyramid	6	6	10	5	0	1	0	0	0	5	0	1	0	0	0
Triangular cupola	9	8	15	4	3	0	1	0	0	6	3	0	0	0	0
Square cupola	12	10	20	4	5	0	0	1	0	8	4	0	0	0	0
Pentagonal cupola	15	12	25	5	5	1	0	0	1	10	5	0	0	0	0
Pentagonal rotunda	20	17	35	10	0	6	0	0	1	10	10	0	0	0	0
Elongated triangular pyramid	7	7	12	4	3	0	0	0	0	4	3	0	0	0	0
Elongated square pyramid	9	9	16	4	5	0	0	0	0	4	5	0	0	0	0
Elongated pentagonal pyramid	11	11	20	5	5	1	0	0	0	5	5	1	0	0	0
Gyroelongated square pyramid	9	13	20	12	1	0	0	0	0	0	5	4	0	0	0
Gyroelongated pentagonal pyramid	11	16	25	15	0	1	0	0	0	0	5	6	0	0	0
Triangular bipyramid	5	6	9	6	0	0	0	0	0	2	3	0	0	0	0
Pentagonal bipyramid	7	10	15	10	0	0	0	0	0	0	5	2	0	0	0
Elongated triangular bipyramid	8	9	15	6	3	0	0	0	0	2	6	0	0	0	0
Elongated square bipyramid	10	12	20	8	4	0	0	0	0	0	10	0	0	0	0
Elongated pentagonal bipyramid	12	15	25	10	5	0	0	0	0	0	10	2	0	0	0
Gyroelongated square bipyramid	10	16	24	16	0	0	0	0	0	0	2	8	0	0	0
Elongated triangular cupola	15	14	27	4	9	0	1	0	0	6	9	0	0	0	0
Elongated square cupola	20	18	36	4	13	0	0	1	0	8	12	0	0	0	0
Elongated pentagonal cupola	25	22	45	5	15	1	0	0	1	10	15	0	0	0	0
Elongated pentagonal rotunda	30	27	55	10	10	6	0	0	1	10	20	0	0	0	0
Gyroelongated triangular cupola	15	20	33	16	3	0	1	0	0	0	9	6	0	0	0
Gyroelongated square cupola	20	26	44	20	5	0	0	1	0	0	12	8	0	0	0
Gyroelongated pentagonal cupola	25	32	55	25	5	1	0	0	1	0	15	10	0	0	0

Continued: The polyhedron profile statistic (with 15 features) for the 92 Johnson solids

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Gyroelongated pentagonal rotunda	30	37	65	30	0	6	0	0	1	0	20	10	0	0	0
Gyrobifastigium	8	8	14	4	4	0	0	0	0	4	4	0	0	0	0
Triangular orthobicupola	12	14	24	8	6	0	0	0	0	0	12	0	0	0	0
Square orthobicupola	16	18	32	8	10	0	0	0	0	0	16	0	0	0	0
Square gyrobicupola	16	18	32	8	10	0	0	0	0	0	16	0	0	0	0
Pentagonal orthobicupola	20	22	40	10	10	2	0	0	0	0	20	0	0	0	0
Pentagonal gyrobicupola	20	22	40	10	10	2	0	0	0	0	20	0	0	0	0
Pentagonal orthocupolarotunda	25	27	50	15	5	7	0	0	0	0	25	0	0	0	0
Pentagonal gyrocupolarotunda	25	27	50	15	5	7	0	0	0	0	25	0	0	0	0
Pentagonal orthobirotunda	30	32	60	20	0	12	0	0	0	0	30	0	0	0	0
Elongated triangular orthobicupola	18	20	36	8	12	0	0	0	0	0	18	0	0	0	0
Elongated triangular gyrobicupola	18	20	36	8	12	0	0	0	0	0	18	0	0	0	0
Elongated square gyrobicupola	24	26	48	8	18	0	0	0	0	0	24	0	0	0	0
Elongated pentagonal orthobicupola	30	32	60	10	20	2	0	0	0	0	30	0	0	0	0
Elongated pentagonal gyrobicupola	30	32	60	10	20	2	0	0	0	0	30	0	0	0	0
Elongated pentagonal orthocupolarotunda	35	37	70	15	15	7	0	0	0	0	35	0	0	0	0
Elongated pentagonal gyrocupolarotunda	35	37	70	15	15	7	0	0	0	0	35	0	0	0	0
Elongated pentagonal orthobirotunda	40	42	80	20	10	12	0	0	0	0	40	0	0	0	0
Elongated pentagonal gyrobirotunda	40	42	80	20	10	12	0	0	0	0	40	0	0	0	0
Gyroelongated triangular bicupola	18	26	42	20	6	0	0	0	0	0	6	12	0	0	0
Gyroelongated square bicupola	24	34	56	24	10	0	0	0	0	0	8	16	0	0	0
Gyroelongated pentagonal bicupola	30	42	70	30	10	2	0	0	0	0	10	20	0	0	0
Gyroelongated pentagonal cupolarotunda	35	47	80	35	5	7	0	0	0	0	15	20	0	0	0
Gyroelongated pentagonal birotunda	40	52	90	40	0	12	0	0	0	0	20	20	0	0	0
Augmented triangular prism	7	8	13	6	2	0	0	0	0	2	5	0	0	0	0
Biaugmented triangular prism	8	11	17	10	1	0	0	0	0	0	6	2	0	0	0
Triaugmented triangular prism	9	14	21	14	0	0	0	0	0	0	3	6	0	0	0
Augmented pentagonal prism	11	10	19	4	4	2	0	0	0	6	5	0	0	0	0
Biaugmented pentagonal prism	12	13	23	8	3	2	0	0	0	2	10	0	0	0	0

Continued: The polyhedron profile statistic (with 15 features) for the 92 Johnson solids

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Augmented hexagonal prism	13	11	22	4	5	0	2	0	0	8	5	0	0	0	0
Parabiaugmented hexagonal prism	14	14	26	8	4	0	2	0	0	4	10	0	0	0	0
Metabiaugmented hexagonal prism	14	14	26	8	4	0	2	0	0	4	10	0	0	0	0
Triaugmented hexagonal prism	15	17	30	12	3	0	2	0	0	0	15	0	0	0	0
Augmented dodecahedron	21	16	35	5	0	11	0	0	0	15	5	1	0	0	0
Parabiaugmented dodecahedron	22	20	40	10	0	10	0	0	0	10	10	2	0	0	0
Metabiaugmented dodecahedron	22	20	40	10	0	10	0	0	0	10	10	2	0	0	0
Triaugmented dodecahedron	23	24	45	15	0	9	0	0	0	5	15	3	0	0	0
Metabidiminished icosahedron	10	12	20	10	0	2	0	0	0	2	6	2	0	0	0
Tridiminished icosahedron	9	8	15	5	0	3	0	0	0	6	3	0	0	0	0
Augmented tridiminished icosahedron	10	10	18	7	0	3	0	0	0	4	6	0	0	0	0
Augmented truncated tetrahedron	15	14	27	8	3	0	3	0	0	6	9	0	0	0	0
Augmented truncated cube	28	22	48	12	5	0	0	5	0	16	12	0	0	0	0
Biaugmented truncated cube	32	30	60	16	10	0	0	4	0	8	24	0	0	0	0
Augmented truncated dodecahedron	65	42	105	25	5	1	0	0	11	50	15	0	0	0	0
Parabiaugmented truncated dodecahedron	70	52	120	30	10	2	0	0	10	40	30	0	0	0	0
Metabiaugmented truncated dodecahedron	70	52	120	30	10	2	0	0	10	40	30	0	0	0	0
Triaugmented truncated dodecahedron	75	62	135	35	15	3	0	0	9	30	45	0	0	0	0
Gyrate rhombicosidodecahedron	60	62	120	20	30	12	0	0	0	0	60	0	0	0	0
Parabigyrate rhombicosidodecahedron	60	62	120	20	30	12	0	0	0	0	60	0	0	0	0
Metabigyrate rhombicosidodecahedron	60	62	120	20	30	12	0	0	0	0	60	0	0	0	0
Trigyrate rhombicosidodecahedron	60	62	120	20	30	12	0	0	0	0	60	0	0	0	0
Diminished rhombicosidodecahedron	55	52	105	15	25	11	0	0	1	10	45	0	0	0	0
Paragyrate diminished rhombicosidodecahedron	55	52	105	15	25	11	0	0	1	10	45	0	0	0	0
Metagyrate diminished rhombicosidodecahedron	55	52	105	15	25	11	0	0	1	10	45	0	0	0	0
Bigyrate diminished rhombicosidodecahedron	55	52	105	15	25	11	0	0	1	10	45	0	0	0	0
Parabidiminished rhombicosidodecahedron	50	42	90	10	20	10	0	0	2	20	30	0	0	0	0
Metabidiminished rhombicosidodecahedron	50	42	90	10	20	10	0	0	2	20	30	0	0	0	0
Gyrate bidiminished rhombicosidodecahedron	50	42	90	10	20	10	0	0	2	20	30	0	0	0	0

Continued: The polyhedron profile statistic (with 15 features) for the 92 Johnson solids

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Tridiminished rhombicosidodecahedron	45	32	75	5	15	9	0	0	3	30	15	0	0	0	0
Snub disphenoid	8	12	18	12	0	0	0	0	0	0	4	4	0	0	0
Snub square antiprism	16	26	40	24	2	0	0	0	0	0	0	16	0	0	0
Sphenocorona	10	14	22	12	2	0	0	0	0	0	6	4	0	0	0
Augmented sphenocorona	11	17	26	16	1	0	0	0	0	0	3	8	0	0	0
Sphenomegacorona	12	18	28	16	2	0	0	0	0	0	4	8	0	0	0
Hebesphenomegacorona	14	21	33	18	3	0	0	0	0	0	4	10	0	0	0
Disphenocingulum	16	24	38	20	4	0	0	0	0	0	4	12	0	0	0
Bilunabirotunda	14	14	26	8	2	4	0	0	0	4	10	0	0	0	0
Triangular hebesphenorotunda	18	20	36	13	3	3	1	0	0	0	18	0	0	0	0

Table A3.3: The polyhedron profile statistic (with 15 features) for the 92 Johnson solids.

Appendix 3.4

The following table shows the polyhedron profile statistic (with 15 features) for the 13 Catalan solids:

Solid Name	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
Triakis Tetrahedron	8	12	18	12	0	0	0	0	0	4	0	0	4	0	0
Rhombic Dodecahedron	14	12	24	0	12	0	0	0	0	8	6	0	0	0	0
Triakis Octahedron	14	24	36	24	0	0	0	0	0	8	0	0	0	6	0
Tetrakis Hexahedron	14	24	36	24	0	0	0	0	0	0	6	0	8	0	0
Deltoidal Icositetrahedron	26	24	48	0	24	0	0	0	0	8	18	0	0	0	0
Disdyakis dodecahedron	26	48	72	48	0	0	0	0	0	0	12	0	8	6	0
Rhombic triacontahedron	32	30	60	0	30	0	0	0	0	20	0	12	0	0	0
Triakis icosahedron	32	60	90	60	0	0	0	0	0	20	0	0	0	0	12
Pentakis dodecahedron	32	60	90	60	0	0	0	0	0	0	0	12	20	0	0
Pentagonal icositetrahedron	38	24	60	0	0	24	0	0	0	32	6	0	0	0	0
Deltoidal hexecontahedron	62	60	120	0	60	0	0	0	0	20	30	12	0	0	0
Disdyakis triacontahedron	62	120	180	120	0	0	0	0	0	0	30	0	20	0	12
Pentagonal hexecontahedron	92	60	150	0	0	60	0	0	0	80	0	12	0	0	0

Table A3.4: The polyhedron profile statistic (with 15 features) for the 13 Catalan solids.

Appendix 3.5

As discussed in Chapter 3, the polyhedron profile statistic with 15 features cannot alone differentiate all standard polyhedral shapes. The solids inside parentheses below show the same polyhedron profile statistic. The profiles are different across parentheses.

1. (Cuboctahedron, Triangular orthobicupola)
2. (Square gyrobicupola, Square orthobicupola)
3. (Pentagonal orthobicupola, Pentagonal gyrobicupola)
4. (Pentagonal orthocupolarotunda, Pentagonal gyrocupolarotunda)
5. (Elongated triangular gyrobicupola, Elongated triangular orthobicupola)
6. (Elongated pentagonal orthobicupola, Elongated pentagonal gyrobicupola)
7. (Elongated pentagonal orthocupolarotunda, Elongated pentagonal gyrocupolarotunda)
8. (Elongated pentagonal orthobirotunda, Elongated pentagonal gyrobirotunda)
9. (Parabiaugmented hexagonal prism, Metabiaugmented hexagonal prism)
10. (Metabiaugmented dodecahedron, Parabiaugmented dodecahedron)
11. (Parabiaugmented truncated dodecahedron, Metabiaugmented truncated dodecahedron)
12. (Elongated square gyrobicupola, Rhombicuboctahedron)
13. (Pentagonal orthobirotunda, Icosidodecahedron)
14. (Parabidiminished rhombicosidodecahedron, Metabidiminished rhombicosidodecahedron, Gyrate bidiminished rhombicosidodecahedron)
15. (Gyrate rhombicosidodecahedron, Parabigyrate rhombicosidodecahedron, Metabigyrate rhombicosidodecahedron, Trigyrate rhombicosidodecahedron, Rhombicosidodecahedron)
16. (Triaugmented truncated dodecahedron, Gyrate rhombicosidodecahedron, Parabigyrate rhombicosidodecahedron, Metabigyrate rhombicosidodecahedron, Trigyrate rhombicosidodecahedron, Rhombicosidodecahedron)

The following solids inside parentheses show the same polyhedron profile statistic, same edge adjacency matrix and same face adjacency matrix (N_v = No. of vertices).

1. (Elongated triangular orthobicupola, Elongated triangular gyrobicupola), $N_v = 18$
2. (Elongated square gyrobicupola, Rhombicuboctahedron) , $N_v = 24$
3. (Elongated pentagonal orthobicupola, Elongated pentagonal gyrobicupola) , $N_v = 30$
4. (Elongated pentagonal orthocupolarotunda, Elongated pentagonal gyrocupolarotunda) , $N_v = 35$
5. (Elongated pentagonal orthobirotunda, Elongated pentagonal gyrobirotunda), $N_v =40$
6. (Parabiaugmented truncated dodecahedron, Metabiaugmented truncated dodecahedron) , $N_v = 70$
7. (Parabigyrate rhombicosidodecahedron, Metabigyrate rhombicosidodecahedron), $N_v = 60$
8. (Paragyrate diminished rhombicosidodecahedron, Metagyrate diminished rhombicosidodecahedron) , $N_v = 55$

Appendix for Chapter 4

Appendix 4.1

The following table provides some basic statistical measures on the edge lengths from individual metabolosomes. The measurements are in pixel where 1 pixel = 9.62 Å (approximately).

Metabolosome	Edge Count	Min	Max	Range	Mean	SD
1	26	46.10	150.50	104.41	96.45	35.90
2	23	27.49	121.78	94.30	78.52	26.19
3	28	69.20	161.31	92.10	123.47	23.71
4	26	43.46	169.38	125.92	105.98	37.61
5	26	42.98	127.62	84.64	87.71	20.57
6	22	74.54	186.17	111.63	125.76	29.47
7	24	44.90	129.22	84.32	86.14	23.60
8	24	43.57	131.58	88.01	94.64	23.77
9	25	65.76	170.38	104.62	113.14	27.71
10	22	50.35	134.31	83.95	87.08	20.77
11	28	42.07	116.77	74.70	81.79	22.29
12	23	49.92	152.50	102.58	88.90	22.00
13	19	42.30	152.33	110.03	96.75	29.19
14	24	60.05	121.00	60.95	87.29	16.25
15	25	36.96	120.59	83.63	77.35	23.70
16	22	43.46	164.94	121.48	106.57	38.66
17	23	38.61	143.00	104.39	105.00	26.93
18	28	45.74	107.76	62.01	73.79	16.00
19	16	38.66	156.30	117.64	90.16	34.60
20	26	36.35	151.77	115.42	90.56	30.26
21	22	31.61	105.19	73.58	62.53	21.03
22	26	43.05	113.58	70.53	81.57	19.50
23	26	49.67	155.37	105.70	113.59	26.44
24	23	72.84	161.75	88.91	114.55	27.33
25	24	62.16	149.57	87.41	93.46	24.01
26	23	40.99	177.39	136.40	102.00	34.00
27	24	55.29	140.52	85.23	91.49	23.00
28	25	35.34	111.86	76.52	63.09	20.16
29	20	47.22	84.31	37.09	68.53	11.95
30	27	29.81	111.13	81.32	68.46	23.25

Table A4.1: Some basic measurements on the edge lengths from 30 metabolosomes.

Appendix 4.2

The following table provides some basic statistical measures on the triangular face areas from 30 metabolosomes. The measurements are in square pixel where 1 pixel is approximately 9.62 Å.

Metabolosome	**Face Count**	**Min (×10^2)**	**Max (×10^2)**	**Range (×10^2)**	**Mean (×10^2)**	**SD (×10^2)**
1	16	11.37	57.38	46.01	35.40	15.06
2	14	5.14	47.91	42.78	24.87	12.47
3	16	33.71	98.67	64.95	64.06	20.01
4	16	12.27	68.50	56.22	42.35	15.90
5	12	21.61	54.97	33.35	34.66	10.23
6	12	35.64	91.84	56.20	60.68	22.21
7	12	13.86	47.83	33.98	30.14	11.15
8	12	17.72	56.17	38.45	38.34	11.44
9	13	23.03	88.64	65.61	52.34	20.02
10	12	14.69	52.84	38.16	30.73	10.86
11	16	8.87	45.60	36.73	27.64	10.26
12	14	16.93	53.89	36.96	32.13	10.91
13	8	19.76	71.50	51.74	42.67	17.42
14	12	21.41	39.10	17.69	30.75	5.56
15	11	13.67	34.11	20.44	25.62	6.82
16	12	14.59	99.49	84.89	46.17	27.13
17	10	14.84	73.54	58.70	46.91	18.86
18	16	13.35	40.16	26.82	22.89	7.22
19	8	13.33	57.63	44.30	30.06	15.44
20	16	13.91	56.02	42.11	31.90	14.30
21	12	6.58	32.18	25.60	15.47	8.19
22	16	11.69	41.49	29.80	27.40	10.44
23	12	23.12	79.46	56.34	50.28	15.94
24	14	30.23	80.55	50.32	51.79	15.22
25	12	20.11	48.04	27.93	33.24	9.35
26	10	25.68	64.44	38.76	44.86	12.26
27	12	16.43	56.23	39.80	34.45	11.62
28	14	7.87	21.55	13.67	14.70	4.23
29	10	13.39	28.32	14.93	20.78	4.94
30	14	6.78	30.88	24.11	15.60	6.88

Table A4.2: Some basic measurements on the triangular face areas from 30 metabolosomes.

Appendix for Chapter 5

Appendix 5.1

As discussed in Chapter 5, we developed a structural distance model which contains parameters associated with weights and robustness of the features of a polyhedron. A possible set of values we used for shape prediction is provided here**.**

Features	l_1	l_2	w
Number of Vertices	1	1	0.5387
Number of Faces	1	4	0.0252
Number of Edges	1	1	0.0252
Number of Triangles	1	2	0.0252
Number of Quadrilaterals	4	2	0.0252
Number of Pentagons	4	1	0.1333
Number of Hexagons	1	1	0.0252
Number of Octagons	1	1	0.0252
Number of Decagons	1	1	0.0252
Number of Vertices with 3 Edges	1	1	0.0252
Number of Vertices with 4 Edges	4	1	0.0252
Number of Vertices with 5 Edges	1	1	0.0252
Number of Vertices with 6 Edges	1	1	0.0252
Number of Vertices with 8 Edges	1	1	0.0252
Number of Vertices with 10 Edges	1	1	0.0252

Table A5.1: A possible set of values for parameters in the structural distance model.

Appendix 5.2

The name of the predicted shapes for metabolosomes: the 'Nearest' solid has the minimum structural distance; ' Second Nearest' and 'Third Nearest' solids are based on second and third minimum structural distances and visual inspection on metabolosome structures.

Metabolosome	Nearest Solid	Second Nearest Solid	Third Nearest Solid
1	Augmented sphenocorona (J_{87})	Icosahedron (P)	
2	Sphenocorona (J_{86})	Gyro-elongated square bipyramid (J_{17})	Augmented sphenocorona (J_{87})
3	Sphenomegacorona (J_{88})	Icosahedron (P)	
4	Augmented sphenocorona (J_{87})		
5	Elongated pentagonal bipyramid (J_{16})	Sphenomegacorona (J_{88})	Icosahedron (P)
6	Sphenocorona (J_{86})	Augmented sphenocorona (J_{87})	Gyroelongated square bipyramid (J_{17})
7	Augmented sphenocorona (J_{87})	Sphenomegacorona (J_{88})	
8	Augmented sphenocorona (J_{87})	Sphenocorona (J_{86})	
9	Augmented sphenocorona (J_{87})	Icosahedron (P)	
10	Sphenocorona (J_{86})	Gyroelongated square bipyramid (J_{17})	Augmented sphenocorona(J_{87})
11	Sphenomegacorona (J_{88})	Icosahedron (P)	
12	Sphenocorona (J_{86})	Gyroelongated square bipyramid (J_{17})	Augmented sphenocorona(J_{87})
13	Metabidiminished icosahedron (J_{62})		
14	Augmented sphenocorona (J_{87})	Gyroelongated pentagonal pyramid(J_{11})	
15	Augmented sphenocorona (J_{87})		
16	Sphenocorona (J_{86})	Gyroelongated square bipyramid (J_{17})	
17	Augmented sphenocorona (J_{87})	Sphenomegacorona (J_{88})	
18	Sphenomegacorona (J_{88})	Icosahedron (P)	
19	Biaugmented triangular prism (J_{50})	Snub disphenoid (J_{84})	
20	Augmented sphenocorona (J_{87})		
21	Sphenocorona (J_{86})	Gyroelongated square bipyramid (J_{17})	
22	Augmented sphenocorona (J_{87})	Icosahedron (P)	Sphenomegacorona (J_{88})
23	Cubeoctahedron (A)	Icosahedron (P)	
24	Gyroelongated square bipyramid (J_{17})	Sphenocorona (J_{86})	Augmented sphenocorona (J_{87})
25	Augmented sphenocorona (J_{87})	Sphenomegacorona (J_{88})	Icosahedron (P)
26	Augmented sphenocorona (J_{87})	Sphenomegacorona (J_{88})	Elongated pentagonal bipyramid
27	Augmented sphenocorona (J_{87})	Icosahedron (P)	
28	Augmented sphenocorona (J_{87})	Icosahedron (P)	
29	Metabidimisinhed icosahedron (J_{62})		
30	Augmented sphenocorona (J_{87})	Sphenomegacorona (J_{88})	Icosahedron (P)

Table A5.2: Predicted solids' names, obtained through the structural distance model. 'A' stands for Archimedean solids, 'P' stands for platonic solids and 'J' stands for Johnson solids, followed by the Johnson solid number.

Appendix for Chapter 6

Appendix 6.1

As described in Chapter 6, from each of the metabolosomes, 231 features were extracted. Here, the completely visible features are provided from 30 incomplete metabolosomes. 'M' stands for metabolosome.

M	N_V	N_F	N_E	N_F^3	N_F^4	N_F^5	N_F^6	N_F^8	N_F^{10}	N_V^3	N_V^4	N_V^5	N_V^6	N_V^8	N_V^{10}
1	7	16	22	5	0	0	0	0	0	0	1	5	0	0	0
2	7	15	21	3	0	0	0	0	0	0	3	3	0	0	0
3	9	16	23	5	0	0	0	0	0	0	4	2	0	0	0
4	8	15	21	6	0	0	0	0	0	0	2	3	0	0	0
5	9	15	23	1	3	0	0	0	0	0	4	2	0	0	0
6	7	14	17	3	0	0	0	0	0	0	3	1	0	0	0
7	8	14	21	4	0	0	0	0	0	0	3	3	0	0	0
8	8	14	21	2	2	0	0	0	0	0	3	3	0	0	0
9	8	13	20	2	1	0	0	0	0	0	4	1	0	0	0
10	7	13	19	2	1	0	0	0	0	0	5	1	0	0	0
11	9	17	24	5	1	0	0	0	0	0	2	5	0	0	0
12	8	15	20	4	1	0	0	0	0	0	3	2	0	0	0
13	7	10	15	0	0	0	0	0	0	3	2	0	0	0	0
14	8	14	21	3	0	0	0	0	0	0	6	1	0	0	0
15	9	15	21	4	1	0	0	0	0	0	5	1	0	0	0
16	8	14	21	4	1	0	0	0	0	0	4	3	0	0	0
17	9	14	22	5	1	0	0	0	0	0	6	2	0	0	0
18	10	18	24	9	0	0	0	0	0	0	2	6	0	0	0
19	6	10	14	2	0	0	0	0	0	0	4	0	0	0	0
20	9	17	23	6	1	0	0	0	0	0	2	4	0	0	0
21	8	14	21	4	1	0	0	0	0	0	5	2	0	0	0
22	8	16	23	4	0	0	0	0	0	0	2	4	0	0	0
23	9	16	21	4	1	0	0	0	0	0	4	1	0	0	0
24	8	15	21	8	0	0	0	0	0	0	2	4	0	0	0
25	8	13	19	1	1	0	0	0	0	0	3	1	0	0	0
26	9	13	20	2	3	0	0	0	0	1	3	1	0	0	0
27	9	13	20	5	1	0	0	0	0	0	4	1	0	0	0
28	8	15	22	5	0	0	0	0	0	0	4	2	0	0	0
29	7	11	17	3	0	0	0	0	0	2	2	1	0	0	0
30	9	16	23	7	0	0	0	0	0	0	3	4	0	0	0

Table A6.1: Some characteristic features from the metabolosomes.

Appendix 6.2

This section presents misclassification probabilities for different truncation proportions, different feature combinations and comparisons among them. The distributions of overall misclassification probabilities for 55 solids from the Bayes classifiers, LDA and SVM are calculated based on different truncation proportions. These probabilities are calculated using all 231 features from a *TPPS* (truncated polyhedron profile statistic).

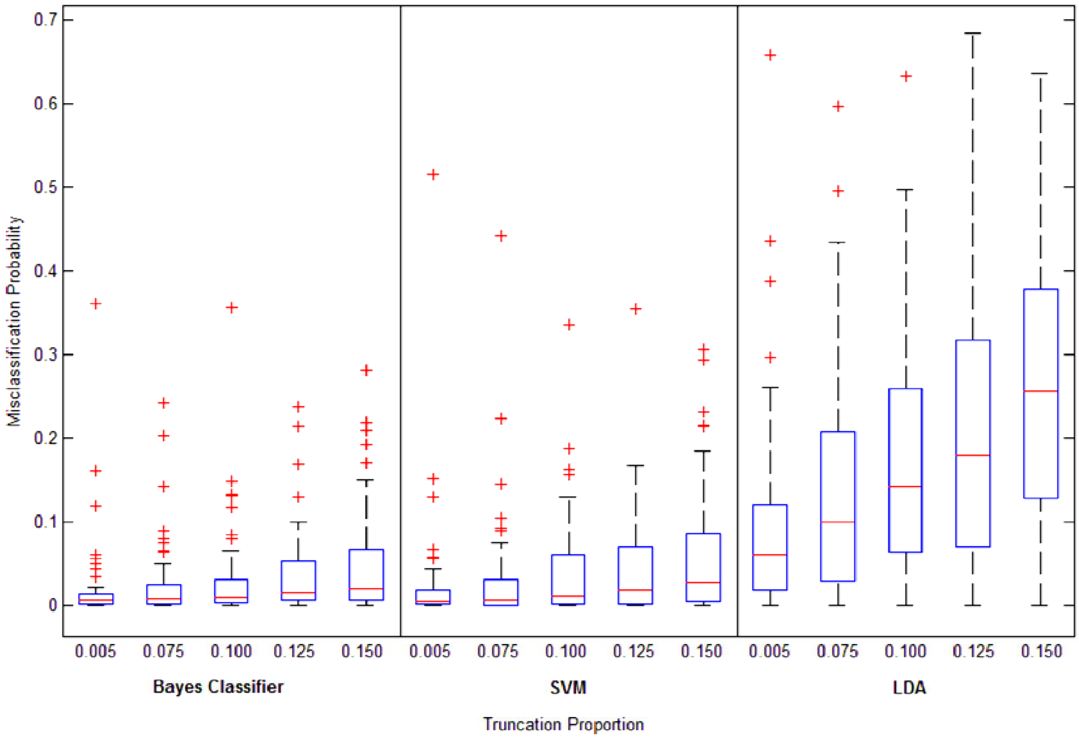


Figure A6.1: Misclassification probabilities from three classifiers for varying truncation proportions.

Intuitively, higher truncation proportions remove more 'information' from a polyhedron, so the misclassifications are also expected to be higher. This is true for all of the Bayes classifiers, LDA and SVM. Figure A6.1 displays this observation.

The following table summarizes these probabilities for LDA, SVM and the Bayes Classifiers. As discussed before, the Bayes classifiers and SVM work better than LDA for this classification problem, irrespective of truncation proportion.

Bayes Classifier	**Truncation Proportion**				
	0.0500	**0.0750**	**0.1000**	**0.1250**	**0.1500**
Minimum	0.0000	0.0000	0.0000	0.0000	0.0000
Maximum	0.3616	0.2416	0.3560	0.2380	0.2808
Mean	0.0217	0.0257	0.0323	0.0385	0.0480
Standard Deviation	0.0546	0.0475	0.0577	0.0524	0.0645

LDA	**Truncation Proportion**				
	0.1000	**0.1500**	**0.2000**	**0.2500**	**0.3000**
Minimum	0.0000	0.0000	0.0000	0.0000	0.0000
Maximum	0.6580	0.5976	0.6336	0.6840	0.6364
Mean	0.0999	0.1444	0.1791	0.2046	0.2568
Standard Deviation	0.1252	0.1415	0.1521	0.1712	0.1773

SVM	**Truncation Proportion**				
	0.1000	**0.1500**	**0.2000**	**0.2500**	**0.3000**
Minimum	0.0000	0.0000	0.0000	0.0000	0.0000
Maximum	0.5160	0.4424	0.3360	0.3544	0.3068
Mean	0.0250	0.0325	0.0394	0.0459	0.0606
Standard Deviation	0.0735	0.0701	0.0621	0.0657	0.0788

Table A6.2: The summary of misclassification probabilities, calculated for varied truncation proportion and across three methods.

Appendix 6.3

Here we showed the impact of feature selection on misclassification probabilities. Let us fix the truncation proportion = 0.10. We define three feature sets as: Set_1 = $\{N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*\}$, Set_2 = $\{N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*, ATN_V^*, ATN_F^*\}$ and Set_3 = $\{N_{TV}, N_{TF}, N_{TE}, TN_V^*, TN_F^*, ATN_V^*, ATN_F^*, eADJt, fADJt\}$. It is expected that the inclusion of more features improves the error rates. Figure A6.2 shows the misclassification probabilities from different feature sets across three methods.

As a simple measure to evaluate the impacts of excluding features from TPPS on misclassification probabilities, we estimated the k, such that $L^*(k) \leq 0.10$, calculated across above three feature sets and three classifiers. Table A6.3 gives the result. It shows the impact is large.

	Set_1	**Set_2**	**Set_3**
Bayes	37	44	50
LDA	14	11	19
SVM	32	39	48

Table A6.3: The number of solids with misclassification probabilities ≤ 0.10.

Appendix 6.4

In the following table, the solids are ordered based on the posterior probabilities, and first three solids (predicted by LDA) with their posterior probabilities are provided.

Metabolosome	1st Solid	2nd Solid	3rd Solid	P(1st Solid)	P(2nd Solid)	P(3rd Solid)
1	17	87	88	0.6608	0.3339	0.0053
2	11	17	51	0.9917	0.0051	0.0017
3	11	88	87	0.9999	0.0001	0.0000
4	11	17	87	0.9842	0.0109	0.0048
5	11	16	86	0.6577	0.3423	0.0000
6	86	16	84	0.6694	0.3280	0.0014
7	86	11	16	0.9312	0.0506	0.0172
8	11	86	51	0.8410	0.1500	0.0062
9	86	16	10	0.8606	0.1366	0.0024
10	16	86	50	0.8075	0.1493	0.0283
11	87	88	17	0.5380	0.4197	0.0423
12	16	86	11	0.9963	0.0035	0.0002
13	52	63	3	0.8734	0.1226	0.0030
14	16	27	99	0.9954	0.0025	0.0009
15	16	27	99	0.9778	0.0155	0.0067
16	86	10	16	0.8827	0.1161	0.0008
17	16	86	10	0.9930	0.0049	0.0021
18	17	87	88	0.7330	0.2637	0.0034
19	15	14	50	0.7063	0.2937	0.0000
20	87	88	17	0.5750	0.4108	0.0142
21	86	10	16	0.9353	0.0548	0.0097
22	88	87	11	0.6465	0.3342	0.0151
23	16	11	86	0.9996	0.0004	0.0000
24	11	17	51	0.4444	0.3714	0.1804
25	16	86	11	0.9997	0.0003	0.0000
26	16	86	84	0.6543	0.3455	0.0001
27	16	86	10	0.9764	0.0236	0.0000
28	16	86	11	0.9718	0.0243	0.0039
29	62	9	50	1.0000	0.0000	0.0000
30	11	17	87	0.8909	0.0603	0.0452

Table A6.4: The posterior probability based ordered solids and their corresponding probabilities. Only top three solids with maximum posterior probabilities are shown.

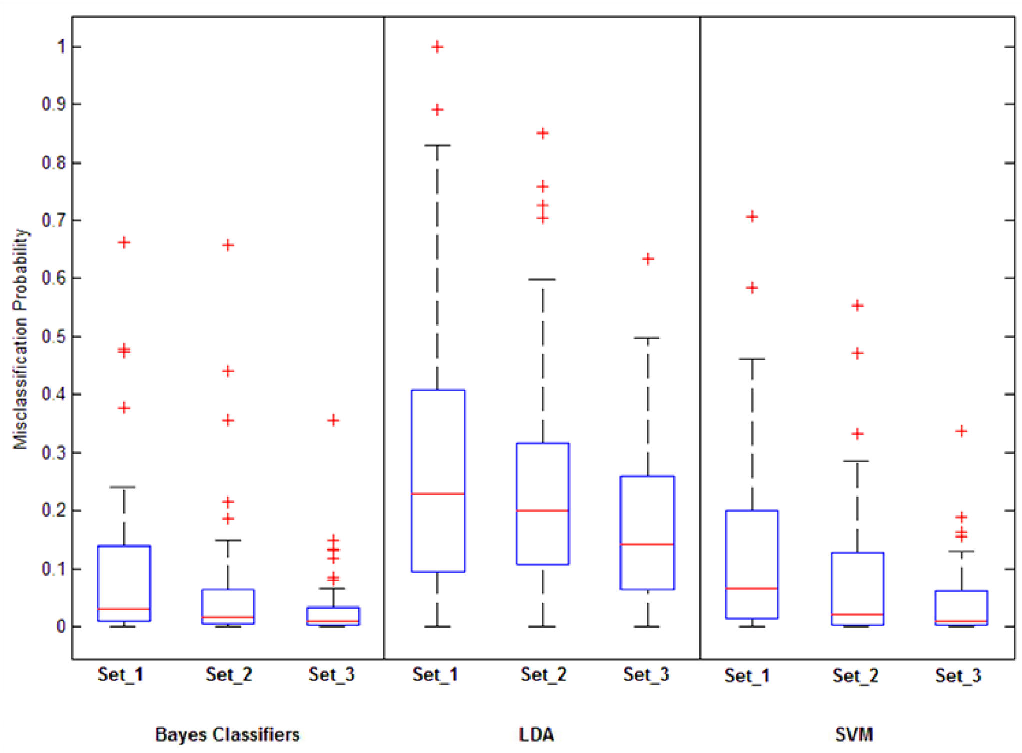


Figure A6.2: Misclassification probabilities from three classifiers for different feature sets.

Publications by the Author

Book Chapter

Encyclopedia of Mathematics and Society (Engineering, Technology, and Medicine), Chapter: *Search Engines*, Salem Press, 2012.

Research Papers

Rubin GD, Roos JE, Tall M, Harrawood B, Bag S, Donald LL, Seaman DM, Koweek LM, McAdams HP, Napel S and Choudhury KR. *Characterizing Search, Recognition and Decision in the Detection of Lung Nodules in CT Scans: Elucidation with Eye Tracking*. Radiology, 2014. 274(1), 276-286.

Bag S, Liang M, Prentice MB, Choudhury KR. *Classification of polyhedral shapes from individual incomplete cryo-electron tomography reconstructions*. (Submitted to BMC Bioinformatics, April 2015).

Liang M, Bag S, Choudhury KR, Jensen GJ, McDowall AW, Warren MJ, Prentice MB. *Bacterial microcompartments in heterotrophic bacteria are non-uniform polyhedra resembling Johnson solids*. (Submitted to Journal of Molecular Biology, April 2014).

Conference Presentations

New methods for predicting structure from cryo-EM tomography. Bag S, Liang M, Prentice MB and Choudhury KR. Poster presentation on Duke Basic Science Day, October 2013.

Testing for the shape of a polyhedral cellular inclusion. Bag S, Roy Choudhury K, Liang M and Prentice MB. Poster presentation at Joint Statistical Meeting 2011.