AUTOMATED MORPHOLOGY ANALYSIS OF NANOPARTICLES

A Dissertation

by

CHIWOO PARK

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2011

Major Subject: Industrial Engineering

Automated Morphology Analysis of Nanoparticles

Copyright 2011 Chiwoo Park

AUTOMATED MORPHOLOGY ANALYSIS OF NANOPARTICLES

A Dissertation

by

CHIWOO PARK

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Yu Ding
	Jianhua Z. Huang
Committee Members,	Abhijit Deshmukh
	Hong Liang
Head of Department,	Brett A. Peters

August 2011

Major Subject: Industrial Engineering

ABSTRACT

Automated Morphology Analysis of Nanoparticles. (August 2011)

Chiwoo Park, B.S., Seoul National University

Co–Chairs of Advisory Committee: Dr. Yu Ding
Dr. Jianhua Z. Huang

The functional properties of nanoparticles highly depend on the surface morphology of the particles, so precise measurements of a particle's morphology enable reliable characterizing of the nanoparticle's properties. Obtaining the measurements requires image analysis of electron microscopic pictures of nanoparticles. Today's labor-intensive image analysis of electron micrographs of nanoparticles is a significant bottleneck for efficient material characterization. The objective of this dissertation is to develop automated morphology analysis methods.

Morphology analysis is comprised of three tasks: separate individual particles from an agglomerate of overlapping nano-objects (image segmentation); infer the particle's missing contours (shape inference); and ultimately, classify the particles by shape based on their complete contours (shape classification). Two approaches are proposed in this dissertation: the divide-and-conquer approach and the convex shape analysis approach. The divide-and-conquer approach solves each task separately, taking less than one minute to complete the required analysis, even for the largest-sized micrograph. However, its separating capability of particle overlaps is limited, meaning that it is able to split only touching particles. The convex shape analysis approach solves shape inference and classification simultaneously for better accuracy, but it requires more computation time, ten minutes for the biggest-sized electron micrograph. However, with a little sacrifice of time efficiency, the second approach

achieves far superior separation than the divide-and-conquer approach, and it handles the chain-linked structure of particle overlaps well.

The capabilities of the two proposed methods cannot be substituted by generic image processing and bio-imaging methods. This is due to the unique features that the electron microscopic pictures of nanoparticles have, including special particle overlap structures, and large number of particles to be processed. The application of the proposed methods to real electron microscopic pictures showed that the two proposed methods were more capable of extracting the morphology information than the state-of-the-art methods. When nanoparticles do not have many overlaps, the divide-and-conquer approach performed adequately. When nanoparticles have many overlaps, forming chain-linked clusters, the convex shape analysis approach performed much better than the state-of-the-art alternatives in bio-imaging.

The author believes that the capabilities of the proposed methods expedite the morphology characterization process of nanoparticles. The author further conjectures that the technical generality of the proposed methods could even be a competent alternative to the current methods analyzing general overlapping convex-shaped objects other than nanoparticles.

Dedicated to My Family, Friends and Colleagues

ACKNOWLEDGMENTS

First, I thank my dissertation co-advisors, Dr. Yu Ding and Dr. Jianhua Z. Huang, who have enthusiastically guided my research and given me the benefit of their expertise. I will be forever grateful to them for teaching me how to become an independent researcher and a well-rounded academic colleague. Their enthusiasm has been the driving force behind my successful studies at Texas A&M University.

I also would like to acknowledge the research group of Dr. Hong Liang, including Dr. Subrata Kundu and Dr. David Huitink, who kindly provided the transmission electron microscopy images of nanoparticles for this research work. They gave freely of their time and expertise to discuss my ideas and experiments. Dr. Liang suggested the research problem of this dissertation and then helped me to develop and refine the direction of my research. Thank you!

I also express sincere appreciation to Dr. Abhijit V. Deskmukh for serving on my committee and for encouraging me to explore several diverse viewpoints. Each time we met, he challenged me with several fundamental and insightful questions. Answering the questions was very helpful to improving the quality of my research. I am also grateful to Dr. Jim Ji for his technical assistance with image processing. As an expert in bio-medical imaging, he introduced me to a multitude of research ideas and literature related to image processing in his lecture and our discussion.

I am thankful to the past and current members in the Advanced Metrology Laboratory, especially Dr. Jung-Jin Cho, Dr. Haifeng (Heidi) Xia, Dr. Abhishek K. Shrivastava, Dr. Eunshin Byon, Mr. Arash Pourhabib, Mr. Lokesh Kulkarni. Also, special thanks to my best friends, Dr. Young Myoung Ko, Dr. Soondo Hong, Mr. Sangdo Choi and Mr. Baik Song An. Your encouragement, support, feedback and our good times together made the time fly.

Last but not least, I owe my deepest gratitude to my mother, father, sister and brother. Your endless love has always supported me. You are my greatest source of joy and the source of my strength.

TABLE OF CONTENTS

CHAPTER Page

I INTRODUCTION 1

I.1. Nanoparticles and Their Morphology-related Properties 1
I.2. The Morphology Analysis Problem 1
I.3. The Objective of the Dissertation 3
I.4. Image Segmentation 4
I.5. Shape Inference and Classification 5
I.6. Main Contribution and Organization of the Dissertation 7

II STATE OF THE ART 9

II.1. Particle Characterization Technique 9
II.2. Morphology Measurement Methods in Nano-Imaging . . 12
II.3. Cell Segmentation Methods in Bio-medical Imaging . . . 16
II.4. General Image Processing Methods 18
II.4.1. Image Segmentation Methods 18
II.4.2. Shape Classification Methods 20

III A DIVIDE-AND-CONQUER APPROACH FOR ANALYZING WEAKLY OVERLAPPED NANOPARTICLES 22

III.1. Decomposition of Composite Contours 22
III.2. Invariant Curve Spaces for Shape Feature Extraction . . 26
III.2.1. Location- and Scale-invariant Curve Space 28
III.2.2. Shape-Similarity Measure on the Curve Space . . . 30
III.3. Define Reference Shapes 32
III.3.1. Dimension Reduction of Parametric Curves 33
III.3.2. Semi-supervised Shape Clustering 36
III.4. Principal Component Analysis on the Curve Space for Shape Inference 41
III.5. Results and Discussion 44

IV A CONVEX SHAPE ANALYSIS FOR ANALYZING CHAIN-LINKED NANOPARTICLES 53

IV.1. Limitation of the Divide-and-Conquer Approach 54
IV.2. Convex Decomposition of Chain-linked Nanoparticles . . 55
IV.2.1. Ultimate Erosion for Overlapping Convex Sets . . 56

CHAPTER Page

IV.2.2. Extraction and Association of Contour Evidences . 65
IV.3. Gaussian Mixture Model for Contour Inference and Shape Classification 68
IV.3.1. Expectation Maximization via the `ECM` algorithm . 71
IV.3.2. Approximate Data Parameterization 74
IV.4. Results and Discussion 76
IV.4.1. Results of Shape Classification 80
IV.4.2. Discussion 82

V CONCLUSION 84

REFERENCES 89

APPENDIX A 101

APPENDIX B 104

APPENDIX C 109

APPENDIX D 113

VITA 114

LIST OF TABLES

TABLE Page

1 Particle characterization techniques 13

2 Summary on the related research 15

3 Comparison of the divide-and-conquer approach with `ImageJ` in terms of particle identification rate. 52

4 Comparison of performances on nanoparticle identification 80

LIST OF FIGURES

FIGURE Page

1 Decomposition of a composite contour. 23

2 A failing case of the contour decomposition method. 23

3 Reflex in a composite contour. 25

4 Convex hulls of composite edges 26

5 Illustration of decomposition of composite contours. 27

6 Representation of nanoparticle boundaries 29

7 Basic idea of feature extraction. 32

8 Rotational invariance feature extraction 36

9 Results of the semi-supervised learning 40

10 Identification results from our method for Image 1 46

11 Identification results from `ImageJ` for Image 1 46

12 Identification results from our method for Image 2 47

13 Identification results from `ImageJ` for Image 2 47

14 Identification results from our method for Image 3 48

15 Identification results from `ImageJ` for Image 3 48

16 Identification results from our method for Image 4 49

17 Identification results from `ImageJ` for Image 4 49

18 Identification results from our method for Image 5 50

19 Identification results from `ImageJ` for Image 5 50

20 Identification results from our method for Image 6 51

FIGURE Page

21 Identification results from `ImageJ` for Image 6 51

22 Ultimate erosion for convex sets 56

23 Intuitive examples explaining Assumption 3 58

24 Electron micrographs picturing the chain-linked clusters of nanoparticles. 59

25 Practical Implication of Lemma 7 61

26 Classification of edge pixels by the relevance to markers 67

27 Results from low-degree overlapping cases 77

28 Results from medium-degree overlapping cases. 77

29 Results from medium-degree overlapping cases (continued). 78

30 Results from high-degree overlapping cases. 78

31 Results of contour inference 81

32 Results of shape classification 82

33 Different configurations of overlaps with different level of noises σ. . . 105

34 Performance of the UECS ($\rho = 0.3$). 106

35 Performance of the UECS ($\rho = 0.2$). 107

36 Performance of the UECS ($\rho = 0.1$). 108

CHAPTER I

INTRODUCTION

I.1. Nanoparticles and Their Morphology-related Properties

The British and several international standard organizations define nanoparticles as ultrafine particles having one or more dimensions on the order of 100 nm or less (BSI, 2005; ISO, 2009). At this order of magnitude, the functional properties of nanoparticles are significantly different from the properties of the bulk material having the same chemical composition. This is mainly because the fraction of atoms or molecules located at the surface of the particles increases as the size of the particles decreases below a critical size (100 nm); the atoms or molecules at the surface are more reactive to chemical bonding (Hosokawa, 2007).

In addition, nanoparticles of different shapes have different crystal facets exposed at the surface, and different facets have different degrees of chemical reactivity (Narayanan and El-Sayed, 2004). Hence, both the shape and the size of nanoparticles are important factors in determining the properties of the nanoparticles. For this reason, material scientists attempt to synthesize nanoparticles with controlled size and shape in order to have the functional properties desired for applications.

I.2. The Morphology Analysis Problem

The precise design and control of a nanoparticle synthesis process requires accurate analysis of the size and shape of the nanoparticles produced because the analysis result provides the feedback necessary to control or fine-tune the synthesis process.

The journal model is *IIE Transactions.*

In this dissertation, obtaining the size and shape characteristics of nanoparticles is called *morphology analysis of nanoparticles*.

More than 400 different techniques exist for counting, sizing and characterizing nanoparticles, but most of them are applicable only to obtaining size estimates of nanoparticles; these techniques are reviewed in Section II.1.

One way to conduct both shape and size analysis is to perform the image analysis on electron microscopic pictures, hereafter called electron micrographs, taken with either a transmission electron microscope (TEM), a scanning electron microscope (SEM), or an atomic force microscope (AFM). The advantage of using the micrographs is that sizes and shapes of individual nanoparticles can be captured, while most size-measuring instruments can capture only the average size for a population of nanoparticles. In particular, TEM is widely used to profile the size and shape characteristics of nanoparticles (Lacava *et al.*, 2001). Unless stated otherwise, from now onward in this dissertation, the term morphology analysis implies that the size and shape information of nanoparticles are to be extracted from a TEM micrograph.

However, in using TEM, it may take several attempts to prepare a diluted solvent with an appropriate particle concentration. If the concentration is too low, only a few nanoparticles in a micrograph can be observed, which may not be enough to represent the whole population. Conversely, if the concentration is too high, nanoparticles are heavily and severe overlaps will make it difficult to measure individual nanoparticles. Once the diluted solvent is ready, the solvent is imaged to a grayscale micrograph. It is then necessary to measure the morphologies of individual nanoparticles on the micrograph and to classify the particles by their shapes.

The current practice is to use image processing tools such as ImageJ Particle Analyzer (`http://rsbweb.nih.gov/ij`) and AxioVision (`http://www.zeiss.com/`), which have been popularly used for biomedical image processing. The final results are a particle size distribution and a particle shape distribution.

However, the current practice is highly labor-intensive, and requires manual counting process owing to the low accuracy of the image processing tools. Because hundreds to thousands of nanoparticles may exist in a single micrograph, the manual process takes a significant amount of time and effort, and it is almost impossible to contour the exact surfaces of nanoparticles. This is the major reason for developing an automated analytical method.

The low accuracy of the image processing tools largely stems from their incapability of handling particle overlaps. The tools fail to recognize individual nanoparticles from a cluster of overlapping particles. Hence, it is important for a morphology analysis method to be capable of handling a wide-ranging degree of particle overlap.

I.3. The Objective of the Dissertation

The objective of this dissertation is to formulate an automated morphology analysis method capable of measuring and classifying precise morphologies of individual nanoparticles on a micrograph under severe overlaps among nanoparticles. The automated method has three main components: (1) *image segmentation*, (2) *shape inference*, and (3) *shape classification*. The morphology analysis method needs an image segmentation component in order to separate overlapping particles into individual pieces, but there are several other issues. Overlaps among nanoparticles occlude partial surface contours of other particles so simply splitting the overlaps does not provide the desired morphology analysis. The occluded parts must be inferred using some prior knowledge of the morphologies of nanoparticles. This presents a *shape inference* problem on the occluded parts. In addition, the inference on the occluded parts can be definitely enhanced by utilizing information about possible shapes of nanoparticles. This raises the need to classify the particle's shape, i.e. *shape classification.*

I.4. Image Segmentation

Image segmentation is the process of partitioning an image into a set of non-overlapping regions that are meaningful to a particular application (Haralick and Shapiro, 1992). The segmentation process utilizes various features revealed in an image such as colors, textures, and edges, and finally identifies several homogeneous sub-regions in terms of the features.

Image segmentation is formally described as follows. Let I denote the set of coordinates of image pixels (x, y) and let $F(I)$ denote a homogeneity measure of I, where $F(I)$ is a measure of similarity among image features in I; $F(I) = true$ implies that the image features in I are similar. The image segmentation of I for predicate F is to partition I into disjoint subsets $C_1^{seg}, \ldots, C_n^{seg}$ such that

- $\bigcup_{i=1}^{n} C_i^{seg} = I$ with $C_i^{seg} \bigcap C_j^{seg} = \emptyset$ for $i \neq j$.
- $F(C_i^{seg}) = true$ for all i's.
- $F(C_i^{seg} \bigcup C_j^{seg}) = false$ when C_i^{seg} and C_j^{seg} are neighboring.

In the morphology analysis problem, the definition of image segmentation is changed by the following three problem-specific constraints. First, in the bright field mode of the TEM imaging, the image contrast is formed by the absorption of electrons in the specimen; thicker regions of the specimen will appear darker (Fultz and Howe, 2008). Therefore, the region occupied by the specimen (i.e. nanoparticle) can be easily separated from the remaining region (i.e. background) by thresholding the grayscale image intensities. The image segmentation needs to be performed only on the region of the specimen. As such, I in this context can be narrowed to be the region of the specimen, rather than the whole micrograph. Second, according to the theory of equilibrium shapes of crystals (Wortis, 1988), the morphology of a nanoparticle is most likely convex. Therefore, I can be further thought of as a union of convex sets; i.e. if C_i is the convex region of particle i, $I = \bigcup_{i=1}^{n} C_i$. Third, nanoparticles often overlap so that a simple partition of I may not result in C_i. In fact, the result of the simple partition, C_i^{seg}, is a subset of C_i. In the morphology analysis problem, C_i^{seg} is restricted to a convex subset of C_i, not just an arbitrary subset of C_i.

According to the three constraints, the image segmentation is defined as a simple partition that splits I into the minimal number of convex sets $\{C_1^{seg}, C_2^{seg}, \ldots, C_n^{seg}\}$. Expressed using a predicate, the image segmentation is to find the minimal number of convex sets, C_i^{seg}'s, such that

- $\bigcup_{i=1}^{n} C_i^{seg} = I$ with $C_i^{seg} \bigcap C_i^{seg} = \emptyset$ for $i \neq j$.
- $F^{cv}(C_i^{seg}) = true$ if all i's.
- $F^{cv}(C_i^{seg} \bigcup C_j^{seg}) = false$ for $i \neq j$,

where F^{cv} is the predicate for maximal convexity, defined as for $C \in I$,

$$F^{cv}(C) = true \text{ if } C \text{ is a maximal convex set in } I; false \text{ otherwise.}$$

I.5. Shape Inference and Classification

Shape inference is the statistical inference problem that estimates missing parts of a geometric shape given the partial observation of the shape. In morphology analysis, shape inference is a necessary step to find the actual morphologies of nanoparticles using the image segmentation result C_i^{seg}. As discussed in the previous section, the image segmentation result C_i^{seg} does not represent the morphology of a nanoparticle due to the occlusion caused by particle overlaps. Solving the shape inference problem is necessary to recover the complete morphology of nanoparticles.

Since the complete morphology of a nanoparticle is convex, the objective of the shape inference in the morphology analysis problem is to find a convex superset of C_i^{seg}. So far, the shape inference problem is still ill-posed, because there could be an infinite number of convex sets which contain the subset C_i^{seg}. In order to make the problem well-posed, it is necessary to add some constraints to the shape of the convex superset of C_i^{seg}, so that the inference problem provides a unique solution. The shape constraints come in many different forms: some methods restrict the smoothness of the final shape (Kass *et al.*, 1988; Chan and Vese, 2001), while other methods restrict the final shape to one of the pre-determined reference shapes (Chen *et al.*, 2002; Foulonneau *et al.*, 2009). The latter case raises the need for solving a shape classification problem in order to choose one reference shape to use for restricting the solution (shape).

Morphology analysis concerns a finite number of geometric shapes of nanoparticles. The number and categories of these shapes can also be pre-determined according to the engineering understanding of the nanoparticles in question. For example, commonly observed shapes of nanoparticles are spheres, cubes, tetrahedra and rods (Huitink *et al.*, 2010). This dissertation assumes that the geometric shapes of nanoparticles are circles, rectangles, triangles or rods, which are the two-dimensional projection of spheres, cubes, tetrahedra and rods respectively. Under this assumption, both shape inference and classification can be solved either separately (Section III.4) or jointly (Section IV.3).

Suppose that multiple reference shapes $R_1, R_2, \ldots, R_m$ are given. The shape inference is to infer the missing part C_i^{miss} of complete shape $C_i = (C_i^{miss}, C_i^{seg})$: if

C_i is of the kth reference shape R_k,

$$C_i^{miss*} = \arg \max_{\forall C_i^{miss}} \{d(C_i, R_k); C_i \text{ is convex}\},$$

where $d(C_i, C_k)$ is a shape similarity measure. The shape classification determines whether C_i is of the kth reference shape R_k by solving

$$k = \arg \min_{\forall j} d(C_i, R_j).$$

Note that the shape classification for determining k is solved when C_i is given and the shape inference for determining C_i is solved when k is fixed. Therefore, solving the two problems jointly (see Section IV.3) is ideal. However, the sequential solution approach, i.e. first solving the classification problem with a rough estimate of C_i and then inferring C_i, is also possible when the particle overlaps are not significant and the rough estimate of C_i is reliable (Section III.4).

I.6. Main Contribution and Organization of the Dissertation

The main contribution of this dissertation is to formalize the morphology analysis problem as three components of image segmentation, shape inference and classification, and to provide two approaches to solving the problem. Each approach has different capabilities in terms of time efficiency and separation of nanoparticle overlaps. According to extensive empirical studies of both approaches using real electron micrographs, the approaches provide automated morphology analysis of nanoparticles deemed as reliable and satisfactory by the dissertation's material science collaborators.

This dissertation is organized as follows. Chapter II reviews the state-of-the-art research in the morphology analysis of nanoparticles. The chapter also reviews a large amount of literature about cell segmentation problems in bio-imaging because the problem is similar to the morphology analysis problem in terms of technical ingredients. This chapter includes discussions about the limitation of the state-of-the-art approaches to the morphology analysis problem.

Chapter III presents the first approach, divide-and-conquer, to separately solve the three components of morphology analysis. Because it solves each component separately, it is efficient as it takes less than a minute to obtain the final analysis results for all electron micrographs tested. However, its separating capability of particle overlaps is limited. This chapter explains why the method is capable only when the degree of particle overlaps is low enough that the largest part of the connected particle contour not occluded by overlaps remains significant, i.e. as much as 80% or more of its whole contour.

Chapter IV presents the second approach, convex shape analysis, that is able to separate more severe particle overlaps and obtain accurate morphologies. It solves shape inference and classification jointly for better accuracy. As a result, this approach requires more computation time, ten minutes for the biggest-sized electron micrograph. With a little sacrifice of time efficiency, the second approach achieves far superior separation capability than the divide-and-conquer approach presented. Chapter IV explains why the second approach is particularly competent to handle the chain-linked structure of particle overlaps, which are frequently observed in real samples of nanoparticles. In addition, the separation capability of the second approach is justified on the theoretical base of mathematical morphology.

Chapter V summarizes the contribution of this dissertation with its broad impact on nanoscience and engineering. The chapter also discusses the future research direction to extend this dissertation study.

CHAPTER II

STATE OF THE ART

This chapter reviews existing technology and literature related to morphology analysis of nanoparticles. The first section compares capabilities and limitations of instruments available for measuring size and shape of nanoparticles, and discusses why the image analysis of electron micrographs is popularly used. The second and third section review the literature about the image analysis of electron micrographs for nanoparticle analysis and for bio-cell analysis respectively. The last section introduces some general image processing algorithms or tools applicable to image analysis of electron micrographs.

II.1. Particle Characterization Technique

Geometric size and shape information of nanoparticles are obtained by several different microscopic technologies. For summaries, please refer to Table 1.

Light or X-ray scattering is frequently used to obtain the average size of an ensemble of nanoparticles. Dynamic light scattering (DLS), also known as photon correlation spectroscopy, is a technique for sizing nanoparticles in liquid phase. DLS measures the time-dependent fluctuation in the intensity of light scattered by nanoparticles, converts the fluctuation into particle movement speed, and finally applies the Stokes-Einstein equation to convert the particle movement speed into the mean particle size (Pecora, 1985; Nobbmann and Morfesis, 2009). Nanoparticle tracking analysis (NTA) developed by NightSight Ltd. is a variant of DLS. It uses the same physical principle as DLS, but captures the particle movement speed directly by tracking particle movement with laser illuminated microscope. By doing so, NTA performs better than DLS in the analysis of complex and polydispersed samples (Filipe *et al.*, 2010).

In the meanwhile, DLS has wider size range of detectable particles; 1-1000 nm for DLS and 30-1000 nm for NTA (Filipe *et al.*, 2010).

X-ray diffraction (XRD) is a particle sizing technique based on a physical phenomenon that the broadening of the X-ray diffraction lines occurs when particle size is smaller than 100 nm (Jones, 1938). The broadened amount of the diffraction lines is linked to the mean particle size by the Sherrer formula. The formula is exact when the shape of nanoparticles is spherical (Patterson, 1939).

Ultrasonic signal attenuation or gas absorption can also be used for sizing nanoparticles. Acoustic attenuation spectrometry (AAS) is used for measuring the average size of particles (size range: 10 nm - 1000 μm) in suspension or emulsion; it senses the spectra of ultrasonic waves attenuated by particles, and it correlates the amount of attenuation to particle size (Alba *et al.*, 1999). In particular, the spectroscopy works better in a suspension or emulsion of highly concentrated particles than DLS, so it does not need sample dilution process which is required for DLS.

Brunauer-Emmett-Teller (BET) method can be an option for obtaining the average particle size. It is based on the gas absorption phenomenon occurred at the surface of particles. The method measures the surface area of nanoparticles by observing gas absorption, and it estimates the mean particle size from the surface area after making an assumption regarding the shape of particles. BET method is especially useful for sizing particles when the size distribution is broad (Hosokawa, 2007), where DLS and XRD do not produce very accurate measurements.

The particle sizing instruments explained above produce one value to represent a sample of nanoparticles. In other words, they provide the average size of an ensemble of nanoparticles rather than individual sizes or the size distribution of individual nanoparticles. In addition, they suppress the size of a three-dimensional particle to a one-dimensional value, called ‘equivalent size’. The definitions of the equivalent size are different according to the physical principles used in the instruments, e.g. for DLS and NTA, it is the Stoke radius. These different definitions usually produce different size measurements unless the particle's morphology is uniformly spherical. Hence, their use for analyzing a particle size distribution is limited.

When measuring sizes or shapes of individual nanoparticles is required, various types of microscopes followed by image analysis can be used. A popular choice is the use of electron microscopes such as transmission electron microscope (TEM) and scanning electron microscope (SEM). Although they capture only two-dimensional projection images of three-dimensional nanoparticles, they are popularly used in the nanoparticle research because of their ultra-high resolution; the highest resolution is 0.05 nm for TEM or 1 nm for SEM (Allard and O'Keefe, 2004; Boyes, 2005). Applying tomography technology to conventional TEM or SEM images enables revealing three-dimensional structures of nanopartices (Koster *et al.*, 2000), but the technology is not yet mature nor broadly available among most nanoparticle research laboratories.

Scanning probe microscopy (SPM) is another powerful technique for characterizing the three-dimensional morphology of nanoparticles. This type of microscope uses a very small probing tip that scans the surface of a specimen. Popular instruments of SPM includes atomic force microscope (AFM) and scanning tunneling microscope (STM). A main advantage of SPM is that it can produce three-dimensional morphology. In addition, it is capable of imaging nanoparticles in liquid phase, which cannot be observed by electron microscopes. However, SPM generates a convolution image of a probing tip with a specimen, so its lateral dimension apts to be overestimated (Villarrubia, 1997).

In spite of many advantages, the application of SPM to the morphology analysis is limited. SPM has rounding effect on sharp edges and corners, so small triangular particles are not very differentiable from spherical particles of equivalent size. In addition, if nanoparticles are very closely located or form complicated aggregates, SPM cannot produce true lateral dimension. For more detailed comparison between SPM and electron microscopes, please refer to Vujtek *et al.* (2010).

Therefore, in nanoparticle research, the image analysis of electron micrographs is regarded as a reasonable option. This is evident by the sheer amount of research papers studying nanoparticles in which the morphology information of the particles comes from transmission electron microscopes (Ahmadi *et al.*, 1996; Puntes *et al.*, 2001; Chithrani *et al.*, 2006).

II.2. Morphology Measurement Methods in Nano-Imaging

Once two-dimensional projection images of nanoparticles are generated, the next step is to the measure sizes and shapes of individual nanoparticles from the images. In most nanoparticle research, the image analysis step is performed manually. The manual analysis is very time-consuming and it is also susceptible to subjectivity of human analyzers. Thus, an automated method for the quantitative image analysis is necessary for quick and accurate morphology analysis.

A popular tool used for morphology analysis is `ImageJ`. The tool is a branched version of NIH Image, which has been developed by the National Institute Health (NIH) for bio-medical imaging, and it is widely used in the analysis of TEM or SEM nanoparticle images (Eustis *et al.*, 2005; Siqueira *et al.*, 2008; Hindson *et al.*, 2011). Since `ImageJ` is a general image analysis tool, the ways of using the tool in particle analysis may vary for different types of images. Mostly used is a sequence of the following operations; first separate nanoparticles from the background by global thresholding, then apply watershed segmentation to separate overlapping nanoparticles, and finally execute a function called ‘particle analyzer.’ More details about

Table 1. Particle characterization techniques. XRD: X-ray diffraction, DLS: dynamic light scattering, NTA: nanoparticle tracking analysis, AAS: acoustic attenuation spectroscopy, BET: Brunauer-Emmett-Teller, TEM: transmission electron microscopy, SEM: scanning electron microscopy, AFM: atomic force microscopy, STM: scanning tunneling microscope.

	Sample Type	**Information**	**Size Range**
XRD	Powder	Average Size	1 - 100 nm
DLS	Emulsion or Suspension (Diluted)	Average Size	1 - 1000 nm
NTA	Emulsion or Suspension	Average Size	30 - 1000 nm
AAS	Emulsion or Suspension	Average Size	10 nm - 1000 μm
BET	Powder	Average Size	Not available
TEM	Powder, Dried ‘electron-transparent’-thin	Size and Shape (2d projection)	0.05 nm and larger
SEM	Powder, Dried only (semi)-conductive	Size and Shape (2d projection)	1 nm and larger
AFM	Power, Emulsion, or Suspension	Size and Shape	Lateral: 5 nm and larger , Vertical: 0.1 nm and larger
STM	Power, Emulsion, or Suspension	Size and Shape	Lateral: 0.1 nm and larger , Vertical: 0.01 nm and larger

these steps are described at Woehrle *et al.* (2006).

The `ImageJ` particle analyzer works well if nanoparticles are well separated without overlaps, but its analysis capability is limited otherwise. Therefore, after applying the `ImageJ` particle analyzer, human analyzers need to do significant amount of manual work for refining the result from `ImageJ`. Detailed discussion about the accuracy of `ImageJ` will be presented in Section III.5.

More sophisticated image analysis methods to supersede `ImageJ` are necessary, but there are only a limited amount of literature about the automated image analysis. Fisker *et al.* (2000) applied a deformable template model, also known as active contour, to estimate particle size distribution, where the shape of nanoparticles is assumed to be an ellipsis in the algorithm of separating overlapping nanoparticles. Gontard *et al.* (2006) proposed a local thresholding of electron micrographs for identifying particles with spatially varying intensities. Chen and Ho (2008) used sequentially the Laplacian edge detection algorithm and a radius estimation algorithm to estimate the size distribution of spherical nanoparticles, which do not overlap. Glotov (2008) analyzed very complicated chain-structured nanoparticle aggregates by applying the proposed circular decomposition algorithm to the outline of aggregates, but the results of the proposed method still need to be refined by human operators.

The electron micrographs of nanoparticles we come across usually have a large number of particles, with various types of shapes and different degree of overlaps. The methods reviewed above fall short of handling the electron micrographs of these generic characteristics. The capabilities and limitations of the methods are summarized in the first row of Table 2.

Table 2. Summary on the related research. Image analysis methods potentially usable for the morphology analysis of nanoparticles.

Method	**What-to-do**	**Limitation**
Nano-imaging methods		
Circle or ellipse template (Fisker *et al.*, 2000; McFarland and Van Duyne, 2003; Glotov, 2008; Chen and Ho, 2008)	• analyze chain branched aggregates composed of spherical or elliptical objects	• only for spherical or elliptical objects
Cell segmentation methods		
Morphological segmentation (Malpica *et al.*, 1997; Tek *et al.*, 2005; Cheng and Rajapakse, 2009; Umesh Adiga and Chaudhuri, 2001; Lin *et al.*, 2005)	• use morphological operators • work for convex shapes	• no statistical reasoning on occlusion • oversegmentation
Graph cut (Shi and Malik, 2000; Felzenszwalb and Huttenlocher, 2004; Daněk *et al.*, 2009)	• regard a digital image as a graph • partition the graph by intensity similarities	• no statistical reasoning on occlusion • undersegmentation for overlapping objects with similar intensities • time consuming
Active contour (Bamford and Lovell, 1998; Nath *et al.*, 2006; Chen *et al.*, 2002; Foulonneau *et al.*, 2009; Hodneland *et al.*, 2009)	• find regions with uniform intensities • no preprocessing necessary • apply prior knowledge on shapes	• cannot handle overlaps without prior shape information • hard to learn many contour functions with prior shape information
Convergent index filtering (Marcuzzo *et al.*, 2009; Quelhas *et al.*, 2010)	• use image gradient and its direction • work for identifying convex objects • segment overlapping objects efficiently	• does not work well when the size distribution of objects is wide

II.3. Cell Segmentation Methods in Bio-medical Imaging

There exists plenty of research addressing the bio-cell segmentation problem. Many of them are about a single-cell segmentation. What is reviewed here is the literature about multiple cell segmentation, which involves the separation of individual cells from other cells; this line of research is directly related to our objective.

A classical approach for the cell segmentation problem is to use the morphological image segmentation methods, represented by the watershed method and its variants (Malpica *et al.*, 1997; Tek *et al.*, 2005; Cheng and Rajapakse, 2009). The approach first finds markers pointing to the approximate locations of cells and then segments an image region into several influence zones of markers. An alternative approach is to segment the image region using the watershed transform (Umesh Adiga and Chaudhuri, 2001; Lin *et al.*, 2005). Doing so usually oversegments a region, so the regions should be merged into meaningful groups by some ad-hoc rules. Watershed segmentation generally works well, but this line of methods does not provide any inference on the missing parts, which makes it difficult to fulfill the final objective of morphology analysis.

Graph-cut methods have also been applied to cell segmentation. The method constructs a graph by treating each image pixel as a node. Each pair of nodes is connected by an edge with the similarity between pixel intensities as its cost. It finds a normalized minimum cut of the graph, which naturally segments images (Shi and Malik, 2000; Felzenszwalb and Huttenlocher, 2004). This approach does not separate overlapping objects well, especially when the overlapping objects have similar intensity levels. Hence, the estimated mean radius of cells was utilized for the purpose of separating the overlapping cells (Daněk *et al.*, 2009), but their method is only applicable to spherically shaped objects.

Active contour is another school of methods applied to the cell segmentation (Bamford and Lovell, 1998; Nath *et al.*, 2006). Active contour is originally designed to segment a single object of complicated shape from the background (Kass *et al.*, 1988), but the level-set based active contour (Chan and Vese, 2001) or the multiphase active contour (Vese and Chan, 2002) could be used to segment multiple objects. However, if any shape constraints is not put on the contours, active contour methods work only for segmenting non-overlapping objects or those overlapped but of different colors. There have been several attempts to put shape constraints for handling overlaps and occlusions of objects (Chen *et al.*, 2002; Foulonneau *et al.*, 2009; Lecumberry *et al.*, 2010). The limitation of these methods is their computation inefficiency. In order to separate the individual objects from the overlapping agglomerates, one needs to put a shape constraint on every single object. But, doing so will significantly increase the number of level-set functions, to be the same as the number of objects in a micrograph, because each level-set function can only be associated with one shape constraint. It is computationally difficult to coordinate hundreds of level-set functions so that they do not locate in the exactly same region. In fact, coordinating level-set functions is so far practically feasible only for a small number (e.g. four) of level-set functions (Hodneland *et al.*, 2009; Zhang and Matuszewski, 2009).

Recently, convergent index filtering and its advanced version, the sliding band filter (SBF), have been used for a noise-robust cell segmentation (Marcuzzo *et al.*, 2009; Quelhas *et al.*, 2010). A major difficulty to apply SBF is that the range of object sizes should be known *a priori*. The size range can be roughly estimated by morphological granulometric analysis (Matheron, 1975), but if the range is wide or several objects are located in a specified size range, SBF will fail to separate different objects or to oversegment one big object.

All of the aforementioned methods solve problems similar to ours, but their effectiveness in morphology analysis is limited by the way they handle overlapping objects or their computation inefficiency. Overlapping nanoparticles have similar image intensities, which makes the graph-cut methods a less ideal candidate. Nanoparticles often manifest themselves in different sizes, and consequently, hundreds of particles in a micrograph could present a rather broad range of object size, rendering the SBF approach ineffective. The large number of particles in a micrograph that need to be separated also presents a computational challenge for the active contour methods to be practically applied. Simple segmentation methods such as the watershed method will not give us the desired morphology information of particles because the simple methods do not provide inference on the missing parts caused by particle overlaps. The capabilities and limitations of the methods are summarized in Table 2.

II.4. General Image Processing Methods

This section reviews general image processing algorithms, which can be applied to morphology analysis of nanoparticles.

II.4.1. Image Segmentation Methods

General image segmentation methods can largely be categorized into three classes: (1) *feature thresholding*, (2) *edge detection* and (3) *region-based segmentation. Feature thresholding* is the simplest approach to image segmentation. One or more threshold values are determined, and a feature value at each image pixel is compared with the threshold values in order to determine which subregion X_i the image pixel belongs to. The histograms of colors or gray intensities in an image are sometimes used for obtaining the threshold values (Gonzalez and Woods, 2002). Using clustering techniques with multiple types of image features is a multi-dimensional extension of the feature thresholding method (Fu and Mui, 1981). Graph cut methods (Shi and Malik, 2000; Felzenszwalb and Huttenlocher, 2004; Daněk *et al.*, 2009) can be regarded as a clustering approach. It models an image as a graph of image pixels with the similarity of image features as the costs of links, and finds minimal (cost) cuts of the graph.

Edge detection identifies discontinuities in image features, called edges, and then uses the edges as boundaries of subregion X_i. There exist many edge detection algorithms such as Sobel operator, Prewitt operator, Roberts cross operator and Canny's algorithm (Al-amri *et al.*, 2010). There are several issues in this type of image segmentation. First, the edges detected are too fragmented to represent a complete contour of one sub-region. Therefore, the use of edge-linking algorithms may be needed to connect the fragmented edges to a complete contour, e.g. Hough transform and local edge processing (Nagabhushana, 2006). Second, when objects overlap, forming a mixture of multiple objects, the edges detected also become a mixture of multiple contours corresponding to the objects. Separating the mixture into individual contours is not a trivial problem. Section III.1 presents an algorithm that separates the mixture of contours.

Region-based segmentation uses one of the following approaches: (1) splits an image into subregions having high degree of homogeneity by using a combination of simple partitioning operations, i.e. recursive quad-tree partitioning (Lee, 1986; Bhanu and Parvin, 1987); (2) merges neighboring pixels or grows smaller regions into larger regions satisfying some homogeneity criteria (Burt *et al.*, 1981; Browning and Tanimoto, 1982; Beaulieu and Goldberg, 1989; Hojjatoleslami and Kittler, 2002); or (3) uses a combination of partition-and-merges (Pietikainen *et al.*, 1982; Cheevasuvit *et al.*, 1986). Morphological image segmentation methods such as watershed transformation (Serra, 1983) and waterfall algorithm (Beucher, 1994) fall in the category of the region-based segmentation. The segmentation method presented in Section IV.2 is a morphological image segmentation method specialized for segmenting a mixture of overlapping convex sets.

II.4.2. Shape Classification Methods

A common underlying procedure in the shape classification is to extract the geometric information of subregion X_i in a form of shape representation, then to compare the shape representation with the representations of pre-defined reference shapes with respect to a similarity measure, and finally to choose the most similar reference shape. There exist various approaches to shape classification depending on different forms of *shape representations* and different definitions of shape *similarity measures*.

In statistical shape analysis (Dryden and Mardia, 1993), a shape is represented by a finite number of landmark points located on the contour of the object and its similarity is measured by the Procrustes distance. One big issue is that obtaining the landmark points is not straightforward.

There are approaches using the Hausdorff distance of various shape representations: Fourier descriptor (Kauppinen *et al.*, 2002), chain code (Saghri and Freeman, 1981), medial axis transformation (Philbrick, 1968), curvature scale space (Zhang and Lu, 2003; Wang and Teoh, 2007), and invariant moments (Zhao and Chen, 1997). For a comprehensive review of shape representations and classifications, please refer to Loncaric (1998); Zhang and Lu (2004). None of these methods except Fourier descriptor and invariant moments can be used when some parts of a shape is not available due to occlusion or other reasons.

Active contour methods discussed in the previous section represent the contour of geometric shape as a parametric curve or a level set of smooth functions. The shape similarity is measured by computing the distance between parametric curves or comparing the moments of the interior regions of two contours. The methods naturally solve shape classification together with shape inference, but they are more appropriate to handle a small number of complicated shapes.

CHAPTER III

A DIVIDE-AND-CONQUER APPROACH FOR ANALYZING WEAKLY OVERLAPPED NANOPARTICLES

The image processing softwares currently used for morphology analysis such as `ImageJ` are ineffective because of particle overlaps, as discussed in Section II.2. To work around the issue of particle overlaps, three problems of image segmentation, shape inference and classification should be considered together. This chapter presents a divide-and-conquer approach to solving these three problems in a step-by-step fashion. To the best of our knowledge, this approach is one of the first attempts to provide an automated procedure for morphology analysis of nanoparticles. This chapter is based on Park *et al.* (2011).

This chapter is organized as follows. Section III.1 describes how image segmentation is solved by decomposing mixtures of multiple contours. Section III.2 explains an invariant curve space as a way of shape representation, and Section III.3 defines a set of reference shapes. The definition of reference shapes is necessary for performing shape classification and inference. Section III.4 explains how shape inference and classification can be performed on the invariant curve space. Finally, Section III.5 presents the empirical evidence, showing that the new approach is more accurate than `ImageJ`.

III.1. Decomposition of Composite Contours

Image segmentation is an algorithm to partition an image into subregions. In morphology analysis, a subregion should be the region of one nanoparticle. As discussed in Section I.4, the imaging process of TEM makes it easy to separate the region I occupied by nanoparticles from the background. However, it is not straightforward to

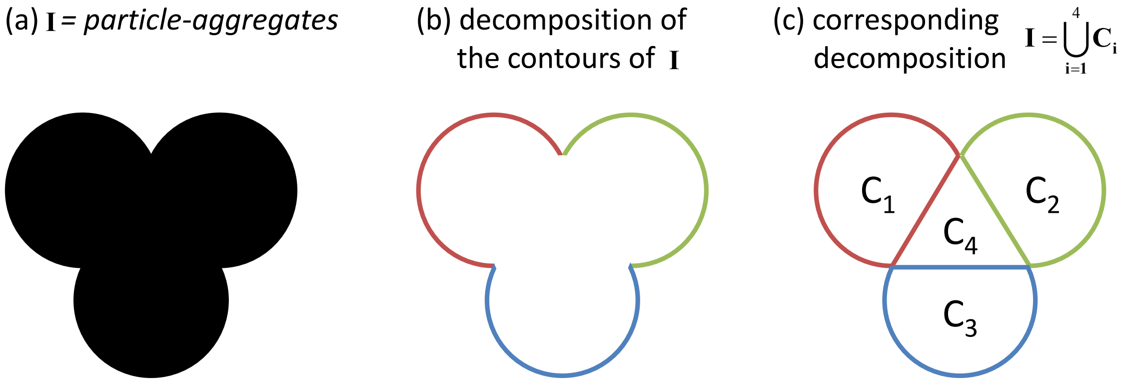


Fig. 1. Decomposition of a composite contour.

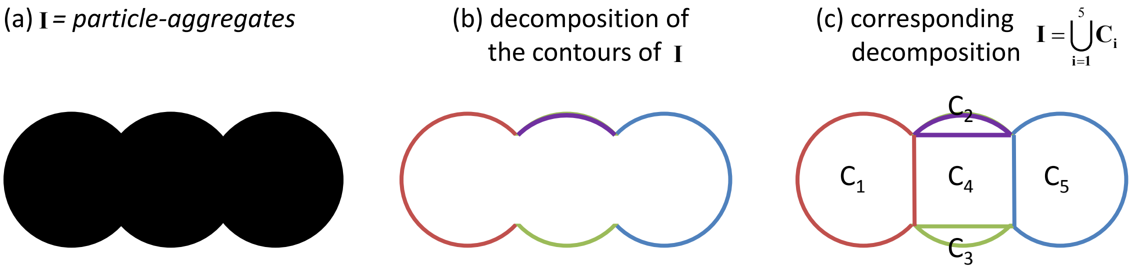


Fig. 2. A failing case of the contour decomposition method.

separate I into subregions so that one subregion corresponds to one particle, because of the overlaps among nanoparticles. This section presents a method to partition I by decomposing the contour of particle-aggregates (the boundary of I) into the contours of individual particles.

Figure 1 shows the key idea of applying the contour decomposition of I for partitioning I. The contour of I is the contour of the union of convex sets, because nanoparticle are geometrically convex. The contour decomposition of I is that the contour of I, ∂I, is partitioned into the minimal number of subpieces $\{\partial J_1, \ldots, \partial J_m\}$ such that

- $\partial I = \bigcup_{i=1}^{m} \partial J_i$ and

- conv(∂J_i) is a subset of I for $i = 1, \ldots, m$,

where conv(J) is the convex hull of J, i.e. the small convex set containing J. Since conv(∂J_i) is always a subset of one convex set composing I, the following property holds.

Proposition 1. *(Contour-to-convex-set decomposition equivalence) The decomposition of the contour ∂I is equivalent to the minimal convex decomposition of I into $\{C_1, \ldots, C_n\}$ if for every C_k there exists only one $i \in \{1, \ldots, m\}$ such that conv(∂J_i) $\subset C_k$.*

The 'if'-condition does not always hold. An example that does not satisfy the 'if'-condition is in Figure 2. In this example, three nanoparticles overlaps in a chain-link cluster. When the contour decomposition method is applied, the result would look like part (c) in the figure, namely that the contour of the middle particle is split into two parts by the other two particles. In this section, only the cases satisfying the 'if'-condition are considered; for the chain-link structure of particle overlap, it will be considered in Chapter IV.

The contour-decomposition process uses the test of whether conv(∂J_i) is in I. To be more specific, the contour-decomposition process breaks a composite contour at the points on the contour where conv(∂J_i) is not in I. The breaking points mathematically correspond to points where the internal angle between the left tangent vector and the right tangent vector is larger than π; please see Figure 3. Such points are called reflexes in computational geometry (Keil and Snoeyink, 2002).

Finding reflexes in a contour of a non-convex shape is as easy as computing the left and right tangent vectors at every point on the contour. However, if a contour is corrupted with image noises, it is not very easy to differentiate real reflexes from the reflexes falsified by noises. This is because the tangent vector is purely a local

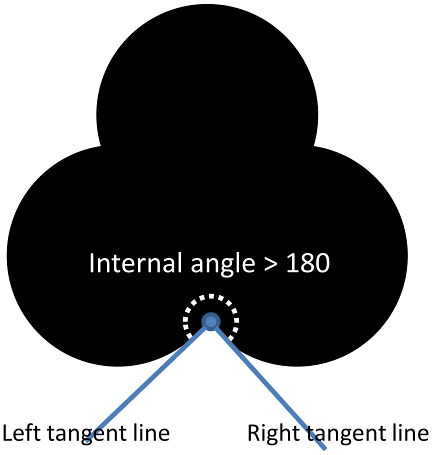


Fig. 3. Reflex in a composite contour.

geometric property, which is vulnerable to noise corruption. What is proposed here is a heuristic algorithm robust to noise.

Let $I \subset \mathbb{R}^2$ be non-convex. The heuristic algorithm to find reflexes of I uses the distance function d_I defined on the boundary ∂I that measures the minimum distance from a point $x \in \partial I$ to the boundary of conv(I), denoted by ∂conv(I). For $x \in \partial I$,

$$d_I(x) = \min\{||x - y|| : y \in \partial \text{conv}(I)\}.$$

The heuristic algorithm is based on the heuristic finding that the distance function defined above is locally maximized at reflexes. This understanding suggests that it is only necessary to take the local maxima of the distance function in order to identify the reflexes.

In the implementation of the heuristic algorithm, the `Qhull` algorithm is used to find a convex hull (Barber *et al.*, 1996). Convex hulls of a few examples of composite boundaries are illustrated in Figure 4. To locate the local maxima of the distance function to the convex hull, wavelet smoothing is first applied to the distance function in order to remove noises and the local maxima of the smoothed distance function is

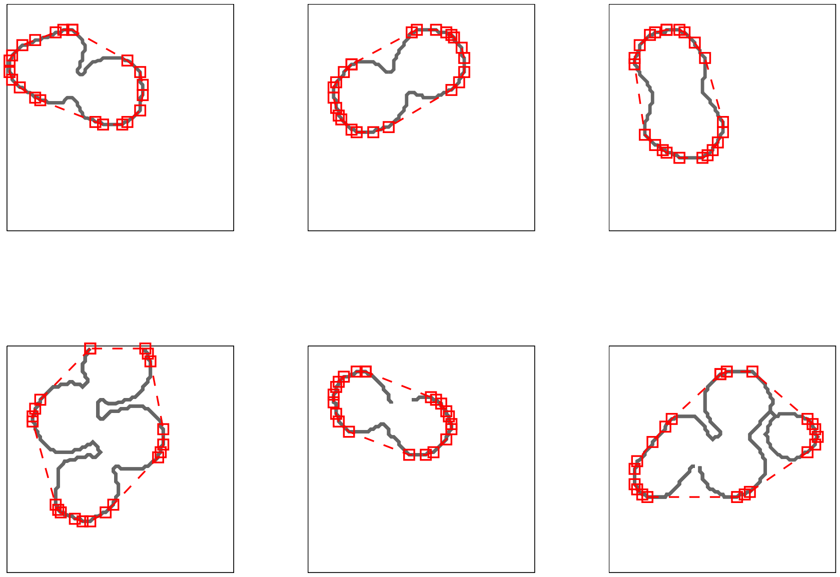


Fig. 4. Convex hulls of composite edges. The dotted lines with rectangular markers are convex hulls and the rectangular markers are the extreme points of the convex hulls.

identified by finding downward zero-crossing of the first derivative of the smoothed distance function (Yang *et al.*, 2009). Doing so helps filter out the reflexes caused by noises. Some results of applying the contour-decomposition process are illustrated at the fourth column of Figure 5.

III.2. Invariant Curve Spaces for Shape Feature Extraction

To characterize the size and the shape distributions of nanoparticles in a TEM image, one should have the complete contours of nanoparticles. However, as seen on the fourth column of Figure 5, the contour-decomposition process gives only the partial information of the contours, so the complete contours should be inferred using the partial information. As discussed in Section I.5, for shape inference, defining the reference shapes $R_1, \ldots, R_k$ as well as a shape-similarity measure is necessary. Subsection

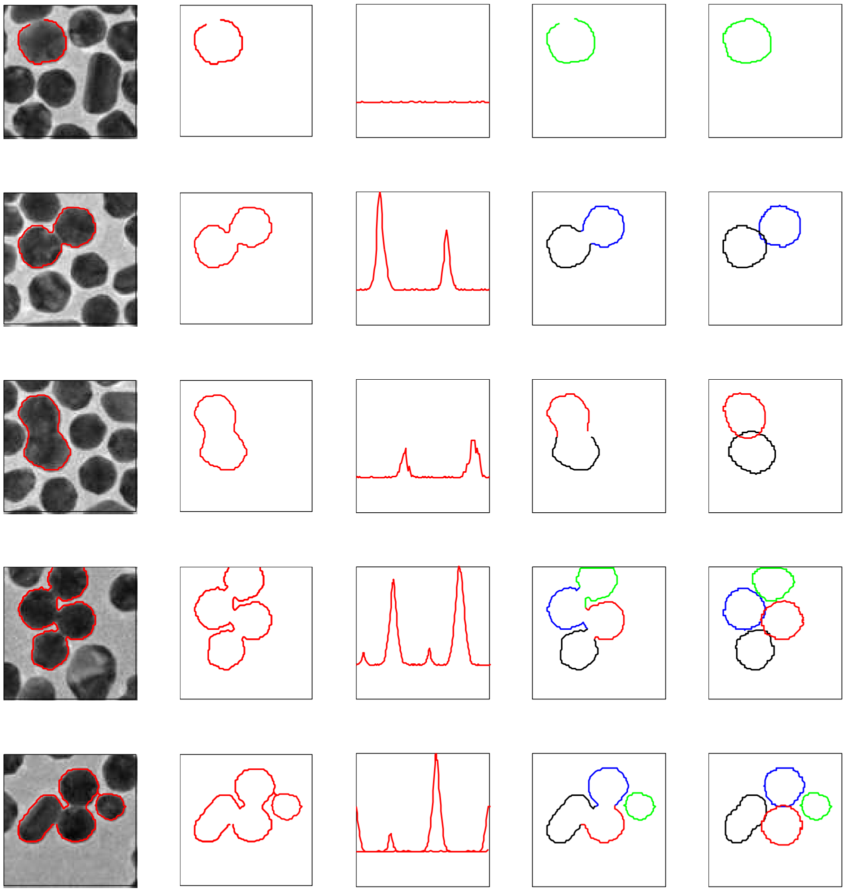


Fig. 5. Illustration of decomposition of composite contours.

III.2.1 presents the representation of a shape by a parametric curve f and Subsection III.2.2 defines the similarity measure d_f between two shapes, which is to measure the similarity between the two corresponding parametric curves. Later, based on the curve space with the pair (f, d_f), Subsection III.3 discusses how to define reference shapes using the sample shapes from real electron micrographs.

III.2.1. Location- and Scale-invariant Curve Space

Following Dryden and Mardia (1998), a geometric shape is defined as all the geometrical information that remains when location, scale and rotational effects are filtered out from an object. If an object is (topologically) simply connected, its geometric information is summarized by its contour.

In the two dimensional space, a contour is naturally represented by a parametric curve of a one-dimensional parameter t, $f : [0,1] \rightarrow \mathbb{R}^2$, which is a map from the parameter $t \in [0,1]$ to the coordinate $(x(t), y(t))$ of one point on the contour. Similarly, in the divide-and-conquer approach, a contour is represented by a parametric curve that maps the parameter t to the polar coordinate of $(x(t), y(t))$, denoted by $(\theta(t), r(t))$. The polar coordinate is defined with respect to the center of mass of an object, where $r(t)$ is the distance from the center to point $(x(t), y(t))$ and $\theta(t)$ is the angle between the x-axis and the line from the center to point $(x(t), y(t))$. If t is an angular parameter, i.e. $t = \theta(t)$, the parametric curve f maps

$$t \mapsto (\theta(t), r(t)) \quad \Leftrightarrow \quad \theta(t) \mapsto (\theta(t), r(t)).$$

In other words, $f = \mathcal{I}_\theta \otimes r$, implying that $f(\theta(t)) = (\mathcal{I}_\theta(\theta(t)), r(\theta(t)))$, where $\mathcal{I}_\theta$ is an identity map over $[0, 2\pi]$ and function r maps $\theta(t)$ to $r(t)$. Since $\mathcal{I}_\theta$ is the same for every contour, a contour is uniquely identified by function $r : [0, 2\pi] \rightarrow \mathbb{R}^+$.

In a discrete setting, the function r is represented in the form of $\{r_i\}$, where

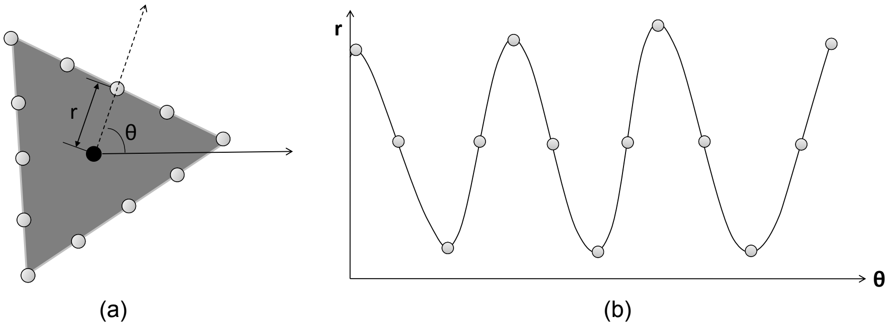


Fig. 6. Representation of nanoparticle boundaries. in the left figure, the black dot is the gravity center of the triangle and the gray points are pixels sampled from the boundary of the triangle, which correspond to the gray points in the right figure.

$r_i = r(\theta_i)$ for uniformly distributed θ_i over $[0, 2\pi]$. This is indeed what will be used in this chapter. That is to say, a parametric curve refers to the set of values $\{r_i\}$.

For shape analysis, a basic requirement for the function r is its invariance under rotation, location, and scaling. The function is already location-invariant because it is defined relative to the center of mass. To meet the invariance under scaling, r is normalized as follows.

$$\overline{r} = \frac{1}{\int_0^{2\pi} r(\theta) d\theta} r.$$

In the discrete setting, the denominator is simply the arithmetic mean of $\{r_i\}$. Note that using the mean distance from the center of mass is a popular way to measure the size of a shape, a.k.a centroid size, for the purpose of scaling (Dryden and Mardia, 1998, pages 23-24). The scaled parametric curve is shown in Figure 6 in the r-θ coordinate system. Since both r and θ are defined relative to the center of mass, they are invariant to translation. This parametric curve is scale invariant because the distance is normalized. However, the parametric curve is not yet rotation invariant; this issue will be addressed in Subsection III.2.2.

The parametric curve characterizes convex shapes such as polygons and circles very well. For example, the curve with three modes shown in Figure 6(b) corresponds to the triangular shape in Figure 6(a). Similarly, the representing curve has four modes for a rectangle and so forth for other polygons. Nanoparticles are geometrically convex as discussed, so using the parametric curve is appropriate for the morphology analysis.

III.2.2. Shape-Similarity Measure on the Curve Space

This section defines a shape-similarity measure on parametric curves. The shape-similarity must be independent of the rotation effect on parametric curves. Hence, the rotationally invariant distance between two parametric curves is first defined, and subsequently, the shape similarity measure is defined using the rotationally invariant distance.

Let f_i and f_j be two parametric curves. Please recall that a parametric curve is a function of θ. The rotation of shape by Δ is equivalent to the circular shift of the parametric curve by Δ along its domain θ. The circular shift of θ by Δ is a function $s_\Delta(\theta)$ that maps

$$\theta \mapsto \theta + \Delta - 2\pi \left\lfloor \frac{\theta + \Delta}{2\pi} \right\rfloor .$$

A rotationally invariant distance between f_i and f_j is defined as

$$d_f(f_i, f_j) = \min_{\Delta \in [0, 2\pi]} ||f_i - f_j \circ s_\Delta||, \tag{1}$$

where $|| \cdot ||$ is the L_2 norm, and $f_j \circ s_\Delta$ is a composition of function f_j and function s_Δ.

Let $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ be the parametric curves in the discrete form, i.e. $\boldsymbol{f}_i = (f_{i1}, f_{i2}, \ldots, f_{ip})^t$, where $f_{ij} = f_i(\theta_j)$ for uniformly distributed θ_j over $[0, 2\pi]$. A rotationally invariant distance between $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ is defined as

$$d_f(\boldsymbol{f}_i, \boldsymbol{f}_j) = \min_{\Delta=1,2,\ldots,p} \|\boldsymbol{f}_i - s_\Delta \circ \boldsymbol{f}_j\|, \tag{2}$$

where $\|\cdot\|$ is the Euclidean distance and $s_\Delta \circ \boldsymbol{v}$ is an operator circularly shifting each element of $\boldsymbol{v}$ downward by Δ elements, which is defined by: for $v = (v_1, \ldots, v_p)^t \in \mathbb{R}^p$,

$$s_\Delta \circ \boldsymbol{v} = (v_{\Delta+1}, \ldots, v_p, v_1, \ldots, v_\Delta)^t.$$

A shape can be defined as an equivalence class of parametric curves under the rotationally invariant distance, i.e.,

$$[\boldsymbol{f}_i] = \{\boldsymbol{f} \in \mathbb{R}^p; d_f(\boldsymbol{f}, \boldsymbol{f}_i) = 0\}.$$

The shape space is the set of $[\boldsymbol{f}_i]$'s. The shape space forms a curved manifold, instead of a Euclidean space.

In the curved manifold, the distance is defined by the geodesic distance rather than the Euclidean distance. For this reason, the shape similarity measure in the curve space should also be based on the geodesic distance in the curve space. The geodesic distance is the length of the shortest arc connecting two points over the curvature in a manifold. For neighboring points, the Euclidean distance, i.e. d_f, provides a good approximation to the geodesic distance and for far apart points, the geodesic distance can be approximated by adding a sequence of local hops between neighboring points. To get approximate geodesic distances, first construct is a graph of data points having edges with the distance d_f as weights only for k-nearest neighbors and then define the geodesic distance as the distance of the shortest path between a pair of two data points. In the sequel, $g(\cdot, \cdot)$ is used to denote the geodesic distance.

For example, please see the left subfigure in Figure 7. Suppose that $\boldsymbol{f}_i$'s are in a p-dimensional subspace embedded in $\mathbb{R}^n$ (also called p-manifold.) The subspace is

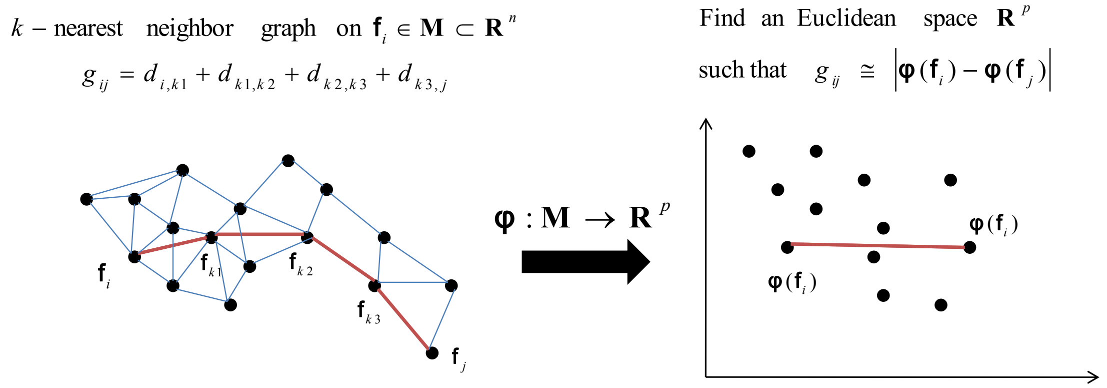


Fig. 7. Basic idea of feature extraction.

roughly approximated by a surface generated from meshes of the k-nearest neighborhood graph, so the distance in the subspace is the distance in the graph, which is geodesic distance $g(\cdot,\cdot)$. In the figure, the geodesic distance between $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ is approximately the summation of the weights of the edges that are on the shortest path from $\boldsymbol{f}_i$ to $\boldsymbol{f}_j$ on the graph.

In summary, the shape space is the set of equivalence classes of parametric curves equipped with the shape distance g as its shape-similarity measure.

III.3. Define Reference Shapes

Suppose that the contours of n sample nanoparticles are extracted from electron micrographs. The corresponding parametric curves are denoted by $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n$, where $\boldsymbol{f}_i = (f_{i1}, f_{i2}, \ldots, f_{ip})^t$ is the p-dimensional vector as in the discrete setting of the parametric curve. Defining m reference shapes from the n parametric curves is done by grouping the parametric curves into m subgroups. A reference shape is in turn defined by a set of parametric curves which belong to one of the subgroups. The set of the curves can be used to compute the mean and the variation of a reference shape, corresponding to that specific subgroup.

Grouping n parametric curves into m groups demands a shape clustering method. There is a technical difficulty that hinders accurate shape clustering. The dimension of the resulting parametric curves from a particle is high. It is well known that clustering analysis methods using similarity measure works poorly in high dimension spaces (Steinbach *et al.*, 2003, pages 12-13). To address this issue, Subsection III.3.1 presents the non-linear dimension reduction method to project the parametric curves to a low-dimensional feature space, and Subsection III.3.2 presents a clustering method on the dimension-reduced feature space for grouping the low-dimensional projection of the parametric curves by shapes.

III.3.1. Dimension Reduction of Parametric Curves

A projection of $\boldsymbol{f}_i$ onto a low dimensional space is defined by an embedding map $\boldsymbol{\phi} : \mathbb{R}^p \to \mathbb{R}^q$ such that $q \ll p$. It is desirable that the embedding map is an isometry, i.e. the Euclidean distance in the low-dimensional space is equivalent to the shape distance g in the shape space. In order to find the embedding map, our strategy is to create an $n \times n$ dissimilarity matrix by applying the shape distance g to every pair of the n parametric curves, and then apply the multidimensional scaling (MDS; Kruskal and Wish, 1978) to obtain $\boldsymbol{\phi}$.

Given the dissimilarities $g_{ij} = g(\boldsymbol{f}_i, \boldsymbol{f}_j)$ produced by the geodesic distance, the low dimensional features $\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_n$ of curves $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n$ should be found such that the Euclidean distances between the features are close to the corresponding geodesic distances between the curves, that is,

$$g_{ij}^2 = (\boldsymbol{\phi}_i - \boldsymbol{\phi}_j)^t (\boldsymbol{\phi}_i - \boldsymbol{\phi}_j). \tag{3}$$

Denote by $\boldsymbol{G}$ the dissimilarity matrix whose (i, j)-entry is g_{ij} and denote the doubly centered geodesic distance matrix

$$\tau(\boldsymbol{G}^2) = -\frac{1}{2}\boldsymbol{H}\boldsymbol{G}^2\boldsymbol{H}^t, \tag{4}$$

where $\boldsymbol{G}^2$ is the matrix whose elements are the squares of the elements of $\boldsymbol{G}$, and $\boldsymbol{H}$ is the $n \times n$ centering matrix with $(\boldsymbol{H})_{ij} = \delta_{ij} - \frac{1}{n}$. If the kernel matrix $\tau(\boldsymbol{G}^2)$ is positive semi-definite, the classical MDS gives an explicit solution to the embedding problem. Let $\tau(\boldsymbol{G}^2) = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^t$ be the eigen-decomposition of $\tau(\boldsymbol{G}^2)$, then the collection of q-dimensional features $\boldsymbol{X} = [\boldsymbol{\phi}_1, .., \boldsymbol{\phi}_n]^t$ is given by $\boldsymbol{X} = \boldsymbol{V}[\ ,1:q]\,\boldsymbol{\Lambda}[1:q, 1:q]^{1/2}$, where $\boldsymbol{V}[\ ,1:q]$ is the first q columns of $\boldsymbol{V}$ and $\boldsymbol{\Lambda}[1:q, 1:q]$ is the $q \times q$ upper-left corner of $\boldsymbol{\Lambda}$. Unfortunately, because of the use of geodesic distances in defining $\boldsymbol{G}$, the matrix $\tau(\boldsymbol{G}^2)$ is not guaranteed to be positive semi-definite. As a remedy, the author uses a constant-shifting method that is well studied in metric MDS and replace $\boldsymbol{G}$ in (4) by the matrix $\widetilde{\boldsymbol{G}}$ with entries $\tilde{g}_{ij} = g_{ij} + c(1 - \delta_{ij})$, where δ_{ij} is the Kronecker delta, and c is the largest eigenvalue of the matrix

$$\begin{bmatrix} 0 & 2\tau(\boldsymbol{G}^2) \\ -\boldsymbol{I} & -4\tau(\boldsymbol{G}) \end{bmatrix}. \tag{5}$$

According to Cailliez (1983), the matrix $\tau(\widetilde{\boldsymbol{G}}^2)$ is positive semi-definite. Therefore, the classical MDS procedure described above can be applied to the modified kernel matrix $\tau(\widetilde{\boldsymbol{G}}^2)$ to obtain the desired nonlinear embeddings. All necessary steps for the nonlinear embedding is summarized in Algorithm 1.

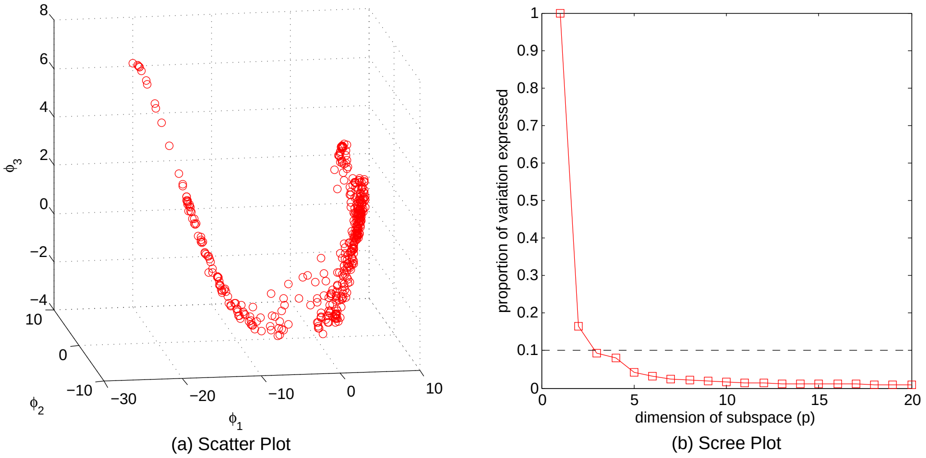
In the nonlinear embedding method, there are two tuning parameters whose choices are critical for the results: the number of nearest neighbors, k, and the reduced dimension, q. To compute the geodesic distances, it is necessary to decide on the number of nearest neighbors, k. If k is too large, it would cause the creation of "short circuit" edges that shortcut the true geometry of a manifold reflecting the non-linear

Algorithm 1. Feature Extraction by Isomap.

```
1. Construct a $k$-nearest neighbor graph on $\{\boldsymbol{f}_i; i = 1, \ldots, n\}$.
2. Compute $\boldsymbol{G}$ with $(\boldsymbol{G})_{ij} = g(\boldsymbol{f}_i, \boldsymbol{f}_j)$
3. Compute $\tau(\boldsymbol{G})$ using equation (4) and compute $c$ by taking the largest
     eigenvalue of matrix (5).
4. Compute $\tau(\widetilde{\boldsymbol{G}}^2)$ using equation (4) and substituting in $\widetilde{\boldsymbol{G}}^2 = \boldsymbol{G} + c(\mathbf{1}_n - \boldsymbol{I}_n)$,
     where $\mathbf{1}_n$ is an $n \times n$ matrix of ones and $\boldsymbol{I}_n$ is an $n \times n$ identity matrix.
5. Perform eigen-decomposition: $\tau(\widetilde{\boldsymbol{G}}^2) = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^t$.
6. Obtain $\boldsymbol{X} = \boldsymbol{V}[\ ,1:q]\,\boldsymbol{\Lambda}[1:q, 1:q]^{1/2}$ for a specific dimension $q$.
```

structure of data; if k is too small, it will cause the manifold to fragment into a large number of disconnected clusters. Following Samko *et al.* (2006), k is chosen by maximizing $|\rho(\boldsymbol{D}, \Phi_{k,q})|$, where $\boldsymbol{D}$ and $\Phi_{k,q}$ are the matrices of the Euclidean distances between a pair of points in the original space and the feature space, respectively, and $\rho(\cdot, \cdot)$ is the linear correlation coefficient. Note that $\Phi_{k,q}$ depends on q, the dimension of the space of the embeddings. Samko *et al.* (2006) argued that the data set has its intrinsic dimension, and subsequently, they showed empirically that q does not change even if k changes. Hence, q is first estimated for an arbitrary (but reasonable) choice of k, and then, the optimal k is chosen with this q.

In the application to electron micrographs, the 420 (i.e., $n = 420$) parameterized curves were extracted, where each curve was represented by a 315-dimensional vector (i.e., $p = 315$). Subsequently, the shape similarity measure d_f was computed for every pair of the n parametric curves and constructed the graph structure retaining edges among the k-nearest neighbors with $k = 12$. Finally, $\boldsymbol{f}$'s are projected to a rotationally invariant subspace of dimension three (i.e., $q = 3$) to obtain the features



(a) Scatter Plot

(b) Scree Plot

Fig. 8. Rotational invariance feature extraction. The left panel is the scatter plot showing the projection of the 420 curves in the three-dimensional feature space. The right panel shows a scree plot of the kernel matrix.

$\boldsymbol{\phi}(\boldsymbol{f})$. The number of neighbors was chosen by maximizing the correlation criterion given in the previous paragraph. The dimension of the low-dimensional subspace was chosen by using the scree plot of the kernel matrix $\tau(\widetilde{\boldsymbol{G}}^2)$; see Figure 8(b). Figure 8(a) shows the scatter plot of the 420 curves distributed in the feature space.

III.3.2. Semi-supervised Shape Clustering

The low-dimensional features obtained by the nonlinear embedding can be used as inputs to clustering algorithms to group the parametric curves into m groups. However, our experience shows that generic clustering methods do not work well in our context. Such methods tend to overly partition a dataset into far more number of groups than what is needed in the nanomaterial research. A semi-supervised learning approach is adopted; domain experts are asked to determine the number of reference shapes and manually pick a small number of particles from each reference shape; then these labeled cases are used as training data. It is known that semi-supervised learning is able to significantly increase the accuracy of clustering by using a small number of labeled data, together with a large number of unlabeled data (Zhu, 2005).

There are various semi-supervised learning methods in the literature such as using a generative model with the EM algorithm, self-training, information regularization, and graph-based semi-supervised learning; see Zhu (2005) for a comprehensive review. The graph-based approach proposed by Zhu *et al.* (2003) is employed for the morphology analysis. The basic strategy is as follows: Labeled and unlabeled data are represented as vertices in a connected graph, where each edge is assigned a weight that measures the similarity between the two data points connected by the edge; the method produces a label function that is smooth on the graph and correctly matches the known label. The method in Zhu *et al.* (2003) was originally designed for binary classification in this dissertation. The author extended it for multiclass classification.

Suppose that l labeled points $(\boldsymbol{\phi}_1, t_1), \ldots, (\boldsymbol{\phi}_l, t_l)$ from K classes, and u unlabeled points $(\boldsymbol{\phi}_{l+1}, t_{l+1}), \ldots, (\boldsymbol{\phi}_m, t_m)$ are given with $m = l+u$, where $\boldsymbol{\phi}_i \in \mathbb{R}^q$ is the feature for the i-th case and $t_i \in \{0, 1, \ldots, K-1\}$ is the label associated to $\boldsymbol{\phi}_i$; for unlabeled cases, the t_i values are unknown. Construct a connected graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{V}$ is the collection of nodes corresponding to the m data points, where the coordinates of a data point is specified by $\boldsymbol{\phi}_i$, and $\boldsymbol{E}$ is the collection of edges. The edge connecting $\boldsymbol{\phi}_i$ and $\boldsymbol{\phi}_j$ is weighted by the similarity measure

$$w_{ij} = \exp\left\{ -\sum_{d=1}^{p} \frac{(\phi_{id} - \phi_{jd})^2}{\sigma_d^2} \right\},$$

where ϕ_{id} is the d-th component of the feature vector $\boldsymbol{\phi}_i$, and σ_d is a scale for the d-th feature whose choice will be discussed later. Note that the σ_d-scaled Euclidean distance in the feature space corresponds to the geodesic distance in the original (curved) data space by the definition of the feature mappings. Therefore, the weightings, w_{ij}, reflect well the similarities between data points in the original data. Our use of geodesic distance is a major difference from Zhu *et al.* (2003), which defines the weightings using the scaled Euclidean distance in the data space.

A vector-valued label function $\boldsymbol{h} = (h_0, \ldots, h_{K-1})$ on $\boldsymbol{G}$ with certain properties is constructed, which assigns labels based on h. An unlabeled feature $\boldsymbol{\phi}$ is assigned the label k^* if $h_{k^*}(\boldsymbol{\phi}) = \max_{k=0,\ldots,K-1} h_k(\boldsymbol{\phi})$. It is required that the label function can reproduce the true label for the labeled data, that is, $h_k(\boldsymbol{\phi}_i) = \delta_{t_i,k}$, $i = 1, \ldots, l$, where δ is the Kronecker delta. Moreover, it is desirable to choose the label function $\boldsymbol{h}$ such that unlabeled points have the same labels as their neighboring points in the graph. These considerations motivate us to obtain the label function by minimizing with respect to $\boldsymbol{h}$ the following loss function

$$L(\boldsymbol{h}) = \frac{1}{2}\sum_{i,j} w_{ij}\|\boldsymbol{h}(\boldsymbol{\phi}_i) - \boldsymbol{h}(\boldsymbol{\phi}_j)\|^2 = \frac{1}{2}\sum_{k}\sum_{i,j} w_{ij}\{h_k(\boldsymbol{\phi}_i) - h_k(\boldsymbol{\phi}_j)\}^2,$$

subject to the constraints that $h_k(\boldsymbol{\phi}_i) = \delta_{t_i,k}$, $i = 1, \ldots, l$.

Let $\boldsymbol{W}$ denote the matrix whose (i,j)-th entry is w_{ij} and let $\boldsymbol{h}_k = (h_k(\boldsymbol{\phi}_1), ..., h_k(\boldsymbol{\phi}_m))^t$. The loss function can be rewritten as a quadratic form

$$L(\boldsymbol{h}) = \frac{1}{2}\sum_{k} \boldsymbol{h}_k^t \, \Delta \, \boldsymbol{h}_k, \tag{6}$$

where $\boldsymbol{\Delta} = \boldsymbol{D} - \boldsymbol{W}$, and $\boldsymbol{D}$ is the $m \times m$ diagonal matrix whose i-th diagonal entry is $d_i = \sum_j w_{ij}$. To present the solution of this minimization problem, the matrix $\boldsymbol{W}$ and $\boldsymbol{D}$ and the vector $\boldsymbol{h}_k$ are written in block forms according to labeled parts and unlabeled parts

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{ll} & \boldsymbol{W}_{lu} \\ \boldsymbol{W}_{ul} & \boldsymbol{W}_{uu} \end{bmatrix}, \ \boldsymbol{D} = \begin{bmatrix} \boldsymbol{D}_{ll} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{D}_{uu} \end{bmatrix}, \ \boldsymbol{h}_k = \begin{bmatrix} \boldsymbol{h}_k^{(l)} \\ \boldsymbol{h}_k^{(u)} \end{bmatrix}, \tag{7}$$

where $\boldsymbol{O}$'s denote matrices of zeros whose dimensions can be determined from the context. Note that $\boldsymbol{h}_k^{(l)}$ is given by the constraint $h_k(\boldsymbol{\phi}_i) = \delta_{t_i,k}$ for $i = 1, .., l$. Ignoring the term that is purely determined by $\boldsymbol{h}_k^{(l)}$, the loss function can be written as

$$\frac{1}{2}(\boldsymbol{h}_k^{(u)})^t(\boldsymbol{D}_{uu} - \boldsymbol{W}_{uu})\boldsymbol{h}_k^{(u)} - (\boldsymbol{h}_k^{(l)})^t\boldsymbol{W}_{ul}\boldsymbol{h}_k^{(u)}.$$

The following closed-from expression that minimizes $\boldsymbol{h}_k^{(u)}$ is obtained.

$$\boldsymbol{h}_k^{(u)} = (\boldsymbol{D}_{uu} - \boldsymbol{W}_{uu})^{-1}\boldsymbol{W}_{ul}^t\boldsymbol{h}_k^{(l)}. \tag{8}$$

Given the label function in (8), the label $k^* = \arg\max_k h_k(\boldsymbol{\phi}_{l+i})$ is assigned to the unlabeled feature $\boldsymbol{\phi}_{l+i}$ for $i = 1, \ldots, u$.

To choose a suitable scale σ_d in the weighting function, the author extended the heuristic rule by Zhu *et al.* (2003) to the multiclass setting. A σ_d that can make the most confident decision of labels is preferred. For a K-vector $\boldsymbol{p}$ with $\boldsymbol{p}'\boldsymbol{1}_K = 1$, the Shannon's entropy $H[\boldsymbol{p}]$ measures the uncertainty associated with the random variable whose probability distribution is $\boldsymbol{p}$. Thus a small value of $H[\boldsymbol{p}]$ is associated with a case that the random variable is more focused on certain value. Since the label function after appropriate normalization induces a probability distribution on the labels, the author proposed to find σ_d by minimizing the following average entropy

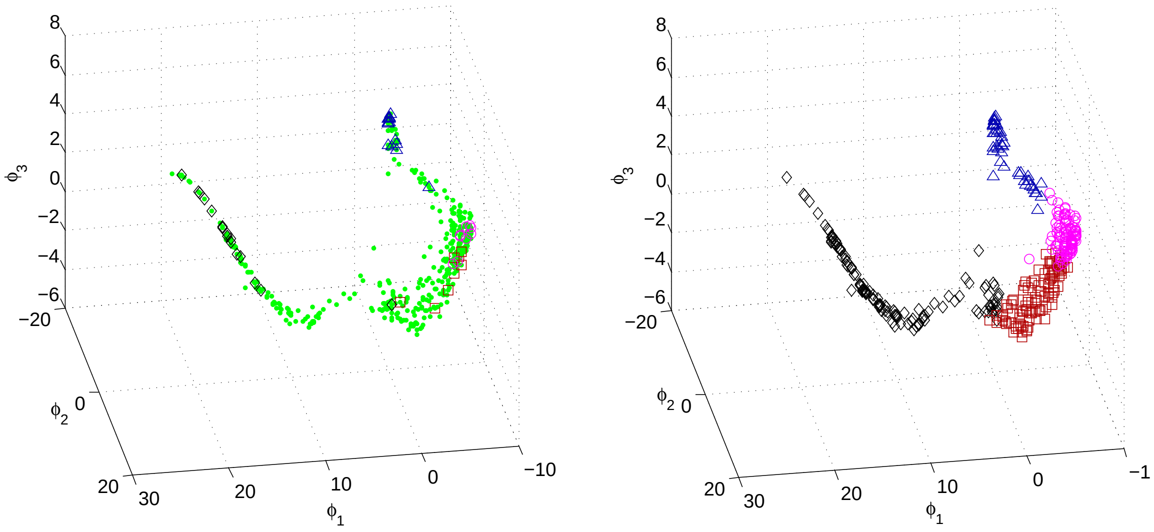
$$H(\boldsymbol{h}) = \frac{1}{u}\sum_{i=l+1}^{l+u} H_i\{\boldsymbol{h}(\boldsymbol{\phi}_i)\}, \tag{9}$$

where

$$H_i\{\boldsymbol{h}(\boldsymbol{\phi}_i)\} = H\left[\frac{h_0(\boldsymbol{\phi}_i)}{\sum_k h_k(\boldsymbol{\phi}_i)}, \frac{h_1(\boldsymbol{\phi}_i)}{\sum_k h_k(\boldsymbol{\phi}_i)}, \ldots, \frac{h_{K-1}(\boldsymbol{\phi}_i)}{\sum_k h_k(\boldsymbol{\phi}_i)}\right]$$

is the entropy associated with the i-th unlabeled case. This minimization problem can be solved by the gradient descent algorithm. Derivation of the gradients is straightforward and omitted.

In the application to the real electron micrographs, the author asked domain experts to determine the number of reference shapes and manually pick ten particles from each reference shape to form labeled data. According to the experts, it is



(a) Initial Assignment of Labels

(b) After the Semi supervised Learning

Fig. 9. Results of the semi-supervised learning. Each marker represents a low-dimensional embedding of a parametric curve. The triangular, rectangular, circular and diamond-shaped markers represent respectively triangular, rectangular, circular and rod-shaped particles, respectively. The left panel shows a mix of a few manually assigned shape labels and the unassigned ones, which is marked using small dots.

sufficient to distinguish a gold nanoparticle into one of the four shapes: triangles, rectangles, circles, and rods. These four shapes were assigned labels 0–3 respectively. Figure 9 presents the results of applying our semi-supervised learning procedure.

As a result of the clustering, the reference shapes are defined as a set of parametric curves that are classified to the zth reference shape,

$$R_z = \{\boldsymbol{f}_i; z = \arg\max_j h_j(\boldsymbol{\phi}_i), \boldsymbol{\phi}_i := \phi(\boldsymbol{f}_i)\}.$$

III.4. Principal Component Analysis on the Curve Space for Shape Inference

The reference shapes $R_1, \ldots, R_m$ are determined in the previous section, along with a shape-similarity measure. This section discusses the shape inference problem. Note that the shape classification problem must be considered together with the shape inference problem, as explained in Section I.5, because each of them requires the result from the other. In the divide-and-conquer approach, the shape classification problem is solved using only the incomplete contours, and the shape inference is then solved using the result of the classification problem.

A k-nearest neighbor (k-NN) classifier is used to classify a nanoparticle with incomplete contour information. A rough estimate of the gravity center for the nanoparticle is obtained by taking the center of a circle fitted with the incomplete contour, where the circle minimizes the mean square distance between itself and the incomplete contour. With this estimated gravity center, the method in Section III.2 can be used to yield a partially observed parametric curve representation of the particle for further analysis. Note that our gravity center estimate would be inaccurate if the missing part is significant and such inaccuracy would have a big impact on subsequent analysis. More sophisticated method needs to be developed for such a situation, which is left out for future research. The subsequent analysis is applied only to those particle contours whose missing part, if there is any, is no more than twenty percent of the entire contour, measured in terms of polar angles. In Section IV.3, a better classification method will be presented, which is able to perform the shape classification under larger portions of the missing part.

To apply a k-NN classifier, the author modified the rotationally invariant distance d_f defined in Subsection III.2.2 for it to be used with the incomplete contours. For a complete parametric curve $\boldsymbol{f}_i \in \mathbb{R}^p$ and the partially-observed curve $\boldsymbol{f}^*_{obs} \in \mathbb{R}^s$, the modified distance is defined as

$$d^*(\boldsymbol{f}_i, \boldsymbol{f}^*_{obs}) = \min_{\Delta=1,..,p} \|\boldsymbol{f}^*_{obs} - \boldsymbol{f}_i(\Delta, s)\|, \tag{10}$$

where $\boldsymbol{f}_i(\Delta, s)$ is a circularly completed subpart of $\boldsymbol{f}_i$ starting from Δ and having length s. That is, the distance is the minimum of the distances between $\boldsymbol{f}^*_{obs}$ and all possible continuous subparts of $\boldsymbol{f}_i$ with the same length as $\boldsymbol{f}^*_{obs}$.

Based on d^*, the k nearest neighbors of $\boldsymbol{f}^*_{obs}$ are chosen among $\{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_n\}$, denoted by $\boldsymbol{f}^{(1)}, \ldots, \boldsymbol{f}^{(k)}$, and then estimate the shape label of $\boldsymbol{f}^*_{obs}$ by a majority vote of these neighbors.

$$z = \arg\max_{t=1,\ldots,k} |\{\boldsymbol{f}^{(j)} \in R_t\}|,$$

where $|A|$ is the cardinality of A. The neighborhood size k can be determined by a ten-fold cross validation.

Once R_z, the reference shape of $\boldsymbol{f}^*_{obs}$, is defined, R_z can be used to recover the missing part of the incomplete contour $\boldsymbol{f}^*_{obs}$. Note that R_z is the set of the complete contours classified to the zth reference shape (Subsection III.3.2). The shape inference procedure takes the following steps. First, the method of functional principal component analysis by Huang *et al.* (2008) is applied to summarize the variations of the complete contours in R_z. Next, the observed part of the incomplete contour ($\boldsymbol{f}^*_{obs}$)

is used to obtain the corresponding principal component scores. Finally, combine the principal component basis functions and the principal component scores to fill in the missing part for $\boldsymbol{f}^*_{obs}$.

Let $\boldsymbol{f}^* = (\boldsymbol{f}^{*t}_{obs}, \boldsymbol{f}^{*t}_{mis})^t \in \mathbb{R}^p$ be a partially observed curve, where $\boldsymbol{f}^*_{obs} \in \mathbb{R}^s$ and $\boldsymbol{f}^*_{mis} \in \mathbb{R}^{p-s}$ denote respectively its observed and unobserved parts. Let $\boldsymbol{f}_{z,1}, .., \boldsymbol{f}_{z,r} \in R_z$ and let $\bar{\boldsymbol{f}}$ be the sample mean of these curves. Consider the following expansion

$$\boldsymbol{f}_i = \bar{\boldsymbol{f}} + \boldsymbol{v}_1 u_{1i} + \cdots + \boldsymbol{v}_k u_{ki}, \qquad i = 1, \ldots, r, \tag{11}$$

where $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ are the principal component curves and $u_{1i}, \ldots, u_{ki}$ are the corresponding scores. The principal component curves are obtained by sequentially minimizing a regularized least squares criterion that penalizes the roughness of the curves (Huang *et al.*, 2008). In particular,

$$\boldsymbol{v}_1 = \arg\max_{v} \left\{ \sum_{i=1}^{r} \|\boldsymbol{f}_i - \bar{\boldsymbol{f}} - \boldsymbol{v}_1 u_{1i}\|^2 + \alpha \sum_{i=1}^{r} u_{1i}^2 \, \boldsymbol{v}^t \Omega \boldsymbol{v} \right\},$$

where α is a penalty parameter and Ω is a penalty matrix, defined in Huang *et al.* (2008, page 687). Subsequent principal component curves are obtained similarly by using the residuals after removing preceding components. Following Huang *et al.* (2008), the Generalized Cross-Validation (GCV) is used to select the penalty parameter α, and the number of principal components is chosen so that the majority (i.e., 99%) of the sample variation is accounted for. The reason of using roughness penalties is to ensure that the recovered particle boundaries are smooth. Note that (11) can be rewritten in a matrix form as $\boldsymbol{f}_i = \bar{\boldsymbol{f}} + \boldsymbol{V}\boldsymbol{u}_i$ where $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ and $\boldsymbol{u}_i = (u_{1i}, \ldots, u_{ki})^t$. Similarly, by partitioning $\bar{\boldsymbol{f}}$ and $\boldsymbol{V}$, the partially observed curve has the expansion

$$\begin{bmatrix} \boldsymbol{f}^*_{obs} \\ \boldsymbol{f}^*_{mis} \end{bmatrix} = \begin{bmatrix} \bar{\boldsymbol{f}}_{obs} \\ \bar{\boldsymbol{f}}_{mis} \end{bmatrix} + \begin{bmatrix} \boldsymbol{V}_{obs} \\ \boldsymbol{V}_{mis} \end{bmatrix} \boldsymbol{u}^*. \tag{12}$$

A regression using the first part (i.e. *obs* part) of the system is executed to obtain the vector of principal score $\boldsymbol{u}^*$, and the vector is in turn plugged into the second part (i.e. *mis* part) to get the missing part of the curve. See the last column in Figure 5 for some examples of the recovered boundaries.

III.5. Results and Discussion

The ultimate objective of the morphology analysis problem is to obtain the summary statistics of the morphology of nanoparticles, which usually includes three major distributions: (a) the size distribution (the size of a particle is characterized by the length of the longest axis of the corresponding boundary.), (b) the shape distribution, and (c) the distribution of the aspect ratios, defined as the length of the perimeter of a boundary divided by the area of the same boundary. The three statistics are widely adopted in nano science and engineering to characterize the morphology of nanoparticles, and are believed to strongly affect the physical or chemical properties of the nanoparticles (El-Sayed, 2001; Nyiro-Kosa *et al.*, 2009). For example, the aspect ratio is considered an important parameter relevant to certain macro-level material properties because physical and chemical reactions are believed to frequently occur on the surface of particles so that as the aspect ratio of a nanoparticle gets larger, those reactions are more active.

Now reported are the results of applying our procedure to six actual electron micrographs under different scales. One image consists of Palladium (Pd) nanoparticles prepared by microwaving a Palladium solution with a surfactant. The remaining five images contain gold nanoparticles reduced from a gold salt solution heated and stirred, adding different ratios of citrate concentration. The results are the distributions of size, shape, and aspect ratio in the form of histogram. We also applied the `ImageJ` to the same six electron micrographs, and we compared the result from `ImageJ` with that from the divide-and-conquer approach for each micrograph. Figure 10 and 11 is the comparison for the first micrograph; Figure 12 and 13 for the second one; Figure 14 and 15 for the third one; Figure 16 and 17 for the fourth one; Figure 18 and 19 for the fifth one; and Figure 20 and 21 for the sixth one.

For those particles that are successfully identified and classified, they are labeled by an integer number in the image. One can observe that our procedure recognizes the majority. Most of the unlabeled particles are those located on the border of the image, of which a significant portion was not observed. Domain experts deem these particles unnecessary to be identified so that they are intentionally removed before our analysis. Our classification results of particle type are also verified by domain experts and deemed satisfactorily accurate. This verification is done manually by the domain experts, who looked through each image, compared the identification result with the original image, and then counted the number of correctly and incorrectly identified subjects. This manual verification appears the only valid way for the time being.

As part of the verification process, the author compared the accuracy of our method with the imaging tool `ImageJ` that is popularly used in the nanotechnology research. The author chose six electron micrographs and categorized the images into two groups by the frequency of particle overlaps; 'slight and rare', and 'slight but pervasive'. The images in the former group (the images in Figure 10 through Figure 14) have few overlaps among nanoparticles, and the images in the latter group (the images in Figure 16 through Figure 20) contain many nanoparticles entangled with other ones. Note that, according to the way of classifying electron micrographs in Chapter IV, both of these groups mostly correspond to the cases for low degree of particle overlaps; Image 6 corresponds to medium degree.

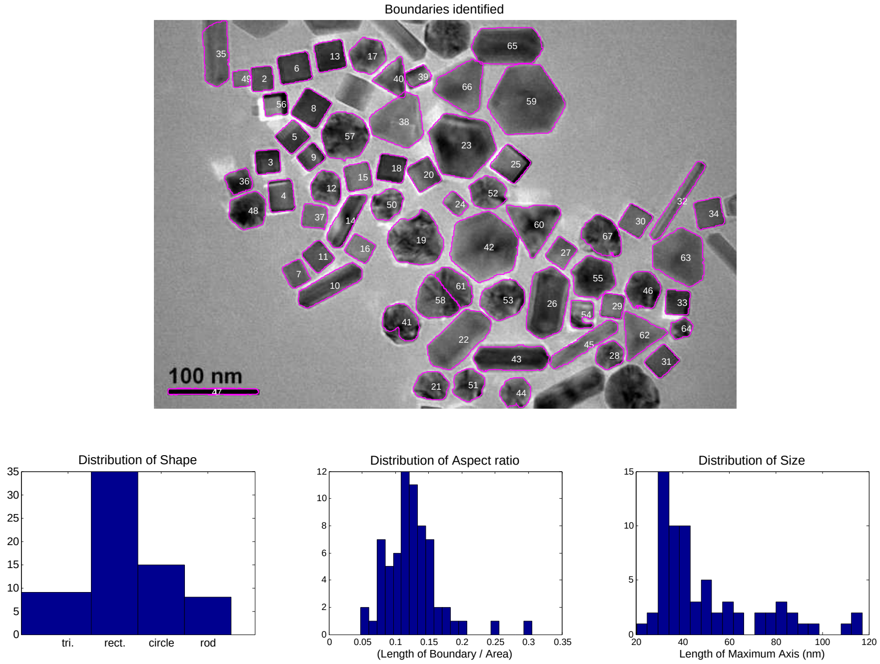


Fig. 10. Identification results from our method for Image 1. 66 particles manually identified, and 62 identified by our method, identification rate = 94%. The top figure shows the boundaries of the identified ones.

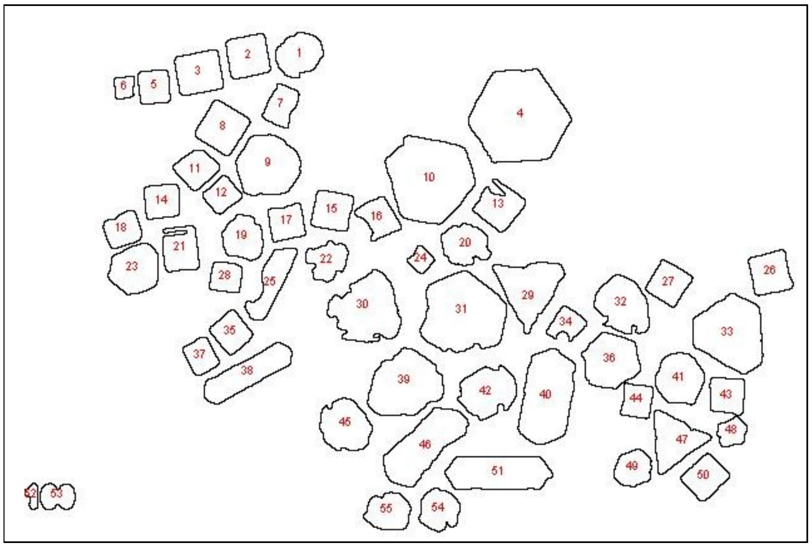


Fig. 11. Identification results from `ImageJ` for Image 1: Only the identified particles are shown. Out of the 66 particles, 52 are identified. Identification rate = 79%.

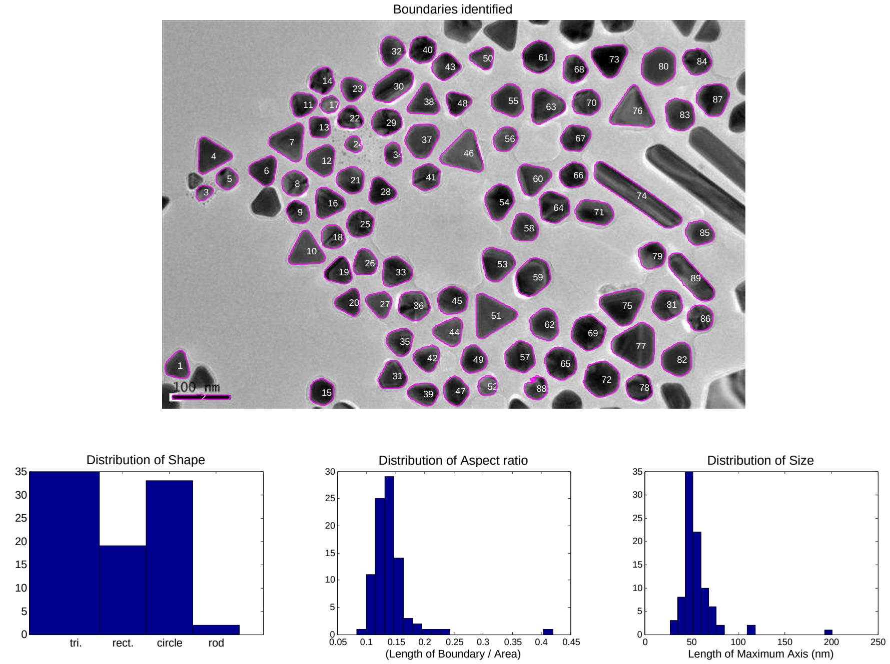


Fig. 12. Identification results from our method for Image 2. 91 particles manually identified, and 88 identified by our method, identification rate = 97%. The top figure shows the boundaries of the identified ones.

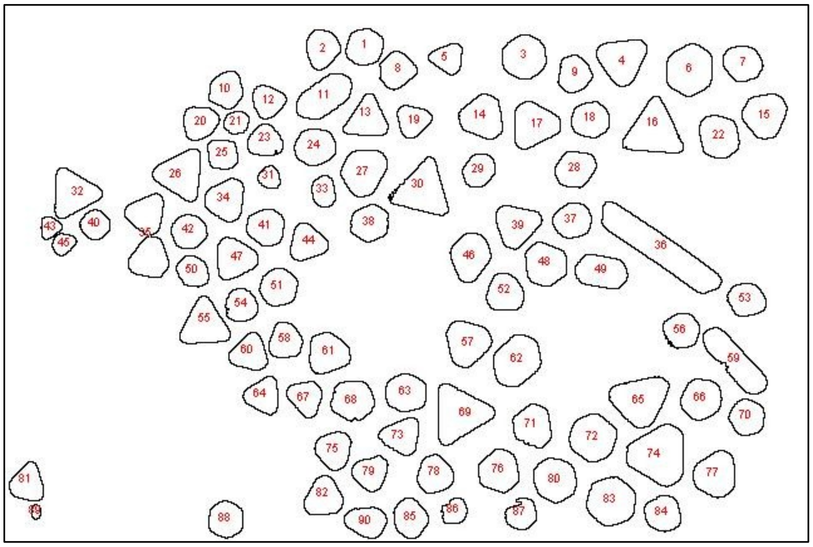


Fig. 13. Identification results from `ImageJ` for Image 2: Only the identified particles are shown. Out of the 91 particles, 87 are identified. Identification rate = 96%.

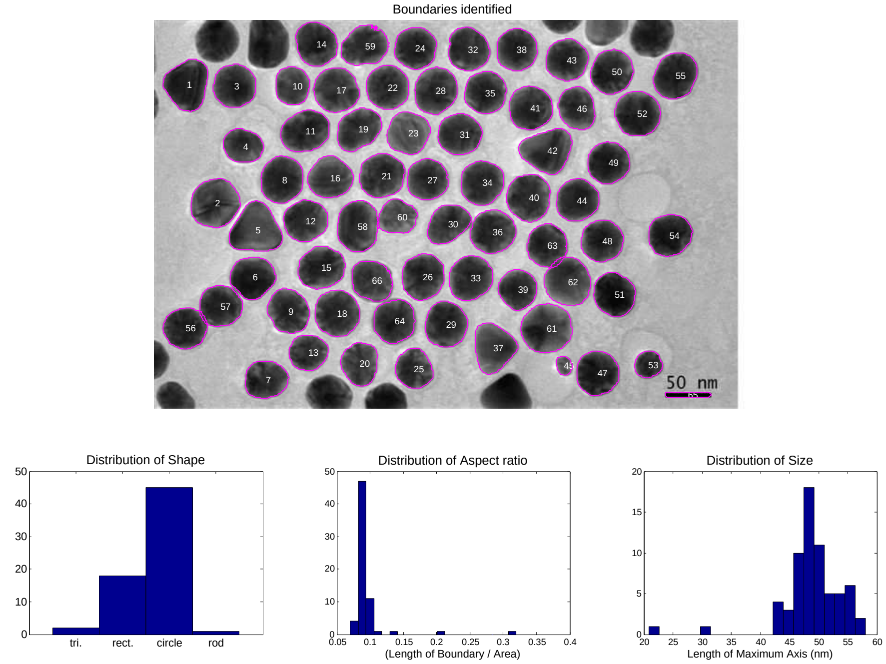


Fig. 14. Identification results from our method for Image 3. 64 particles manually identified, and 64 identified by our method, identification rate = 100%. The top figure shows the boundaries of the identified ones.

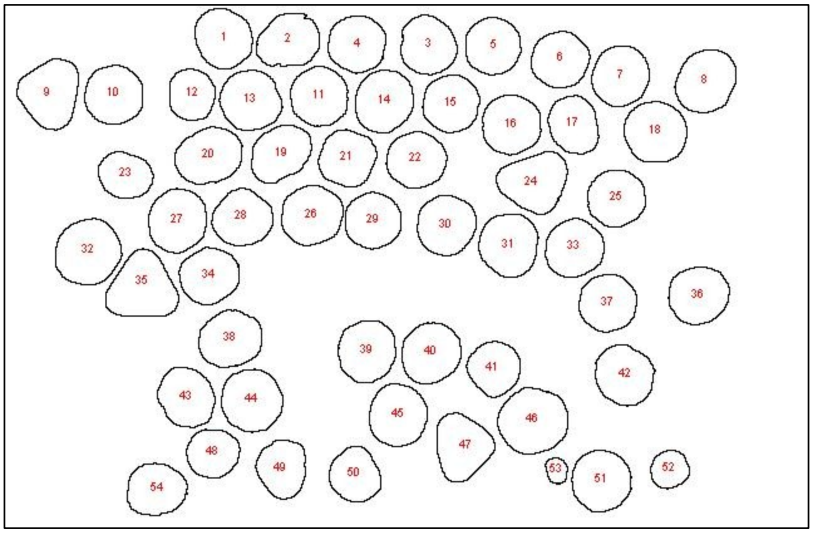


Fig. 15. Identification results from `ImageJ` for Image 3: Only the identified particles are shown. Out of the 64 particles, 53 are identified. Identification rate = 83%.

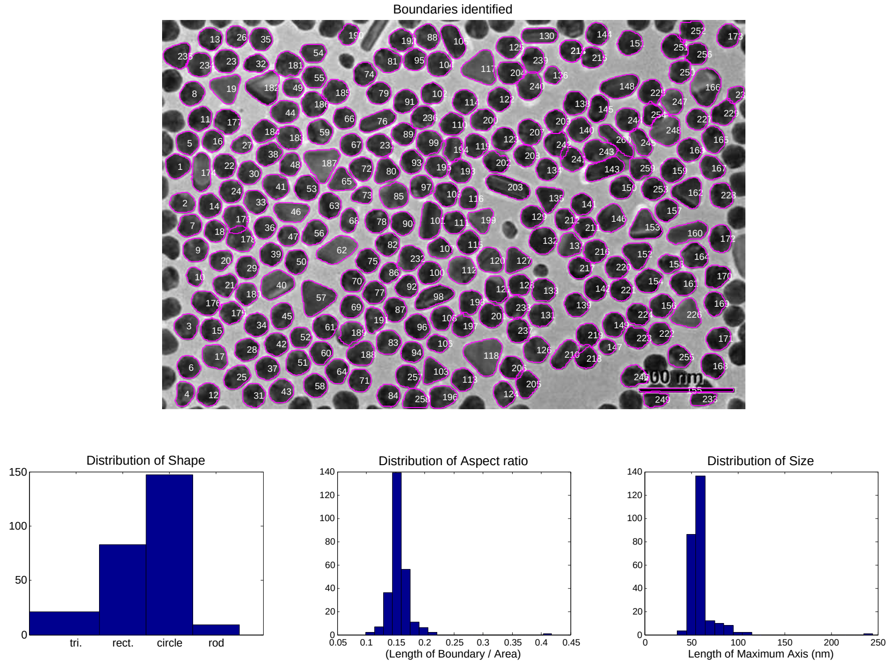


Fig. 16. Identification results from our method for Image 4. 259 particles manually identified, and 252 identified by our method, identification rate = 97%. The top figure shows the boundaries of the identified ones.

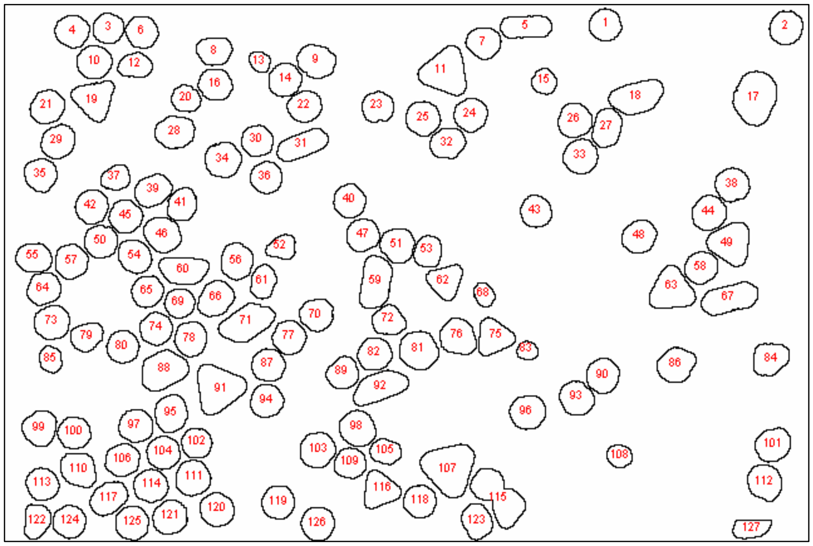


Fig. 17. Identification results from `ImageJ` for Image 4: Only the identified particles are shown. Out of the 259 particles, 124 are identified. Identification rate = 47.8%.

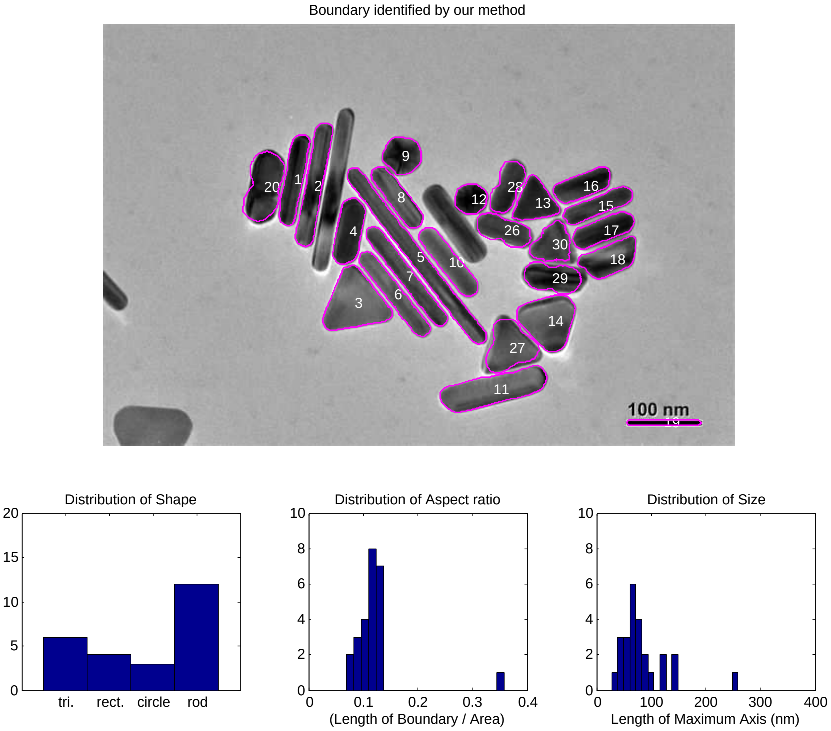


Fig. 18. Identification results from our method for Image 5. 26 particles manually identified, and 24 identified by our method, identification rate = 92%. The top figure shows the boundaries of the identified ones.

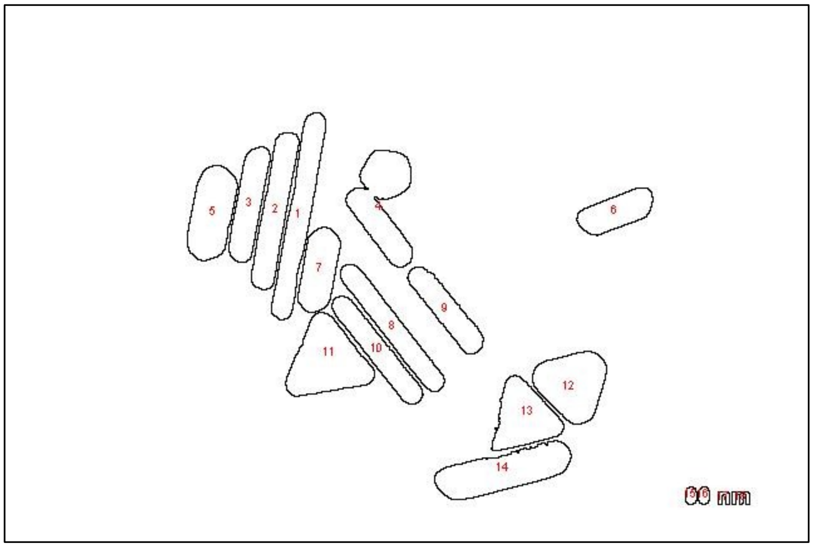


Fig. 19. Identification results from `ImageJ` for Image 5: Only the identified particles are shown. Out of the 26 particles, 13 are identified. Identification rate = 50%.

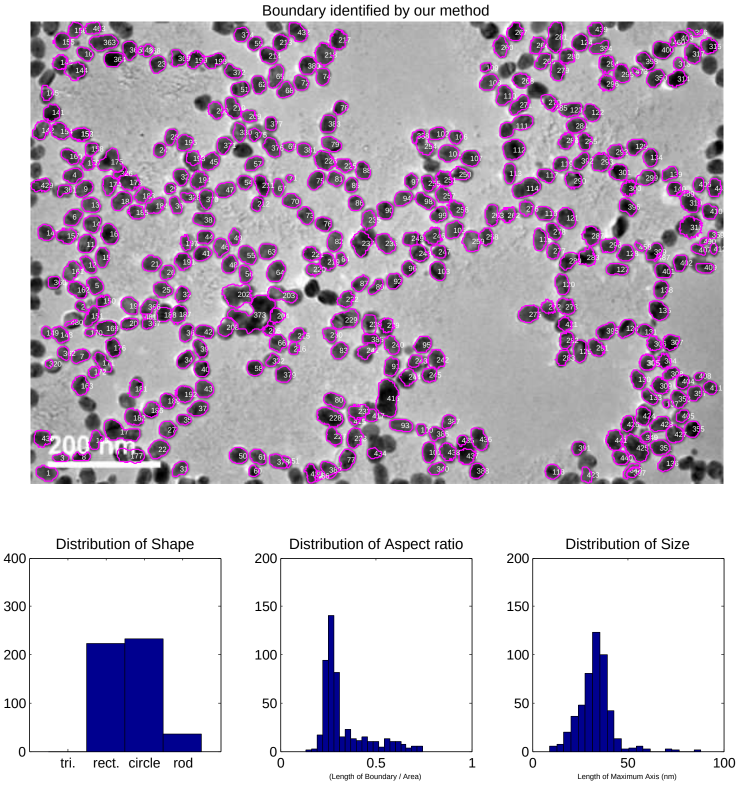


Fig. 20. Identification results from our method for Image 6. 396 particles manually identified, and 363 identified by our method, identification rate = 91.7%. The top figure shows the boundaries of the identified ones.

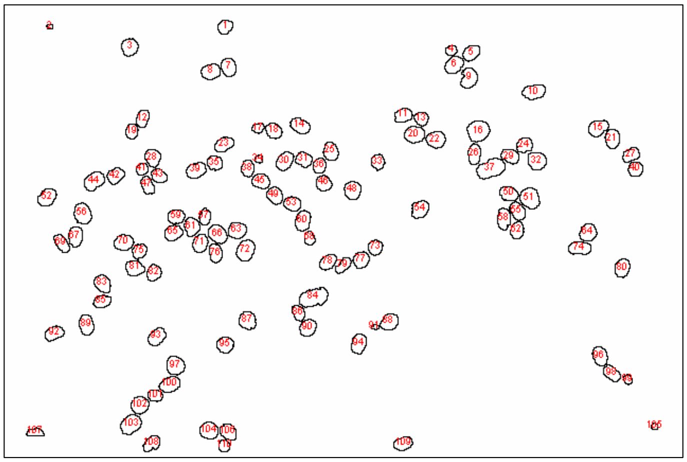


Fig. 21. Identification results from `ImageJ` for Image 6: Only the identified particles are shown. Out of the 396 particles, 110 are identified. Identification rate = 27.8%.

Table 3. Comparison of the divide-and-conquer approach with `ImageJ` in terms of particle identification rate.

Samples	Overlap	Total # of particles	# of the identified ones	
			Our method	`ImageJ`
Image 1 (Figure 10)	Slight/Rare	66	62	52
Image 2 (Figure 12)	Slight/Rare	91	88	87
Image 3 (Figure 14)	Slight/Rare	64	64	53
Image 4 (Figure 16)	Medium/Pervasive	259	252	124
Image 5 (Figure 18)	Slight/Pervasive	26	24	13
Image 6 (Figure 20)	Slight/Pervasive	396	363	110

Table 3 summarizes the numbers of nanoparticles identified by the proposed method and `ImageJ` for the six electron micrographs. For three electron micrographs in 'slight and rare' group, the proposed method identified 94 - 100% of the total particles, comparing with 78 - 95% identification rates of `ImageJ`. For three other electron micrographs in 'slight but pervasive' group, the proposed method obtained 91.7 - 97% identification rates, whereas `ImageJ`'s identification rates are ranged 28 - 50%. Considering frequent occurrence of overlaps in the electron micrographs of nanoparticles, the existing software tool cannot be more than a supporting tool. The high identification rate of our method can facilitate a great deal the nano-material exploration.

CHAPTER IV

A CONVEX SHAPE ANALYSIS FOR ANALYZING CHAIN-LINKED NANOPARTICLES

In Chapter III, the divide-and-conquer approach was examined, which sequentially solves the image segmentation, the shape classification and the shape inference problems. The sequential approach is designed for fast extraction of particle size and shape distributions when particle overlaps are not very severe. However, solving the three inter-related problems sequentially may produce less ideal outcomes in the morphology analysis than solving them all together, because the quality of outcome from one problem can affect the next step of the problem-solving.

Natural enhancements in the divide-and-conquer approach can be possible in two ways. The first one is to develop more accurate methods for each of the three component problems. The second one is to combine all three component problems or some of them into one unified framework. This chapter presents a better approach than the divide-and-conquer approach by following both enhancement directions. The new method enhances the solving of shape inference and shape classification problems by combining them into one problem, and enhances the solving of the image segmentation problem by introducing a new morphological erosion procedure and association of evidence.

This chapter starts in Section IV.1 with the discussion about the limitation of the divide-and-conquer approach. The enhanced image segmentation method is presented in Section IV.2. The new formulation for the combined shape classification and inference problem is presented in Section IV.3. Section IV.4 is devoted to showing the enhanced performance of the new approach with comparison to the divide-and-conquer approach and other state-of-the-art methods.

IV.1. Limitation of the Divide-and-Conquer Approach

The divide-and-conquer approach has two major limitations in terms of handling various cases of particle overlaps. The first limitation is related to the capability of the image segmentation method used in the divide-and-conquer approach, which provides the desirable solution only if the condition in Proposition 1 holds. However, there are many cases where Proposition 1 does not satisfy. For an example, please look at Figure 2. In this example, three nanoparticles overlaps in a chain-link cluster. When the contour decomposition method in the divide-and-conquer approach is applied, the result would look like part (c), namely that the contour of the particle in the middle is split into two parts by the other two particles; this leads to over-segmentation. Such problems occur whenever nanoparticles are linked in a chain structure. Since the chain-linked structure is often observed in actual electron micrographs of nanoparticles, an image segmentation method having the capability of properly handling the chain-linked overlaps is pressingly needed.

The second limitation is related to the sequential way of solving shape classification and shape inference in the divide-and-conquer approach. Please recall that the gravity center of an incomplete contour needs to be estimated for the k-NN shape classification (Section III.4). The rough estimate of the center is inaccurate when the degree of particle overlaps is high and the missing part of the incomplete contour is significant. An inaccurate estimate of the center leads to inaccurate shape classification and shape inference. The inaccurate center estimate is mainly caused by that no accurate inference of the missing contour is available when the shape classification is performed. Therefore, if one can solve the shape classification along with the shape inference, the problem would be much alleviated, and a unified formulation of shape inference and classification can handle particle overlaps at a much higher degree. The combined formulation will be presented in Section IV.3.

IV.2. Convex Decomposition of Chain-linked Nanoparticles

As discussed in Section I.4, the image segmentation in the morphology analysis becomes a convex decomposition problem, namely segmenting a complicated morphology into convex sub-pieces. In the meanwhile, the divide-and-conquer approach could suffer from over-segmentation in the cases of chain-linked structure of particle overlaps. This section presents a new convex decomposition approach that enhances the experience of image segmentation.

The basic strategy is to identify the convex subsets of individual convex sets composing a chain-structure (one subset per convex set). The convex subsets are called the markers of the individual convex sets. The contour segments to be associated with the markers are called the contour evidences, because the contour segments are in fact the evidences used for inferring the complete contours later in the shape inference and classification problem. Subsection IV.2.1 explains how to find the markers of individual convex pieces while Subsection IV.2.2 explains how to associate the contour segments with the markers.

The proposed convex decomposition method is built upon the existing morphological segmentation method, called ultimate erosion (Dougherty, 1994, UE). But the proposed method has a couple of key differences from UE. First, the stopping criterion for iterative erosion is different, and the one used in the proposed method produces more robust segmentation results for convex sets. Second, the proposed method skips the marker-growing step used in a typical morphological segmentation method. Rather, it directly identifies the contour evidences relevant to each marker by detecting contour segments and relating the segments as contour evidences to

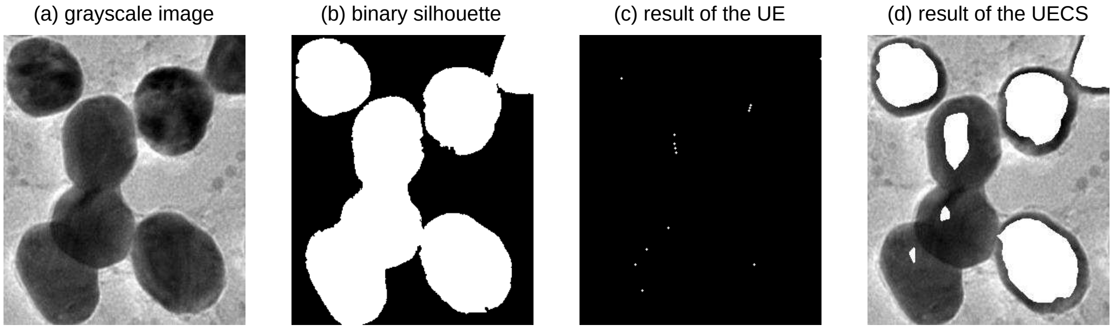


Fig. 22. Ultimate erosion for convex sets. (a) the original grayscale image, (b) binary silhouette of clustered convex shaped objects, (c) markers identified by the ultimate erosion (UE) (d) markers identified by the ultimate erosion for convex sets (UECS).

one of the markers. To relate the contour segments, an evidence-to-marker relevance measure is defined, which is also different from those used in the marker-growing approach.

IV.2.1. Ultimate Erosion for Overlapping Convex Sets

Suppose that a binary silhouette of overlapping objects from a grayscale image is given (please see Figure 22-(b) for an example). Since nanoparticles have only convex morphologies, the binary silhouette is a union of the convex silhouettes of individual particles. This section describes how to decompose the binary silhouette into disjoint convex sets.

Suppose that there are n nanoparticles in a micrograph. Let I be the binary silhouette of the particles and C_i be a set of pixels in the interior or on the boundary of nanoparticle i, where C_i should be a convex set due to the convexity of the particle's morphology. As such $I = \bigcup_{i=1}^{n} C_i$. Given a non-empty set I, the objective is to obtain a connected subset for each C_i, called the marker of C_i, so that the markers are pairwise disjoint. The marker plays an important role to locate C_i in a micrograph and to guide the particle segmentation. A possible way to produce the marker is to use a morphological erosion. A morphological erosion is performed by using a set operator called a Minkowski subtraction in the mathematical morphology (Schneider, 1993), defined as follows.

Definition 2. *(Schneider, 1993, Page 133) A Minkowski subtraction to I with respect to Δ is*

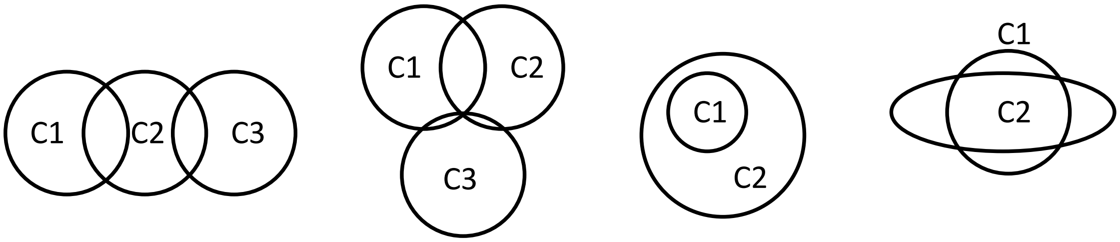
$$I \ominus \Delta \equiv \{x : \Delta_x \subset I\},$$

where Δ_x is the translation of Δ by x, i.e. $\{x + y : y \in \Delta\}$.

Intuitively, if $\Delta = B(0, 1)$, where $B(x, r)$ is a closed ball in $\mathbb{R}^2$ centered at x with radius r, the result of the set operator is equivalent to peeling off I from its boundary by size one. Repeated applications of the operator may disconnect the junctions of overlapping objects. The question is when to stop the morphological erosion. A popular choice is to keep applying the erosion process to each object just before it is completely removed. This is called the ultimate erosion (Dougherty, 1994, Chapter II).

In this section, the author shows that the ultimate erosion (UE) is capable of identifying exactly one marker for each C_i in I when C_i's are convex and when I is specially structured as described in Assumption 3. The author also shows that the UE is susceptible to over-segmentation when C_i is corrupted by noise so it is not perfectly convex. The author proposes a noise-robust morphological erosion process with an earlier stopping criterion than that of the UE. The author calls the erosion process ‘ultimate erosion for convex sets’, shortly UECS.

Assumption 3. *(Chained cluster of overlapping objects) The intersection of every three of the n convex sets composing I is at most one point and for every pair $i \neq j$, $C_i \backslash C_j$ is not empty and it is connected.*



(a) Cases satisfying Assumption 1 C1 ∩ C2 ∩ C3 = empty

(b) C1 ∩ C2 ∩ C3 = non-empty

(c) C1 \ C2 is empty

(d) C1 \ C2 is disconnected

Fig. 23. Intuitive examples explaining Assumption 3. (a) and (b) satisfy the assumption, and (c) and (d) violate the assumption.

Intuitively, the assumption is related to the degree of overlaps among particles; please refer to Figure 23 for an illustration. In real micrographs, one could observe that many overlapping nanoparticles satisfy the assumption. For examples, please see Figure 24. The exemplary micrographs depict the complicated chain-linked clusters of overlapping nanoparticles. The chain-linked clusters of nanoparticles are separable by the UE defined below.

Definition 4. *(Ultimate erosion, UE (Dougherty, 1994, Page 72)) The ultimate erosion to I is an iterative process to update $I^{(t)}$:*

Initialization: *Start with $I^{(0)} = I$.*

Iteration t***:*** *For the ith connected component $A_i^{(t-1)}$ in $I^{(t-1)}$, compute R_i and update $I^{(t)} = \bigcup_i R_i$.*

$$R_i = \begin{cases} A_i^{(t-1)} \ominus B(0,1) & \textit{if } A_i^{(t-1)} \ominus B(0,1) \neq \emptyset \\ A_i^{(t-1)} & \textit{otherwise} \end{cases}$$

End: *The iteration stops when $I^{(t)} = I^{(t-1)}$.*

Each connected set resulting from the UE becomes a marker. If there is one marker for each C_i, the author says that I is separable by the UE. In the following

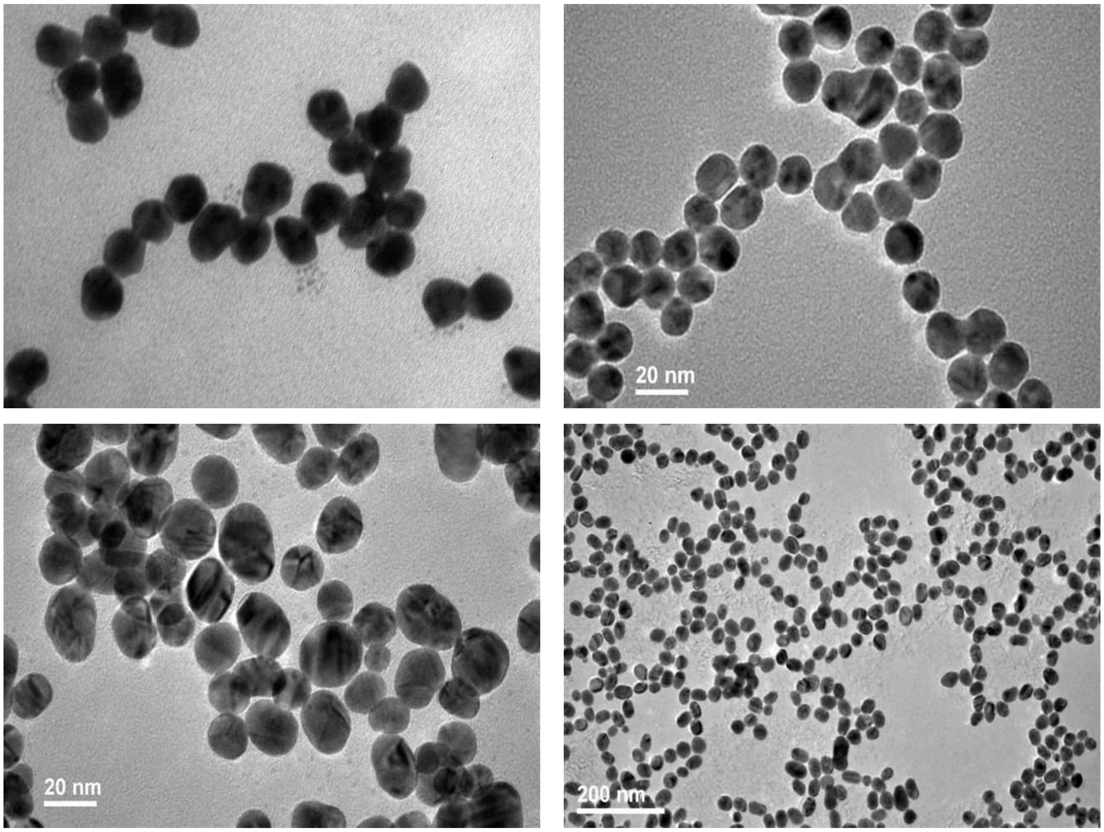


Fig. 24. Electron micrographs picturing the chain-linked clusters of nanoparticles.

text, the author will prove that I is separable by the UE under some regularity conditions on I.

First, new notations and functions are defined to characterize the process of the UE. For a connected set C in $\mathbb{R}^2$, let $d_C(x)$ be the distance function from $x \in \mathbb{R}^2$ to ∂C such that $d_C(x) = \inf\{||x - y|| : y \in \partial C\}$, where ∂C is the boundary of C.

The UE repeatedly applies the Minkowski subtraction to C with respect to $B(0,1)$. Let us denote by $C \ominus rB(0,1)$ the r successive applications of the Minkowski subtraction to C with respect to $B(0,1)$. The result of the iterative erosion process is characterized by the level-set of the distance function $d_C(x)$.

Lemma 5. *Let C be a connected set in $\mathbb{R}^2$.* $C \ominus rB(0,1) = C \ominus B(0,r) = \{x \in C :$

$d_C(x) \geq r\}$.

Proof. The proof is straightforward by the definition of Minkowski subtraction. Please see (Dougherty, 1994, page 324). □

Note that a marker resulting from the UE is a local maximum of the distance function by definition. Hence, it is important to understand the modes of $d_C(x)$. Define $r_C = \sup_{x \in C} d_C(x)$ and Ω_C as a subset of C satisfying

$$\Omega_C = \{x \in C : d_C(x) = r_C\}.$$

The points in Ω_C are the global maxima of $d_C(x)$ by definition. If C is convex, Ω_C is equivalent to the set of all local maxima of $d_C(x)$. This is implied in the following lemma (for its proof, please refer to Appendix A. Same for Lemma 7.)

Lemma 6. *If C is a convex set, the function $d_C(x)$ has the following properties: for $x \in \Omega_C$ and $y \in C$,*

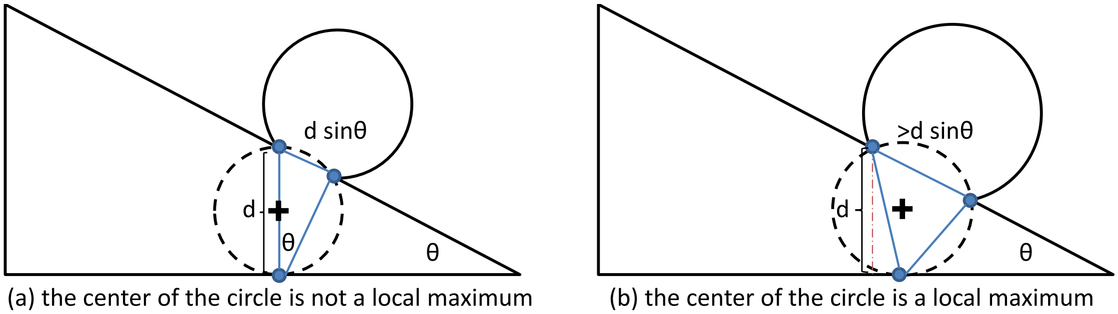
(i) $d_C(\lambda x + (1-\lambda)y)$ is a concave function of λ on $[0,1]$.

(ii) $d_C(\lambda x + (1-\lambda)y)$ is a monotone (increasing) function of λ on $[0,1]$.

The author has so far discussed about the local maxima of the distance function $d_C(x)$ for a single convex set C. Based on the results, the local maxima of the distance function will be studied for a union of convex sets I, which is essential for understanding the separation capability of the UE.

Let $C_1, C_2, \ldots, C_n$ be n convex sets in $\mathbb{R}^2$ with $I = \bigcup_{i=1} C_i$, satisfying Assumption 3. Let $U_i = \text{conv}(C_i \backslash \bigcup_{j \neq i} C_j)$. It is easy to show that $I = \bigcup_{i=1}^{n} U_i$. A supporting set $S(x)$ is defined by

$$S(x) = \{y \in \partial I : ||x - y|| = d_I(x)\} \quad \text{for } x \in I.$$



(a) the center of the circle is not a local maximum

(b) the center of the circle is a local maximum

Fig. 25. Practical Implication of Lemma 7. (a) does not satisfy the 'if-and-only-if' condition for the local maxima of $d_I(x)$ and (b) does satisfy.

Lemma 7. *A point x in U_i is a local maximum of $d_I(x)$ if and only if there exists $\epsilon > 0$ such that $B(x, \epsilon) \subset \{\sum_k \lambda_k z_k : z_k \in S(x), \sum_k \lambda_k = 1, \lambda_k \geq 0\}$.*

Lemma 7 is about where the local maxima of the distance function $d_I(x)$ locate. The UE stops when it identified the local maxima, which become the markers for C_i's. Therefore, the lemma is essential for showing the separation capability of the UE in Lemma 8. Before introducing Lemma 8, the author wants to elaborate the practical meaning of the 'if-and-only-if' condition in Lemma 7 by an example. The condition is related to the size of the allowable overlappings between particles. Please look at Figure 25. In the figure, a triangle and a circle overlap. The author wants to check if the location at the cross (x) is a local maximum of $d_I(x)$. The solid dots are the elements in the supporting set $S(x)$, which are on the dotted circles centered at x. Let θ be the angle between the following two lines: a line connecting two intersection points of the triangle and the circle, a tangent line to the dotted circle. If the length of the former line is greater than $d \sin\theta$, x is a local maximum satisfying the 'if-and-only-if' condition of Lemma 7 (Figure 25-(b)); otherwise, it is not a local maximum (Figure 25-(a)). In this example, the triangle has another local maximum and the circle has its own local maximum. Therefore, if the size of the overlapping region is greater than $d \sin\theta$ in this example, there are at least three local maxima for two objects, which results in three markers (over-segmentation). In general, if the size of the overlapping region, specified by the length of the line cutting through an overlapping region, is greater than a certain threshold (e.g. $d \sin \theta$ here), the over-segmentation will happen due to excessive number of local maxima in the distance function caused by the overlaps.

Lemma 8. *(Separability of the UE) Suppose that* I *is a union of* n *overlapping convex sets, namely* $C_1, \ldots, C_n$*, satisfying Assumption 3. Then,* I *is separable by the ultimate erosion (UE) if and only if* U_i *is not empty and there exists at most one* $x \in U_i$ *and* $\epsilon > 0$ *such that* $B(x, \epsilon) \subset \{\sum_k \lambda_k z_k : z_k \in S(x), \sum_k \lambda_k = 1, \lambda_k \geq 0\}$ *for* $i = 1, \ldots, n$*.*

Proof. I is separable by UE if and only if for every U_i there exists exactly one marker, which is equivalent to that there is only one local optimum of distance function $d_I(x)$ in U_i for $i = 1, \ldots, n$. By Lemma 7, the latter condition is in turn equivalent to the if-and-only-if condition of this lemma. $\square$

As implied by the conditions in Lemma 8, the UE is capable in handling a chain-linked cluster of convex sets when C_i is perfectly convex, but it is prone to producing more than one marker for each C_i when C_i is not perfectly convex (e.g., due to noise corruption). Having more than one marker for C_i leads to over-segmentation of nanoparticles, e.g. Figure 22-(c). To avoid over-segmentation, the iterative erosion process should stop before the process separates small non-convexities caused by noise. The author proposes a modified ultimated erosion process that stops at an earlier time than the UE and shows that it is as capable as the UE in terms of separating a mixture of convex sets into individual ones. The author calls this modified process the ultimate erosion for convex sets (UECS).

Definition 9. *(Ultimate erosion for convex sets, UECS) The ultimate erosion to* I

is an iterative process to update $I^{(t)}$*:*

Initialization: *Start with* $I^{(0)} = I$*.*

Iteration t***:*** *For the* i*th connected component* $A_i^{(t-1)}$ *in* $I^{(t-1)}$*, compute* R_i *and update* $I^{(t)} = \bigcup_i R_i$*.*

$$R_i = \begin{cases} A_i^{(t-1)} \ominus B(0,1) & \text{if } A_i^{(t-1)} \text{ is not convex and} \\ A_i^{(t-1)} & \text{otherwise} \end{cases}$$

End: *The iterations stop when* $I^{(t)} = I^{(t-1)}$*.*

The UECS stops the erosion process when the marker is still convex. When I conforms with Assumption 3 and when the 'if-and-only-if' condition in Lemma 8 holds, the equivalence of the UECS and the UE in terms of identifying markers is indirectly shown by proving the following lemma; it implies that the UE cannot identify more markers than the UECS when C_i is perfectly convex.

Lemma 10. *Suppose that* I *is a union of* n *overlapping convex sets, namely* $C_1, \ldots, C_n$*. If* I *is convex, then none of* C_i *is separable by UE.*

Proof. By Lemma 6, if I is convex, $d_I(x)$ is a convex function on I. Therefore, $I \ominus rB = \{d_I(x) \geq r\}$ is convex, which is connected for every $r > 0$. □

Theorem 11. *(Separability of the UECS) Suppose that* I *is a union of* n *overlapping convex sets, namely* $C_1, \ldots, C_n$*, satisfying Assumption 3. Then,* I *is separable by the UECS if and only if there exists* $\epsilon > 0$ *such that* $B(x, \epsilon) \subset \{\sum_k \lambda_k z_k : z_k \in S(x), \sum_k \lambda_k = 1, \lambda_k \geq 0\}$ *for* $i = 1, \ldots, n$*.*

Proof. By Lemma 10, the separation capability of the UE is equivalent to that of the UECS. Therefore, the if-and-only-if condition for the separability of the UECS is the same as the condition for the separability of the UE in Lemma 8. □

Different from the UE, the UECS can avoid over-segmentation by using a noise-robust measure of 'convexity' (or conversely, a measure of concavity) that determines the stopping criterion of erosion operations. Many previous works have already been reported regarding how to measure the size of concavity in digital images (Ronse, 1989). Rosenfeld (Rosenfeld, 1985) compared three popular measures for the concavity in terms of their sensitivity to the coarseness of digital grids. According to the comparison, the following concavity measure is the least sensitive to the coarseness.

Definition 12. *(Rosenfeld, 1985, Page 72) Let I be a connected set. If $O = conv(I)$, $V = O \backslash I$ is called a concavity of I. Suppose that V consists of m connected sets. If the boundary of the jth connected set is denoted by V_j, the size of concavity V is defined by*

$$c(V) = \max_{j=1,\ldots,m} \frac{d(V_j \bigcap \partial O, V_j \bigcap \partial I)}{l(V_j \bigcap \partial O)},$$

where $d(X,Y) = \max_{x \in X} \min_{y \in Y} \|x - y\|$ and $l(L)$ is the length of a line segment L.

The concavity measure ranges in $[0, 0.5]$ and its largeness implies that V is not convex. If the measure is applied to the UECS, the stopping criterion of the UECS is defined by comparing the measure with a threshold ρ, i.e. continue erosion if $c(V) > \rho$ or stop otherwise.

The choice of ρ determines the noise-robustness degree of the UECS. If ρ is large, the UECS will be more robust to boundary protrusion and intrusion by noise but it will lose its separation capability. Conversely, if ρ is too small, the UECS will be less robust but it is more capable in separating overlaps. Empirically, the author observed that ρ in between 0.2 and 0.3 worked well with real micrographs. An exemplary result from the UECS ($\rho = 0.2$) is presented in Figure 22-(d), where the markers are depicted as the white regions inside the nanoparticles. However, the choice of ρ could be dependent on the field of applications. The results of the UECS with different choices of ρ under different degrees of particle overlaps and different noise levels are presented in Appendix B. This will be a good guideline for practitioners to choose the ρ appropriate for their own applications.

In summary, in the stage of image segmentation, the author proposed a modified ultimate erosion process, UECS, for noise-robust marker generation. The essence of UECS is to introduce a new stopping criterion to the existing UE process. The author showed that its separation capability is the same as that of the UE in the noise-free cases, and empirically showed that the UECS is less prone to over-segmentation where the particle images are corrupted by noises.

IV.2.2. Extraction and Association of Contour Evidences

Once the markers are obtained, most existing image segmentation methods grow the markers by repeated applications of geodesic dilations to the markers, and the growth of a marker usually stops when it collides with the growth of other markers. For instance, marker-controlled watershed follows this approach (Dougherty, 1994). The contours of the grown-up markers are used as the contour evidences to infer the complete contours of objects. In such an approach, the growing process can be regarded as an implicit way to get the contour evidences for the final inference of the complete contours. Different from the marker growth approach, the proposed approach retrieves the contour evidences in two steps; first, extract all the edge pixels from an image, and secondly, associate the edge pixels with each individual marker according to a relevance measure. This edge-to-marker association is used as our contour evidences. This section focuses on defining the relevance measure for this evidence association.

Suppose that there are n markers from the UECS, denoted by $\{T_1, T_2, ..., T_n\}$, where T_i is the marker of C_i and it is represented by a set of point coordinates. There are also m edge pixel coordinates detected by an arbitrary edge detection method, which is denoted by $E = \{e_1, \ldots, e_m\}$. Note that the edges are the locations where image intensities abruptly change and they have been used as evidences of object's contours previously, e.g. in Gonzalez and Woods (2002). In order to measure the relevance of e_i to T_j, denoted by $\mathrm{rel}(e_i, T_i)$, the author defines a compound measure rather than a simple measure like distance alone. A component composing the compound measure is a distance from e_i to T_j, the same as what is used in the marker-growing approach. The author defines the distance measure in order to exclude the edge points that locate closely to an irrelevant marker by chance. The distance is defined with respect to I (the same I used in the previous section) as

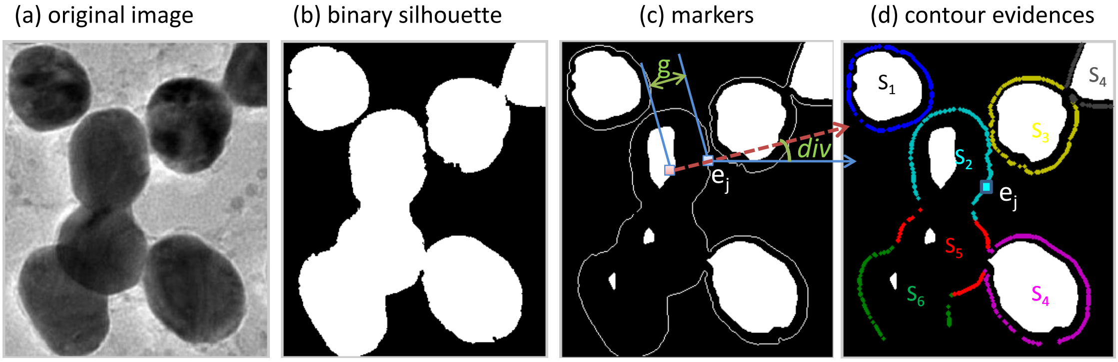
$$g(e_j, T_i) = \min_{x \in T_i} g_j(x), \tag{13}$$

where $g_j(x)$ is the Euclidean distance $|e_j - x|$ if the line from e_j to x entirely resides within I and ∞ when any portion of the line is outside I. By the convexity of C_i, if e_j is a substance of the C_i's contour, the line from $x \in T_i$ to e_j must be in C_i and also in I. Such treatment helps avoid over-emphasizing the markers irrelevant to but close to e_j.

The other component in the compound measure is the divergence index of e_j from T_i, which compares the direction of intensity gradient at e_j with the direction of line from $x \in T_i$ to e_j. Technically, it is expressed as a cosine function:

$$div(e_j, T_i) = \min_{x \in T_i} \frac{\vec{g}(e_j) \cdot \vec{l}(x, e_j)}{\|\vec{g}(e_j)\| \|\vec{l}(x, e_j)\|},$$

where $\vec{g}(e_j)$ is the direction of intensity gradient at e_j, and $\vec{l}(x, e_j)$ is the direction of line from $x \in T_i$ to e_j. The use of the divergence index is motivated by how electron micrography works. In a typical electron micrograph, the regions occupied by nanoparticles have lower image intensities than the background. For this reason,



(a) original image (b) binary silhouette (c) markers (d) contour evidences

Fig. 26. Classification of edge pixels by the relevance to markers. (c) illustrates how to compute the geodesic distance (g) and the divergence index (div) between edge pixel e_j and marker T_i. (d) is the classification result, where different classes are labeled as different colors of edge pixels.

if e_j is a substance of the C_i's contour, the gradient at e_j diverges from T_i. Since C_i is convex, the gradient direction is very close to the vector direction from T_i to e_j, i.e. the cosine of the angle between the two directions is close to being maximized. In Figure 26-(c), the solid-line arrow outbound from e_j is $\vec{g}(e_j)$, the (image intensity) gradient vector at e_j, and the dotted-line arrow represents $\vec{l}(x, e_j)$, the straight line from T_i to e_j. The divergence index is simply the cosine of the angle between the two vectors.

Compounding $g(e_j, T_i)$ and $div(e_j, T_i)$, the author defines the relevance measure of e_j to T_i as

$$\mathrm{rel}(e_j, T_i) = \frac{1}{1 + g(e_j, T_i)/nIter} + \frac{div(e_j, T_i) + 1}{2}, \tag{14}$$

where each of the two terms is normalized to $(0, 1]$ and $nIter$ is the number of erosion iterations in the UECS. If $i = \arg\max_k \mathrm{rel}(e_j, T_k)$, e_j becomes an element of the contour evidences for C_i.

IV.3. Gaussian Mixture Model for Contour Inference and Shape Classification

Suppose that there are a set of m_i edge points as the contour evidences for C_i, which are denoted by $\{\mathbf{e}_{i1}, \mathbf{e}_{i2}, \ldots, \mathbf{e}_{im_i}\}$, where $\mathbf{e}_{ij}$ is a 2×1 vector since our images are two dimensional. The markers identified in Section IV.2 are used to locate the C_i's but they will not be used explicitly in the subsequent inference.

The objective here is to infer a contour, fitted to the evidences and regulated by the prior shape knowledge (known reference shapes). One difficulty is that a contour can have several possible (convex) shapes, whereas most previous research only dealt with a single predetermined reference shape. To deal with multiple reference shapes, the author proposes an approach that performs simultaneously the shape classification and contour inference.

A contour for C_i is assumed to be a uniform periodic B-spline curve with order d and p control points; for $t \in [0, 1]$,

$$\boldsymbol{f}_i(t) = \sum_{h=0}^{p-1} \phi_{h,d}(t) \boldsymbol{p}_{i,h}, \tag{15}$$

where t is a parameter to identify a point on the curve, $\phi_{h,d}(t)$ is the hth periodic B-spline bending function, and $\boldsymbol{p}_{i,h} \in \mathbb{R}^2$ is the hth control point.

Suppose that $\mathbf{e}_{ij}$ is a noisy observation of $\boldsymbol{f}_i(t)$ at a B-spline parameter value t_{ij}, i.e.

$$\mathbf{e}_{ij} = \boldsymbol{f}_i(t_{ij}) + \boldsymbol{\epsilon}_{ij}, \quad \boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I}_2),$$

where the parameter value t_{ij} is unknown and it needs to be estimated. The problem assigning t_{ij} to each data point $\mathbf{e}_{ij}$ is called the data parameterization problem in the literature (Hoschek, 1988; Goshtasby, 2000; Saux and Daniel, 2003). For the time being, t_{ij} is assumed known. The m_i contour evidences collectively is denoted by a $2m_i \times 1$ vector $\mathbf{x}_i$, which is formed by binding $\mathbf{e}_{ij}$ in rowwise. The likelihood of $\boldsymbol{f}_i(t)$

given $\mathbf{x}_i$, or equivalently, the likelihood of $\boldsymbol{p}_{i,h}$'s given $\mathbf{x}_i$, is then

$$P(\mathbf{x}_i|\boldsymbol{p}_i, \sigma^2) = \mathcal{N}(\mathbf{x}_i; \boldsymbol{\Phi}_i\boldsymbol{p}_i, \sigma^2\boldsymbol{I}_{2m_i}),$$

where $\boldsymbol{\Phi}_i$ is a $2m_i \times 2p$ matrix with $\phi_{r,d}(t_{ij})\boldsymbol{I}_2$ as its (j,h)th 2×2 submatrix and $\boldsymbol{p}_i$ is a $2p \times 1$ vector binding $\boldsymbol{p}_{i,h}$ in rowwise.

The $\boldsymbol{p}_i$ is affected by shape information. It also varies with the pose parameters such as scaling, shifting and rotation of shapes. Hence, before constraining $\boldsymbol{p}_i$ with shape information, the pose parameters are separated from $\boldsymbol{p}_i$. Given the scale parameter s, rotation angle θ and horizontal-vertical shifts $\boldsymbol{c}$, the model for $\boldsymbol{p}_{i,h}$ is

$$\boldsymbol{p}_{i,h} = \frac{1}{s}R_\theta\tilde{\boldsymbol{p}}_{i,h} + \boldsymbol{c},$$

where $\tilde{\boldsymbol{p}}_{i,h}$ is the normalized shape feature independent of the pose parameters, $\boldsymbol{c}$ is a two-dimensional column vector of horizontal and vertical shifts, and R_θ is a transformation matrix for a rotation by θ in counter-clock wise. The model for the whole feature $\boldsymbol{p}_i$ is

$$\boldsymbol{p}_i = \frac{1}{s}\boldsymbol{Q}_\theta\tilde{\boldsymbol{p}}_i + \boldsymbol{H}\boldsymbol{c}, \tag{16}$$

where $\boldsymbol{Q}_\theta$ is a Kronecker product of the $p\times p$ identity matrix with R_θ and $\boldsymbol{H} := \mathbf{1}_p\otimes\boldsymbol{I}_2$ is a Kronecker product of the $p \times 1$ vector of ones with a 2×2 identity matrix.

The normalized shape feature $\tilde{\boldsymbol{p}}_i$ should be constrained by the prior shape knowledge, which is summarized as follows. The shapes of nanoparticles are grouped into K possible shapes. If particle i belongs to the kth shape group, $\tilde{\boldsymbol{p}}_i$ follows a Gaussian distribution with $\boldsymbol{\mu}_k$ as its mean and $\boldsymbol{\Sigma}_k$ as its variance-covariance matrix. The hidden group membership vectors $\boldsymbol{g}_i = (g_{i1}, \ldots, g_{iK})^t$ are defined such that g_{ik} is equal to one, if particle i belongs to shape group k, and zero otherwise. Then, the prior distribution on $\tilde{\boldsymbol{p}}_i$ given $\boldsymbol{g}_i$ is then

$$P(\tilde{\boldsymbol{p}}_i|\boldsymbol{g}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{k=1}^{K} \mathcal{N}(\tilde{\boldsymbol{p}}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{g_{ik}},$$

where $\boldsymbol{\mu}$ is a set of $\boldsymbol{\mu}_k$'s and $\boldsymbol{\Sigma}$ is a set of $\boldsymbol{\Sigma}_k$'s. Since $\boldsymbol{p}_i$ is the linear transformation of $\tilde{\boldsymbol{p}}_i$, $\boldsymbol{p}_i$ is also characterized as Gaussian with parameters depending on the pose parameters. The pose parameters might depend on the shape group that particle i belongs to. Hence, they are defined separately for each shape group by s_{ik}, θ_{ik} and $\boldsymbol{c}_{ik}$, and they are denoted collectively for all k's by $\boldsymbol{s}_i$, $\boldsymbol{\theta}_i$ and $\boldsymbol{c}_i$. Using (16), $\boldsymbol{p}_i$ is distributed as

$$\begin{aligned} &P(\boldsymbol{p}_i|\boldsymbol{g}_i, \boldsymbol{s}_i, \boldsymbol{\theta}_i, \boldsymbol{c}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &= \prod_{k=1}^{K} \mathcal{N}\left(\boldsymbol{p}_i; \boldsymbol{A}_{ik}\boldsymbol{\mu}_k + \boldsymbol{H}\boldsymbol{c}_{ik}, \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t\right)^{g_{ik}}, \end{aligned} \tag{17}$$

where $\boldsymbol{A}_{ik} = \frac{1}{s_{ik}}\boldsymbol{Q}_{\theta_{ik}}$. A multinomial distribution is put on the hidden matrix $\boldsymbol{g}_i$ as its prior distribution, i.e.,

$$P(\boldsymbol{g}_i|\boldsymbol{\alpha}) = \prod_{k=1}^{K} \alpha_k^{g_{ik}}, \tag{18}$$

where $\sum_k \alpha_k = 1$ and $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_K)^t$.

The ultimate goal of this section is to obtain the contours of convex shapes C_i's and to determine which shape group C_i belongs to. This problem is corresponding to estimating the hidden variables $\boldsymbol{Z} = \{\boldsymbol{p}_i, \boldsymbol{g}_i; i = 1, \ldots, N\}$. However, the hidden variables depend on the unknown parameters $\boldsymbol{\Theta} = \{\sigma^2, \boldsymbol{\alpha}, \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}, \{\boldsymbol{s}_i, \boldsymbol{\theta}_i, \boldsymbol{c}_i\}\}$. The unknown parameters are estimated by maximizing the following marginal likelihood of observing contour evidences $\boldsymbol{X} = \{\mathrm{x}_i; i = 1, \ldots, N\}$ with respect to $\boldsymbol{\Theta}$,

$$P(\boldsymbol{X}|\boldsymbol{\Theta}) = \int_{\boldsymbol{Z}} P(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\Theta}) d\boldsymbol{Z}. \tag{19}$$

Given the estimated $\widehat{\boldsymbol{\Theta}}$, $\boldsymbol{Z}$ is estimated by taking its posterior mode, a maximizer of the following posterior distribution,

$$P(\boldsymbol{Z}|\widehat{\boldsymbol{\Theta}}, \boldsymbol{X}) = \frac{P(\boldsymbol{X}, \boldsymbol{Z}|\widehat{\boldsymbol{\Theta}})}{P(\boldsymbol{X}|\widehat{\boldsymbol{\Theta}})}. \tag{20}$$

Conceptually, solving (20) involves two complicated optimization problems, which are not easy to handle. In practice, (20) is solved through an iterative solver, the expectation conditional maximization (`ECM`), which has been proved to converge to a local maximum (Meng and Rubin, 1993).

IV.3.1. Expectation Maximization via the `ECM` algorithm

When $\mathbf{\Theta}$ is considered as the unknown parameters, the complete likelihood of the parameters given contour evidences $\mathbf{X}$ and hidden variables $\mathbf{Z}$ is as follows:

$$\begin{aligned} P(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta}) = \prod_{i=1}^{n}\prod_{k=1}^{K}[&\alpha_k \mathcal{N}(\boldsymbol{p}_i; \boldsymbol{A}_{ik}\boldsymbol{\mu}_k + \boldsymbol{H}\boldsymbol{c}_{ik}, \boldsymbol{A}_{ik}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^t) \\ &\mathcal{N}(\mathbf{x}_i; \mathbf{\Phi}_i\boldsymbol{p}_i, \sigma^2\boldsymbol{I}_{2m_i})]^{g_{ik}}. \end{aligned}$$

At iteration t, the expectation maximization (EM) algorithm first computes the expected value of the complete log-likelihood function with respect to the posterior distribution, $P(\mathbf{Z}|\mathbf{X}, \mathbf{\Theta})$ (E-step). The expected log likelihood is

$$\begin{aligned} &E_{\mathbf{Z}}[\log P(\mathbf{X}, \mathbf{Z}|\mathbf{\Theta})] \\ &\propto \sum_{i=1}^{N}\sum_{k=1}^{K}\beta_{ik}[2\log\alpha_k - \log\det(\mathbf{\Sigma}_k) + 4p\log s_{ik} \\ &- (\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik})^t\mathbf{\Sigma}_k^{-1}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik}) \\ &+ 2s_{ik}\boldsymbol{m}_{ik}^t\boldsymbol{Q}_{\theta_{ik}}\mathbf{\Sigma}_k^{-1}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik}) \\ &- s_{ik}^2\mathrm{trace}(\boldsymbol{Q}_{\theta_{ik}}\mathbf{\Sigma}_k^{-1}\boldsymbol{Q}_{\theta_{ik}}^t\mathbf{\Delta}_{ik}) - 2m_i\log\sigma^2 \\ &- \sigma^{-2}\boldsymbol{x}_i^t\boldsymbol{x}_i + 2\sigma^{-2}\boldsymbol{\gamma}_{ik}^t\mathbf{\Phi}_i^t\boldsymbol{x}_i \\ &- \sigma^{-2}\mathrm{trace}(\mathbf{\Phi}_i^t\mathbf{\Phi}_i\mathbf{\Delta}_{ik})], \end{aligned} \tag{21}$$

where $\beta_{ik} = \alpha_k q_{ik}$, $\boldsymbol{\Delta}_{ik} = \boldsymbol{S}_{ik} + \boldsymbol{m}_{ik}\boldsymbol{m}_{ik}^t$,

$$\begin{aligned}
q_{ik} &= \mathcal{N}\left(\mathbf{x}_i; \boldsymbol{\Phi}_i \boldsymbol{A}_{ik}\boldsymbol{\mu}_k + \boldsymbol{H}\boldsymbol{c}_{ik}, \sigma^2 \boldsymbol{I}_{2m_i} + \boldsymbol{\Phi}_i \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k \boldsymbol{A}_{ik}^t \boldsymbol{\Phi}_i^t\right), \\
\boldsymbol{m}_{ik} &= \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k \boldsymbol{A}_{ik}^t \boldsymbol{\Phi}_i^t \left(\sigma^2 \boldsymbol{I}_{2m_i} + \boldsymbol{\Phi}_i \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k \boldsymbol{A}_{ik}^t \boldsymbol{\Phi}_i^t\right)^{-1} \\
&\qquad (\boldsymbol{x}_i - \boldsymbol{\Phi}_i \boldsymbol{A}_{ik}\boldsymbol{\mu}_k - \boldsymbol{H}\boldsymbol{c}_{ik}) + \boldsymbol{A}_{ik}\boldsymbol{\mu}_k + \boldsymbol{H}\boldsymbol{c}_{ik}, \\
\boldsymbol{S}_{ik} &= (\boldsymbol{A}_{ik}^{-t}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{A}_{ik}^{-1} + \sigma^{-2}\boldsymbol{\Phi}_i^t \boldsymbol{\Phi}_i)^{-1}.
\end{aligned}$$

A lengthy derivation of the expectation is moved to Appendix C. In the M-step, the expectation in (21) is maximized with respect to $\boldsymbol{\Theta}$. The first order necessary condition (FONC) with respect to α_k, along with the constraint $\sum_k \alpha_k = 1$, gives us the local optimum for α_k,

$$\alpha_k = \frac{\sum_{i=1}^N \beta_{ik}}{\sum_{i=1}^N \sum_{k'=1}^K \beta_{ik'}}.$$

The local optimum for σ^2 also comes directly from the FONC,

$$\sigma^2 = \frac{\sum_{\forall i,k} \beta_{ik}\left[\boldsymbol{x}_i^t \boldsymbol{x}_i - 2\boldsymbol{m}_{ik}^t \boldsymbol{\Phi}_i^t \boldsymbol{x}_i + \mathrm{trace}(\boldsymbol{\Phi}_i^t \boldsymbol{\Phi}_i \boldsymbol{\Delta}_{ik})\right]}{2\sum_{i=1}^N (m_i \sum_{k=1}^K \beta_{ik})}.$$

The author cannot derive the closed form expressions of the local optima for $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$, θ_{ik}, s_{ik} and $\boldsymbol{c}_{ik}$, since their FONCs are entangled with one another in complicated forms. Instead, the M-step can be solved iteratively by the Newton Raphson, but the author wants to avoid expensive iterations as well. There are two other possible options to proceed the M-step without iterations; the first one is to improve the expected log-likelihood (21) rather than to maximize it for every M-step, resulting in a GEM algorithm (Dempster *et al.*, 1977), and another one is to use the `ECM` algorithm (Meng and Rubin, 1993). The first option does not in general converge appropriately, but the second option does. For this reason, the author takes the second option.

The ECM algorithm partitions $\boldsymbol{\Theta}$ into L subgroups and solves L optimizations, where each optimization maximizes (21) with respect to one subgroup of $\boldsymbol{\Theta}$, provided that the other groups remained to be their previous values. In this paper, $L = 5$ with five subgroups of parameters $\{\boldsymbol{\mu}_k\}$, $\{\boldsymbol{\Sigma}_k\}$, $\{\theta_{ik}\}$, $\{s_{ik}\}$ and $\{\boldsymbol{c}_{ik}\}$. From the FONCs with respect to each subgroup, we have the following solutions for the M-step of the ECM algorithm

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^N \beta_{ik} s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t (\boldsymbol{m}_{ik} - \boldsymbol{H}\boldsymbol{c}_{ik})}{\sum_{i=1}^N \beta_{ik}},$$

$$\begin{aligned}
\boldsymbol{\Sigma}_k &= \frac{\sum_{i=1}^N \beta_{ik}(\boldsymbol{\mu}_{ik} + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik})(\boldsymbol{\mu}_{ik} + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik})^t}{\sum_{i=1}^N \beta_{ik}} \\
&+ \frac{s_{ik}^2 \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{\Delta}_{ik} \boldsymbol{Q}_{\theta_{ik}} - s_{ik}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik})\boldsymbol{m}_{ik}^t \boldsymbol{Q}_{\theta_{ik}}}{\sum_{i=1}^N \beta_{ik}} \\
&- \frac{s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{m}_{ik}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik})^t}{\sum_{i=1}^N \beta_{ik}}, \\
\boldsymbol{c}_{ik} &= (\boldsymbol{H}^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H})^{-1} \boldsymbol{H}^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \left(\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{m}_{ik} - \frac{\boldsymbol{\mu}_k}{s_{ik}} \right), \\
s_{ik} &= \frac{v + \sqrt{v^2 + 8p\beta_{ik}u}}{2u},
\end{aligned}$$

where

$$\begin{aligned}
u &= \mathrm{trace}[\boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{Q}_{\theta_{ik}}^t (\boldsymbol{\Delta}_{ik} + \boldsymbol{H}\boldsymbol{c}_{ik}\boldsymbol{c}_{ik}^t \boldsymbol{H}^t - 2\boldsymbol{H}\boldsymbol{c}_{ik}\boldsymbol{m}_{ik}^t)] \\
v &= (\boldsymbol{m}_{ik} - \boldsymbol{H}\boldsymbol{c}_{ik})^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k.
\end{aligned}$$

The solution for θ_{ik} is more complicated. The equation governing $x = \sin(\theta_{ik})$ is

$$\begin{aligned}
4(a^2 + b^2)x^4 + 8(bd - ac)dx^3 + 4(c^2 + d^2 - a^2 - b^2)x^2 \\
+ 4(ac - 2bd)x + a^2 - 4d^2 = 0,
\end{aligned}$$

where

$$\begin{aligned}
a =& s_{ik}\mathrm{trace}((\boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p*}^t + \boldsymbol{I}_{p*} \boldsymbol{\Sigma}_k^{-1})(\boldsymbol{\Delta}_{ik} + \boldsymbol{H}\boldsymbol{c}_{ik}\boldsymbol{c}_{ik}^t \boldsymbol{H}^t)), \\
b =& s_{ik}\mathrm{trace}((\boldsymbol{\Sigma}_k^{-1} - \boldsymbol{I}_{p*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p*}^t)(\boldsymbol{\Delta}_{ik} + \boldsymbol{H}\boldsymbol{c}_{ik}\boldsymbol{c}_{ik}^t \boldsymbol{H}^t)) \\
&- 2s_{ik}\mathrm{trace}((\boldsymbol{\Sigma}_k^{-1} - \boldsymbol{I}_{p*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p*}^t)\boldsymbol{H}\boldsymbol{c}_{ik}\boldsymbol{m}_{ik})), \\
c =& (\boldsymbol{m}_{ik} - \boldsymbol{H}\boldsymbol{c}_{ik})^t \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k, \\
d =& (\boldsymbol{m}_{ik} - \boldsymbol{H}\boldsymbol{c}_{ik})^t \boldsymbol{I}_{p*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k,
\end{aligned}$$

and $\boldsymbol{I}_{p*}$ is an anti-symmetric matrix defined as

$$\boldsymbol{I}_{p*} = \boldsymbol{I}_p \otimes \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

The solution of the above quartic equation can be obtained by means of a method discovered by Ferrari (Tignol, 2001, pages 22–24).

Once a convergence is attained from iterations of the E-steps and the M-steps through the ECM, the posterior distributions for $\boldsymbol{g}_i$ and $\boldsymbol{p}_i$ given the converged parameters $\boldsymbol{\Theta}$ and evidences $\mathbf{X}$ are obtained by

$$\begin{aligned} &P(g_{ik} = 1|\mathbf{X}, \boldsymbol{\Theta}) \propto \beta_{ik} \\ &P(\boldsymbol{p}_i|\mathbf{X}, \boldsymbol{\Theta}) \propto \sum_{k=1}^{K} \beta_{ik} \mathcal{N}(\boldsymbol{p}_i; \boldsymbol{m}_{ik}, \boldsymbol{S}_{ik}). \end{aligned}$$

Accordingly, the posterior mode of $\boldsymbol{g}_i$ is $\widehat{g}_{ik} = 1$ if $k = \arg\max_{k'} \beta_{ik'}$ and 0 otherwise. The posterior mode for $\boldsymbol{p}_i$ is given by

$$\widehat{\boldsymbol{p}}_i = \left(\sum_{k=1}^{K} \beta_{ik} \boldsymbol{S}_{ik}^{-1} \right)^{-1} \sum_{k=1}^{K} \beta_{ik} \boldsymbol{S}_{ik}^{-1} \boldsymbol{m}_{ik}.$$

Finally, the shape group of C_i is determined to be k if $\widehat{g}_{ik} = 1$ and reconstruct the contour by plugging $\widehat{\boldsymbol{p}}_i$ into (15).

IV.3.2. Approximate Data Parameterization

It has been assumed that the spline parameter value t_{ij} for evidence $\mathbf{e}_{ij}$ is known. In real problems, however, it is unknown and needs to be obtained. In the literature, assigning a parameter value t_{ij} to data point $\mathbf{e}_{ij}$ is called data parameterization, where several methods were available, including the chord length parameterization, the centripetal method (Epstein, 1976; Lee, 1989), or the intrinsic parameterization (Hoschek, 1988). Among these methods, the chord length parameterization is easy to use and efficient in computation. But, it requires the ordering information of the points to be parameterized, which is not available. On the other hand, the intrinsic parametrization is more general but it will add another family of parameters to the already large set of parameters under estimation in the ECM. In the end, the author chooses to base our approach on the chord length parameterization but use an approximation to get the ordering of the data points.

The basic idea of our approximate chord length parameterization is as follows: find a convex hull inscribing the evidences and then use the parameterization of points on the convex hull to get the approximate spline parameter t_{ij}. The detailed procedure is as follows: given a set of contour evidences for C_i, $\{\mathbf{e}_{i1}, \mathbf{e}_{i2}, \ldots, \mathbf{e}_{im_i}\}$,

1. Find a convex hull inscribing the contour evidences by the `Qhull` algorithm (Barber *et al.*, 1996).

2. Sequentially order all points on the convex hull in counter-clock wise (or clockwise) into $\mathbf{q}_0, \ldots, \mathbf{q}_L$ and then parameterize the points by the chord-length parameterization: the parameter t_l for $\mathbf{q}_l$ is assigned as

$$t_l = \frac{\sum_{s=1}^{l} |\mathbf{q}_s - \mathbf{q}_{s-1}|}{\sum_{s=1}^{L} |\mathbf{q}_s - \mathbf{q}_{s-1}|}$$

3. In order to get t_{ij}, find the point closest to $\mathbf{e}_{ij}$ among the points in the convex hull, say $\mathbf{q}_s$. Set $t_{ij} = t_s$.

The main advantage of this parameterization is its simplicity and computationally efficiency. In addition, the parameterization is not effected by the noisy contour evidences which locate inside an nanoparticle. When the noisy evidences locate outside an nanoparticle, the approximate data parameterization may be distorted. This problem can be addressed if the convex hull inscribing the majority of points except for a few outliers is taken.

IV.4. Results and Discussion

The results of segmentation and contour inference of the twelve micrographs are presented in Figures 27 through 30. Each figure has four columns. The first column is the original micrograph. Its binary silhouette is obtained by applying the alternative sequence filtering (Gonzalez and Woods, 2002), followed by the Otsu's optimum global thresholding (Otsu, 1979). The UECS proposed in Subsection IV.2.1 was applied to the binary silhouette for obtaining the second column, where a white-colored connected region implies one marker. Contour evidences (pixels at the boundaries of particles) have first been extracted by the Canny's edge detection method (Canny, 1986), and then they were associated with the markers by using the procedure in Subsection IV.2.2. After the association, the algorithm filters out some noise edge outliers based on the mean and standard deviation of g defined in equation (13). In the third column of the figures, the association to different markers is illustrated by different colors of the contour evidences. The last column shows the final result from the contour inference proposed in Section IV.3.

The proposed UECS correctly identified one marker per particle for most of the cases from Figure 27 through 29. Figure 30 has severe overlaps among the nanoparticles, and as a result, the UECS sometimes identified one marker for multiple nanoparticles. The association between the markers and the edge pixels look reasonable, although some noise edge pixels have been classified as valid contour evidences. The inference results were matched well with the original image. In some cases where the contour evidences are not sufficient, especially for the nanoparticles cropped by image borders, the shapes of the nanoparticles cannot be inferred correctly, and the recovered contours do not look good. The author wants to note that most of those cases cannot be easily handled even by human vision.

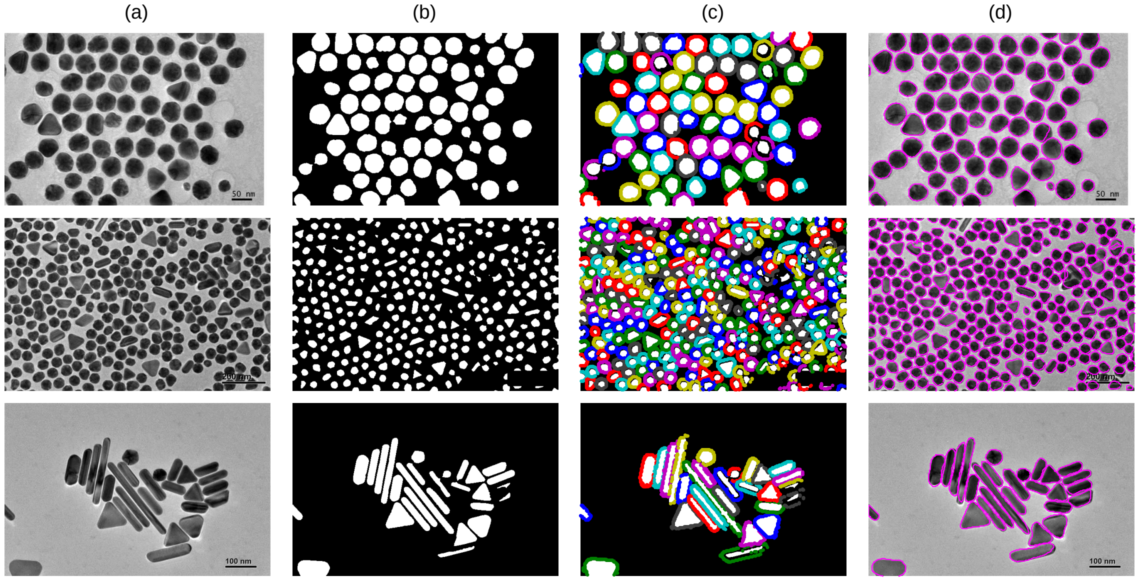


Fig. 27. Results from low-degree overlapping cases. (a) original images, (b) markers from the UECS, (c) contour evidences (Section IV.2.2) , (d) final contours by the ECM

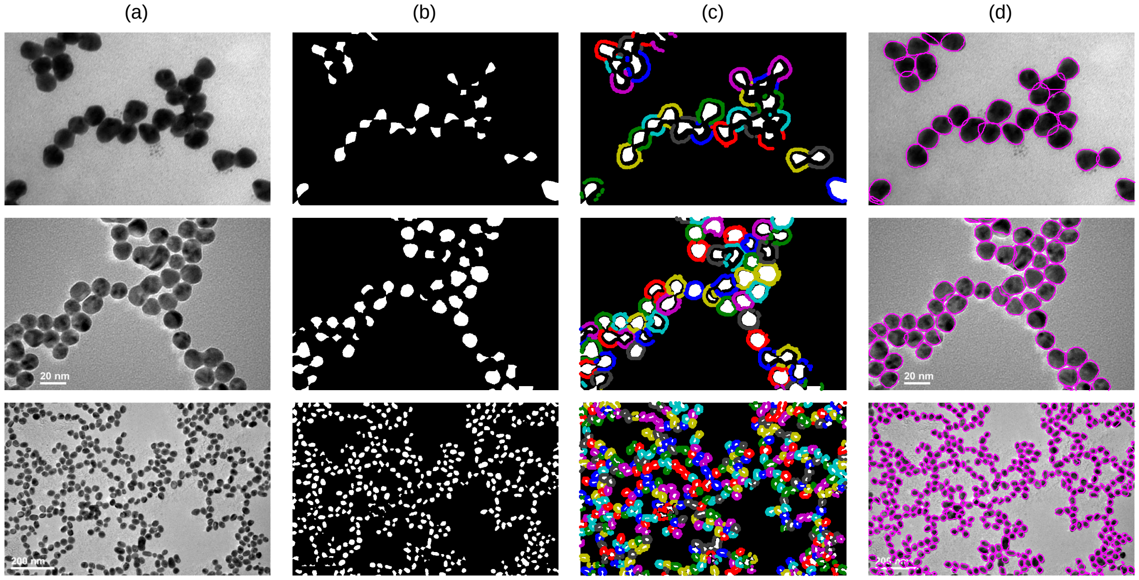


Fig. 28. Results from medium-degree overlapping cases.

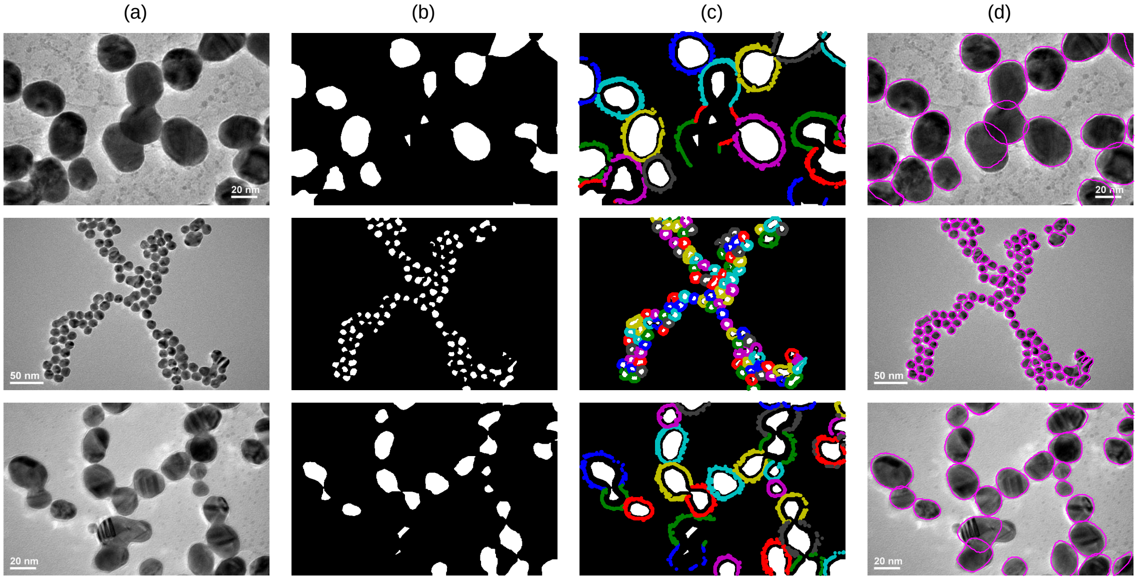


Fig. 29. Results from medium-degree overlapping cases (continued).

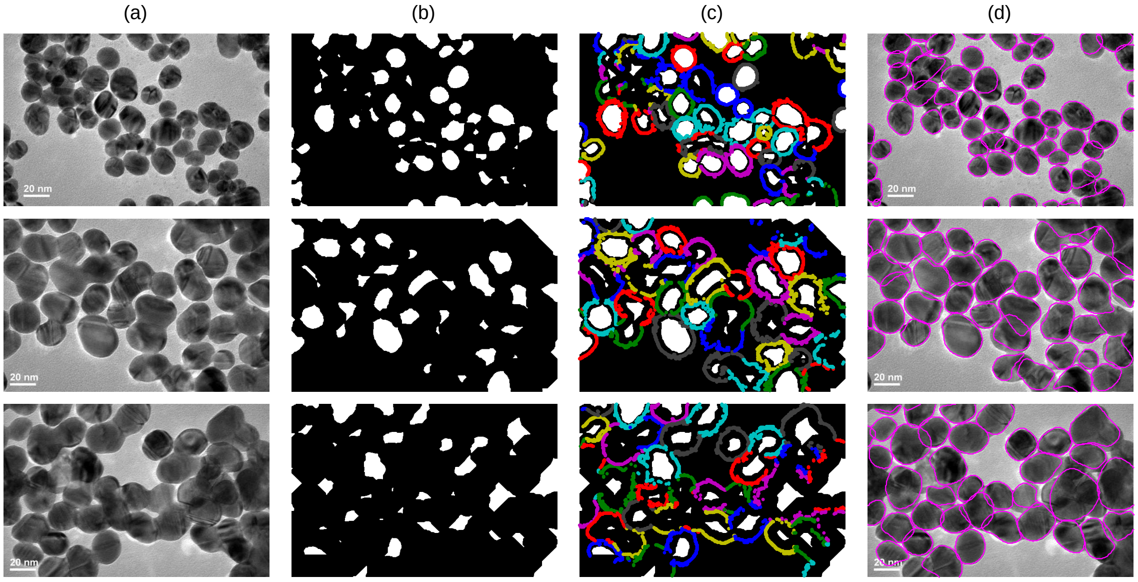


Fig. 30. Results from high-degree overlapping cases.

The accuracy of the proposed method is quantitatively compared with the divide-and-conquer approach and the four state-of-the-art methods introduced in Section II.3: marker-controller watershed segmentation (Malpica *et al.*, 1997, MCWS), normalized-cut (Shi and Malik, 2000, N-Cut), multi-phase active contour (Vese and Chan, 2002, MPAC), sliding band filter (Quelhas *et al.*, 2010, SBF). The results of `ImageJ` were not included for the comparison, because `ImageJ` performed far worse than the other methods.

For each of the twelve micrographs, the author manually counted the total number of nanoparticles and the number of the particles correctly inferred by each of the methods in comparison. The numbers are tabularized in Table 4. The convex shape analysis method was the best performer in most cases (only one exception is the first row of Figure 6). The N-cut and the MPAC method did not perform well, because their image segmentation are guided by image intensities but overlapping particles have too similar image intensities to be separated. It is not easy to guide these methods by multiple reference shapes due to the prohibitive computational demand from the current technology. The SBF method performed comparably well but it struggled with the cases that the particle sizes vary drastically. The third micrograph in Figure 27 corresponds to such a case.

Figure 31 presents one exemplary result of the contour inference performed by the convex shape analysis and the four competing methods. The under-segmentation could be clearly observed in the MPAC and the N-Cut, and over-segmentation was observed in the MCWS due to its sensitivity to noise. The contour estimation by the SBF looks rough, but this roughness can be smoothed by curve fitting. SBF also missed a few particles.

The author recorded the total computation times spent by the methods for the same micrograph used in Figure 31. The SBF spent the longest time, 541 seconds;

Table 4. Comparison of performances on nanoparticle identification. CSA: convex shape analysis, DCA: divide-and-conquer approach, N-Cut: normalized cut method, MPAC: multiphase active contour, MCWS: marker-controlled watershed, SBF: sliding band filter.

Samples	Degree of overlap	# of particles	# of the correctly identified ones					
			CSA	DCA	N-Cut	MPAC	MCWS	SBF
Figure 27, row 1	Low	64	64	64	62	57	64	57
Figure 27, row 2	Low	259	251	252	236	151	235	167
Figure 27, row 3	Low	26	26	24	20	17	20	1
Figure 28, row 1	Medium	28	26	7	6	2	20	28
Figure 28, row 2	Medium	52	48	29	29	22	39	43
Figure 28, row 3	Medium	396	374	363	235	84	358	199
Figure 29, row 1	Medium	19	17	7	10	8	6	15
Figure 29, row 2	Medium	108	103	24	56	44	70	99
Figure 29, row 3	Medium	29	25	14	12	13	14	19
Figure 30, row 1	High	63	54	21	31	12	34	42
Figure 30, row 2	High	44	34	10	23	11	23	28
Figure 30, row 3	High	45	33	0	20	6	20	25

the N-cut spent 256 seconds, and the MPAC spent 273 seconds. The MCWS took less than one second. The DCA method took 14 seconds, and the CSA method took 72 seconds, where 28 seconds are for the UECS and association of contour evidences, and the rest of time is for contour inference and shape classification. The convex shape analysis method is not the fastest among the five methods. However, please note that the convex shape analysis method performed shape inference and classification along side with segmentation, while the other four methods performed either segmentation/inference (MPAC) or only segmentation (the other three methods).

IV.4.1. Results of Shape Classification

The author chose four micrographs among the twelve micrographs to evaluate the accuracy of shape classification. The four figures have various types of particle shapes. The classification results are presented by labeling the nanoparticles with one char-

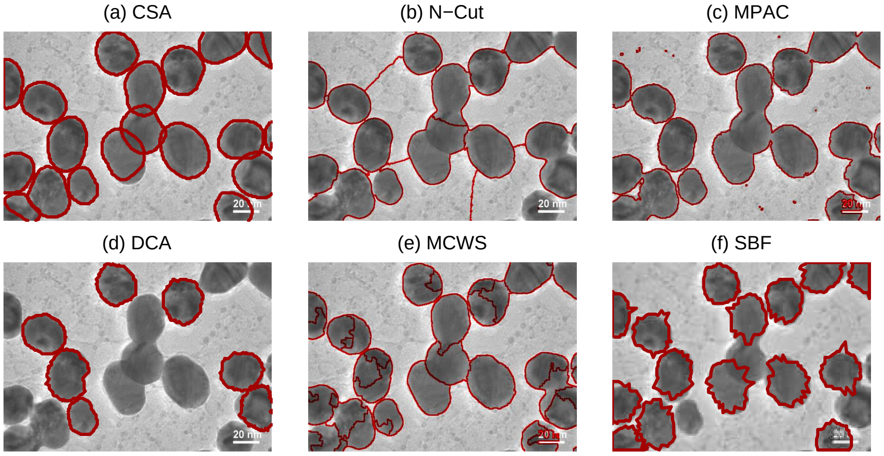


Fig. 31. Results of contour inference. The red lines correspond to the contours identified by the convex shape analysis method and the four state-of-the-art methods.

acter symbol representing shape classes; 't' = triangle, 'b' = rectangle, 'c' = circle and 'r' = rod. Please refer to Figure 32.

The automated classification outcomes is compared with what human would classify the shapes. In the top-left figure, the result was accurate except for two miss-classifications; the convex shape analysis method classified a triangle as a circle and classified a circle as a triangle. Such miss-classification are also observed in a few other cases at the bottom-left figure and the bottom-right figure. The circle-to-triangle misclassification is mostly caused by insufficient contour evidences. The other type of miss-classification was caused by a faulty data parameterization for the spline curves in the ECM. Looking for a more capable data parameterization is certainly desirable but does not appear to be a simple task. The author leaves out this issue to our future research. Overall, the author believes that our automated method did the shape classification reasonably well.

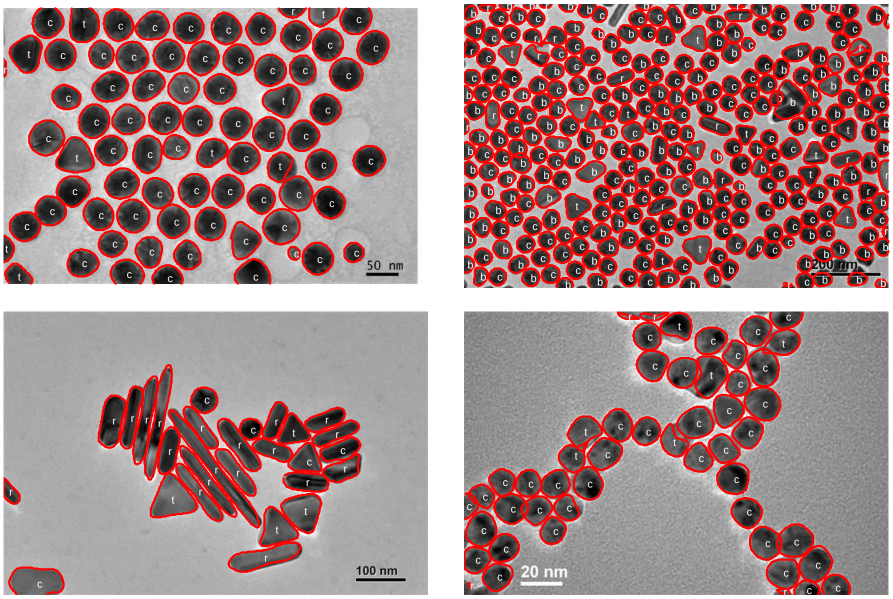


Fig. 32. Results of shape classification. Each particle's shape is labeled as: t = triangle, b = rectangle, c = circle and r = rod.

IV.4.2. Discussion

In this chapter, the author proposed a convex shape analysis approach to tackle the contour inference and shape classification problems for automating the morphology analysis of overlapping nanoparticles in electron micrographs. The unique contributions of this approach are:

1. to propose a modified ultimate erosion process with an earlier stopping time (UECS) to separate overlapping convex objects,

2. to provide the justification on the use of UECS in terms of its separation capability for a chained-link cluster of convex objects,

3. to propose a new way to convert the segmented edge pixels into contour evidences by using a compound marker-to-edge relevance measure, and

4. to integrate the ECM with the UECS and the evidence association, which al-

lows us to solve a complicated image segmentation and recognition problem. Although the ECM-solution approach for shape classification and inference is not entirely new by itself (Tu *et al.*, 2008), our proposed model and solution procedure can solve the morphology analysis problem for a large number of overlapping nanoparticles, evolving an equally large number of contours with guidance of multiple reference shapes. To our best knowledge, there is no other method that has such capability.

The proposed method was tested with twelve electron micrographs of overlapping nanoparticles. The result shows that the proposed method performs better than the existing methods when both accuracy and computation efficiency are considered. It is not surprising that the proposed method performed better, because it is specially designed for the morphology analysis of nanoparticles in the sense that it is more capable of segmenting the chain-aggregate structure of overlapping convex-shaped nano-objects. The author would like to re-iterate here that the two morphological characteristics (chain-aggregated and convex-shaped) are particularly relevant to nanoparticle's formation, as told by the physical laws governing the formation process.

CHAPTER V

CONCLUSION

The objective of this dissertation research is to automate the morphology analysis of nanoparticles in electron micrographs. One technical challenge is that nanoparticles mostly overlap. The current practice is a labor-intensive manual counting process supported by general image processing softwares such as `ImageJ`. In order to design the automated morphology analysis method with enhanced accuracy, the author formulated the morphology analysis problem with three components: *image segmentation* for decomposing particle overlaps, *shape inference* for inferring particle morphology hidden by overlaps, and *shape classification* for grouping particles by their morphologies.

The three components of the morphology analysis are linked in the sense that solving each component problem could benefit, and be benefited by, the solving of the other component problems. In other words, solving the image segmentation and the shape classification requires the result of the shape inference, whereas the shape inference requires solving the shape classification. Thus, developing one unified formulation is ideal. However, current methodological development cannot solve such a unified formulation for electron micrographs having hundreds or thousands of nanoparticles.

In Chapter III, the author proposed the divide-and-conquer approach that solves the three components of the morphology analysis sequentially. The divide-and-conquer approach is especially designed for providing a computationally efficient solution of the morphology analysis, which is able to process electron micrographs having a large number of weakly overlapping nanoparticles. This divide-and-conquer approach is not capable of handling nanoparticles of high degrees of overlaps for two reasons. First, the image segmentation component is not very accurate under significant particle overlaps. Second, the shape classification component is solved by using a rough estimate of the gravity center of a nanoparticle, and the rough estimate is no longer accurate when the degree of particle overlaps is high.

In Chapter IV, the author proposed an alternative approach, the convex shape analysis method, which is more capable of handling particle overlaps than the divide-and-conquer approach. It improves morphology analysis in two ways. First, the image segmentation method in the divide-and-conquer approach suffers from over-segmentation in the cases where particle overlaps in a chain structure. The convex shape analysis method associates fragmented contour segments with one of the markers of nanoparticles to form a coherent piece of evidence for an individual particle. A challenge facing the new image segmentation method is about how to identify a unique marker for each particle. For that purpose, the author proposed a modified ultimate erosion process, which uses a new stopping criterion of the iterative erosion process. The erosion process is especially capable of identifying markers for the chain-linked structure of particle overlaps. Second, the new approach jointly solves shape inference and shape classification, which produces greater accuracy of classification and inference.

The application of the proposed methods to real electron micrographs showed that the two proposed methods were capable of extracting the morphology information, effectively decomposing the various degrees of particles overlaps, and satisfactorily inferring the morphologies hidden by particle overlaps.

The author believes that the proposed methods contribute to the following application fields.

1. **Efficient characterization of nanoparticles**: Automating the otherwise

time-consuming morphology analysis helps accelerate the characterization process of nanoparticles. The robustness of the proposed methods to a wide range of overlaps among the particles makes it possible to reduce the trial-and-error iterations for preparing samples with an appropriate concentration level.

Moreover, the high quality of morphology analysis from the proposed methods could help inspectors to design a testing procedure for quality assurance, or process designers to calibrate nanoparticle synthesis processes. Fast and accurate morphology analysis makes it possible to accumulate the large amount of data and build up statistics demanded for acceptance testing.

2. **Applications to general image segmentation**: This dissertation only applies the two proposed methods for morphology analysis of nanoparticles. However, the proposed methods also work for general image segmentation to separate overlapping convex objects. Such generality makes the proposed methods valuable alternatives to be used in, for example, cell segmentation problems. The research showed empirically that the proposed convex shape analysis method in Chapter IV outperforms the state-of-the-art methods currently available for cell segmentation.

The author suggests three directions for future research.

1. **Control charting with geometric data**: The final result of the morphology analysis problem includes the particle size and shape distributions that summarize the characteristic of geometrical shapes of nanoparticles. A natural question that quality engineers might ask is "*how do the geometric characteristics of nanoparticles from two different processes differ*". Particularly, it is important to answer whether the geometrical characteristic of the current process is significantly different from that of the normal process. Answering that

question requires the development of a hypothesis testing on samples of geometric shapes. A natural extension of this dissertation study is to develop this hypothesis testing with appropriate statistics.

2. **Characterizing the dynamic changes of particle morphology evolution**: The primary goal of particle characterization is to fine-tune the synthesis process of nanoparticles such that the synthesized nanoparticles have the desired morphology. The author believes that analyzing the changes of particle morphology during the synthesis process helps understand how nanoparticles grow during the synthesis, and this understanding in turn helps find control factors in the synthesis process. The analysis requires obtaining the time-series data of particle morphology, extracting the morphology information of nanoparticles for each time frame in the time series, and combining the series of the morphology-at-fixed-time into a dynamic morphology change model or a particle growth model. The technology to obtain the time-series data of particle growth is already available in literature (Zheng *et al.*, 2009). The proposed methods in this dissertation become valuable tools that can extract the morphology of nanoparticles at each snapshot when an electron micrograph is taken. Something not yet available but precisely needed, beyond this dissertation study, is a dynamic morphology change model in order to provide the understanding of how particles grow in their synthesis process.

3. **Characterizing the three-dimensional morphology**: Most available methods only conduct two-dimensional characterizations of particle's morphology, because the micrographs produced by conventional electron microscopes are the two-dimensional projections of the three-dimensional objects. The emerging technologies, 3D TEM and 3D SEM, takes a series of two-dimensional pro-

jection images from different observing angles and combines them to a three-dimensional image. One way to obtain the 3D morphology analysis is to stitch together the series of 2D results with the reconstruction of the 3D objects. Although the undertaking does not appear straightforward, the research required can build upon the work described in this dissertation.

REFERENCES

Ahmadi, T., Wang, Z., Green, T., Henglein, A. and El-Sayed, M. (1996) Shape-controlled synthesis of colloidal platinum nanoparticles. *Science*, **272**(5270), 1924–1926.

Al-amri, S., Kalyankar, N. and Khamitkar, S. (2010) Image segmentation by using edge detection. *International Journal on Computer Science and Engineering*, **2**(03), 804–807.

Alba, F., Crawley, G., Fatkin, J., Higgs, D. and Kippax, P. (1999) Acoustic spectroscopy as a technique for the particle sizing of high concentration colloids, emulsions and suspensions. *Colloids and Surfaces A: Physicochemical and Engineering Aspects*, **153**(1-3), 495–502.

Allard, L. and O'Keefe, M. (2004) Sub-Ångstrom metrology of resolution in aberration-corrected transmission electron microscopes using the A-OK standard test specimens. *Microscopy and Microanalysis*, **10**(S03), 66–67.

Bamford, P. and Lovell, B. (1998) Unsupervised cell nucleus segmentation with active contours. *Signal Processing*, **71**(2), 203–213.

Barber, C., Dobkin, D. and Huhdanpaa, H. (1996) The Quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, **22**(4), 469–483.

Beaulieu, J.M. and Goldberg, M. (1989) Hierarchy in picture segmentation: a step-wise optimization approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(2), 150–163.

Beucher, S. (1994) Watershed, hierarchical segmentation and waterfall algorithm. *Mathematical Morphology and its Applications to Image Processing*, 69–76.

Bhanu, B. and Parvin, B. (1987) Segmentation of natural scenes. *Pattern Recognition*, **20**(5), 487–496.

Boyes, E. (2005) Potential, practicalities and problems of aberration minimized FE-SEM. *Microscopy and Microanalysis*, **11**(S02), 2112–2113.

Browning, J. and Tanimoto, S. (1982) Segmentation of pictures into regions with a tile-by-tile method (split and merge approach). *Pattern Recognition*, **15**(1), 1–10.

BSI, B.S.I. (2005) Vocabulary-nanoparticles. *PAS-71:2005*.

Burt, P., Hong, T. and Rosenfeld, A. (1981) Segmentation and estimation of image region properties through cooperative hierarchial computation. *IEEE Transactions on Systems, Man and Cybernetics*, **11**(12), 802–809.

Cailliez, F. (1983) The analytical solution of the additive constant problem. *Psychometika*, **48**(2), 305–308.

Canny, J. (1986) A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(6), 679–698.

Chan, T. and Vese, L. (2001) Active contours without edges. *IEEE Transactions on Image Processing*, **10**(2), 266–277.

Cheevasuvit, F., Maitre, H. and Vidal-Madjar, D. (1986) A robust method for picture segmentation based on a split-and-merge procedure. *Computer Vision, Graphics and Image Processing*, **34**(3), 268–281.

Chen, L. and Ho, C. (2008) Development of nanoparticle shape measurement and analysis for process characterization of TiO_2 nanoparticle synthesis. *Review on Advanced Material Science*, **18**(8), 677–684.

Chen, Y., Tagare, H., Thiruvenkadam, S., Huang, F., Wilson, D., Gopinath, K., Briggs, R. and Geiser, E. (2002) Using prior shapes in geometric active contours in a variational framework. *International Journal of Computer Vision*, **50**(3), 315–328.

Cheng, J. and Rajapakse, J. (2009) Segmentation of clustered nuclei with shape markers and marking function. *IEEE Transactions on Biomedical Engineering*, **56**(2009), 741–748.

Chithrani, B., Ghazani, A. and Chan, W. (2006) Determining the size and shape dependence of gold nanoparticle uptake into mammalian cells. *Nano Letters*, **6**(4), 662–668.

Daněk, O., Matula, P., Ortiz-De-Solórzano, C., Muñoz-Barrutia, A., Maška, M. and Kozubek, M. (2009) Segmentation of touching cell nuclei using a two-stage graph cut model, in *Proceedings of the 16th Scandinavian Conference on Image Analysis*, Oslo, Norway, pp. 410–419.

Dempster, A., Laird, N., Rubin, D. *et al.* (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, **39**(1), 1–38.

Dougherty, E. (1994) *Digital Image Processing Methods*, CRC Press, New York, USA.

Dryden, I. and Mardia, K. (1993) Multivariate shape analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, **55**(3), 460–480.

Dryden, I. and Mardia, K. (1998) *Statistical Shape Analysis*, Wiley, West Sussex, UK.

El-Sayed, M.A. (2001) Some interesting properties of metals confined in time and nanometer space of different shapes. *Accounts of Chemical Research*, **34**(4), 257–264.

Epstein, M. (1976) On the influence of parametrization in parametric interpolation. *SIAM Journal on Numerical Analysis*, **13**(2), 261–268.

Eustis, S., Hsu, H. and El-Sayed, M. (2005) Gold nanoparticle formation from photochemical reduction of Au_3^+ by continuous excitation in colloidal solutions. A proposed molecular mechanism. *The Journal of Physical Chemistry B*, **109**(11), 4811–4815.

Felzenszwalb, P. and Huttenlocher, D. (2004) Efficient graph-based image segmentation. *International Journal of Computer Vision*, **59**(2), 167–181.

Filipe, V., Hawe, A. and Jiskoot, W. (2010) Critical evaluation of nanoparticle tracking analysis (NTA) by NanoSight for the measurement of nanoparticles and protein aggregates. *Pharmaceutical Research*, **27**(5), 796–810.

Fisker, R., Carstensen, J., Hansen, M., Bødker, F. and Mørup, S. (2000) Estimation of nanoparticle size distributions by image analysis. *Journal of Nanoparticle Research*, **2**(3), 267–277.

Foulonneau, A., Charbonnier, P. and Heitz, F. (2009) Multi-reference shape priors for active contours. *International Journal of Computer Vision*, **81**(1), 68–81.

Fu, K. and Mui, J. (1981) A survey on image segmentation. *Pattern Recognition*, **13**(1), 3–16.

Fultz, B. and Howe, J. (2008) *Transmission Electron Microscopy and Diffractometry of Materials*, Springer Verlag, Berlin, Germany.

Glotov, O. (2008) Image processing of the fractal aggregates composed of nanoparticles. *Russian Journal of Physical Chemistry A, Focus on Chemistry*, **82**(13), 2213–2218, (in English).

Gontard, L., Dunin-Borkowski, R., Ozkaya, D., Hyde, T., Midgley, P. and Ash, P. (2006) Crystal size and shape analysis of Pt nanoparticles in two and three dimensions. *Journal of Physics: Conference Series*, **26**, 367–370.

Gonzalez, R. and Woods, R. (2002) *Digital Image Processing, 3rd ed.*, Prentice Hall, Upper Saddle River, NJ, USA.

Goshtasby, A. (2000) Grouping and parameterizing irregularly spaced points for curve fitting. *ACM Transactions on Graphics*, **19**(3), 185–203.

Haralick, R. and Shapiro, L. (1992) *Computer and Robot Vision. volume 1*, Addison-Wesley, Reading, MA, USA.

Hindson, J., Saghi, Z., Hernandez-Garrido, J.C., Midgley, P. and Greenham, N. (2011) Morphological study of nanoparticle-polymer solar cells using high-angle annular dark-field electron tomography. *Nano Letters*, **11**(2), 904–909.

Hodneland, E., Tai, X. and Gerdes, H. (2009) Four-color theorem and level set methods for watershed segmentation. *International Journal of Computer Vision*, **82**(3), 264–283.

Hojjatoleslami, S. and Kittler, J. (2002) Region growing: a new approach. *IEEE Transactions on Image Processing*, **7**(7), 1079–1084.

Hoschek, J. (1988) Intrinsic parameterization for approximation. *Computer Aided Geometric Design*, **5**(1), 27–31.

Hosokawa, M. (2007) *Nanoparticle Technology Handbook*, Elsevier Science Ltd., Oxford, UK.

Huang, J., Shen, H. and Buja, A. (2008) Functional principal components analysis via penalized rank one approximation. *Electronic Journal of Statistics*, **2**, 678–695.

Huitink, D., Kundu, S., Park, C., Mallick, B., Huang, J. and Liang, H. (2010) Nanoparticle shape evolution identified through multivariate statistics. *The Journal of Physical Chemistry A*, **114**(17), 5596–5600.

ISO, I.S.O. (2009) Nanotechnologies – terminology and definitions for nano-objects – nanoparticle, nanofibre and nanoplate. *ISO/TS 27687:2008*.

Jones, F. (1938) The measurement of particle size by the X-ray method. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, **166**(924), 16–43.

Kass, M., Witkin, A. and Terzopoulos, D. (1988) Snakes: active contour models. *International Journal of Computer Vision*, **1**(4), 321–331.

Kauppinen, H., Seppanen, T. and Pietikainen, M. (2002) An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(2), 201–207.

Keil, M. and Snoeyink, J. (2002) On the time bound for convex decomposition of simple polygons. *International Journal of Computational Geometry and Applications*, **12**(3), 181–192.

Koster, A., Ziese, U., Verkleij, A., Janssen, A. and De-Jong, K. (2000) Three-dimensional transmission electron microscopy: a novel imaging and characterization technique with nanometer scale resolution for materials science. *The Journal of Physical Chemistry B*, **104**(40), 9368–9370.

Kruskal, J. and Wish, M. (1978) *Multidimensional Scaling (Series: Quantitative Applications in the Social Sciences)*, Sage Publications, Newbury Park, CA, USA.

Lacava, L., Lacava, B., Azevedo, R., Lacava, Z., Buske, N., Tronconi, A. and Morais, P. (2001) Nanoparticle sizing: a comparative study using atomic force microscopy, transmission electron microscopy, and ferromagnetic resonance. *Journal of Magnetism and Magnetic Materials*, **225**(1-2), 79–83.

Lecumberry, F., Pardo, A. and Sapiro, G. (2010) Simultaneous object classification and segmentation with high-order multiple shape models. *IEEE Transactions on Image Processing*, **19**(3), 625 –635.

Lee, C.H. (1986) Recursive region splitting at hierarchical scope views. *Computer Vision, Graphics and Image Processing*, **33**(2), 237–258.

Lee, E. (1989) Choosing nodes in parametric curve interpolation. *Computer-Aided Design*, **21**(6), 363–370.

Lin, G., Chawla, M., Olson, K., Guzowski, J., Barnes, C. and Roysam, B. (2005) Hierarchical, model-based merging of multiple fragments for improved three-dimensional segmentation of nuclei. *Cytometry Part A*, **63**(1), 20–33.

Loncaric, S. (1998) A survey of shape analysis techniques. *Pattern Recognition*, **31**(8), 983–1001.

Malpica, N., de Solórzano, C., Vaquero, J., Santos, A., Vallcorba, I., García-Sagredo, J. and del Pozo, F. (1997) Applying watershed algorithms to the segmentation of clustered nuclei. *Cytometry Part A*, **28**(4), 289–297.

Marcuzzo, M., Quelhas, P., Mendonça, A. and Campilho, A. (2009) Evaluation of symmetry enhanced sliding band filter for plant cell nuclei detection in low contrast noisy fluorescent images, in *Proceedings of the 6th Scandinavian Conference on Image Analysis and Recognition*, Halifax, Canada, pp. 824–831.

Matheron, G. (1975) *Random Sets and Integral Geometry*, Wiley, New York, USA.

McFarland, A. and Van Duyne, R. (2003) Single silver nanoparticles as real-time optical sensors with zeptomole sensitivity. *Nano Letters*, **3**(8), 1057–1062.

Meng, X. and Rubin, D. (1993) Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika*, **80**(2), 267.

Nagabhushana, S. (2006) *Computer Vision and Image Processing*, New Age International, New Delhi, India.

Narayanan, R. and El-Sayed, M. (2004) Shape-dependent catalytic activity of platinum nanoparticles in colloidal solution. *Nano Letters*, **4**(7), 1343–1348.

Nath, S., Palaniappan, K. and Bunyak, F. (2006) Cell segmentation using coupled level sets and graph-vertex coloring, in *Proceedings of the 9th International Conference on Medical Image Computing and Computer-Assisted Intervention*, Copenhagen, Denmark, pp. 101–108.

Nobbmann, U. and Morfesis, A. (2009) Light scattering and nanoparticles. *Materials Today*, **12**(5), 52–54.

Nyiro-Kosa, I., Nagy, D. and Posfai, M. (2009) Size and shape control of precipitated magnetite nanoparticles. *European Journal of Mineralogy*, **21**(2), 293–302.

Otsu, N. (1979) A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, **9**(1), 62–66.

Park, C., Huang, J., Huitink, D., Kundu, S., Mallick, B., Liang, H. and Ding, Y. (2011) A multi-stage, semi-automated procedure for analyzing the morphology of nanoparticles. *IIE Transactions,*, accepted.

Patterson, A. (1939) The Scherrer formula for X-ray particle size determination. *Physical Review*, **56**(10), 978.

Pecora, R. (1985) *Dynamic Light Scattering: Applications of Photon Correlation Spectroscopy*, Springer Verlag, Berlin, Germany.

Philbrick, O. (1968) Shape description with the medial axis transformation, in *Pictorial Pattern Recognition (Proceedings of the Symposium on Automated Photointerpretation)*, Thompson, Washington, DC, pp. 395–407.

Pietikainen, M., Rosenfeld, A. and Walter, I. (1982) Split-and-link algorithms for image segmentation. *Pattern Recognition*, **15**(4), 287–298.

Puntes, V., Krishnan, K. and Alivisatos, A. (2001) Colloidal nanocrystal shape and size control: The case of cobalt. *Science*, **291**(5511), 2115–2117.

Quelhas, P., Marcuzzo, M., Mendonca, A.M. and Campilho, A. (2010) Cell nuclei and cytoplasm joint segmentation using the sliding band filter. *IEEE Transactions on Medical Imaging*, **29**(8), 1463–1473.

Ronse, C. (1989) A bibliography on digital and computational convexity (1961-1988). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(2), 181–190.

Rosenfeld, A. (1985) Measuring the sizes of concavities. *Pattern Recognition Letters*, **3**(1), 71–75.

Saghri, J. and Freeman, H. (1981) Analysis of the precision of generalized chain codes for the representation of planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-3**(5), 533–539.

Samko, O., Marshall, A. and Rosin, P. (2006) Selection of the optimal parameter value for the Isomap algorithm. *Pattern Recognition Letters*, **27**(9), 968–979.

Saux, E. and Daniel, M. (2003) An improved Hoschek intrinsic parametrization. *Computer Aided Geometric Design*, **20**(8-9), 513–521.

Schneider, R. (1993) *Convex Bodies: the Brunn-Minkowski Theory*, Cambridge Univ Press, New York, USA.

Serra, J. (1983) *Image Analysis and Mathematical Morphology*, Academic Press, Inc., Orlando, FL, USA.

Shi, J. and Malik, J. (2000) Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(8), 888–905.

Siqueira, G., Bras, J. and Dufresne, A. (2008) Cellulose whiskers versus microfibrils: influence of the nature of the nanoparticle and its surface functionalization on the thermal and mechanical properties of nanocomposites. *Biomacromolecules*, **10**(2), 425–432.

Steinbach, M., Ertoz, L. and Kumar, V. (2003) Challenges of clustering high dimensional data. *New Vistas in Statistical Physics - Applications in Econophysics, Bioinformatics, and Pattern Recognition*, 207–312.

Tek, F., Dempster, A. and Kale, I. (2005) Blood cell segmentation using minimum area watershed and circle radon transformations, in *Proceedings of the 7th International Symposium on Mathematical Morphology*, Paris, France, pp. 441–454.

Tignol, J. (2001) *Galois' Theory of Algebraic Equations*, World Scientific Publishing Company, Toh Tuck Link, Singapore.

Tu, Z., Zheng, S. and Yuille, A. (2008) Shape matching and registration by data-driven EM. *Computer Vision and Image Understanding*, **109**(3), 290–304.

Umesh Adiga, P. and Chaudhuri, B. (2001) An efficient method based on watershed and rule-based merging for segmentation of 3-D histo-pathological images. *Pattern Recognition*, **34**(7), 1449–1458.

Vese, L. and Chan, T. (2002) A multiphase level set framework for image segmentation using the Mumford and Shah model. *International Journal of Computer Vision*, **50**(3), 271–293.

Villarrubia, J. (1997) Algorithm for scanned probe microscope image simulation, surface reconstruction, and tip estimation. *Journal of Research-National Institute of Standards and Technology*, **102**, 425–454.

Vujtek, M., Kubinek, R. and Zboril, R. (2010) The Role of atomic force microscopy in nanoparticle research. *Journal of Advanced Microscopy Research*, **5**(2), 67–77.

Wang, Y. and Teoh, E. (2007) 2D affine-invariant contour matching using B-spline model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**(10), 1853–1858.

Woehrle, G., Hutchison, J., Ozkar, S. and Finke, R. (2006) Analysis of nanoparticle transmission electron microscopy data using a public-domain image-processing program, Image. *Turkish Journal of Chemistry*, **30**(1), 1–13.

Wortis, M. (1988) *Chemistry and Physics Of Solid Surfaces*, Springer-Verlag, Berlin, Germany.

Yang, C., He, Z. and Yu, W. (2009) Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis. *BMC Bioinformatics*, **10**(4), 1–13.

Zhang, D. and Lu, G. (2003) A comparative study of curvature scale space and Fourier descriptors for shape-based image retrieval. *Journal of Visual Communication and Image Representation*, **14**(1), 39–57.

Zhang, D. and Lu, G. (2004) Review of shape representation and description techniques. *Pattern Recognition*, **37**(1), 1–19.

Zhang, Y. and Matuszewski, B. (2009) Multiphase active contour segmentation constrained by evolving medial axes, in *Proceedings of the 16th IEEE International Conference on Image Processing*, IEEE, Cairo, Egypt, pp. 2993–2996.

Zhao, D. and Chen, J. (1997) Affine curve moment invariants for shape recognition. *Pattern Recognition*, **30**(6), 895–901.

Zheng, H., Smith, R., Jun, Y., Kisielowski, C., Dahmen, U. and Alivisatos, A. (2009) Observation of single colloidal platinum nanocrystal growth trajectories. *Science*, **324**(5932), 1309–1312.

Zhu, X. (2005) Semi-supervised learning literature survey, Technical Report 1530, Computer Sciences, University of Wisconsin-Madison.

Zhu, X., Ghahramani, Z. and Lafferty, J. (2003) Semi-supervised learning using Gaussian fields and harmonic functions, in *Proceedings of the 20th International Conference on Machine Learning*, Washington, DC, pp. 912–919.

APPENDIX A

THEORETICAL STUDY ON THE SEPARATION CAPABILITY OF THE ULTIMATE EROSION

This appendix is dedicated to proving the two key lemmas related to the separation capability of the ultimate erosion (UE).

Proof of Lemma 6.

(i) Let A_λ be a set of points inside the circle that is centered at $\lambda x + (1-\lambda)y$ with radius $d_C(\lambda x + (1-\lambda)y)$. Then, both A_0 and A_1 are in C according to the definition of the distance function $d_C(x)$. Since C is convex, $\mathrm{conv}(A_0 \cup A_1) \subset C$, where $\mathrm{conv}(A)$ is a minimal convex set including A i.e. the convex hull of A. For $\lambda \in [0,1]$, the maximal circle centered at $\lambda x + (1-\lambda)y$ in $\mathrm{conv}(A_0 \cup A_1)$ has radius $\lambda d_C(x) + (1-\lambda)d_C(y)$. By the definition of $d_C(x)$, $d_C(\lambda x + (1-\lambda)y) \geq \lambda d_C(x) + (1-\lambda)d_C(y)$.

(ii) Suppose that $d_C(\lambda x + (1-\lambda)y)$ is not monotone increasing for $x \in \Omega_C$ and $y \in C \backslash \Omega_C$. If then, there exists $\lambda_1, \lambda_2 \in [0,1]$ so that $\lambda_2 > \lambda_1$ and $d_C(\lambda_1 x + (1-\lambda_1)y) > d_C(\lambda_2 x + (1-\lambda_2)y)$. Note that $A_\lambda \subset C$ for $\lambda \in [0,1]$. Since C is convex and A_{λ_1} and A_1 are in C, $\mathrm{conv}(A_{\lambda_1} \cup A_1) \subset C$. The convex hull includes a circle centered at $\lambda_2 x + (1-\lambda_2)y$ with radius,

$$\begin{aligned}
&\frac{1-\lambda_2}{1-\lambda_1} d_C(x) + \frac{\lambda_2-\lambda_1}{1-\lambda_1} d_C(\lambda_1 x + (1-\lambda_1)y) \\
&> \left(\frac{1-\lambda_2}{1-\lambda_1} + \frac{\lambda_2-\lambda_1}{1-\lambda_1}\right) d_C(\lambda_2 x + (1-\lambda_2)y) \\
&> d_C(\lambda_2 x + (1-\lambda_2)y).
\end{aligned}$$

This is a contradiction. □

Proof of Lemma 7.

First, some notations are defined. Let $W = \{\sum_k \lambda_k z_k : z_k \in S(x), \sum_k \lambda_k = 1, \lambda_k \geq 0\}$. If $B(x,\epsilon) \subset W$, for $y \in B(x,\epsilon)\backslash\{x\}$, there exists $\lambda_k \geq 0$'s such that

$$\begin{aligned} ||y-x||^2 &= \left\langle y-x, \sum_k \lambda_k z_k - x \right\rangle \\ &= \sum_k \lambda_k \left\langle y-x, z_k - x\right\rangle > 0, \end{aligned}$$

where $\langle \cdot, \cdot \rangle$ is an inner product. Therefore, there exists at least one $z_j \in S(x)$ such that $\langle y-x, z_j-x\rangle$ is greater than zero.

$$\begin{aligned} d_I(y)^2 = ||z_j - y||^2 &= \langle z_j - x + x - y, z_j - x + x - y\rangle \\ &= ||z_j - x||^2 + ||y-x||^2 - 2\left\langle z_j - x, y - x\right\rangle \\ &= d_I(x)^2 + \epsilon^2 - 2\epsilon d_I(x)\cos\theta \\ &= d_I(x)^2 + \epsilon(\epsilon - 2d_I(x)\cos\theta). \end{aligned} \tag{A.22}$$

Since $\langle y-x, z_j-x\rangle$ is positive, $d_I(x)\cos(\theta)$ is a positive constant and ϵ can be chosen so that $\epsilon < 2d_I(x)\cos(\theta)$. Then, $d_I(y)^2 < d_I(x)^2$. Since $y \in \partial B(x,\epsilon)$ is arbitrary, x is a local maximum, which completes the 'if' part.

The 'only-if' part is shown by contradiction. Suppose that x is a local maximum but there is no $\epsilon > 0$ such that $B(x,\epsilon) \subset W$. That is, x is not in the interior of W. If x is not in W, $z \in W$ can be chosen to be closest to x. Let $y = \frac{\epsilon}{||x-z||}(x-z) + x$. Since x is a local maximum, by (A.22), there exists at least one $z_j \in S(x)$ such that $\langle y-x, z_j - x\rangle$ is positive. Since $\langle y-x, z_j-x\rangle = \epsilon/||x-z||\left\langle x-z, z_j-x\right\rangle$, $\langle x-z, z_j-x\rangle$ is also positive. Since $z_j, z \in W$ and W is convex, $z' = \lambda z + (1-\lambda)z_j$

is in W for $\lambda \in [0, 1]$.

$$\begin{aligned} ||z' - x||^2 =& \langle \lambda z + (1-\lambda) z_j - x, \lambda z + (1-\lambda) z_j - x \rangle \\ =& \langle \lambda(z - x) + (1-\lambda)(z_j - x), \\ & \quad \lambda(z - x) + (1-\lambda)(z_j - x) \rangle \\ =& \lambda^2 ||z - x||^2 + (1-\lambda)^2 ||z_j - x||^2 \\ & - 2\lambda(1-\lambda) \langle x - z, z_j - x \rangle . \end{aligned}$$

If $\lambda \in [0, 1]$ is chosen such that $(1 - \lambda^2)||z - x||^2 = (1 - \lambda)^2 ||z_j - x||^2$, the last line is smaller than $||z - x||^2$ since $\langle x - z, z_j - x \rangle$ is positive, meaning that $||z' - x||^2 < ||z - x||^2$. This is a contradiction, because $z \in W$ is chosen to be the closest to x. If x is on the boundary of W, $y \in B(x, \epsilon)$ can be chosen such that $y - x$ is orthogonal to the boundary of W at x, and the same logic can be applied. $\square$

APPENDIX B

GUIDELINES FOR DETERMINATION OF ρ IN THE UECS

The results of the UECS with different choices of ρ under different degrees of particle overlaps and different noise levels are presented. This will be a good guideline for practitioners to choose the ρ appropriate for their own applications. Figure 33 shows different settings of two overlapping rectangles under the different levels of noises artificially generated. Each setting of the overlap is specified by the degree of overlap (doo), which is proportional to the negative distance between the centers of the two rectangles. Specifically, if the distance is x and the length of the diagonal of the rectangle is d, the doo is related to the size of overlap relative to the rectangle size as

$$doo(x) = 5(1 - \frac{x}{d}) + 6.$$

The noise level, σ, is specified by the maximum allowable size of intrusion or extrusion. The $\sigma = 6$ corresponds to six percents of the diagonal length d. The combinations of different doo and σ values provide twenty four experimental settings.

Figures 34 through 36 show the results of the UECS for these experimental settings, where $\rho \in \{0.1, 0.2, 0.3\}$ and $\sigma \in \{0, 2, 4, 6\}$. The UECS with $\rho = 0.2$ generally keeps a good balance between under-segmentation (caused by two objects located closely) and over-segmentation (caused by image noises).

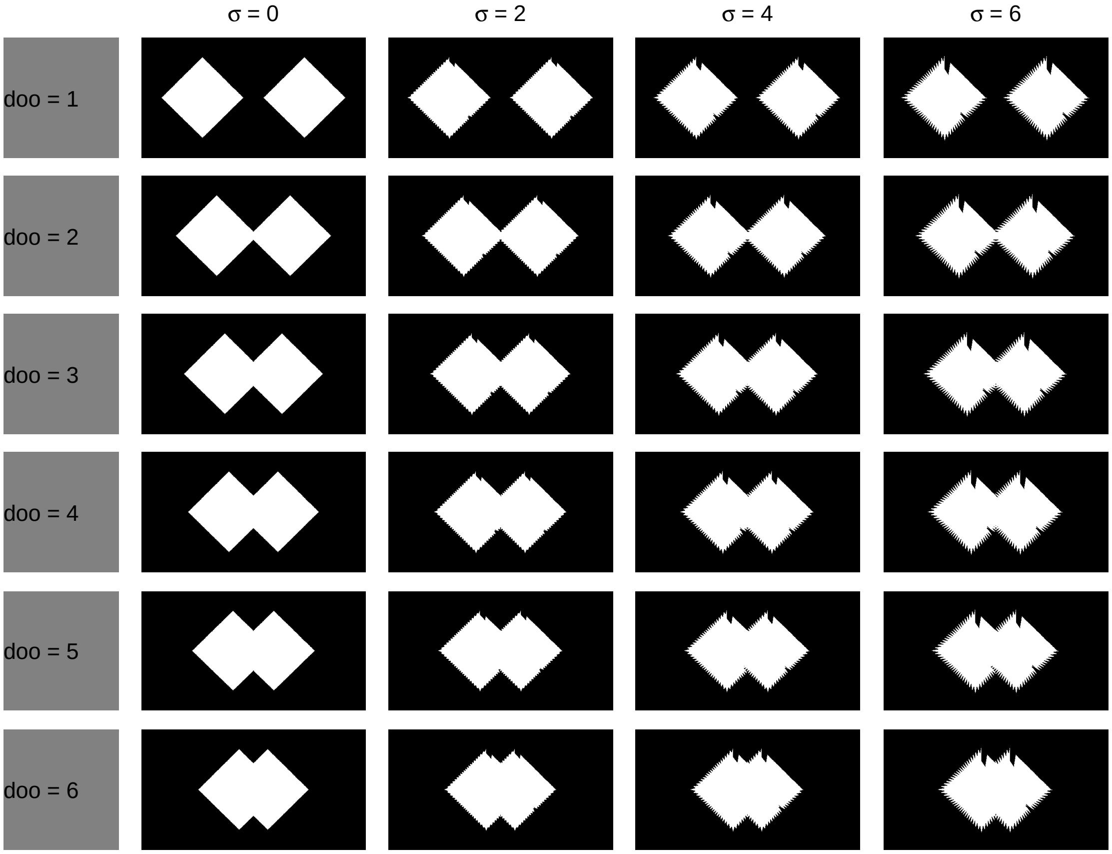


Fig. 33. Different configurations of overlaps with different level of noises σ. The ‘doo’ stands for the degree of overlap, inversely proportional to the distance between the centers of two rectangles.

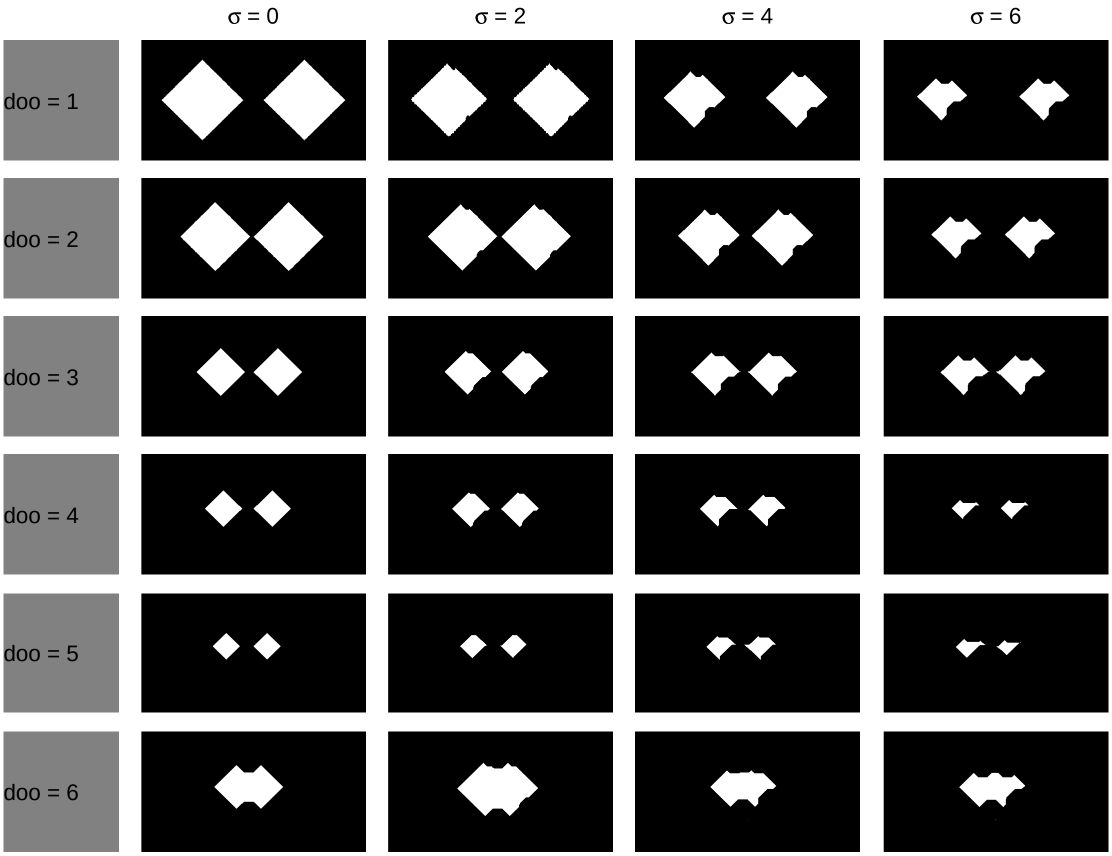


Fig. 34. Performance of the UECS ($\rho = 0.3$).

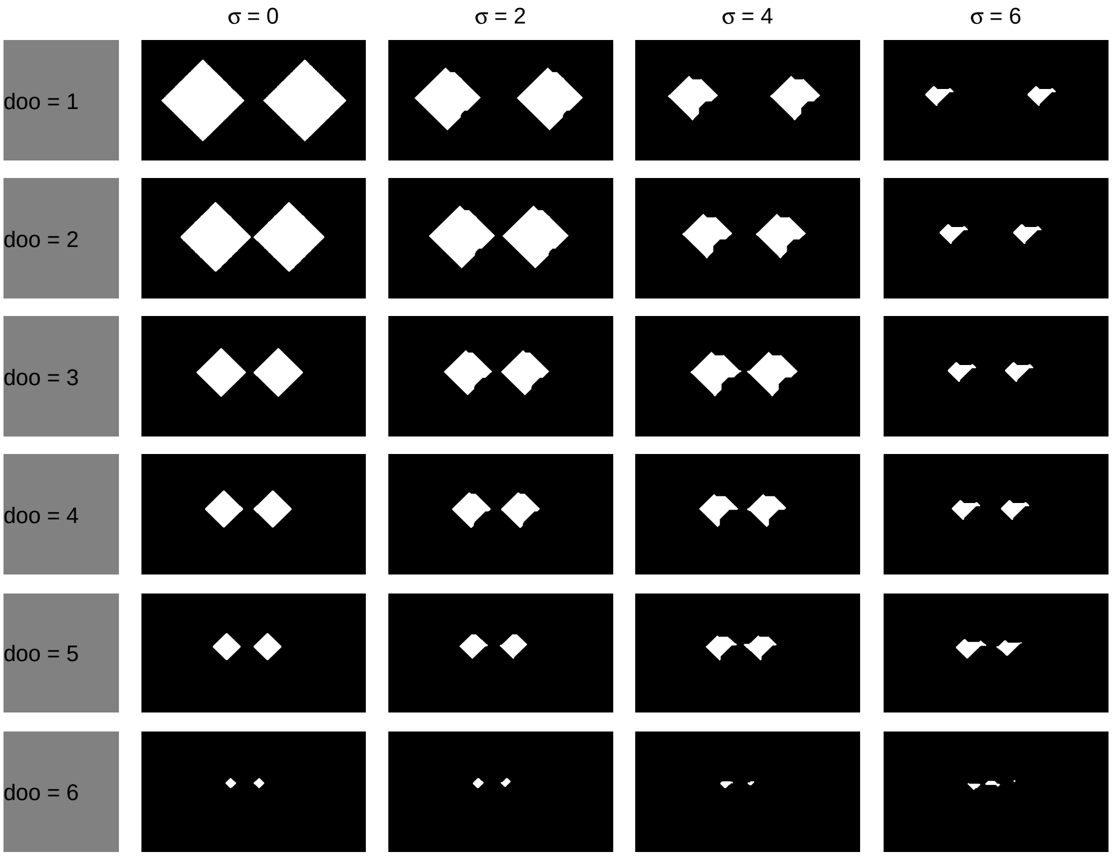


Fig. 35. Performance of the UECS ($\rho = 0.2$).

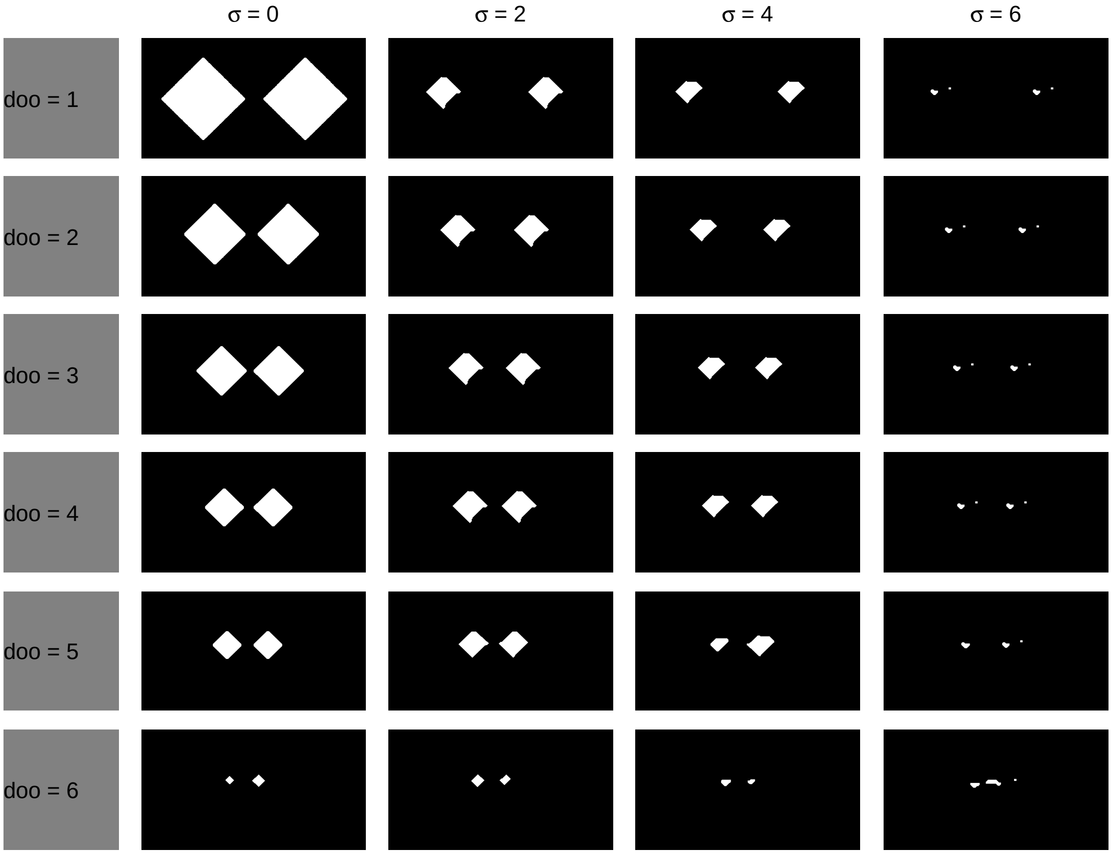


Fig. 36. Performance of the UECS ($\rho = 0.1$).

APPENDIX C

FULL DERIVATION OF THE E-STEP

At iteration t, the expectation maximization (EM) algorithm first computes the expected value of the complete log-likelihood function with respect to the posterior distribution, $P(\mathbf{Z}|\mathbf{X}, \boldsymbol{\Theta}^{(o)})$, where $\boldsymbol{\Theta}^{(o)}$ is an old estimate of $\boldsymbol{\Theta}$ (E-step). The expected value is given by (21). This appendix presents the full derivation of the expected value.

The complete log likelihood is

$$\begin{aligned}\log P(\mathbf{X}, \mathbf{Z}|\boldsymbol{\Theta}) =& \sum_{i=1}^{N}\sum_{k=1}^{K} g_{ik}\left[\log \alpha_k + \log \mathcal{N}(\boldsymbol{p}_i; \boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}, \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)\right.\\ &\left. + \log \mathcal{N}(\mathbf{x}_i; \boldsymbol{\Phi}_i\boldsymbol{p}_i, \sigma^2\boldsymbol{I}_{2m_i})\right]\\ =& \sum_{i=1}^{N}\sum_{k=1}^{K} g_{ik}\left[\log \alpha_k - \frac{1}{2}\log|\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t| - \frac{1}{2}\log|\sigma^2\boldsymbol{I}_{2m_i}|\right.\\ &- \frac{1}{2}(\boldsymbol{p}_i - \boldsymbol{A}_{ik}\mu_k - \boldsymbol{H}\boldsymbol{c}_{ik})^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{p}_i - \boldsymbol{A}_{ik}\mu_k - \boldsymbol{H}\boldsymbol{c}_{ik})\\ &\left.-\frac{1}{2\sigma^2}(\mathbf{x}_i - \boldsymbol{\Phi}_i\boldsymbol{p}_i)^t(\mathbf{x}_i - \boldsymbol{\Phi}_i\boldsymbol{p}_i)\right] + c,\end{aligned}$$

where c is a constant that does not depend on $\boldsymbol{\Theta}$. Using that $\boldsymbol{A}_{ik}$ is a scaled orthogonal matrix such as $\boldsymbol{A}_{ik} = \frac{1}{s_{ik}}\boldsymbol{Q}_{\theta_{ik}}$ and $\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{Q}_{\theta_{ik}} = \boldsymbol{I}$,

$$\begin{aligned}\log P(\mathbf{X}, \mathbf{Z}|\boldsymbol{\Theta}) =& \sum_{i=1}^{N}\sum_{k=1}^{K} g_{ik}\left[\log \alpha_k - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| + 2p\log s_{ik} - m_i\log\sigma^2\right.\\ &- \frac{1}{2}(\boldsymbol{p}_i - \boldsymbol{A}_{ik}\mu_k - \boldsymbol{H}\boldsymbol{c}_{ik})^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{p}_i - \boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik})\\ &\left.-\frac{1}{2\sigma^2}(\mathbf{x}_i - \boldsymbol{\Phi}_i\boldsymbol{p}_i)^t(\mathbf{x}_i - \boldsymbol{\Phi}_i\boldsymbol{p}_i)\right] + c.\end{aligned} \tag{C.1}$$

The expectation of (C.1) with respect to $P(\mathbf{Z}|\mathbf{X},\boldsymbol{\Theta}^{(o)})$ is

$$
\begin{aligned}
E[\log P(\mathbf{X},\mathbf{Z}|\boldsymbol{\Theta})] = & \sum_{i=1}^{N}\sum_{k=1}^{K} E[g_{ik}]\left[\log\alpha_k - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| + 2p\log s_{ik} - m_i\log\sigma^2\right] \\
& + E[g_{ik}\boldsymbol{p}_i^t](\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
& - \frac{1}{2}E[g_{ik}\boldsymbol{p}_i^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}\boldsymbol{p}_i] \\
& - \frac{1}{2}E[g_{ik}](\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik})^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
& + \frac{1}{\sigma^2}E[g_{ik}\boldsymbol{p}_i^t]\boldsymbol{\Phi}_i^t\mathbf{x}_i \\
& - \frac{1}{2\sigma^2}E[g_{ik}\boldsymbol{p}_i^t\boldsymbol{\Phi}_i^t\boldsymbol{\Phi}_i\boldsymbol{p}_i] \\
& - \frac{1}{2\sigma^2}E[g_{ik}]\mathbf{x}_i^t\mathbf{x}_i \\
= & \sum_{i=1}^{N}\sum_{k=1}^{K} E[g_{ik}]\left[\log\alpha_k - \frac{1}{2}\log|\boldsymbol{\Sigma}_k| + 2p\log s_{ik} - m_i\log\sigma^2\right] \\
& + E[g_{ik}\boldsymbol{p}_i^t](\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
& - \frac{1}{2}\text{trace}((\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}E[g_{ik}\boldsymbol{p}_i\boldsymbol{p}_i^t]) \\
& - \frac{1}{2}E[g_{ik}](\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik})^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
& + \frac{1}{\sigma^2}E[g_{ik}\boldsymbol{p}_i^t]\boldsymbol{\Phi}_i^t\mathbf{x}_i \\
& - \frac{1}{2\sigma^2}\text{trace}(\boldsymbol{\Phi}_i^t\boldsymbol{\Phi}_i E[g_{ik}\boldsymbol{p}_i\boldsymbol{p}_i^t]) \\
& - \frac{1}{2\sigma^2}E[g_{ik}]\mathbf{x}_i^t\mathbf{x}_i.
\end{aligned}
\tag{C.2}
$$

The expression for $E[g_{ik}]$ is given as

$$\begin{aligned}
E[g_{ik}] =& P(g_{ik}=1|\mathbf{x}_i,\mathbf{\Theta}^{(o)}) \\
=& \int P(\boldsymbol{p}_i|\mathbf{x}_i,\mathbf{\Theta}^{(o)},g_{ik}=1)P(g_{ik}=1)d\boldsymbol{p}_i \\
\propto& \alpha_k \int P(\mathbf{x}_i|\boldsymbol{p}_i,\mathbf{\Theta}^{(o)},g_{ik}=1)P(\boldsymbol{p}_i|\mathbf{\Theta}^{(o)})d\boldsymbol{p}_i \\
=& \alpha_k \mathcal{N}\left(\mathbf{x}_i;\mathbf{\Phi}_i\boldsymbol{A}_{ik}\boldsymbol{\mu}_k+\boldsymbol{H}\boldsymbol{c}_{ik},\sigma^2\boldsymbol{I}_{2m_i}+\mathbf{\Phi}_i\boldsymbol{A}_{ik}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^t\mathbf{\Phi}_i^t\right),
\end{aligned}$$

where the integration is evaluated by the Gaussian identity in Section D. For notational convenience, the following is defined

$$q_{ik} = \mathcal{N}\left(\mathbf{x}_i;\mathbf{\Phi}_i\boldsymbol{A}_{ik}\boldsymbol{\mu}_k+\boldsymbol{H}\boldsymbol{c}_{ik},\sigma^2\boldsymbol{I}_{2m_i}+\mathbf{\Phi}_i\boldsymbol{A}_{ik}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^t\mathbf{\Phi}_i^t\right).$$

Using $\sum_{k'=1}^{K} E[g_{ik}] = 1$,

$$\beta_{ik} := E[g_{ik}] = \frac{\alpha_k q_{ik}}{\sum_{k'=1}^{K}\alpha_{k'}q_{ik'}}.$$

Since $P(\boldsymbol{p}_i|g_{ik}=1,\mathbf{\Theta}^{(o)}) = \mathcal{N}(\boldsymbol{p}_i;\boldsymbol{A}_{ik}\mu_k+\boldsymbol{H}\boldsymbol{c}_{ik},\boldsymbol{A}_{ik}\Sigma_k\boldsymbol{A}_{ik}^t)$ and $P(\boldsymbol{x}_i|\boldsymbol{p}_i,\mathbf{\Theta}^{(o)}) = \mathcal{N}(\mathbf{x}_i;\mathbf{\Phi}_i\boldsymbol{p}_i,\sigma^2\boldsymbol{I}_{2m_i})$, the following posterior distribution can be obtained by simply applying the Gaussian identity in Section D as

$$\begin{aligned}
P(\boldsymbol{p}_i|g_{ik}=1,\boldsymbol{X}_i,\mathbf{\Theta}^{(o)}) &= \mathcal{N}(\boldsymbol{m}_{ik},\boldsymbol{S}_{ik}), \\
\text{where}\quad \boldsymbol{m}_{ik} &= \boldsymbol{A}_{ik}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^t\mathbf{\Phi}_i^t\left(\sigma^2\boldsymbol{I}_{2m_i}+\mathbf{\Phi}_i\boldsymbol{A}_{ik}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^t\mathbf{\Phi}_i^t\right)^{-1} \\
&\quad (\boldsymbol{x}_i-\mathbf{\Phi}_i\boldsymbol{A}_{ik}\boldsymbol{\mu}_k-\boldsymbol{H}\boldsymbol{c}_{ik})+\boldsymbol{A}_{ik}\boldsymbol{\mu}_k+\boldsymbol{H}\boldsymbol{c}_{ik}, \\
\boldsymbol{S}_{ik} &= (\boldsymbol{A}_{ik}^{-t}\mathbf{\Sigma}_k\boldsymbol{A}_{ik}^{-1}+\sigma^{-2}\mathbf{\Phi}_i^t\mathbf{\Phi}_i)^{-1}.
\end{aligned}$$

Then, the following results can be obtained:

$$\begin{aligned}
E[g_{ik}\boldsymbol{p}_i^t] &= P(g_{ik}=1|\mathbf{x}_i,\mathbf{\Theta}^{(o)})\boldsymbol{m}_{ik} = \beta_{ik}\boldsymbol{m}_{ik}, \\
E[g_{ik}\boldsymbol{p}_i\boldsymbol{p}_i^t] &= P(g_{ik}=1|\mathbf{x}_i,\mathbf{\Theta}^{(o)})(\boldsymbol{S}_{ik}+\boldsymbol{m}_{ik}\boldsymbol{m}_{ik}^t) = \beta_{ik}(\boldsymbol{S}_{ik}+\boldsymbol{m}_{ik}\boldsymbol{m}_{ik}^t).
\end{aligned}$$

To sum up, putting the following expression into (C.2),

$$\begin{aligned}
\beta_{ik} &= \alpha_k q_{ik} \\
\boldsymbol{\gamma}_{ik} &= \beta_{ik}\boldsymbol{m}_{ik} \\
\boldsymbol{\Delta}_{ik} &= \beta_{ik}(\boldsymbol{S}_{ik} + \boldsymbol{m}_{ik}\boldsymbol{m}_{ik}^t),
\end{aligned}$$

the expected log likelihood is

$$\begin{aligned}
\text{(C.2)} =& \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{K}\beta_{ik}\left[2\log\alpha_k - \log|\boldsymbol{\Sigma}_k| + 4p\log s_{ik} - 2m_i\log\sigma^2\right. \\
&+ 2\boldsymbol{m}_{ik}^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
&- \operatorname{trace}((\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}\boldsymbol{\Delta}_{ik}) \\
&- (\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik})^t(\boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k\boldsymbol{A}_{ik}^t)^{-1}(\boldsymbol{A}_{ik}\mu_k + \boldsymbol{H}\boldsymbol{c}_{ik}) \\
&\left.+2\sigma^{-2}\boldsymbol{m}_{ik}^t\boldsymbol{\Phi}_i^t\mathbf{x}_i - \sigma^{-2}\operatorname{trace}(\boldsymbol{\Phi}_i^t\boldsymbol{\Phi}_i\boldsymbol{\Delta}_{ik}) - \sigma^{-2}\mathbf{x}_i^t\mathbf{x}_i\right] \\
\propto& \sum_{i=1}^{N}\sum_{k=1}^{K}\beta_{ik}\left[2\log\alpha_k - \log|\boldsymbol{\Sigma}_k| + 4p\log s_{ik} - 2m_i\log\sigma^2\right. \\
&+ 2s_{ik}\boldsymbol{m}_{ik}^t\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{\Sigma}_k^{-1}(\mu_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik}) \\
&- s_{ik}^2\operatorname{trace}(\boldsymbol{Q}_{\theta_{ik}}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{\Delta}_{ik}) \\
&- (\mu_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik})^t\boldsymbol{\Sigma}_k^{-1}(\mu_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t\boldsymbol{H}\boldsymbol{c}_{ik}) \\
&\left.+2\sigma^{-2}\boldsymbol{m}_{ik}^t\boldsymbol{\Phi}_i^t\mathbf{x}_i - \sigma^{-2}\operatorname{trace}(\boldsymbol{\Phi}_i^t\boldsymbol{\Phi}_i\boldsymbol{\Delta}_{ik}) - \sigma^{-2}\mathbf{x}_i^t\mathbf{x}_i\right].
\end{aligned}$$

APPENDIX D

BASIC MATRIX CALCULUS

D.1 Gaussian Identity

The following two product forms of Gaussian distributions are equivalent.

$$\begin{aligned}&\mathcal{N}(\boldsymbol{f};\mu,\Sigma)\mathcal{N}(\boldsymbol{y};\boldsymbol{A}\boldsymbol{f},\boldsymbol{D})\\&\qquad=\mathcal{N}\left(\boldsymbol{y};\boldsymbol{A}\mu,\boldsymbol{D}+\boldsymbol{A}\Sigma\boldsymbol{A}^t\right)\mathcal{N}\left(\boldsymbol{f};\mu+\boldsymbol{B}(\boldsymbol{y}-\boldsymbol{A}\mu),(\Sigma^{-1}+\boldsymbol{A}^t\boldsymbol{D}^{-1}\boldsymbol{A})^{-1}\right),\end{aligned}$$

where $\boldsymbol{B}=\Sigma\boldsymbol{A}^t(\boldsymbol{D}+\boldsymbol{A}\Sigma\boldsymbol{A}^t)^{-1}$.

D.2 Matrix Calculus

$$\begin{aligned}&\frac{d}{d\boldsymbol{A}}\log\det\boldsymbol{A}=\boldsymbol{A}^{-1}\\&\frac{d}{d\boldsymbol{A}}\boldsymbol{a}^t\boldsymbol{A}^{-1}\boldsymbol{b}=-\frac{1}{2}(\boldsymbol{A}^{-1}\boldsymbol{b}\boldsymbol{a}^t\boldsymbol{A}^{-1}+\boldsymbol{A}^{-1}\boldsymbol{a}\boldsymbol{b}^t\boldsymbol{A}^{-1})\\&\frac{d}{d\boldsymbol{A}}\text{trace}(\boldsymbol{A}^{-1}\boldsymbol{B})=-\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{A}^{-1}\\&\frac{d}{d\boldsymbol{A}}(\boldsymbol{a}^t\boldsymbol{A}\boldsymbol{b})=\boldsymbol{b}\boldsymbol{a}^t\\&\frac{d}{d\boldsymbol{A}}\text{trace}(\boldsymbol{X}\boldsymbol{A}\boldsymbol{Y}\boldsymbol{A}^t)=\boldsymbol{Y}\boldsymbol{A}^t\boldsymbol{X}+\boldsymbol{Y}^t\boldsymbol{A}^t\boldsymbol{X}^t\end{aligned}$$

D.3 Rotation Transformation

$$R_\theta=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}=\exp(\theta\boldsymbol{I}_a),\ \text{where }\boldsymbol{I}_a=\begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix},$$

and $\boldsymbol{I}_a$ is an anti-symmetric matrix.

VITA

Chiwoo Park received a B.S. in the department of industrial engineering from the Seoul National University in 2001. His research interest is applied statistics and data mining with applications to nano-imaging, sensor networks and remote sensing. He is a member of IIE, INFORMS, IEEE, and ASA.

He may be reached at his department address: Department of Industrial and Systems Engineering, Texas A&M University, 3131 TAMU, College Station, TX 77843-3131.

The typist for this dissertation was Chiwoo Park.