Segmentation, Inference, and Classification of Partially Overlapping Nanoparticles

Chiwoo Park, *Member, IEEE*, Jianhua Z. Huang, Jim X. Ji, *Senior Member, IEEE*, and
Yu Ding, *Senior Member, IEEE*

Abstract—This paper presents a method that enables automated morphology analysis of partially overlapping nanoparticles in electron micrographs. In the undertaking of morphology analysis, three tasks appear necessary: separate individual particles from an agglomerate of overlapping nanoobjects, infer the particle's missing contours, and, ultimately, classify the particles by shape based on their complete contours. Our specific method adopts a two-stage approach: The first stage executes the task of particle separation, and the second stage simultaneously conducts the tasks of contour inference and shape classification. For the first stage, a modified ultimate erosion process is developed for decomposing a mixture of particles into markers, and then an edge-to-marker association method is proposed to identify the set of evidences that eventually delineate individual objects. We also provide theoretical justification regarding the separation capability of the first stage. In the second stage, the set of evidences becomes inputs to a Gaussian mixture model on B-splines, the solution of which leads to the joint learning of the missing contour and the particle shape. Using 12 real electron micrographs of overlapping nanoparticles, we compare the proposed method with seven state-of-the-art methods. The results show the superiority of the proposed method in terms of particle recognition rate.

Index Terms—Image segmentation, morphology analysis, shape inference, shape classification, nanoparticle analysis

✦

1 INTRODUCTION

THIS paper introduces a new method for separating a large number of partially overlapping convex objects into individual pieces, inferring the missing contours, and classifying the complete contours by shapes. A motivating application is the morphology analysis of nanoparticles in electron micrographs, which requires classifying individual nanoparticles by their sizes and shapes. The functional behavior of nanoparticles is tightly linked to the surface morphology of the particles so that accurately classifying the synthesized nanoparticles is crucial for characterizing the nanoparticle's behavior [1], [2], [3], [4].

There are two major difficulties in nanoparticle morphology analysis. The first one is caused by particle overlap. We frequently observe wide-ranging degrees of particle overlaps in micrographs. The overlaps hide partial contours, hindering the accurate recognition of individual particles. Consequently, separating overlapped particles and inferring the missing contours occluded by the overlaps are the key to an accurate morphology analysis.

The second difficulty is the large number of nanoparticles in a micrograph. It is not uncommon to see 30-100 particles in a magnified image or 300-700 particles in a typical viewfield of $1,024 \times 640$ nm. This quantity makes the morphology analysis different from the segmentation problems that focus on recognizing a small number of complex objects. In order to handle the morphology analysis problem within a practically short time, the method needs to be computationally efficient.

The undertaking of morphology analysis entails the solving of three technical problems: The first is *image segmentation*, aiming at separating individual particles from the overlapping particle agglomerates; the second is *contour inference*, recovering the missing parts of the separated particles; and the third is *shape classification*, classifying the particles by shape. To address these technical problems, we choose to employ a two-stage approach: The first stage solves the image segmentation problem and the second stage solves the joint learning of the missing contour and the particle shape. In the first stage, our segmentation method is a new morphological erosion process, specially tailored for handling convex objects because the theory governing crystal formation tells us that stable nanoparticles are highly likely in convex shapes [5]. In the second stage, the contour inference problem and the shape classification problem are solved simultaneously. The joint learning is formulated as a Gaussian mixture model on B-splines, where both the missing contour and the shape category of the particle are hidden variables. These hidden variables are estimated iteratively by using an expectation-conditional maximization (ECM) algorithm.

The rest of the paper is organized as follows: Section 2 reviews the related work and identifies the competing

- *C. Park is with the Department of Industrial and Manufacturing Engineering, Florida A&M and Florida State University, Office: B319 COE, Tallahassee, FL 32310. E-mail: cpark5@fsu.edu.*
- *J.Z. Huang is with the Department of Statistics, Texas A&M University, Office: 405C Blocker, College Station, TX 77843-3143. E-mail: jianhua@stat.tamu.edu.*
- *J.X. Ji is with the Department of Electrical and Computer Engineering, Texas A&M University, Office: 237L Zachry, College Station, TX 77843-3128. E-mail: jimji@ece.tamu.edu.*
- *Y. Ding is with the Department of Industrial and Systems Engineering, Texas A&M University, Office: 4016 ETB, College Station, TX 77843-3131. E-mail: yuding@iemail.tamu.edu.*

alternatives to be compared. Section 3 describes our approach for the image segmentation problem, while Section 4 formulates the ECM for contour inference and shape classification. In Section 5, we apply the proposed method to several micrographs having different degrees of overlap and compare the results with the competing alternatives. Section 6 concludes the paper.

2 RELATED RESEARCH

Despite its importance, there is only a limited amount of literature about automated morphology analysis of nanoparticles. To the best of our knowledge, all existing methods assumed either circularity of particle's contours [6], [7], [8] or elliptical shape template [9] in order to segment overlapping particles, so their applicability is limited. On the other hand, there exists plenty of research for the biocell segmentation problem. What we choose to review 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.

Morphological image segmentation methods, represented by the watershed and its variants [10], [11], [12], [13], [14], [15], are a classical approach. The approach first finds markers pointing to the approximate locations of cells and then segments an image region into several influence zones of markers. The morphological segmentation methods generally work well, but this line of methods does not provide any inference on the occluded 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 an image [16], [17]. This approach does not separate overlapping objects well, especially when the overlapping objects have similar intensity levels. Hence, Daněk et al. [18] utilized the estimated mean radius of cells for the purpose of separating the overlapping cells, but their method is only applicable to spherically shaped objects.

Active contour is another school of methods applied to the cell segmentation [19], [20]. Active contour is originally designed to segment a single object of complicated shape from the background [21], but the level-set based active contour [22] or the multiphase active contour [23] could be used to segment multiple nonoverlapping objects. To separate overlapping objects, a shape prior constraint is to be associated with every single object [24], [25], [26]. The shape prior level-set method is generally computationally demanding when handling a large number of objects, and consequently, they were mostly applied to the evolving of a small number of level-set functions [27], [28]. The computation cost can be reduced by applying the narrow banding technique [29] to evolve a larger number of level-set functions.

Recently, advanced nonlinear filtering algorithms have been used for a noise-robust cell segmentation: sliding band filter (SBF) [30], [31] and iterative voting method (IVM) [32].

A major difficulty in applying the nonlinear filtering algorithms is that the range of object sizes should be known a priori. The size range can be roughly estimated by morphological granulometric analysis [33]. Still, if the range is wide, the nonlinear filtering algorithms tend to be ineffective to separate different objects and will likely oversegment large objects.

The aforementioned methods have made great advancements in biocell morphology analysis. But the nanoparticle morphology analysis presents a technical challenge through a combination of the following three major factors: the particle overlap, various geometric shapes and sizes, and a large quantity. Overlapping nanoparticles have similar image intensities, which makes both graph cut and active contour methods less ideal candidates. 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 nonlinear filtering approach ineffective. Separating the large quantity of particles needs a computationally effective solution, which makes the active contour methods unattractive. 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. Addressing the challenges presented by nanoparticle morphology analysis still calls for new, more effective approaches.

We propose a two-stage morphology analysis method in this paper which can better address the above-described nanoparticle problem. The contribution of our research is as follows: First, we modify an existing morphological erosion process for handling convex objects and provide theoretical justification on its separation capability. Second, we develop a method for associating the segmented edges with markers and for making them meaningful chunks of evidences to delineate individual objects. Third, the evidences are used as input in the subsequent ECM algorithm to fit the B-spline contours with the guidance of multiple reference shapes (contour inference) as well as to determine which reference shape best conforms with the evidences (shape classification). The proposed method solves the particle segmentation problem separately from the ECM algorithm because incorporating the segmentation problem into the ECM formulation will cause heavy computation and slow convergence of the ECM. We would also like to note a limitation of our method: Since it is designed for handling convex objects, the proposed method will not work for image segmentation of nonconvex shapes, e.g., donut-shaped particles; for such cases and methods that address them, please refer to [34], [35].

3 SEGMENTATION OF PARTICLE OVERLAPS BY CONVEX DECOMPOSITION

According to the theory governing crystal formation [5], nanoparticles are prone to having convex shapes. Hence, at the first stage we need to deal with a convex decomposition problem, namely, segmenting a complicated morphology into convex subpieces.
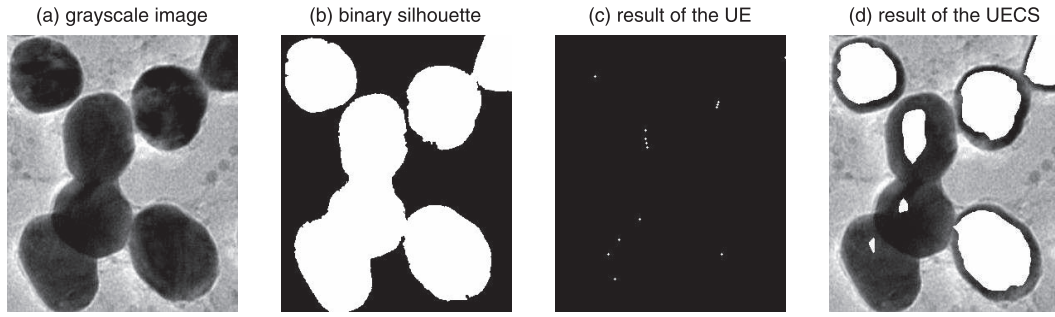
Fig. 1. Ultimate erosion for convex sets: (a) The original grayscale image, (b) binary silhouette of clustered convex shaped objects, (c) markers identified by the ultimate erosion, (d) markers identified by the ultimate erosion for convex sets.

This segmentation task has two parts: finding the markers of individual convex pieces (in Section 3.1) and obtaining the contour evidences to delineate each of the individual convex pieces (in Section 3.2). Our procedure is built upon the existing morphological segmentation method, ultimate erosion (UE) [36], but it has a couple of key differences. First, our stopping criterion for iterative erosion is different, producing more robust segmentation results for convex sets. Second, our approach 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 edges and relating the edges as contour evidences to one of the markers. To relate the edges, we define an evidence-to-marker relevance measure different from the one used in the marker-growing approach.

3.1 Ultimate Erosion for Overlapping Convex Sets

Suppose that we have a binary silhouette of overlapping objects from a grayscale image (please see Fig. 1b for an example). Such a binary silhouette can be easily obtained in real micrographs picturing nanoparticles because these micrographs have a high signal-to-noise ratio. There are usually irregular intensity patterns in the interior of a nanoparticle, but such noises could be effectively removed by using alternative sequence filtering [37]. Since nanoparticles only have 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 we have 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 nonempty set I, we want 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.

The markers are produced through morphological erosion, performed by applying the Minkowski subtraction to I with respect to $B(0, 1)$, where $B(x, r)$ is a closed ball in $\mathbb{R}^2$ centered at x with radius r [38, p. 133]. Conceptually, the result of the erosion operator $B(0, 1)$ is equivalent to peeling off I from its boundary by size one. Repeated applications of the erosion operator may disconnect the junctions of overlapping objects. A key question is when to stop the morphological erosion. A popular choice is to keep applying the erosion operator to each remaining connected set just before it is completely removed. This is called the ultimate erosion [36, p. 72].

Each connected set resulting from UE becomes a marker. If we have one marker for each C_i, we say that I is separable by UE. However, UE is prone to producing more than one marker for each C_i, especially in noisy images, thereby leading to oversegmentation of nanoparticles, e.g., Fig. 1c. To avoid the oversegmentation, we propose a noise-robust morphological erosion process with an earlier stopping criterion than that of UE. We call our erosion process *ultimate erosion for convex sets*, UECS for short.

Definition 3.1 (Ultimate erosion for convex sets). *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 $I^{(t)} = \bigcup_i R_i$:*

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

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

We will show that I is separable by UECS under Assumption 3.2.

Assumption 3.2 (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 is connected.*

Intuitively, the assumption is related to the degree of overlaps among particles; please refer to Fig. 2 for an illustration. In real micrographs, one could observe that many overlapping nanoparticles satisfy the assumption. For examples, please see Fig. 5a. The exemplary micrographs depict the chain-linked clusters formed by overlapping nanoparticles. This type of clusters of nanoparticles is separable by UECS.

For a formal statement, we first introduce some notations. Let C be a connected set in $\mathbb{R}^2$ and 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. We also define a supporting set $S(x)$ by

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

(a)　Cases satisfying Assumption 1　　(b) C1 ∩ C2 ∩ C3 =　　(c) C1 \ C2 is empty　　(d) C1 \ C2 is disconnected
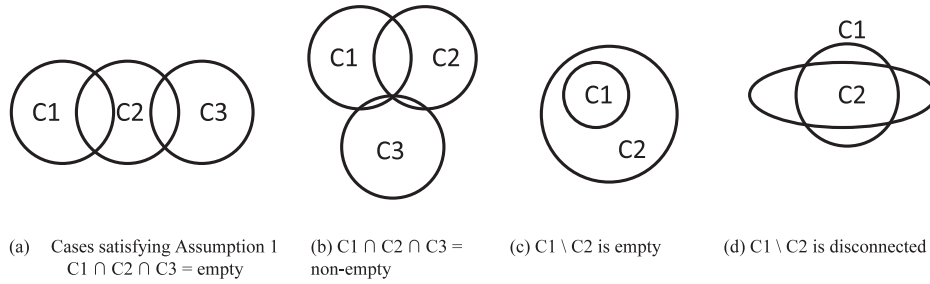　　　C1 ∩ C2 ∩ C3 = empty　　　　non-empty

Fig. 2. Intuitive examples explaining Assumption 3.2: (a) and (b) satisfy the assumption, while (c) and (d) violate the assumption.

The following theorem shows that I is separable by UECS (for its proof, please refer to Appendix A, which can be found in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.163.)

Theorem 3.3 (Separability of UECS). *Suppose that I is a union of n overlapping convex sets, namely, $C_1, \ldots, C_n$, satisfying Assumption 3.2. Then, I is separable by 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 $i = 1, \ldots, n$.*

The *if-and-only-if* condition in the above theorem is related to the size of the allowable overlap between particles. For detailed discussions regarding the implication of this condition, please also refer to Appendix B, available in the online supplemental material.

UECS is less prone to oversegmentation because it uses a noise-robust measure of *convexity* (or conversely, a measure of *concavity*) as its stopping criterion for erosion. Many works were reported regarding how to measure the size of concavity in digital images [39]. Rosenfeld [40] compared three popular measures for concavity in terms of their sensitivity to the coarseness of digital grids, and identified the following concavity measure as the least sensitive one.

Definition 3.4 [40, p. 72]. *Let I be a connected set. If $O = \text{conv}(I)$, $V = O \backslash I$ is called a concavity of I. Suppose that V consists of m connected sets. If we denote the boundary of the jth connected set 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 P is not convex. Using this concavity measure, the stopping criterion of UECS is defined by comparing the measure with threshold ρ, i.e., continue erosion if $c(V) > \rho$, but stop otherwise.

The choice of ρ determines the noise-robustness degree of UECS. If ρ is large, UECS will be more robust to boundary protrusion and intrusion by noise but may lose its separation capability. Conversely, if ρ is too small, UECS will be less robust but more capable of separating the overlaps. Empirically, we observed that ρ in between 0.2 and 0.3 worked well with real micrographs. An exemplary result from UECS ($\rho = 0.2$) is presented in Fig. 1d, where the markers are depicted as the white regions inside the nanoparticles.

3.2 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. Marker-controlled watershed also follows this approach [36]. 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.

Differently from the marker growth approach, we choose to define the contour evidences explicitly; we first extract all the edge pixels from an image and then associate them with each individual marker according to a relevance measure. This edge-to-marker association is used as our contour evidences. This section presents our choice of the relevance measure.

Suppose that we have n markers from UECS, denoted by $\{T_1, T_2, \ldots, T_n\}$, where T_i is the marker of C_i and it is represented by a set of point coordinates in the marker. We also have m edge pixel coordinates detected by an arbitrary edge detection method and 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 [37]. In order to measure the relevance of e_j to T_i, denoted by $\text{rel}(e_j, T_i)$, we define a compound measure rather than a simple measure (e.g., distance). A component composing the compound measure is a distance from e_j to T_i, the same as what is used in the marker-growing approach. We define the distance measure in order to exclude the edge points that locate close to an irrelevant marker by chance. The distance is defined with respect to I (the same I used in the previous section) as

$$g(e_j, T_i) = \min_{x \in T_i} g_j(x), \qquad (1)$$

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 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 overemphasizing the relevance of e_j to markers irrelevant 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

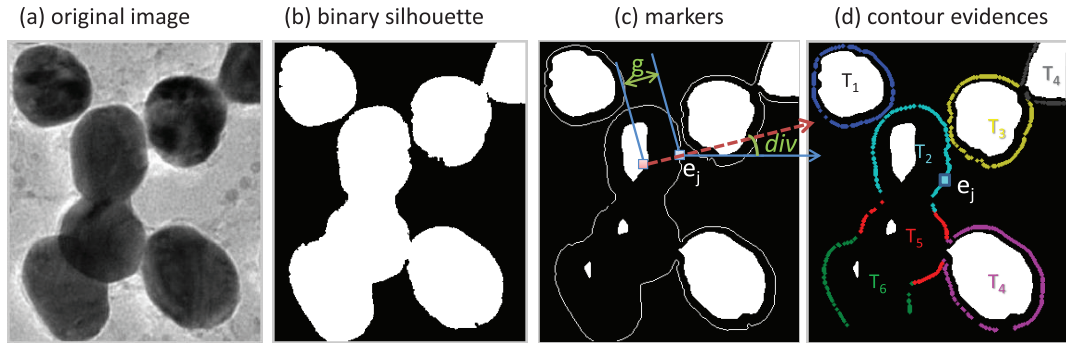| (a) original image | (b) binary silhouette | (c) markers | (d) contour evidences |

Fig. 3. Association of edge pixels by the relevance to markers; (c) illustrates how to compute the distance (g) and the divergence index (*div*) between edge pixel e_j and marker T_i. (d) uses colors to illustrate the association result.

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, if e_j is a substance of 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 Fig. 3c, the solid-line arrow outbound from e_j is the (image intensity) gradient vector at e_j, $\vec{g}(e_j)$, and the dotted-line arrow represents the straight line from T_i to e_j, $\vec{l}(x, e_j)$. The divergence index is simply the cosine of the angle between the two vectors.

Summing up $g(e_j, T_i)$ and $div(e_j, T_i)$ with a weight constant $\lambda \in [0, 1]$, we define the relevance measure of e_j to T_i:

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

where $nIter$ is the number of erosion iterations and both terms are normalized to $(0, 1]$ before being weighted by λ. If $i = \arg\max_k \mathrm{rel}(e_j, T_k)$, e_j becomes an element of the contour evidences for C_i. Throughout this paper, we use $\lambda = 0.5$, i.e., equally weighting the two terms.

By now, one can see that our image segmentation step has two substeps: a modified erosion process (UECS) and the subsequent edge-to-marker association. We acknowledge that there are other robust erosion criteria in terms of marker generation, for example, the extended maxima transform [41]. However, when considering the whole segmentation step altogether, i.e., comparing the proposed UECS followed by edge-to-marker association with the extended maxima transform followed by marker growing, there are two major differences: 1) While both UECS and the extended maxima transform are effective and robust in generating markers, the marker growing step in a watershed method still appears sensitive to noise. The watershed method grows the markers up to the regional minima (watershed lines), which could result in unreasonable

segmentation. 2) The edge-to-marker association performs better when used after UECS than after the extended maxima transform because UECS almost always stops earlier than the extended maxima transform and leaves a sizable marker. The larger markers are preferred and make the edge-to-marker association effective since the association is partially based on a distance measure.

4 CONTOUR EVOLUTION WITH MULTIPLE REFERENCE SHAPES

Suppose that we have 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 we are dealing with 2D images. The markers identified in Section 3 are used to locate C_is but they will not be used explicitly in the subsequent inference.

We want 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, we propose an approach that simultaneously performs 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 $t \in [0, 1]$,

$$\boldsymbol{f}_i(t) = \sum_{h=0}^{p-1} \phi_{h,d}(t) \boldsymbol{p}_{i,h}, \quad (3)$$

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 [42], [43], [44]. For the time being, we assume that t_{ij} is known. We denote m_i contour evidences collectively by a $2m_i \times 1$ vector $\mathbf{x}_i$, which is formed by binding $\mathbf{e}_{ij}$ in a

row-wise fashion. 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 a row-wise fashion.

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, we separate the pose parameters 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, and R_θ is a transformation matrix for a rotation by θ in counterclockwise. The model for the whole feature $\boldsymbol{p}_i$ is

$$\boldsymbol{p}_i = \frac{1}{s} \boldsymbol{Q}_\theta \tilde{\boldsymbol{p}}_i + \boldsymbol{H}\boldsymbol{c}, \tag{4}$$

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.

We constrain the normalized shape feature $\tilde{\boldsymbol{p}}_i$ by the prior shape, 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. We define the hidden group membership vectors $\boldsymbol{g}_i = (g_{i1}, \dots, g_{iK})^t$, where 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

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

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, we denote them separately for each shape group by s_{ik}, θ_{ik}, and $\boldsymbol{c}_{ik}$, and denote them collectively for all ks by $\boldsymbol{s}_i$, θ_i, and $\boldsymbol{c}_i$. Using (4), $\boldsymbol{p}_i$ is distributed as

$$P(\boldsymbol{p}_i|\boldsymbol{g}_i, \boldsymbol{s}_i, \theta_i, \boldsymbol{c}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{p}_i; \boldsymbol{\mu}_{ik}, \boldsymbol{A}_{ik}\boldsymbol{\Sigma}_k \boldsymbol{A}_{ik}^t)^{g_{ik}},$$

where $\boldsymbol{\mu}_{ik} = \boldsymbol{A}_{ik}\boldsymbol{\mu}_k + \boldsymbol{H}\boldsymbol{c}_{ik}$ and $\boldsymbol{A}_{ik} = \frac{1}{s_{ik}}\boldsymbol{Q}_{\theta_{ik}}$. We also put a multinomial distribution on the hidden matrix $\boldsymbol{g}_i$ as its prior distribution, i.e.,

$$P(\boldsymbol{g}_i|\boldsymbol{\alpha}) = \prod_{k=1}^{K} \alpha_k^{g_{ik}}, \tag{5}$$

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

The ultimate goal of this section is to obtain the contours of convex shapes C_is and to determine which shape group

C_i belongs to. This problem corresponds to estimating the hidden variables $Z = \{\boldsymbol{p}_i, \boldsymbol{g}_i; i = 1, \dots, N\}$. However, the hidden variables depend on the unknown parameters $\boldsymbol{\Theta} = \{\sigma^2, \boldsymbol{\alpha}, \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}, \{\boldsymbol{s}_i, \theta_i, \boldsymbol{c}_i\}\}$. We estimate the unknown parameters by maximizing the following marginal likelihood of observing contour evidences $X = \{\mathbf{x}_i; i = 1, \dots, N\}$ with respect to $\boldsymbol{\Theta}$:

$$P(X|\boldsymbol{\Theta}) = \int_Z P(X, Z|\boldsymbol{\Theta}) dZ. \tag{6}$$

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

$$P(Z|\widehat{\boldsymbol{\Theta}}, X) = \frac{P(X, Z|\widehat{\boldsymbol{\Theta}})}{P(X|\widehat{\boldsymbol{\Theta}})}. \tag{7}$$

Conceptually, this solution approach involves two complicated optimization problems, which are not easy to solve. In practice, it is realized through an iterative solver, the expectation conditional maximization, proven to converge to a local maximum [45].

4.1 Expectation Maximization (EM) via the ECM Algorithm

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

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

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

$$\begin{aligned} E_Z&[\log P(X, Z|\boldsymbol{\Theta})] \\ &\propto \sum_{i=1}^{N} \sum_{k=1}^{K} \beta_{ik}[2\log \alpha_k - \log\det(\boldsymbol{\Sigma}_k) + 4p\log s_{ik} \\ &\quad - (\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik})^t \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik}) \\ &\quad + 2s_{ik}\boldsymbol{m}_{ik}^t \boldsymbol{Q}_{\theta_{ik}}\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{\mu}_k + s_{ik}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H}\boldsymbol{c}_{ik}) \\ &\quad - s_{ik}^2 \mathrm{trace}(\boldsymbol{Q}_{\theta_{ik}}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{\Delta}_{ik}) - 2m_i\log\sigma^2 \\ &\quad - \sigma^{-2}\boldsymbol{x}_i^t \boldsymbol{x}_i + 2\sigma^{-2}\gamma_{ik}^t \boldsymbol{\Phi}_i^t \boldsymbol{x}_i \\ &\quad - \sigma^{-2}\mathrm{trace}(\boldsymbol{\Phi}_i^t \boldsymbol{\Phi}_i \boldsymbol{\Delta}_{ik})], \end{aligned} \tag{8}$$

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

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

Here, we omitted a lengthy derivation of the expectation because of page limitation. For the full derivation of all expressions in this section, please refer to Park [46]. In the

M-step, we maximize the expectation in (8) 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:

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

We cannot obtain the closed form expressions of the local optima for $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$, θ_{ik}, s_{ik}, and c_{ik} since their FONCs are entangled with one another in complicated forms. We could instead perform the M-step iteratively by the Newton Raphson, but we want to avoid expensive iterations as well. There are two other possible options to proceed with the M-step without iterations; the first one is to improve the expected log likelihood (8) rather than to maximize it for every M-step, resulting in a GEM algorithm [47], and another one is to use the ECM algorithm [45]. The first option does not in general converge appropriately, but the second option does. For this reason, we take the second option.

The ECM algorithm partitions $\boldsymbol{\Theta}$ into L subgroups and solves L optimizations, where each optimization maximizes (8) with respect to one subgroup of $\boldsymbol{\Theta}$, provided that the other groups remained to be their previous values. For the applications to real micrographs in Section 5, we used $L = 5$ with five subgroups of parameters $\{\boldsymbol{\mu}_k\}$, $\{\boldsymbol{\Sigma}_k\}$, $\{\theta_{ik}\}$, $\{s_{ik}\}$, and $\{c_{ik}\}$. From the FONCs with respect to each subgroup, we have the following solutions for the M-step of the ECM algorithm:

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

$$\boldsymbol{\Sigma}_k = \frac{\sum_{i=1}^{N} \beta_{ik} \left(\boldsymbol{\mu}_{ik} + s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H} c_{ik} \right) \left(\boldsymbol{\mu}_{ik} + s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H} c_{ik} \right)^t}{\sum_{i=1}^{N} \beta_{ik}}$$
$$+ \frac{s_{ik}^2 \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{\Delta}_{ik} \boldsymbol{Q}_{\theta_{ik}} - s_{ik} \left(\boldsymbol{\mu}_k + s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H} c_{ik} \right) \boldsymbol{m}_{ik}^t \boldsymbol{Q}_{\theta_{ik}}}{\sum_{i=1}^{N} \beta_{ik}}$$
$$- \frac{s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{m}_{ik} \left(\boldsymbol{\mu}_k + s_{ik} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H} c_{ik} \right)^t}{\sum_{i=1}^{N} \beta_{ik}},$$

$$c_{ik} = \left(\boldsymbol{H}^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{H} \right)^{-1} \boldsymbol{H}^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \left(\boldsymbol{Q}_{\theta_{ik}}^t \boldsymbol{m}_{ik} - \frac{\boldsymbol{\mu}_k}{s_{ik}} \right),$$

$$s_{ik} = \frac{v + \sqrt{v^2 + 8p\beta_{ik}u}}{2u},$$

where

$$u = \text{trace} \left[\boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{Q}_{\theta_{ik}}^t \left(\boldsymbol{\Delta}_{ik} + \boldsymbol{H} c_{ik} c_{ik}^t \boldsymbol{H}^t - 2\boldsymbol{H} c_{ik} \boldsymbol{m}_{ik}^t \right) \right]$$
$$v = \left(\boldsymbol{m}_{ik} - \boldsymbol{H} c_{ik} \right)^t \boldsymbol{Q}_{\theta_{ik}} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k.$$

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

$$4(a^2 + b^2)x^4 + 8(bd - ac)dx^3 + 4(c^2 + d^2 - a^2 - b^2)x^2 + 4(ac - 2bd)x + a^2 - 4d^2 = 0,$$

where

$$a = s_{ik} \text{trace} \left(\left(\boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p^*}^t + \boldsymbol{I}_{p^*} \boldsymbol{\Sigma}_k^{-1} \right) \left(\boldsymbol{\Delta}_{ik} + \boldsymbol{H} c_{ik} c_{ik}^t \boldsymbol{H}^t \right) \right),$$
$$b = s_{ik} \text{trace} \left(\left(\boldsymbol{\Sigma}_k^{-1} - \boldsymbol{I}_{p^*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p^*}^t \right) \left(\boldsymbol{\Delta}_{ik} + \boldsymbol{H} c_{ik} c_{ik}^t \boldsymbol{H}^t \right) \right)$$
$$- 2s_{ik} \text{trace} \left(\left(\boldsymbol{\Sigma}_k^{-1} - \boldsymbol{I}_{p^*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{I}_{p^*}^t \right) \boldsymbol{H} c_{ik} \boldsymbol{m}_{ik} \right),$$
$$c = (\boldsymbol{m}_{ik} - \boldsymbol{H} c_{ik})^t \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k,$$
$$d = (\boldsymbol{m}_{ik} - \boldsymbol{H} c_{ik})^t \boldsymbol{I}_{p^*} \boldsymbol{\Sigma}_k^{-1} \boldsymbol{\mu}_k,$$

and $\boldsymbol{I}_{p^*}$ is an antisymmetric 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 [48].

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 $\boldsymbol{X}$ are obtained by

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

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, we determine the shape of C_i to be k if $\widehat{g}_{ik} = 1$, and reconstruct the contour by plugging $\widehat{\boldsymbol{p}}_i$ into (3).

4.2 Approximate Data Parameterization

We have 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 [49], [50], or the intrinsic parameterization [42]. 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 we do not have. The intrinsic parametrization, on the other hand, is more general but will add another family of parameters to the already large set of parameters under estimation in the ECM. In the end, we choose 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, $\{\mathbf{e}_{i1}, \mathbf{e}_{i2}, \dots, \mathbf{e}_{im_i}\}$:
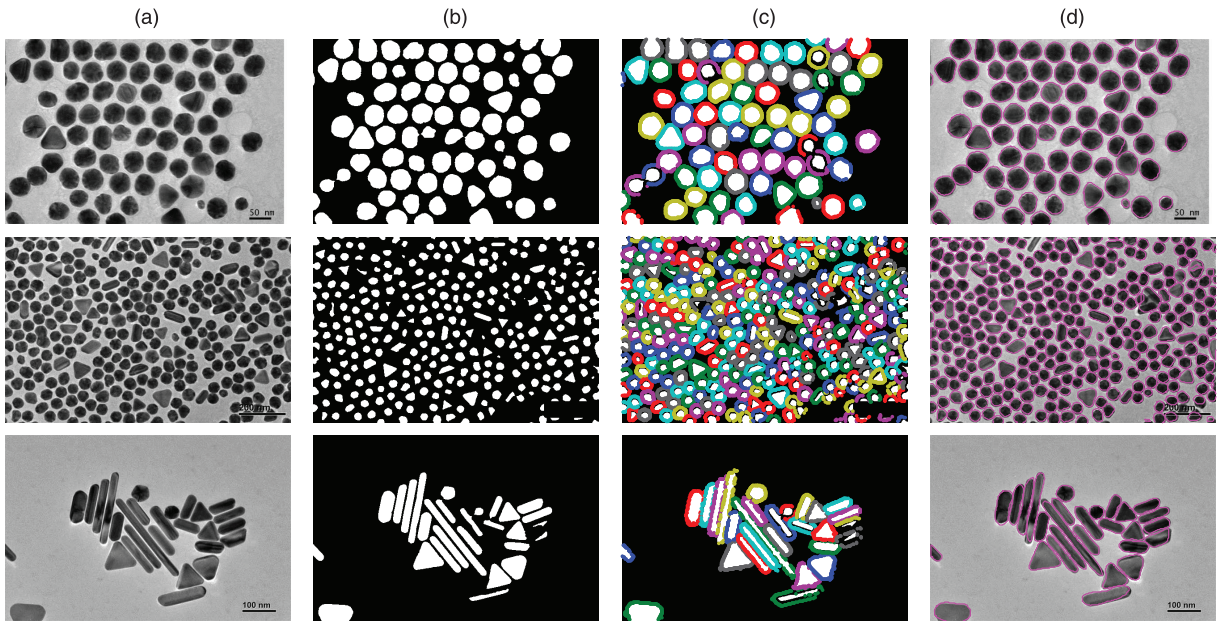
Fig. 4. Results from low-degree overlapping cases. (a) Original images, (b) markers from the UECS, (c) contour evidences (Section 3.2) , (d) final contours by the ECM.

1. Find a convex hull inscribing the contour evidences by the `Qhull` algorithm [51].
2. Sequentially order all points on the convex hull in counterclockwise (or clock wise) 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 computational efficiency. In addition, the parameterization is not affected by the noisy contour evidences which locate inside a nanoparticle. When the noisy evidences locate outside a nanoparticle, the approximate data parameterization may be distorted. This problem can be addressed if we take the convex hull inscribing the majority of points except for a few outliers.

5   APPLICATIONS TO REAL MICROGRAPHS

This section shows how our proposed method works with real electron micrographs and compares its performance with seven state-of-the-art methods: marker-controlled watershed segmentation with h-dome transform [10, WHD], marker-controlled watershed with h-maxima transform [41, WHM], normalized-cut [16, N-Cut], multiphase active contour [23, MPAC], sliding band filter [31, SBF], morphological multiscale method [15, MSD], and iterative voting method [32, IVM].

For the numerical comparison, we implemented WHD, WHM, SBF, and MSD by ourselves. We used the implementation made by the corresponding authors for N-Cut [16] and IVM [32], and the implementation made by Wu [52] for MPAC. For generating markers in the two

watershed methods, in WHD, we took the h-dome transform of an original grayscale image with $h = 6$, and in WHM, we took the extended maxima transform with $h = 8$ on the distance transform of an image.

We chose 12 different real micrographs obtained from a synthesis process of gold nanoparticles. In order to see the recognition quality of nanoparticles having various degrees of overlap, we categorized the micrographs into three groups according to their degrees of overlap: low, medium, and high. The micrographs of "low" overlapping degree have slightly touching among particles. In the cases of the "medium" degree, most nanoparticles overlap and the overlapping structures conform with Assumption 3.2. The high degree cases are when nanoparticles overlap more severely so that the overlapping clearly violates Assumption 3.2.

5.1   Results of Contour Inference

The results of segmentation and contour inference of the 12 micrographs are presented in Figs. 4, 5, 6, and 7. Each figure has four columns. The first column of each figure contains the original micrographs. The corresponding binary silhouettes were obtained by applying the alternative sequence filtering [37], followed by Otsu's optimum global thresholding [53]. The UECS proposed in Section 3.1 was applied to the binary silhouettes for obtaining the second column, where a white-colored connected region implies one marker. Contour evidences (pixels at the boundaries of particles) were first extracted by Canny's edge detection method [54], and then they were associated with the markers by using the procedure in Section 3.2. After the association, the algorithm filtered out some noise edge outliers based on the mean and standard deviation of g defined in (1). In the third column of each figure, 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 4.
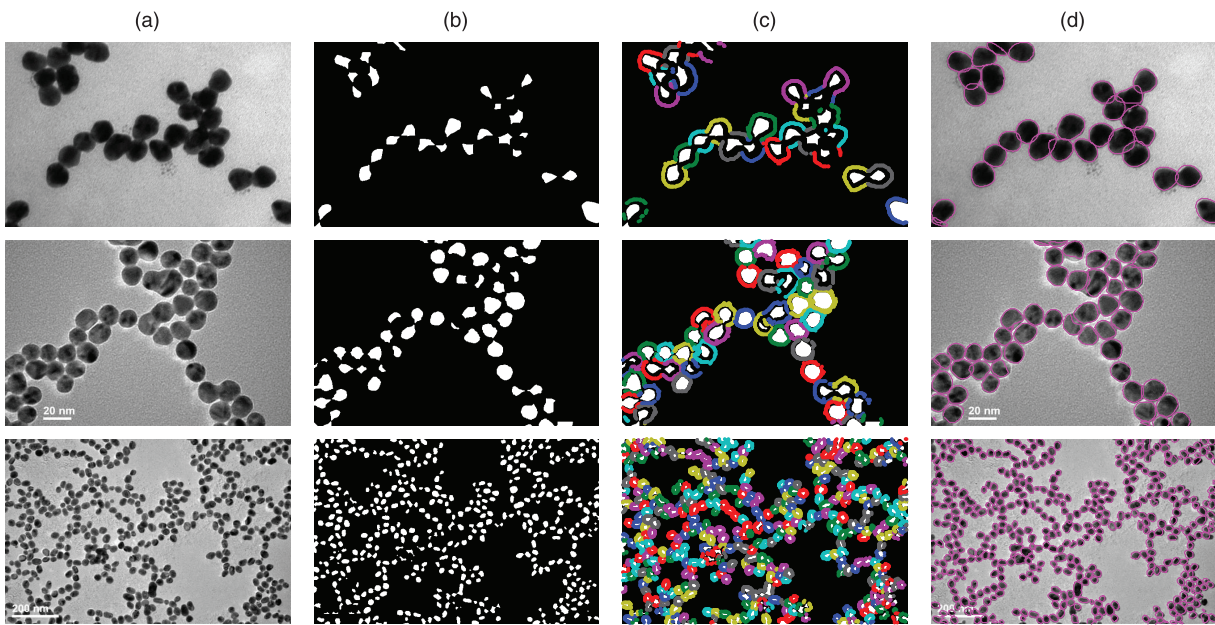
Fig. 5. Results from the medium-degree overlapping cases.

The proposed UECS correctly identified one marker per particle for most of the cases from Figs. 4, 5, and 6. Fig. 7 has severe overlaps among the nanoparticles, and as a result, UECS sometimes identified one marker for multiple nanoparticles.

The association between the markers and the edge pixels generally looks reasonable, although some noise edge pixels have been classified as valid contour evidences. The inference results match 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. We want to note that most of those cases cannot be easily handled even by human vision.

The accuracy of our proposed method was quantitatively compared with the seven methods as mentioned earlier. For each of the 12 micrographs, we manually counted the total number of nanoparticles and the number of the particles correctly separated by each of the methods in comparison. The results are tabularized in Table 1. Overall, our proposed method is the best performer for eight of the 12 micro-graphs, ties for two cases, and performs very similarly to the best performer for the remaining two micrographs.

N-Cut and MPAC suffer from undersegmentation because their image segmentation is guided by image intensities but overlapping particles have similar image
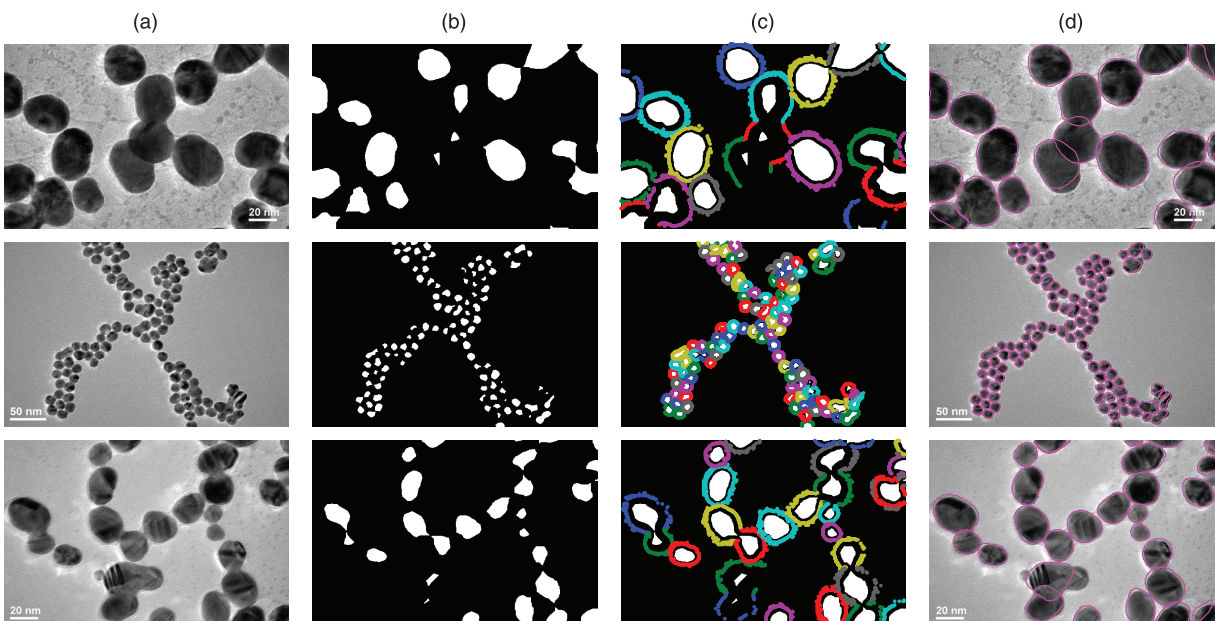

Fig. 6. Results from the medium-degree overlapping cases (continued).
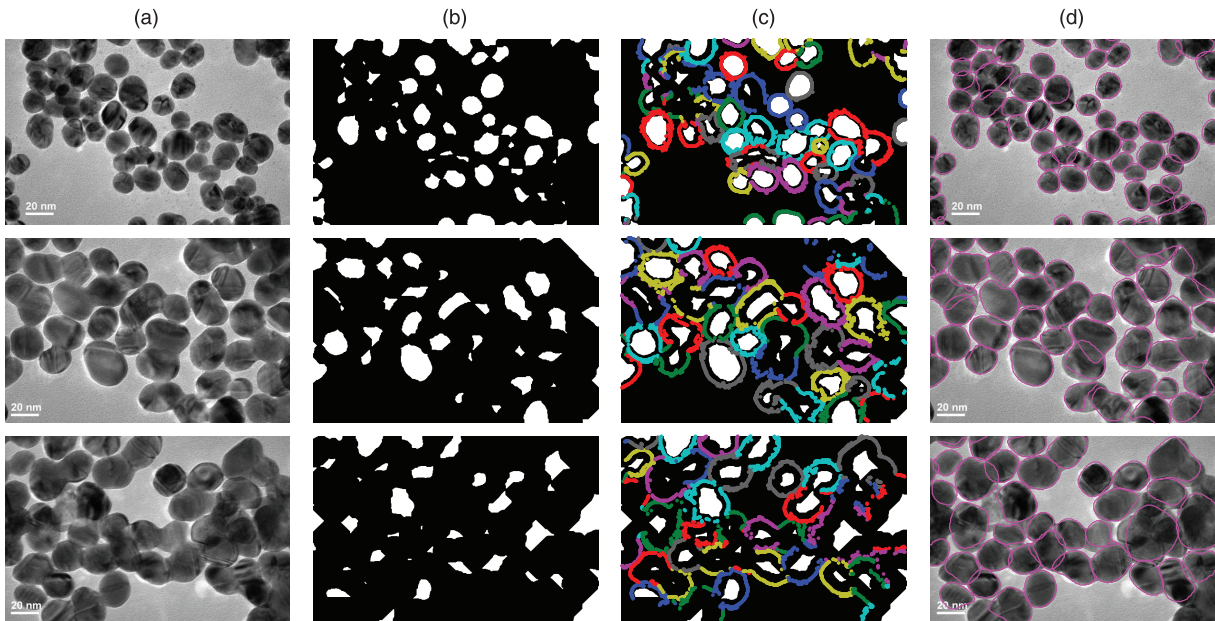
Fig. 7. Results from the high-degree overlapping cases.

TABLE 1
Comparison of Performances on Nanoparticle Recognition

Samples	Degree of overlap	Total # of particles	The number of the correctly recognized ones							
			Our method	N-Cut	MPAC	SBF	IVM	MSD	WHM	WHD
Figure 4, row 1	Low	76	**76**	74	69	69	74	**76**	**76**	**76**
Figure 4, row 2	Low	307	299	284	199	215	291	**300**	296	283
Figure 4, row 3	Low	28	**28**	22	19	1	4	24	24	22
Figure 5, row 1	Medium	28	26	6	2	**28**	25	20	20	20
Figure 5, row 2	Medium	52	**48**	29	22	43	45	**48**	45	39
Figure 5, row 3	Medium	459	**437**	298	147	262	227	429	424	421
Figure 6, row 1	Medium	19	**17**	10	8	6	16	16	15	6
Figure 6, row 2	Medium	108	**103**	56	44	99	85	92	82	70
Figure 6, row 3	Medium	29	**25**	12	13	19	21	23	18	14
Figure 7, row 1	High	63	**54**	31	12	42	42	40	38	34
Figure 7, row 2	High	44	**34**	23	11	28	27	28	28	23
Figure 7, row 3	High	45	**33**	20	6	25	24	22	19	20

intensities (please refer to Fig. 8 for an exemplary result). SBF and IVM perform comparably well but both struggle with the cases where the particle sizes vary drastically. The third micrograph in Fig. 4 corresponds to such a case. Among the two watershed algorithms, WHM works better than WHD, but both are outperformed by the proposed method. That is mainly due to inaccuracy in the marker growing step, leading to unreasonable segmentation (please refer to Fig. 9 for examples). Our method is comparable to MSD but still holds a competitive edge in most cases. The underperformance of MSD, compared with the proposed method, comes mainly from two reasons: undersegmentation and inaccurate marker grow- ing. The undersegmentation is caused by the smoothing step in the marker-generation step of MSD (m-fold dilation) but fine-tuning this smoothing step for avoiding the undersegmentation does not appear to be easy because of the delicate tradeoff between oversegmentation and undersegmentation. The marker-growing step in MSD is similar to the region growing in the watershed. Like the watershed's segmentation shown in Fig. 9, MSD's marker growing step can also lead to unreasonable segmentation.

Fig. 8 presents one exemplary result of the segmentation and contour inference performed by our method and the seven competing methods. We can clearly observe under- segmentation by MPAC and N-Cut. WHD also suffers from oversegmentation due to its sensitivity to noise. The contour estimation by SBF looks rough, but this roughness can be
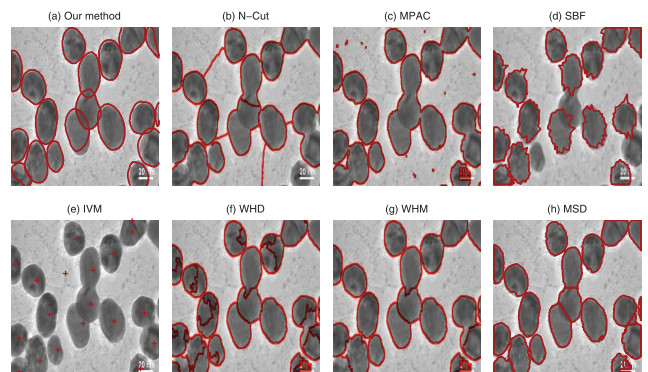

Fig. 8. Results of segmentation. The red lines correspond to the contours identified by our method and six other methods in comparison. The red dots correspond to the center locations identified by IVM.
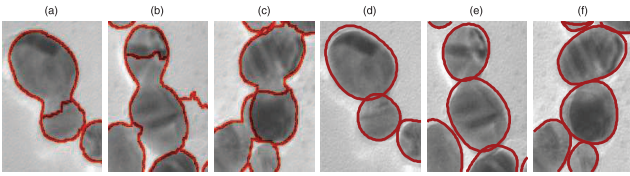
Fig. 9. Comparison of segmentation results: h-maxima followed by marker growing ((a)-(c)) versus UECS followed by edge-to-marker association ((d)-(e)).

smoothed out by curve fitting. SBF also misses a few particles. IVM's segmentation result is presented differently because the author's code produces the center locations of the segmented objects rather than the segmentation boundaries. Generally, IVM, WHM, and MSD perform well.

We recorded the total computation times spent by the methods for the micrograph used in Fig. 8. SBF spends the longest time, 541 seconds, N-cut 256 seconds, MPAC 273 seconds, MSD 78 seconds, and WHD/WHM/IVM take less than 1 second. Our method takes 72 seconds, where 28 seconds are for UECS and association of contour evidences, and the rest of the time is for contour inference and shape classification. Our method is not the fastest among the eight methods. However, please note that our method performs shape inference and classification along with segmentation, while the other seven methods perform either segmentation/inference (MPAC) or only segmentation (the other six methods).

5.2 Results of Shape Classification

We chose four out of the 12 micrographs to evaluate the accuracy of shape classification. The four figures were chosen because they contain various types of particle shapes, while the remaining eight figures contain primarily spherical nanoparticles. The classification results are presented by labeling the nanoparticles with one character symbol representing shape classes; "t" = triangle, "b" = rectangle, "c" = circle, and "r" = rod. Please refer to Fig. 10.
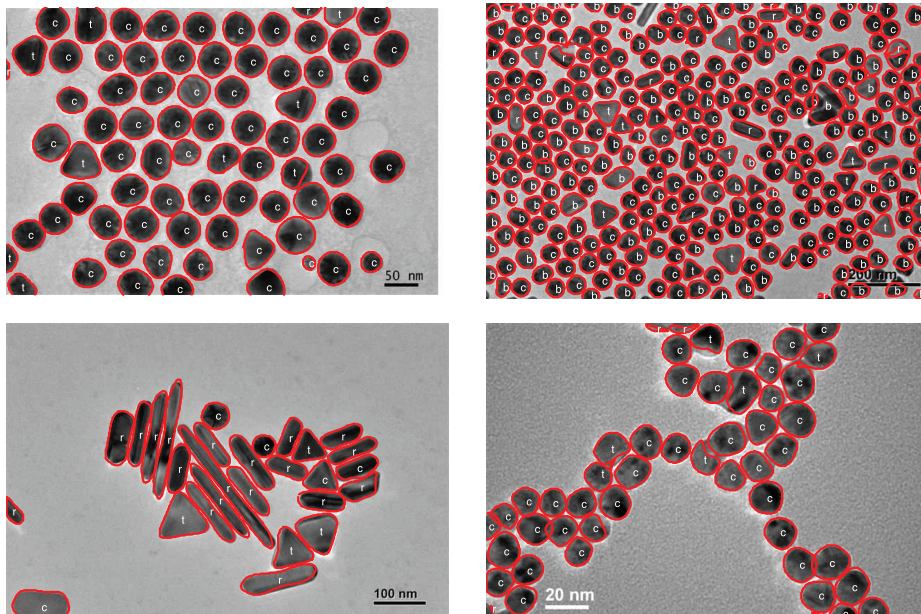
We compare the automated classification outcomes with how humans would classify the shapes. In the top-left figure, the result is accurate except for two missclassifications; our method classifies a triangle as a circle and classifies a circle as a triangle. Such missclassifications are also observed 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 missclassification is 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. We leave this issue to our future research. Overall, we believe that our automated method performs the shape classification task reasonably well.

6 CONCLUSION AND DISCUSSION

In this paper, we have proposed a two-stage approach to tackle the automated morphology analysis problems of overlapping nanoparticles. The unique contributions of this paper are:

1. to propose a modified ultimate erosion process (UECS), followed by an edge-to-marker association, to separate overlapping convex objects,
2. to provide the justification on the use of UECS in terms of its separation capability for a chain-linked 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 UECS and evidence association, which allows 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 [55], our proposed model and solution procedure can solve the

Fig. 10. Shape classification. Each particle's shape is labeled as: t = triangle, b = rectangle, c = circle, and r = rod.

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 12 electron micrographs of overlapping nanoparticles. The results show that the proposed method performs better than the existing methods when both accuracy and computation efficiency are considered. This is not surprising because the proposed method 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 nanoobjects. We have argued, and would like to reiterate 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.

The handling of nanoparticle images are our principal interests in this paper. Nonetheless, we believe that the method could be useful for other image analysis problems of similar nature, i.e., if the overlapping objects are of convex shapes, and the overlap happens in a chain-link configuration, as described in Assumption 3.2. Other applications that could potentially benefit from our method may include some biocell segmentation problems (e.g., [10], [11], [14], [18], [20], [56], [57]) and morphometry analysis of mineral particles (e.g., [58], [59]).

ACKNOWLEDGMENTS

REFERENCES

[1] Z.L. Wang, J.M. Petroski, T.C. Green, and M.A. El-Sayed, "Shape Transformation and Surface Melting of Cubic and Tetrahedral Platinum Nanocrystals," *The J. Physical Chemistry B,* vol. 102, no. 32, pp. 6145-6151, 1998.

[2] M.A. El-Sayed, "Some Interesting Properties of Metals Confined in Time and Nanometer Space of Different Shapes," *Accounts of Chemical Research,* vol. 34, no. 4, pp. 257-264, 2001.

[3] C.L. Nehl, H. Liao, and J.H. Hafner, "Optical Properties of Star-Shaped Gold Nanoparticles," *Nano Letters,* vol. 6, no. 4, pp. 683-688, 2006.

[4] Y. Pan, S. Neuss, A. Leifert, M. Fischler, F. Wen, U. Simon, G. Schmid, W. Brandau, and W. Jahnen-Dechent, "Size-Dependent Cytotoxicity of Gold Nanoparticles," *Small,* vol. 3, no. 11, pp. 1941-1949, 2007.

[5] M. Wortis, *Chemistry and Physics of Solid Surfaces.* Springer-Verlag, 1988.

[6] A. McFarland and R. Van Duyne, "Single Silver Nanoparticles as Real-Time Optical Sensors with Zeptomole Sensitivity," *Nano Letters,* vol. 3, no. 8, pp. 1057-1062, 2003.

[7] O. Glotov, "Image Processing of the Fractal Aggregates Composed of Nanoparticles," *Russian J. Physical Chemistry A, Focus on Chemistry,* vol. 82, no. 13, pp. 2213-2218, 2008.

[8] L.C. Chen and C.C. Ho, "Development of Nanoparticle Shape Measurement and Analysis for Process Characterization of TiO_2 Nanoparticle Synthesis," *Rev. Advanced Material Science,* vol. 18, no. 8, pp. 677-684, 2008.

[9] R. Fisker, J. Carstensen, M. Hansen, F. Bødker, and S. Mørup, "Estimation of Nanoparticle Size Distributions by Image Analysis," *J. Nanoparticle Research,* vol. 2, no. 3, pp. 267-277, 2000.

[10] N. Malpica, C. de Solórzano, J. Vaquero, A. Santos, I. Vallcorba, J. García-Sagredo, and F. del Pozo, "Applying Watershed Algorithms to the Segmentation of Clustered Nuclei," *Cytometry Part A,* vol. 28, no. 4, pp. 289-297, 1997.

[11] F. Tek, A. Dempster, and I. Kale, "Blood Cell Segmentation Using Minimum Area Watershed and Circle Radon Transformations," *Math. Morphology: 40 Years On,* vol. 30, pp. 441-454, 2005.

[12] P. Umesh Adiga and B. Chaudhuri, "An Efficient Method Based on Watershed and Rule-Based Merging for Segmentation of 3-D Histo-Pathological Images," *Pattern Recognition,* vol. 34, no. 7, pp. 1449-1458, 2001.

[13] G. Lin, M. Chawla, K. Olson, J. Guzowski, C. Barnes, and B. Roysam, "Hierarchical, Model-Based Merging of Multiple Fragments for Improved Three-Dimensional Segmentation of Nuclei," *Cytometry Part A,* vol. 63, no. 1, pp. 20-33, 2005.

[14] J. Cheng and J. Rajapakse, "Segmentation of Clustered Nuclei with Shape Markers and Marking Function," *IEEE Trans. Biomedical Eng.,* vol. 56, no. 3, pp. 741-748, Mar. 2009.

[15] O. Schmitt and M. Hasse, "Morphological Multiscale Decomposition of Connected Regions with Emphasis on Cell Clusters," *Computer Vision and Image Understanding,* vol. 113, no. 2, pp. 188-201, 2009.

[16] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[17] P. Felzenszwalb and D. Huttenlocher, "Efficient Graph-Based Image Segmentation," *Int'l J. Computer Vision,* vol. 59, no. 2, pp. 167-181, 2004.

[18] O. Daněk, P. Matula, C. Ortiz-De-Solórzano, A. Muñoz-Barrutia, M. Maška, and M. Kozubek, "Segmentation of Touching Cell Nuclei Using a Two-Stage Graph Cut Model," *Proc. 16th Scandinavian Conf. Analysis,* pp. 410-419, 2009.

[19] P. Bamford and B. Lovell, "Unsupervised Cell Nucleus Segmentation with Active Contours," *Signal Processing,* vol. 71, no. 2, pp. 203-213, 1998.

[20] S. Nath, K. Palaniappan, and F. Bunyak, "Cell Segmentation Using Coupled Level Sets and Graph-Vertex Coloring," *Proc. Ninth Int'l Conf. Image Computing and Computer-Assisted Intervention,* pp. 101-108, 2006.

[21] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models," *Int'l J. Computer Vision,* vol. 1, no. 4, pp. 321-331, 1988.

[22] T. Chan and L. Vese, "Active Contours without Edges," *IEEE Trans. Image Processing,* vol. 10, no. 2, pp. 266-277, Feb. 2001.

[23] L. Vese and T. Chan, "A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model," *Int'l J. Computer Vision,* vol. 50, no. 3, pp. 271-293, 2002.

[24] Y. Chen, H. Tagare, S. Thiruvenkadam, F. Huang, D. Wilson, K. Gopinath, R. Briggs, and E. Geiser, "Using Prior Shapes in Geometric Active Contours in a Variational Framework," *Int'l J. Computer Vision,* vol. 50, no. 3, pp. 315-328, 2002.

[25] A. Foulonneau, P. Charbonnier, and F. Heitz, "Multi-Reference Shape Priors for Active Contours," *Int'l J. Computer Vision,* vol. 81, no. 1, pp. 68-81, 2009.

[26] F. Lecumberry, A. Pardo, and G. Sapiro, "Simultaneous Object Classification and Segmentation with High-Order Multiple Shape Models," *IEEE Trans. Image Processing,* vol. 19, no. 3, pp. 625-635, Mar. 2010.

[27] E. Hodneland, X. Tai, and H. Gerdes, "Four-Color Theorem and Level Set Methods for Watershed Segmentation," *Int'l J. Computer Vision,* vol. 82, no. 3, pp. 264-283, 2009.

[28] Y. Zhang and B. Matuszewski, "Multiphase Active Contour Segmentation Constrained by Evolving Medial Axes," *Proc. 16th IEEE Int'l Conf. Image Processing,* pp. 2993-2996, 2009.

[29] L. Ng and L. Ibanez, "Narrow Band to Image Registration in the Insight Toolkit," *Biomedical Image Registration,* vol. 2717, pp. 271-280, 2003.

[30] M. Marcuzzo, P. Quelhas, A. Mendonça, and A. Campilho, "Evaluation of Symmetry Enhanced Sliding Band Filter for Plant Cell Nuclei Detection in Low Contrast Noisy Fluorescent Images," *Proc. Int'l Conf. Analysis and Recognition,* pp. 824-831, 2009.

[31] P. Quelhas, M. Marcuzzo, A.M. Mendonca, and A. Campilho, "Cell Nuclei and Cytoplasm Joint Segmentation Using the Sliding Band Filter," *IEEE Trans. Medical Imaging,* vol. 29, no. 8, pp. 1463-1473, Aug. 2010.

[32] B. Parvin, Q. Yang, J. Han, H. Chang, B. Rydberg, and M. Barcellos-Hoff, "Iterative Voting for Inference of Structural Saliency and Characterization of Subcellular Events," *IEEE Trans. Image Processing,* vol. 16, no. 3, pp. 615-623, Mar. 2007.

[33] G. Matheron, *Random Sets and Integral Geometry.* Wiley, 1975.

[34] P. Qiu and J. Sun, "Local Smoothing Image Segmentation for Spotted Microarray Images," *J. Am. Statistical Assoc.,* vol. 102, pp. 1129-1144, 2007.

[35] P. Qiu and J. Sun, "Using Conventional Edge Detectors and Post-Smoothing for Segmentation of Spotted Microarray Images," *J. Computational and Graphical Statistics,* vol. 18, pp. 147-1644, 2009.

[36] E. Dougherty, *Digital Image Processing Methods.* CRC, 1994.

[37] R.C. Gonzalez and R.E. Woods, *Digital Image Processing,* third ed. Prentice Hall, 2002.

[38] R. Schneider, *Convex Bodies: The Brunn-Minkowski Theory.* Cambridge Univ. Press, 1993.

[39] C. Ronse, "A Bibliography on Digital and Computational Convexity (1961-1988)," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, no. 2, pp. 181-190, Feb. 1989.

[40] A. Rosenfeld, "Measuring the Sizes of Concavities," *Pattern Recognition Letters,* vol. 3, no. 1, pp. 71-75, 1985.

[41] E. Bengtsson, C. Wahlby, and J. Lindblad, "Robust Cell Image Segmentation Methods," *Pattern Recognition and Image Analysis,* vol. 14, no. 2, pp. 157-167, 2004.

[42] J. Hoschek, "Intrinsic Parametrization for Approximation," *Computer Aided Geometric Design,* vol. 5, no. 1, pp. 27-31, 1988.

[43] A. Goshtasby, "Grouping and Parameterizing Irregularly Spaced Points for Curve Fitting," *ACM Trans. Graphics,* vol. 19, no. 3, pp. 185-203, 2000.

[44] E. Saux and M. Daniel, "An Improved Hoschek Intrinsic Parametrization," *Computer Aided Geometric Design,* vol. 20, nos. 8/9, pp. 513-521, 2003.

[45] X. Meng and D. Rubin, "Maximum Likelihood Estimation via the ECM Algorithm: A General Framework," *Biometrika,* vol. 80, no. 2, pp. 267-278, 1993.

[46] C. Park, "Automate Morphology Analysis of Nanoparticles," PhD dissertation, Texas A&M Univ., 2011.

[47] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Soc., Series B,* vol. 39, no. 1, pp. 1-38, 1977.

[48] J. Tignol, *Galois' Theory of Algebraic Equations.* World Scientific Pub Co., Inc., 2001.

[49] M. Epstein, "On the Influence of Parametrization in Parametric Interpolation," *SIAM J. Numerical Analysis,* vol. 13, no. 2, pp. 261-268, 1976.

[50] E. Lee, "Choosing Nodes in Parametric Curve Interpolation," *Computer-Aided Design,* vol. 21, no. 6, pp. 363-370, 1989.

[51] C.B. Barber, D. Dobkin, and H. Huhdanpaa, "The Quickhull Algorithm for Convex Hulls," *ACM Trans. Math. Software,* vol. 22, no. 4, pp. 469-483, 1996.

[52] R. Wu, "Rex's Tribe of Image Processing at http://sites.google.com/site/rexstribeofimageprocessing/," 2004.

[53] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Systems, Man, and Cybernetics,* vol. SMC-9, no. 1, pp. 62-66, Jan. 1979.

[54] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 6, pp. 679-698, Nov. 1986.

[55] Z. Tu, S. Zheng, and A. Yuille, "Shape Matching and Registration by Data-Driven EM," *Computer Vision and Image Understanding,* vol. 109, no. 3, pp. 290-304, 2008.

[56] Y. Kim, H. Kim, C. Park, T. Park, J. Kim, Y. Choi, Y. Kim, K. Lee, and C. Kang, "Morphological Analysis of Cells by Scanning Electron Microscopy," *Japanese J. Applied Physics-Part 1 Regular Papers and Short Notes,* vol. 47, no. 2, pp. 1325-1328, 2008.

[57] A. Pinidiyaarachchi and C. Wahlby, "Seeded Watersheds for Combined Segmentation and Tracking of Cells," *Proc. 13th Int'l Conf. Image Analysis and Processing,* pp. 336-343, 2005.

[58] S. Al-Thyabat and N. Miles, "An Improved Estimation of Size Distribution from Particle Profile Measurements," *Powder Technology,* vol. 166, no. 3, pp. 152-160, 2006.

[59] W. Wang, "Image Analysis of Size and Shape of Mineral Particles," *Proc. Fourth Int'l Conf. Fuzzy Systems and Knowledge Discovery,* vol. 4, pp. 41-44, 2007.

Chiwoo Park received the BS degree from Seoul National University and the PhD degree from Texas A&M University, all in industrial engineering. He is currently an assistant professor in the Department of Industrial and Manufacturing Engineering at Florida A&M and Florida State University. His research interests include data mining with its applications to nanoimaging and remote sensing. He received the best student paper award at the fourth annual IEEE Conference on Automation Science and Engineering. He is a member of the IEEE and INFORMS.

Jianhua Z. Huang received the BS degree in probability and statistics in 1989 and the MS degree in probability and statistics in 1992 from Beijing University of China and the PhD degree in statistics from the University of California, Berkeley, in 1997. He was on the faculty of the Department of Statistics at the University of Pennsylvania from 1997 to 2004. He is currently a professor in the Department of Statistics at Texas A&M University and is an adjunct professor in the Department of Biostatistics at the MD Anderson Cancer Center. His research interests include computational statistics, statistical machine learning, and applications of statistics in business and engineering. He is a member of the ASA, ICSA, and IMS.

Jim X. Ji received the BS and MS degrees from Tsinghua University and the PhD degree from the University of Illinois at Urbana Champaign (UIUC), all in electrical engineering. He is currently an associate professor in the Department of Electrical and Computer Engineering at Texas A&M University. His research interests include high-speed imaging in MRI, biomedical image processing, and their applications in cancer and cardiac applications. His recent work is focused on MRI with large arrays and compressive sensing MRI. He received the Sundaram Seshu Fellowship from UIUC in 2001, and the US National Science Foundation (NSF) CAREER Award in 2008. He serves as an associate editor of the *IEEE Transactions on Information Technology in Biomedicine* and *Quantitative Imaging in Medicine and Surgery*. He is a senior member of the IEEE.

Yu Ding received the BS degree in precision engineering from the University of Science and Technology of China in 1993, the MS degree in precision instruments from Tsinghua University, China in 1996, the MS degree in mechanical engineering from Pennsylvania State University in 1998, and the PhD degree in mechanical engineering from the University of Michigan in 2001. He is currently a professor of industrial and systems engineering and a professor of electrical and computer engineering at Texas A&M University. His research interests are in the area of systems informatics and control, and quality and reliability engineering. He currently serves as a department editor of *IIE Transactions*. He is a senior member of the IEEE and a member of INFORMS, IIE, and ASME.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.