UC Irvine
UC Irvine Previously Published Works

Title

Recognition of overlapping elliptical objects in a binary image

Permalink

Journal

ISSN

Authors

Zou, Tong
Pan, Tianyu
Taylor, Michael
et al.

Publication Date

DOI

Copyright Information

Peer reviewed

Check for
updates

Recognition of overlapping elliptical objects in a binary image

Tong Zou[1] · Tianyu Pan[1] · Michael Taylor[2] · Hal Stern[1]

Abstract

Recognition of overlapping objects is required in many applications in the field of computer vision. Examples include cell segmentation, bubble detection and bloodstain pattern analysis. This paper presents a method to identify overlapping objects by approximating them with ellipses. The method is intended to be applied to complex-shaped regions which are believed to be composed of one or more overlapping objects. The method has two primary steps. First, a pool of candidate ellipses are generated by applying the Euclidean distance transform on a compressed image and the pool is filtered by an overlaying method. Second, the concave points on the contour of the region of interest are extracted by polygon approximation to divide the contour into segments. Then, the optimal ellipses are selected from among the candidates by choosing a minimal subset that best fits the identified segments. We propose the use of the adjusted Rand index, commonly applied in clustering, to compare the fitting result with ground truth. Through a set of computational and optimization efficiencies, we are able to apply our approach in complex images comprised of a number of overlapped regions. Experimental results on a synthetic data set, two types of cell images and bloodstain patterns show superior accuracy and flexibility of our method in ellipse recognition, relative to other methods.

Keywords Segmentation · Contour · Ellipse fitting · Distance transform · Concave point · Integer programming · Rand index

1 Introduction

Recognizing overlapping objects is a common problem in image analysis and arises in various real-world applications, such as splitting touching cells in medical images [1–4], bubble detection and recognition [5, 6] and bloodstain pattern analysis in forensic science [7]. In cases where the individual objects have approximately oval shapes (cells, bubbles or bloodstains), one approach is to find a representation composed of multiple ellipses to approximate the overlapping objects. The image containing overlapping objects is initially binarized using a proper segmentation method (e.g., Otsu's method [8]); this results in the identification of a number of connected regions. Each region in the binary image, in this paper addressed as a clump, may represent a single object or a union of overlapping objects. The challenging part of the task is to infer the ellipses that compose a clump based only on the contour of the clump. The problem is complex when a clump comprises a large number of objects that heavily overlap [9].

The present work proposes a novel approach that uses an ellipse representation to recognize overlapping objects in a binary image; the approach is outlined in Fig. 1. Details of the approach, referenced in the caption, are described fully in later sections. A brief overview is provided in this introduction. The proposed method consists of two main steps. The first step is *candidate ellipse generation*, based on an efficient modification of techniques in [10]: a pool of candidate ellipses is generated via application of the Euclidean distance transform (EDT) to a series of compressions of the binary image, and then, the pool is reduced greatly in number by an overlaying method. The second step is *optimal ellipse selection*, aiming to select an appropriate combination of ellipses from the pool to fit the image: initially, the contour of the clump is segmented by concave points that are extracted by a polygon approximation algorithm, and then the optimal ellipses are selected from among the candidates

✉ Tong Zou
  zout3@uci.edu

[1] Department of Statistics, University of California, Irvine, Irvine, USA

[2] Institute of Environmental Science and Research, Christchurch, New Zealand
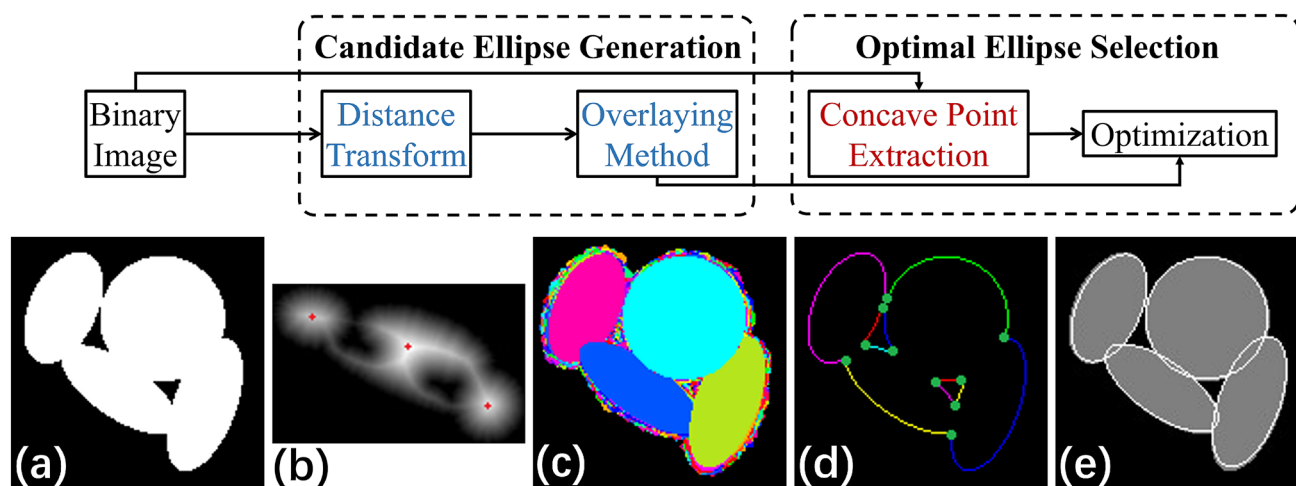
Fig. 1 Flow diagram of the proposed method and output images of each procedure. Words in blue and red denote region-based and contour-based methods, respectively. **a** A synthetic binary image consisting of four ellipses. **b** Result of compressing and EDT of **a** with specified σ and ϕ. Candidate ellipses are then generated from regional maximal pixels (red dots). **c** Result of overlaying all candidate ellip- ses by different colors, according to their 50-percentile score, ellipses with best scores appearing on top. **d** Contour is partitioned by con- cave points (green circles) into segments with different colors. **e** Final result obtained by solving the optimization problem to identify seg- ments with candidate ellipses

by matching them to contour segments using an optimiza- tion framework.

Assessing the performance of object identification meth- ods is itself a challenging task. There may be multiple arrangements of ellipses that all seem to produce a reason- able fit to the clump, yet there is no well-defined criterion to determine which is the best. The existing literature [1, 3, 4, 6, 9] relies on a variety of metrics of overall fit includ- ing Jaccard index, Hausdorff distance and mean absolute contour distance (MAD), or attempts to use concepts from classification (e.g., false positive and false negative counts). The former metrics assess the fit of the collection of ellipses but do not address the accuracy of individual ellipses, lead- ing to optimistic evaluation. The latter concepts are chal- lenging to define and compute, because evaluating fitted ellipses against true objects' arrangement is a problem of judging the quality of the fit rather than determining whether the fit is correct as in a classification problem. We propose the use of the adjusted Rand index (ARI), a performance measure defined in the context of cluster analysis, to assess performance. The use of ARI involves evaluation of the set of ellipses by assessing the degree to which pairs of contour points are assigned correctly to the same ellipse. This is more appropriate and reasonable than the criteria mentioned above for evaluating performance on the object recognition task. Though we use all of the above metrics in this study and perform well on all of them, we believe the ARI is an important contribution to the object identification literature.

The remainder of this paper is organized as follows. Sec- tion 2 reviews related work. Section 3 introduces our frame- work for segmentation of overlapping elliptical objects by

candidate ellipse generation and optimal ellipse selection. Section 4 applies our method to synthetic and real images and compares with two state-of-the-art methods. Conclu- sions are drawn in Sect. 5.

2 Related work

Roughly speaking, approaches to recognizing overlapping objects can be categorized into two broad classes: contour- based methods and region-based methods. Contour-based methods focus on extracting information from the contour pixels of a clump. For example, variants of the Hough trans- form have been widely used in ellipse detection [11–15]. Due to the high dimensionality of the ellipse parameter space, methods based on the Hough transform encounter computation and storage issues, and suffer from poor con- sistency and accuracy because of noise and quantization error [16].

Another class of contour-based methods aims to identify the concave area of the contour, based on which the con- tour is segmented and then fitted by ellipses. In [17], Zafari et al. categorized concave point detection methods into four groups: curvature [2, 18], skeleton, chord [6, 19] and poly- gon approximation [3, 5, 7, 15]. They showed via experi- ment that polygon approximation methods outperform the others. After the contour is partitioned by concave points, most of the methods boil down to the combinatorial problem of grouping segments that potentially come from the same ellipse. The complexity of the problem grows super-expo- nentially as the number of segments increases. Empirical

rules and constraints can be used to rule out unlikely combinations in different applications. For example, in [20], a branch and bound method was employed to minimize a loss function consisting of three weighted criteria: concavity, ellipticity and symmetry. Frequently, grouped segments are fitted by ellipses via least squares [21]. The grouping strategies described above rely on constraints and rules defined by ad hoc parameters. This makes them difficult to apply as the best parameter choices can vary widely across different types of data.

Region-based methods do not restrict themselves to the boundaries of shapes but rather take into account all shape pixels, thus are more tolerant to noise [22]. A group of such methods uses a distance transform as a pre-processing step. For example, the watershed algorithm has been employed on a distance transform of a binary image to find splitting lines [23]. Locating the centers of potential objects, also referred to as seedpoint extraction, is a common strategy for region-based methods. Morphological operations like ultimate erosion [24] and H-minima transform [25, 26] have been used to locate centers. Radial symmetry was proposed as another seedpoint extraction approach in [4, 9]. In [27], the centers of objects were interpreted in a physics perspective as particles dynamically reaching their steady states, with the distance transform mapping serving as a potential well. Of all the region-based methods mentioned above, a common assumption is that objects have approximately circular shape. Given elliptical objects, over-fitting may arise when the eccentricities of the ellipses are large.

In order to accurately detect the centers of ellipses, Talbot et al. [10] extended the Euclidean distance transform (EDT) into an elliptical distance transform (LDT) with two extra parameters indicating the eccentricity and orientation of a class of ellipses. An overlaying method was proposed to discard most of the false positive ellipses detected by the LDT. However, the method is highly sensitive to the choice of its controlling parameter. In [1], Panagiotakis et al. developed a parameter-free region-based method called decremental ellipse fitting algorithm (DEFA) [22] that utilizes the skeleton of a binary image to acquire structural information, and then progressively updates the ellipse-fitting model based on an entropy-based shape-complexity measure that balances model complexity and the fitting error.

To summarize, contour-based methods are effective in distinguishing segments of the clump boundary that represent different objects based on changes in the curvature, but they cannot efficiently merge those segments from the same object without introducing multiple task-specific rules and parameters. Region-based methods are able to match the large-scale features of the clump, but are less sensitive to evidence of overlapping objects provided by the boundary information. The method we propose in Fig. 1 uses the complementary advantages from both region-based and contour-based methods. The algorithm is easy to tune and adapt to different applications because there are only two parameters to be determined, one controlling the concave point extraction algorithm and one impacting the number of final selected ellipses.

3 Method

3.1 Candidate ellipse generation

An ellipse can be characterized by five parameters (x, y, a, b, φ), where (x, y) denote the center coordinates, a and b the radii of the major and minor axes, and φ the angle between the x-axis and the major axis of the ellipse. Let η denote the ratio of a over b. Equivalently, we will use (x, y, b, η, φ) to characterize an ellipse. For clarity, the η and φ of an ellipse will be referred as its shape parameter and orientation parameter to distinguish from parameters of the distance transform defined below.

3.1.1 Distance transform

The primary goal of a distance transform is to identify points within a clump that might serve as centers of ellipses. Denote the binary image containing the clump by $\boldsymbol{B}$, as illustrated in Fig. 2a. The well-known Euclidean distance transform (EDT) of $\boldsymbol{B}$ is a grayscale image in which pixel intensity is the distance from that pixel location to the closest (in Euclidean distance) boundary pixel. Another interpretation of the intensity of a pixel is as the radius of the largest circle that fits inside the clump, with the pixel as its center. Therefore, given a clump composed of circles, the centers of circles can be identified as the local maximum pixels in the distance transform image with their intensities being the radii. However, when the clump consists of overlapping ellipses, their centers usually are not located at the local maxima of the EDT, as shown in Fig. 2b.

In [10], Talbot et al. generalized EDT by modifying the Euclidean distance to any distance metric. The elliptical distance transform (LDT) is a special case when the distance metric between two 2-D vectors $\mathbf{p}$ and $\mathbf{q}$ takes the following form with pre-specified σ and ϕ:

$$d(\mathbf{p}, \mathbf{q}) = \frac{\rho}{\sigma} \sqrt{\cos^2(\theta - \phi) + \sigma^2 \sin^2(\theta - \phi)}, \qquad (1)$$

where ρ and θ denote the Euclidean distance and angle between $\mathbf{p}$ and $\mathbf{q}$.

Then, the pixel intensity of the LDT is the smallest distance (as defined by the above metric) between the pixel and any boundary pixel, or equivalently, is the minor axis radius b of the largest ellipse that fits inside the clump,
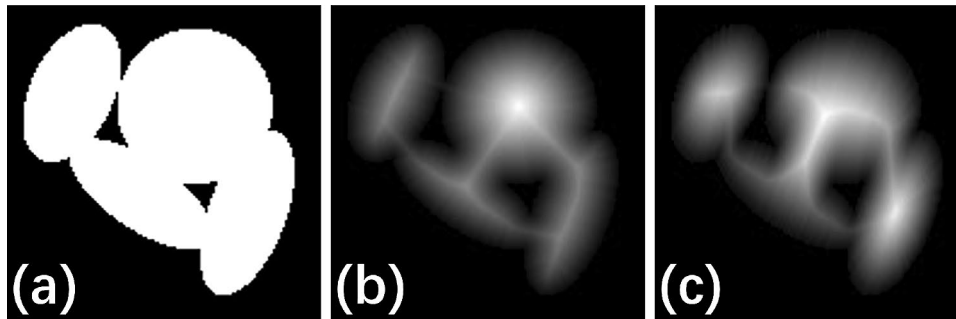
Fig. 2 Distance transforms of a binary image. **a** Binary image containing a clump. **b** Euclidean distance transform: a local maximum point (bright spot) is evident at the center of the circle, while centers of ellipses are not prominent. **c** Elliptical distance transform with the pre-specified σ and ϕ equaling the shape and orientation parameters of the bottom-right ellipse, whose center is more evident than those of other ellipses

with the pixel location as its center (x, y), σ as its shape parameter, and ϕ as its orientation parameter. Thus, centers of ellipses with shape and orientation parameters equaling σ and ϕ can be identified from the LDT(σ, ϕ) transform as the local maximum pixels with their intensities being the minor axis radius, as shown in Fig. 2c. A major weakness of the LDT is that its implementation based on vector propagation [10] is relatively slow, while the EDT, due to its many desirable properties like radial symmetry, has many fast implementations [28].

In this study, we propose an alternative approach to identify candidate ellipses that achieves the benefits of the LDT, while taking advantage of the faster implementation of the EDT. First, the clump is compressed by factor σ along the direction whose angle to the x-axis is ϕ. To do this, the coordinates of pixels inside the clump of B are transformed with an affine matrix T characterized by σ and ϕ:

$$T = S \cdot R \cdot S^{-1}, \tag{2}$$

where $S = \begin{pmatrix} 1/\sigma & 0 \\ 0 & 1 \end{pmatrix}$, and $R = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix}$.

Then, the transformed coordinates are rounded and rearranged on a new binary image with proper size, denoted $B_{\sigma,\phi}$. Running the EDT on $B_{\sigma,\phi}$, with the result denoted $E_{\sigma,\phi}$, we can obtain parameters of candidate ellipses from the local maximum pixels of $E_{\sigma,\phi}$ with their coordinates transformed back by T^{-1} as center coordinates (x, y), their intensity as b, and the LDT values of σ and ϕ as shape and orientation parameters. Figure 3 shows the images of $B_{\sigma,\phi}$ and $E_{\sigma,\phi}$ for three different choices of σ and ϕ.

By iterating this procedure of compression and EDT over a 2D grid search of σ and ϕ values, we are able to establish a pool of candidate ellipses from local maximum pixels of all $E_{\sigma,\phi}$. Given a fine enough grid, the optimal ellipses will be included in this pool. However, a large number of

Fig. 3 Examples using the EDT on affine-transformed image to identify candidate ellipses. The top row are affine-transformed binary images $B_{\sigma,\phi}$, and the bottom row are their EDT results $E_{\sigma,\phi}$ (the size of images is scaled for alignment). For **a** and **b**, $\sigma = 1$ and $\phi = 0$, with the same result as the ordinary EDT. The values of σ and ϕ in **c** and **d** equal the shape and orientation parameters of the bottom-right ellipse, while in **e** and **f** they equal those of the bottom-left ellipse. Notice the shapes of the relevant ellipses turn to circles in $B_{\sigma,\phi}$ and hence their centers become local maximum in $E_{\sigma,\phi}$
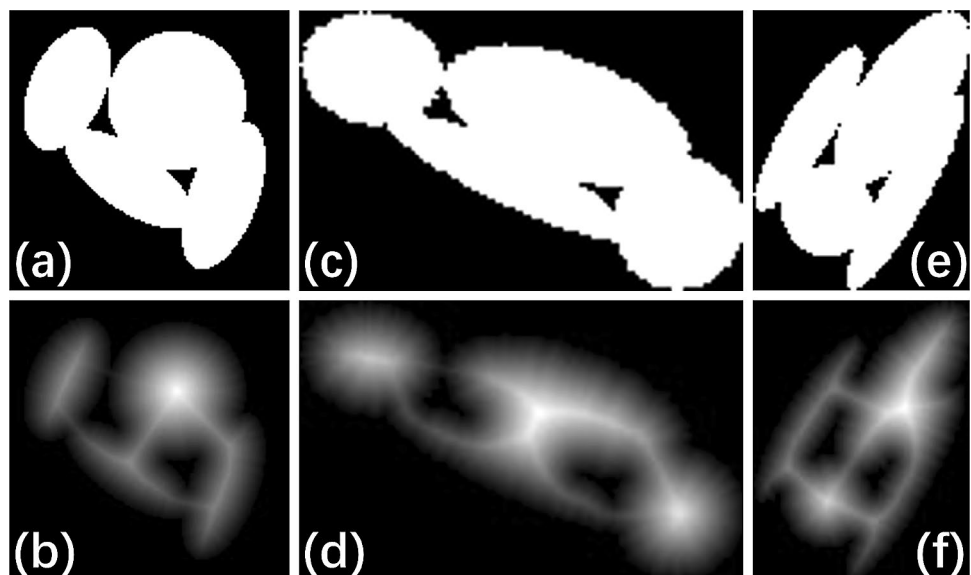
ellipses not well fitted to the clump will also be included. The distance transform of a binary image (either LDT or our method) given certain values of σ and ϕ can contain many local maximum pixels, which means a large number of sub-optimal ellipses (10^4–10^5) included in the candidate pool, making the ellipse selection step computationally intensive. We apply two refinements to reduce the size of the set of candidate ellipses. One effective approach to diminishing the pool is to find the coordinates of local maximum pixels on smoothed versions of $E_{\sigma,\phi}$. This reduces the number of candidate ellipses by an order of magnitude (10^3–10^4). This is still too large. This leads to the next section where a second approach, the overlaying method, is introduced to address the issue.

3.1.2 Overlaying method

To reduce the number of candidate ellipses, we build on the approach of [10] to identify a small subset of ellipses that are better fitted to the clump than the rest. The basic idea is to assign each ellipse a score based on the degree to which the ellipse approximates the contour of the clump and then overlay all ellipses in order of their scores. As a result, the clump is covered by a few of the best-scoring ellipses on the top, while other ellipses completely overlaid or hidden by the better-scoring ellipses are excluded from the candidate pool.

To acquire the score of candidate ellipses, we first compute the EDT of the contour of the clump B, denoted by E. Then for each ellipse, we record the distance from each of its border pixels to the nearest point on the contour of B; these distances are available as the intensities of E. The kth-percentile of the border-pixel distances is assigned to the ellipse as its score, with the choice of k addressed below. A lower score for an ellipse indicates a better fit to the contour. Finally, a new image O is constructed by overlaying all candidate ellipses in decreasing order of their scores. As shown in Fig. 1c with $k = 50\%$, the best fitting ellipses usually appear on the top of O.

The overlaying method can greatly reduce the number of candidate ellipses because we can ignore completely hidden ellipses. But it can suffer from serious under-fitting or over-fitting [3] depending on the choice of the percentile k. As illustrated in Fig. 4, k must be chosen carefully so that an ellipse that appears to fit part of the image well is not completely overlaid by others. Generally, it is difficult to find a single appropriate value of k. In order to avoid missing any essential ellipses, we compute multiple scores with different percentiles (30%, 50% and 70% for this study). And for each percentile k, the corresponding overlay image O_k is constructed. Ellipses appearing on any of the O_k are retained in the candidate pool while others are screened out. This modification is highly effective in ruling out non-optimal ellipses and preserving optimal ellipses, since each O_k only
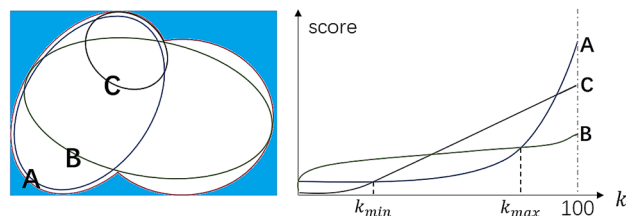

Fig. 4 An illustrative example addressing the choice of k for the overlaying method. On the left is the clump and three candidate ellipses. Ellipse A is a good-fitting ellipse to part of the clump contour. On the right are the scores of the three ellipses for different choices of k. If k is outside the range $[k_{\min}, k_{\max}]$, then ellipse A will receive a poorer (higher) score and be partially overlaid by ellipse B or C. This suggests the importance of considering multiple values of k

contains a small number of ellipses, typically less than 10^2, and the use of multiple percentiles means that optimal ellipses are less likely to be completely overlaid.

3.2 Optimal ellipse selection

Given the candidate ellipses obtained from Sect. 3.1, a matching mechanism is introduced to determine the best group of ellipses to fit the contour of B. To help conduct the matching, the contour is represented as a set of segments defined by concave points. Then, the distance of each segment to each ellipse is computed. Finally, the optimal subset of candidate ellipses is identified by minimizing a distance-based loss function with a penalty on the number of ellipses.

3.2.1 Concave point extraction

Concave points are points of intersection of objects on the contour. A contour segment is represented by the two concave points that define its ends. It is assumed that pixels on the segment are from the same object. In order to extract concave points, a series of steps are carried out, as illustrated in Fig. 5. First, a sequence of coordinates of contour pixels is acquired using a radial sweep algorithm [29] in a clockwise direction (Fig. 5a). Then, the contour is approximated by a polygon represented by its vertices, also addressed as turning points, ruling out collinear points on the contour as being uninformative (Fig. 5b). Finally, a turning point is determined to be a concave point if the sweeping angle from the preceding point to the next turning point outside the clump is less than 180°. The result is shown in Fig. 5c where each segment is presumed to include pixels from a single ellipse.

Here, we provide additional details about the polygon approximation step. In [5, 30], the polygon approximation algorithm keeps three moving pixels p_i, p_j, p_k and computes the distance between p_j and line $\overline{p_i p_k}$, denoted as d_{ik}^j. If the distance exceeds some threshold d_t, then p_j is

Fig. 5 Example of concave point extraction. **a** Contour pixels obtained by radial sweep. **b** Turning points (red and green circles) extracted by polygon approximation and concave points (green circles). **c** Contour segments (in different colors) separated by concave points (green circles)
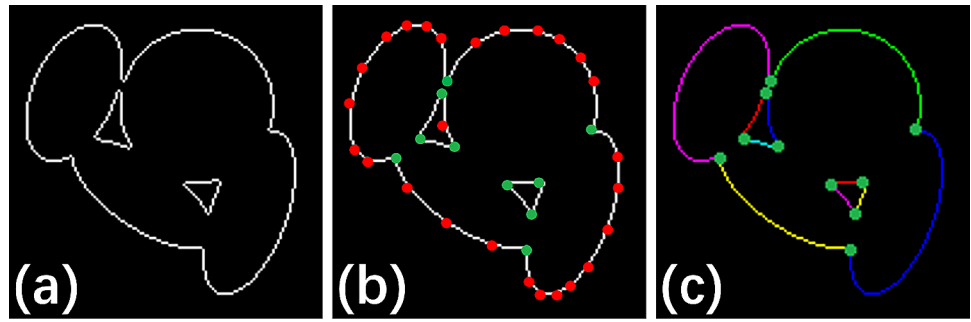
deemed to be a turning point. The smaller d_t is, the more closely the polygon approximates the contour. If d_t is too small, the algorithm can sometimes mistake a collinear point for a turning point, or if d_t is too large, then turning points can be missed, undermining the performance of the optimization step described in Sect. 3.2.2. To address this issue, we modified the algorithm to compute distances of all pixels between p_i and p_k from the line connecting them and compare the highest of these with d_t. In practice, the increase in computation is negligible. Let set $P = \{p_1, p_2, \ldots, p_N\}$ denote the contour pixel sequence and Q contain all detected turning points. The algorithm employed is described in Algorithm 1. The concave points are then obtained as a subset of Q.

Algorithm 1 Modified Polygon Approximation

Input: P (a set of contour pixels)
Output: Q (a set of turning points)

1: $Q = \{p_1\}$
2: $i = 1, k = 3$
3: **while** $k \leq N$ **do**
4: $\quad d_{max} = 0, ind = 0$
5: $\quad$ **for** $j = (i+1) \to (k-1)$ **do**
6: $\quad\quad$ compute $d_{ik}^j \triangleright$ distance between p_j and $\overline{p_i p_k}$
7: $\quad\quad$ **if** $d_{ik}^j > d_{max}$ **then**
8: $\quad\quad\quad d_{max} = d_{ik}^j$
9: $\quad\quad\quad ind = j$
10: $\quad$ **if** $d_{max} > d_t$ **then**
11: $\quad\quad Q = Q \cup \{p_{ind}\}$
12: $\quad\quad i = ind, k = ind + 2$
13: $\quad$ **else**
14: $\quad\quad k = k + 1$

With the proposed algorithm, the result is less sensitive to the choice of distance threshold d_t, which should depend on the scale and roughness of the contour. In the experiments, we use a subset of the data to identify an appropriate d_t. We try to err on the side of small d_t; this can create false positive contour segments, but the optimization procedure can mitigate the effect of these false positives.

3.2.2 Optimization

With contour segments and candidate ellipses available, the optimal ellipse subset can be acquired through a matching procedure. Given a group of ellipses, each contour segment defined by a pair of concave points has to be matched to exactly one ellipse. Each ellipse can be matched to by one or more contour segments simultaneously or by none of them. The matching algorithm requires that we define the distance between a contour segment and an ellipse. Then, the goal is to find a subgroup of ellipses from the candidate pool to minimize the total distance to all contour segments.

Let $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_M\}$ denote the set of contour segments, $\Pi = \{\pi_1, \pi_2, \ldots, \pi_N\}$ the set of candidate ellipses, and $\Pi^* = \{\pi_1^*, \pi_2^*, \ldots, \pi_L^*\} \subseteq \Pi$ the optimal ellipse set.

The distance between contour segment γ_i and candidate ellipse π_j is denoted by d_{ij} and defined as follows: for each pixel of γ_i, find the smallest distance between the pixel and π_j (i.e., the intensity of the pixel on the EDT of π_j's border), and sum them over all pixels of γ_i, which can be written as,

$$d_{ij}(\gamma_i, \pi_j) = \sum_{p \in \gamma_i} \min_{q \in \pi_j} |pq|, \tag{3}$$

where $|pq|$ is the distance between pixel p and q. Then, the loss function of a proposed solution set $\hat{\Pi}$ is defined as the sum of the distances from each contour segment i to its closest ellipse j in $\hat{\Pi}$,

$$L(\hat{\Pi}) = \sum_{i=1}^{M} \min_{\pi_j \in \hat{\Pi}} d_{ij}. \tag{4}$$

Simply minimizing the loss with respect to $\hat{\Pi}$ will cause over-fitting and result in Π being the optimal solution. To avoid this problem, a penalty term $\lambda|\hat{\Pi}|$ is added, where $|\hat{\Pi}|$ is the number of ellipses in $\hat{\Pi}$ and λ is a regularization parameter. This constraint encourages the solution to match multiple contour segments to a smaller set of ellipses and thus avoid over-fitting. The optimal set Π^* is defined as

$$\Pi^* = \underset{\hat{\Pi} \subseteq \Pi}{\operatorname{argmin}} \ L(\hat{\Pi}) + \lambda |\hat{\Pi}|. \tag{5}$$

Proper selection of λ depends on the image size and noise level. As was done for the distance threshold d_t, a subset of the data is used to select the value of λ in Sect. 4.

The minimization problem can be transformed into an integer programming problem. First, some binary variables are introduced to formalize the matching process:

$$x_{ij} = \begin{cases} 1 & \text{if } \gamma_i \text{ is closest to } \pi_j \\ 0 & \text{else}, \end{cases} \tag{6}$$

$$z_j = \begin{cases} 1 & \text{if } \pi_j \text{ is matched to any } \gamma \in \Gamma \\ 0 & \text{else}. \end{cases} \tag{7}$$

Thus, x_{ij} identifies the matching of contour segments i to ellipses j, and z_j identifies the inclusion of ellipse j in the optimal set Π^*. The optimization over the binary variables then has the following form:

$$\min_{x_{ij}, z_j} \ \sum_{i=1}^{M} \sum_{j=1}^{N} d_{ij} x_{ij} + \lambda \sum_{j=1}^{N} z_j, \tag{8}$$

$$\text{s.t.} \quad z_j \leq \sum_{i=1}^{M} x_{ij} \leq M \cdot z_j, \ j = 1, 2, \dots, N, \tag{9}$$

$$0 \leq \sum_{j=1}^{N} x_{ij} \leq 1, \ i = 1, 2, \dots, M. \tag{10}$$

This optimization problem can be solved using a branch and bound method [31], and the optimal set Π^* can be obtained from the values of z_j, i.e., ellipse π_j is included in the optimal group if $z_j = 1$. Note that the effect of a falsely detected concave point can be rectified because the two segments incorrectly partitioned by that point will be assigned to the same ellipse (as described at the end of Sect. 3.2.1).

4 Experiments

4.1 Implementation

For the rest of the paper, we will refer to our method using the acronym DTECMA to indicate Distance Transform-based Ellipse-Contour Matching Algorithm. We implement DTECMA in MATLAB R2017b with DIPimage toolbox 2.9 to conduct basic image processing operations. The toolbox includes the Euclidean distance transform implementation by Mullikin [32]. The binary optimization problem of Sect. 3.2.2 is solved by function `intlinprog` in the optimization toolbox [33].

4.2 Performance on synthetic images

4.2.1 Data description

To evaluate the performance of DTECMA and compare with other approaches, a synthetic dataset of 1600 binary images was generated. Each image contains one connected region, which is a union of multiple overlapping ellipses whose parameters are known. The dataset consists of 8 groups of 200 images, each group having in common the number of ellipses within the images, ranging from 2 to 9. Images with more ellipses present increasing complexity and greater difficulty in ellipse recognition. Ellipses were generated at random locations with parameters a, σ and ϕ uniformly distributed on $(12.5, 37.5)$, $(1, 3)$ and $(0, 180)$, subject to a constraint on the degree of overlap. The overlapping rate of each ellipse, defined as the proportion of the area overlapped with other ellipses, is controlled to stay below 60% to avoid completely overlaid ellipse. Examples of synthetic images and their ground truth are shown in Fig. 9a, b.

4.2.2 Criteria

Several quantitative criteria have been used for evaluation of approaches to identifying overlapping elliptical objects. The Jaccard index is a region-based measure that quantifies the overall coverage of the ellipse-fitting image I_f (the union of the selected ellipses Π^*) to the original image I_o as follows:

$$\text{Jaccard} = \frac{I_f \cap I_o}{I_f \cup I_o}. \tag{11}$$

Higher values of the Jaccard index indicate a better fit. Two other contour-based metrics, Hausdorff distance and mean absolute contour distance (MAD), introduced in [1, 4] are also used. They measure the overall distance between the contours of I_f and I_o. They are defined as:

$$\text{Hausdorff} = \max_{i \in C_f} D(i), \tag{12}$$

$$\text{MAD} = \frac{1}{|C_f|} \sum_{i \in C_f} D(i), \tag{13}$$

where $D(i)$ denotes the minimal Euclidean distance of pixel i to the contour of I_o and C_f denotes the contour of I_f. Lower values of the contour-based measures indicate a better fit. A drawback of all of these metrics is that, instead of checking whether each ellipse is well fitted to the original image, they only evaluate the fit of the set of ellipses. In certain cases, a poor result, where the selected ellipses under-fit or over-fit the image, may still achieve good aggregate measures, as shown in Fig. 6.
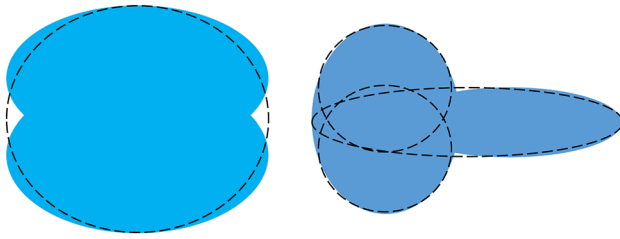
Fig. 6 Examples of under-fitting (left) and over-fitting (right) with good performances as measured by the Jaccard index, Hausdorff distance and MAD. Blue regions denote the original images that both consist of two ellipses, and dashed curves denote the fitting result

To assess performance for each ellipse instead of the whole set, we reconsider the evaluation process from a clustering perspective and propose to use the adjusted Rand index (ARI) [34] as a contour-based performance measure. Given the set of contour pixels of a binary image consisting of overlapping ellipses, the ground truth clustering of the set is defined to be a clustering with pixels from the same ellipse being clustered into the same group and pixels from different ellipses being clustered into different groups (intersecting pixels can be dropped or randomly assigned). The result of an ellipse-fitting algorithm applied to the image also corresponds to a clustering of the contour pixels: each contour pixel is assigned to the nearest ellipse in the solution set and the set of contour pixels assigned to a given ellipse can be viewed as a cluster. For example, in the left-hand side image of Fig. 6 all contour pixels belong to one cluster according to the dashed black line, while in the right-hand side image they are assigned into 3 clusters. In DTECMA, the cluster assignments are given by the binary variables x_{ij}. The Rand index (RI) [35] is a measure of the similarity between two clusterings. Given a set of n elements, the RI between two clusterings X and Y is computed as follows:

$$\text{RI}(X, Y) = \frac{a + b}{\binom{n}{2}}, \tag{14}$$

where a is the number of pairs of elements that are placed in the same cluster in X and in the same cluster in Y, and b is the number of pairs placed in different clusters in X and in different clusters in Y. The ARI is a modified version that corrects the clustering similarity measure for chance agreement under the permutation model [36]:

$$\text{ARI}(X, Y) = \frac{\text{RI}(X, Y) - \mathbb{E}\{\text{RI}(X, Y)\}}{1 - \mathbb{E}\{\text{RI}(X, Y)\}}. \tag{15}$$

The clustering perspective on contour pixels and ARI provides an approach to evaluating how close the fitting result is to the ground truth, which we believe is more objective than visual inspection and more relevant than previous metrics

(see Fig. 6). For this experiment, ARI is used as the primary evaluation criterion, with Jaccard index, Hausdorff distance and MAD serving as supplementary metrics.

4.2.3 Parameter tuning

There are two parameters that have a significant impact on performance of DTECMA. These are the distance threshold d_t of the polygon approximation (Sect. 3.2.1) and the regularization parameter λ of the optimization approach (Sect. 3.2.2). For the simulated data study, we use half of the synthetic images (8 groups of 100), referred to as the validation set, to select these parameters, while the rest of the images, referred to as the test set, are used for evaluation.

The *distance threshold d_t* used in the polygon approximation determines the maximum distance between a turning point and the line connecting its two neighboring turning points. It plays a vital role in concave point extraction. Since the ground truth (ellipse parameters) of the simulated data is known, the true concave points are in fact the intersection points of ellipses on the contour. Concave points of an image can be viewed as a partition of the contour, namely a clustering, so ARI can be used as a performance measure for concave point extraction as it was ellipse fitting. The plot on the left of Fig. 7 shows average ARI as a function of d_t for the 8 groups with different number of ellipses. The results strongly support the choice $d_t = 1$ for this dataset.

The *regularization parameter λ* balances the loss function $L(\hat{\Pi})$ and the number of ellipses. The loss function term will depend on the size of the image, so the proper value of λ should be related to the size of each image. To address this issue, a new parameter λ^* is defined as follows:

$$\lambda = \sqrt{S} \times \lambda^*, \tag{16}$$

where S is the area of the binary image. Now, a value of λ^* can be selected to perform well throughout the dataset. The plot on the right-hand side of Fig. 7 shows the effect of different values of λ^* on the performance of DTECMA in terms of average ARI. λ^* was set to 0.5 to obtain the best performance. It is noteworthy that λ^* values between 0.1 and 1 yield similar performance in terms of ARI, which shows the robustness of the results with respect to this parameter.

4.2.4 Results

Two state-of-the-art methods for detecting overlapping elliptical objects, with code available online, are compared with DTECMA on the synthetic test set. The DEFA model proposed by Panagiotakis et al. [1] is a parameter-free method developed to segment touching cells by ellipse fitting. The method proposed by Zafari et al. [20] integrated three components in an optimization framework that can be solved by
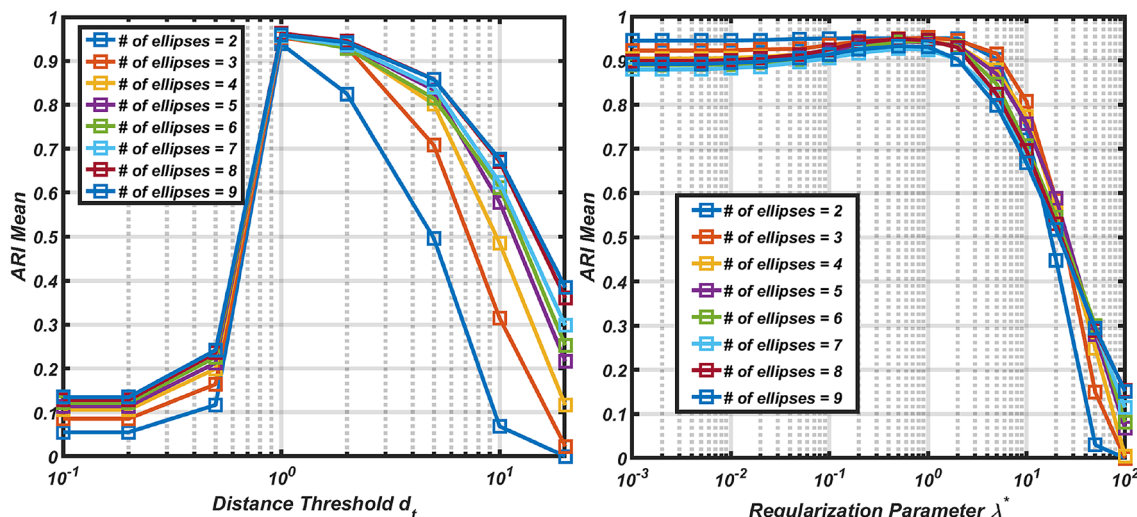
Fig. 7 Parameter setting results for d_t and λ^*. Each point is an average ARI value over 100 images

a branch and bound (BB) algorithm. It applies techniques from the authors' previous work [9, 18] on radial symmetry and concave points, and achieved superior performance among the methods compared in [20]. Model parameters for BB were tuned to achieve the best performance with respect to ARI on the validation set. Both methods take a binary image as input; hence, the comparison does not involve the effect of image thresholding.

Four measures including ARI, Jaccard index, Hausdorff distance and MAD were computed for each of the three methods on each image and averaged within each data group of the test set, as shown in Fig. 8. DTECMA achieved consistently superior performance in all measures. As the number of ellipses in the synthetic images increases, potentially complicating the task, DTECMA remains reliable. Another interesting observation is that BB outperforms DEFA in terms of ARI but not for the three traditional measures. This is due to the limitation of traditional measures described in

Sect. 4.2.2. Figure 9 provides some examples of the fitting results. Compared with BB, DEFA appears prone to serious under-fitting when ellipses overlap too much, but still manages to cover the image nicely, hence gaining higher scores on traditional measures. BB is more accurate than DEFA in ellipse fitting, but sometimes fails to detect crucial concave points and to group small segments, and as a result does worse on the aggregate measures.

4.3 Application to cell splitting

Biological image analysis is often focused on determining the characteristics of each individual cell in an image. Splitting of touching cells is of great importance for optimal performance of quantitative high-throughput automated image analysis [4]. To evaluate DTECMA on images in which the shape of objects is not perfectly elliptical, we apply it to cell splitting on two public data sets
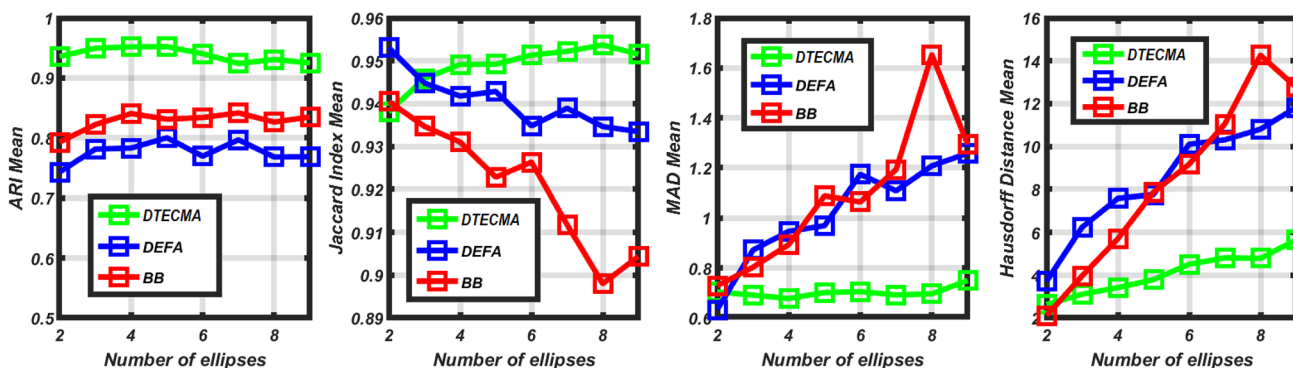

Fig. 8 Results on four performance measures for the synthetic data test set for three different methods. Each point is an average over 100 images
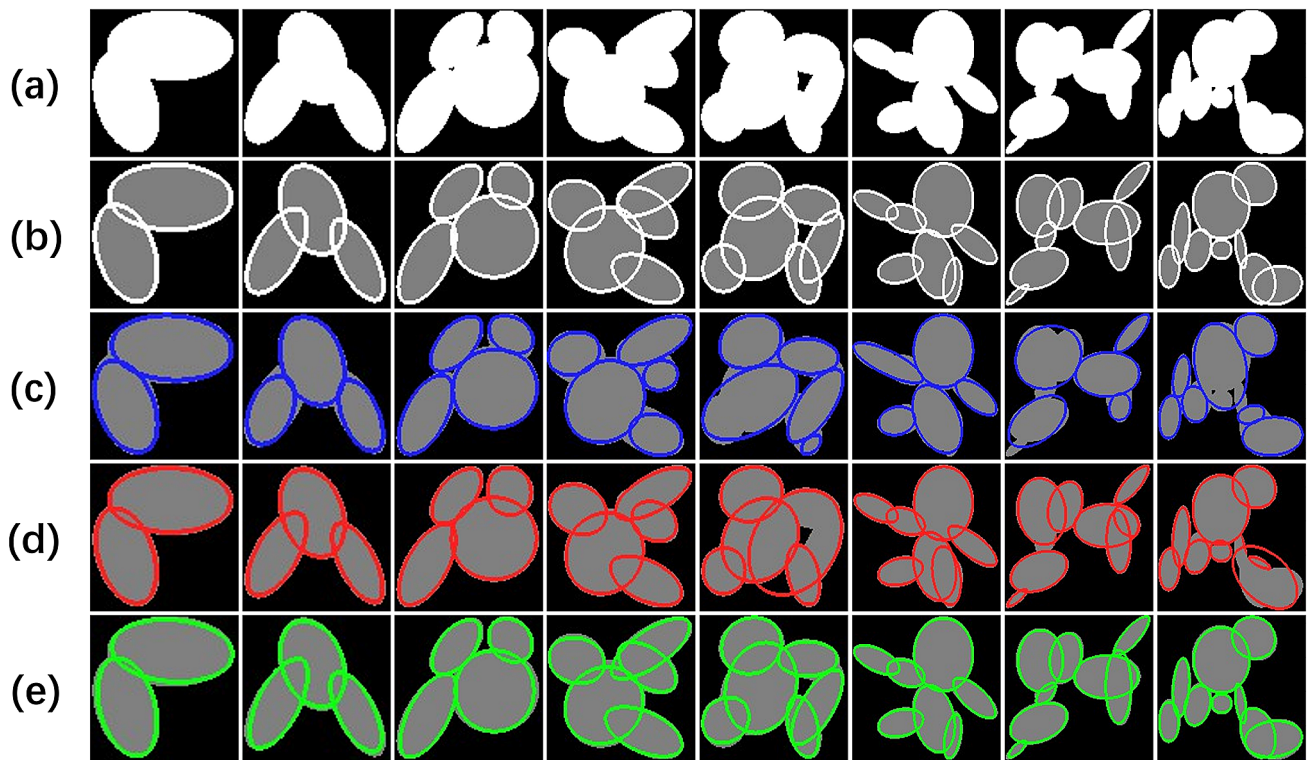
Fig. 9 Examples of ellipse fitting results for synthetic data. **a** Original binary image; **b** ground truth; **c** DEFA model; **d** BB model; **e** DTECMA. The number of true ellipses increases from 2 (leftmost column) to 9 (rightmost column)

Table 1 Basic information for the cell data sets

	U20S	NIH3T3
Number of images	48	49
Image size	1349×1030	1344×1024
Number of cells	1831	2178

for which ground truth is annotated by experts. The result of DTECMA is quantitatively compared with DEFA which has achieved excellent performance on the same data sets [1].

4.3.1 Data description

Two public cell data sets with manual labelling are published in [37] as a benchmark for segmentation algorithms. Table 1 provides basic information about the two data sets. The shape of U20S cells is highly varying and irregular. The NIH3T3 cells are more consistent in shape, but binarization of the NIH3T3 images is more challenging because of the varying image intensity.

4.3.2 Image processing

To transform grayscale images to binary images, we employ the modified Bradley segmentation method described in [1], which is a real-time adaptive thresholding method and has superior segmentation performance on the data sets. After applying an ellipse fitting procedure (e.g., DTECMA or DEFA), touching cells (represented by two ellipses) are split by the line on which each point has the same algebraic distance to both ellipses.

4.3.3 Criteria

For cell data, the ground truth is annotated on the grayscale image instead of the binary image, which thus requires different evaluation criteria than the ones used with the synthetic data. For example, the Jaccard Index, Hausdorff distance and MAD were used in [1] to evaluate segmentation performance but not useful for assessing cell splitting performance. The ARI cannot be directly computed from the available ground truth information and thus cannot be used either. To assess cell splitting performance, we employ the definitions from [1] for computing the number of false positives (FP) (spuriously segmented cells) and the number of false negatives (FN) (cells that have not been segmented).

The FP and FN counts of all images are averaged as a performance measure for cell splitting.

4.3.4 Parameter tuning and results

The two parameters of DTECMA are tuned on each data set using just two images. DTECMA is compared with DEFA and two other top performing methods *Three-step* [38] and LLBWIP [4] that were also used for comparison in [1]. Figure 10 shows an example from the U20S data set. It is worth mentioning that the binarization step plays a vital role in the final results, since it helps the splitting algorithm to distinguish the body of cells from background noise. Despite the shape variation of cells, DTECMA manages to accurately split most touching cells. Table 2 gives the average number of FP and FN for the data sets. We reproduce the result of DEFA and also report the results of *Three-step* [38] and LLBWIP from [1]. DTECMA provides a notable improvement over DEFA.

4.4 Application to bloodstain patterns

A bloodstain pattern is a collection of stains observed in a crime scene. The main objective for bloodstain pattern analysis is to classify patterns with respect to the mechanism or event type that produced the pattern. Analysis of bloodstain patterns frequently relies on measurements of the size, direction and shape of individual stains. For example, recent studies [39, 40] have used an ellipse representation of individual bloodstains to extract features for mechanism determination. A limitation of this approach is that it relies on a clear differentiation of the stain margin and hence ignores overlapping stains. In some patterns, this can severely limit the number of stains available for use and add uncertainty to feature estimation and hence to classification ability. With DTECMA, these overlapping bloodstains can be used to strengthen the analysis.

We demonstrate DTECMA on the impact pattern *HP31* published in [41]. Pre-processing is applied to the pattern by following the procedures in [39], including background

Table 2 Cell splitting results

Methods	U20S		NIH3T3	
	FP	FN	FP	FN
Three-step[a]	0.5	3.9	1.7	11.3
LLBWIP[a]	0.3	2.7	1.5	5.0
DEFA	2.7	0.3	0.7	0.8
DTECMA	1.5	0.2	0.6	0.8

[a]Results are obtained from [1]

subtraction, element segmentation for binarization and morphological operations for smoothing and hole filling. Since the distance transform is very sensitive to defects in shape and topology of the image, smoothing and hole filling are important pre-processing steps for images with high noise level.

DEFA and BB are also applied to the image to compare with DTECMA. Since there is no quantitative approach to decide whether an overlapped bloodstain is correctly separated, parameters of the methods were selected to achieve the best performance based only on visual inspection. Figure 11 displays a part of the *HP31* image and the fitting results for the three methods. The results here are consistent with those in the simulation study. DTECMA seems to perform well, while the other two methods suffer from under-fitting (refer to the insets in Fig. 11).

The example in Fig. 11 demonstrates that DTECMA has the ability to detect and identify overlapping stains. Similar results have been obtained for other images from [41]. Quantitative assessment of the performance of DTECMA in the bloodstain pattern application is not straightforward. Unlike for the synthetic data (Sect. 4.2) and the cell splitting data (Sect. 4.3), there are not, to our knowledge, published bloodstain patterns for which the correct overlapping ellipses are identified. Moreover, there are no published standards for how one might label the images. This is because the primary task is the identification of the mechanism or event that produced the pattern and thus the individual stains are not of primary interest. This suggests a possible alternative

Fig. 10 Ground truth of a U20S cell image (left) and cell splitting results of DTECMA (right)
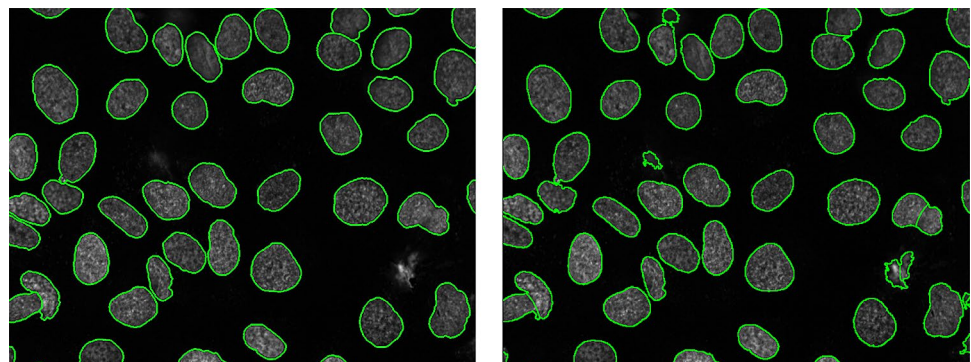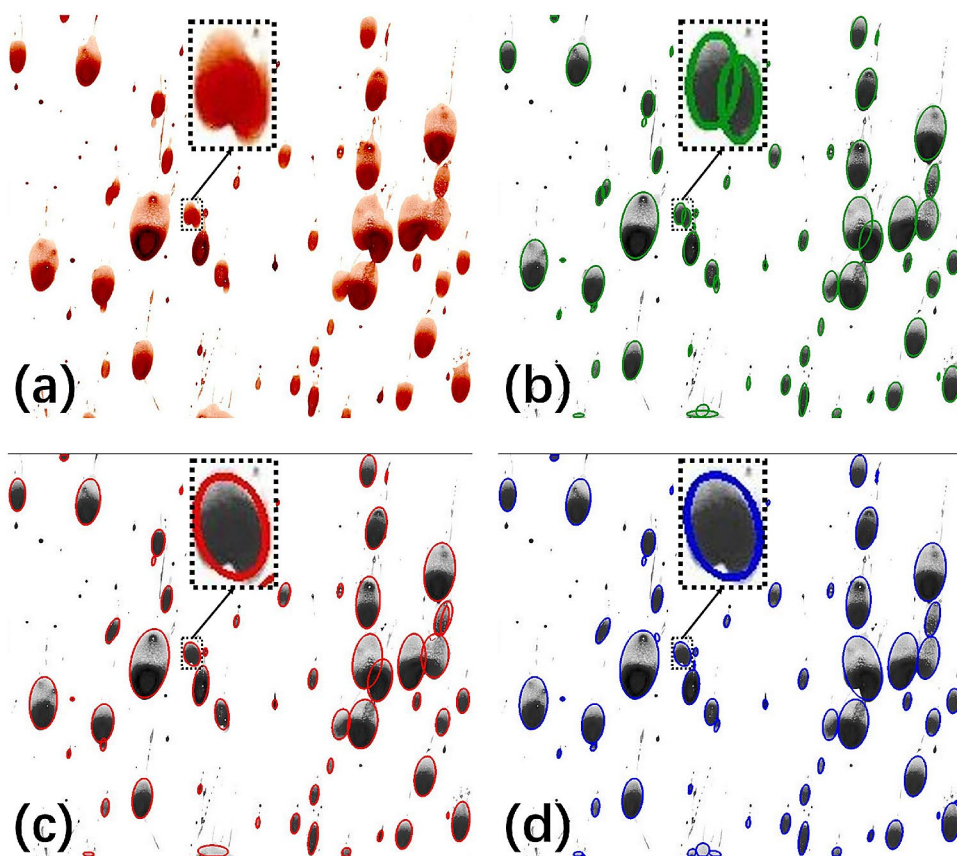
Fig. 11 Ellipse fitting results for a bloodstain pattern. The blood-stain pattern is transformed into a gray scale image to highlight the fitting results in different colors for different methods in images (**b**, **c**, **d**). **a** A part of the bloodstain pattern of *HP31*. **b** Result from DTECMA; **c** result from BB; **d** result from DEFA

approach to assess the performance of DTECMA: one might evaluate whether the overlapping stains identified by DTECMA (which are frequently ignored in practice) help in the classification of bloodstain patterns. This approach, however, is beyond the scope of the current paper and will be addressed in a future study.

5 Conclusions

A method of recognizing overlapping elliptical objects in a binary image is proposed. The method consists of a first step in which candidate ellipses are generated and a second step in which optimal ellipses are selected. The first step generates a large pool of candidate ellipses guaranteed to include most good-fitting ellipses through an exhaustive search in the parameter space defined by a compression and distance transformation. The computational cost of the search for candidate ellipses can be large when the parameter space is finely partitioned. Fortunately, the method can be programmed in parallel since candidate ellipse generation efforts for different parameter settings are independent. Overlaying candidate ellipses on the image effectively reduces the size of the candidate pool through elimination of candidates that are completely

covered. The second step, selection of optimal ellipses, begins with extraction of concave points that separate the contour being analyzed into elliptical segments. The final set of ellipses are identified by framing the selection step as an optimization problem, trading off a loss function that measures the lack of fit to the image contour and the number of ellipses required to obtain the fit. We consider a variety of measures for assessing the performance of our method, including the adjusted Rand index which addresses the ability to correctly identify individual ellipses. An experiment on simulated synthetic data demonstrates that our method outperforms competing methods. Taking advantage of both region-based (ellipse generation) and contour-based (ellipse selection) approaches, and with only two parameters to tune, DTECMA is robust and suitable for various applications, including splitting cells of irregular shapes and representing bloodstain patterns by ellipses.

Availability of data and materials Synthetic data are available but not published yet. Bloodstain patterns and cell images are both published datasets.

Code availability The code of our algorithm is available but not published yet. The competing methods for the purpose of comparison are published.

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interests.

References

1. Panagiotakis C, Argyros AA (2018) Cell segmentation via region-based ellipse fitting. In: 2018 25th IEEE International conference on image processing (ICIP). IEEE, pp 2426–2430
2. Guo X, Yu H, Rossetti B, Teodoro G, Brat D, Kong J (2018) Clumped nuclei segmentation with adjacent point match and local shape-based intensity analysis in fluorescence microscopy images. In: 2018 40th Annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp 3410–3413
3. Bai X, Sun C, Zhou F (2009) Splitting touching cells based on concave points and ellipse fitting. Pattern Recognit 42(11):2434–2446
4. Gharipour A, Liew AWC (2016) Segmentation of cell nuclei in fluorescence microscopy images: an integrated framework using level set segmentation and touching-cell splitting. Pattern Recognit 58:1–11
5. Zhang WH, Jiang X, Liu YM (2012) A method for recognizing overlapping elliptical bubbles in bubble image. Pattern Recognit Lett 33(12):1543–1548
6. De Langlard M, Al-Saddik H, Charton S, Debayle J, Lamadie F (2018) An efficiency improved recognition algorithm for highly overlapping ellipses: application to dense bubbly flows. Pattern Recognit Lett 101:88–95
7. Wójtowicz T, Bułka D (2018) Ellipse detection in forensic blood stain images analysis. Comput Inform 37(1):213–228
8. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):62–66
9. Zafari S, Eerola T, Sampo J, Kälviäinen H, Haario H (2015) Segmentation of overlapping elliptical objects in silhouette images. IEEE Trans Image Process 24(12):5942–5952
10. Talbot H, Appleton B (2002) Elliptical distance transforms and the object splitting problem. In: Proceedings of ISMM2002, pp 229–240
11. Tsuji S, Matsumoto F (1978) Detection of ellipses by a modified Hough transformation. IEEE Trans Comput 8:777–781
12. Yuen H, Illingworth J, Kittler J (1989) Detecting partially occluded ellipses using the Hough transform. Image Vis Comput 7(1):31–37
13. McLaughlin RA (1996) Randomized Hough transform: better ellipse detection. In: Proceedings of digital processing applications (TENCON'96), vol 1. IEEE, pp 409–414
14. Bennett N, Burridge R, Saito N (1999) A method to detect and characterize ellipses using the Hough transform. IEEE Trans Pattern Anal Mach Intell 21(7):652–657
15. Prasad DK, Leung MK, Cho SY (2012) Edge curvature and convexity based ellipse detection method. Pattern Recognit 45(9):3204–3221
16. Mukhopadhyay P, Chaudhuri BB (2015) A survey of Hough transform. Pattern Recognit 48(3):993–1010
17. Zafari S, Eerola T, Sampo J, Kälviäinen H, Haario H (2017) Comparison of concave point detection methods for overlapping convex objects segmentation. In: Scandinavian conference on image analysis. Springer, Berlin, pp 245–256
18. Zafari S, Eerola T, Sampo J, Kälviäinen H, Haario H (2015) Segmentation of partially overlapping nanoparticles using concave points. In: International symposium on visual computing. Springer, Berlin, pp 187–197
19. Farhan M, Yli-Harja O, Niemistö A (2013) A novel method for splitting clumps of convex objects incorporating image intensity and using rectangular window-based concavity point-pair search. Pattern Recognit 46(3):741–751
20. Zafari S, Eerola T, Sampo J, Kälviäinen H, Haario H (2016) Segmentation of partially overlapping convex objects using branch and bound algorithm. In: Asian conference on computer vision. Springer, Berlin, pp 76–90
21. Fitzgibbon A, Pilu M, Fisher RB (1999) Direct least square fitting of ellipses. IEEE Trans Pattern Anal Mach Intell 21(5):476–480
22. Panagiotakis C, Argyros A (2016) Parameter-free modelling of 2D shapes with ellipses. Pattern Recognit 53:259–275
23. Chen Q, Yang X, Petriu EM (2004) Watershed segmentation for binary images with different distance transforms. In: Proceedings of the 3rd IEEE international workshop on haptic, audio and visual environments and their applications, pp 111–116
24. Park C, Huang JZ, Ji JX, Ding Y (2012) Segmentation, inference and classification of partially overlapping nanoparticles. IEEE Trans Pattern Anal Mach Intell 35(3):1–1
25. Cheng J, Rajapakse JC et al (2008) Segmentation of clustered nuclei with shape markers and marking function. IEEE Trans Biomed Eng 56(3):741–748
26. Jung C, Kim C (2010) Segmenting clustered nuclei using H-minima transform-based marker extraction and contour parameterization. IEEE Trans Biomed Eng 57(10):2600–2604
27. Kapaldo J, Han X, Mery D (2018) Seed-point detection of clumped convex objects by short-range attractive long-range repulsive particle clustering. arXiv preprint arXiv:180404071
28. Fabbri R, Costa LDF, Torelli JC, Bruno OM (2008) 2D Euclidean distance transform algorithms: a comparative survey. ACM Comput Surv (CSUR) 40(1):2
29. Reddy PR, Amarnadh V, Bhaskar M (2012) Evaluation of stopping criterion in contour tracing algorithms. Int J Comput Sci Inf Technol 3(3):3888–3894
30. Carmona-Poyato A, Madrid-Cuevas FJ, Medina-Carnicer R, Muñoz-Salinas R (2010) Polygonal approximation of digital planar curves through break point suppression. Pattern Recognit 43(1):14–25
31. Hillier FS, Lieberman GJ (1980) Introduction to operations research. Holden-Day, Inc., San Francisco
32. Mullikin JC (1992) The vector distance transform in two and three dimensions. CVGIP Graph Models Image Process 54(6):526–535
33. The Math Works, Inc. (2019) MATLAB version 9.7.0.1247435 (R2019b). The Math Works, Inc., 2019

34. Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218
35. Rand WM (1971) Objective criteria for the evaluation of clustering methods. J Am Stat Assoc 66(336):846–850
36. Vinh NX, Epps J, Bailey J (2010) Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance. J Mach Learn Res 11(Oct):2837–2854
37. Coelho LP, Shariff A, Murphy RF (2009) Nuclear segmentation in microscope cell images: a hand-segmented dataset and comparison of algorithms. In: 2009 IEEE International symposium on biomedical imaging: from nano to macro. IEEE, pp 518–521
38. Bergeest JP, Rohr K (2012) Efficient globally optimal segmentation of cells in fluorescence microscopy images using level sets and convex energy functionals. Med Image Anal 16(7):1436–1444
39. Arthur RM, Humburg PJ, Hoogenboom J, Baiker M, Taylor MC, de Bruin KG (2017) An image-processing methodology for extracting bloodstain pattern features. Forensic Sci Int 277:122–132
40. Arthur R, Hoogenboom J, Baiker M, Taylor M, de Bruin K (2018) An automated approach to the classification of impact spatter and cast-off bloodstain patterns. Forensic Sci Int 289:310–319
41. Attinger D, Liu Y, Bybee T, De Brabanter K (2018) A data set of bloodstain patterns for teaching and research in bloodstain pattern analysis: impact beating spatters. Data Brief 18:648–654