

RESEARCH PAPER

Open Access



Effective elliptic arc selection from connected edge points

Tomonari Masuzaki* and Yasuyuki Sugaya

Abstract

Extracting edge points from an image and fitting ellipses to them is a fundamental technique for computer vision applications. However, since the extracted edge points sometimes contain non-elliptic arcs such as line segments, it is a very difficult to extract only elliptic arcs from them. In this paper, we propose a new method for extracting elliptic arcs from a spatially connected point sequence. We first fit an ellipse to an input point sequence and segment the sequence into partial arcs at the intersection points of the fitted ellipse. Next, we compute residuals of the fitted ellipse for all input points and select elliptic arcs among the segmented arcs by checking the curvatures of the residual graph. Then, we fit an ellipse to the selected arcs and repeat the above process until the selected arcs do not change. By using simulated data and real images, we compare the performance of our method with existing methods and show the efficiency of our proposed method.

Keywords: Elliptic arc selection, Ellipse fitting, Outlier detection

1 Introduction

A circular object in a scene is projected onto the image plane as an ellipse, and we can compute its 3-D positions from that ellipse [5]. Therefore, detecting circles and ellipses in images is the first step of many computer vision applications including industrial robotic operations and autonomous navigation. For this purpose, many methods for extracting elliptic arcs from an image and for fitting an ellipse to the extracted elliptic arcs are studied [6, 15, 20].

In order to detect ellipses in images, we usually extract edge points from images and fit or estimate the ellipse parameters from them. We classify ellipse detection problems into two stages. One stage is to estimate ellipse parameters accurately for given points. The other is to select elliptic arcs from the extracted edge points.

For the former problem, many methods have been proposed. The simplest method for fitting an ellipse is *least squares (LS)* which minimizes the sum of squares of the ellipse equation. A more accurate method was proposed by Taubin [17]. Moreover, *hyper-LS* [8, 9] and *hyper-renormalization* [10] whose solution has no deviation up to high-order noise terms are also proposed. *Maximum*

likelihood (ML) is also a well-known method, which minimizes the *reprojection error*, i.e., the sum of the distances from data points to the fitted ellipse [4]. Many ML-based methods have been proposed [1, 7, 11, 13].

Ellipse-specific methods also have been proposed [3, 12, 16, 18, 19]. It is Fitzgibbon et al. [3] who first proposed a method that only fits an ellipse.

The latter problem of ellipse detection is to select elliptic arcs from the extracted edge points. Since all the above methods do not consider the presence of non-elliptic arcs, which we call “outliers” in this paper, in the input point sequence, the accuracy of those solutions deteriorate if the input point sequence includes outliers. So, it is an important task to separate elliptic arcs from the input point sequence.

Ad hoc and simple methods for detecting non-elliptic arcs are line fitting and curvature-based methods. However, these methods may detect special targets, for example, line-fitting methods detect only line segments, so they cannot always remove all outliers. Random sample consensus (RANSAC) is a well-known framework for dealing with outliers [2]. RANSAC has problems that it needs a lot of iterations and does not work well if the number of outliers are larger than that of inliers. Yu et al. [20] detected an outlier point sequence from fitting residuals and removed it from the input data. By iteratively

*Correspondence: masuzaki@iim.cs.tut.ac.jp
Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi, Aichi 441-8580, Japan

applying the above procedure, they fitted an ellipse to the remaining inliers. Shao et al. [14] segment an input point sequence into some spatially connected arcs and detect elliptic arcs by fitting ellipses to all the combinations of the segmented arcs.

We propose a new method for extracting elliptic arcs from a spatially connected point sequence. Assuming that input data is a spatially connected sequence of edge points, we segment it into partial arcs by considering the ellipse fitting residuals and detect inliers by computing the curvature of the residual graph of each of the segmented arcs.

Our method has several advantages over existing methods. (1) The proposed method can automatically segment an input point sequence at the intersections of the fitted ellipse and select inlier elliptic arcs by repeating point segmentation and ellipse fitting. (2) Our method involves iterations, but the number of iterations is much less than those of iterative methods like RANSAC and Yu's method. (3) Moreover, in contrast to Yu's method, our method has the possibility of fitting a more accurate ellipse because inliers are selected from all the input data in each iteration step, meaning that the number of data to fit an ellipse does not decrease by iterations. (4) Our method also detect outlier arcs which are smoothly connected to inlier arcs. This type of outliers cannot detect by the curvature-based method.

2 Ellipse fitting

Curves represented by a quadratic equations in x and y in the form

$$Ax^2 + 2Bxy + Cy^2 + 2f_0(Dx + Ey) + f_0^2F = 0, \quad (1)$$

are called "conics," which include ellipses, parabolas, hyperbolas, and their degeneracies such as two lines [5]. The condition that Eq. (1) represents an ellipse is

$$AC - B^2 > 0. \quad (2)$$

Our task is to compute the coefficients A, \dots, F so that the ellipse of Eq. (1) passes through the detected points $(x_\alpha, y_\alpha), \alpha = 1, \dots, N$, as closely as possible (Fig. 1). In Eq. (1), f_0 is a constant that has the order of the image size for stabilizing finite length numerical computation.¹ For a point sequence $(x_\alpha, y_\alpha), \alpha = 1, \dots, N$, we define 6-D vectors

$$\begin{aligned} \xi_\alpha &= (x_\alpha^2, 2x_\alpha y_\alpha, y_\alpha^2, 2f_0 x_\alpha, 2f_0 y_\alpha, f_0^2)^\top, \\ \theta &= (A, B, C, D, E, F)^\top. \end{aligned} \quad (3)$$

The condition that (x_α, y_α) satisfies Eq. (1) is written as

$$(\xi_\alpha, \theta) = 0, \quad (4)$$

where (\mathbf{a}, \mathbf{b}) denotes the inner product of vectors \mathbf{a} and \mathbf{b} . Since vector θ has scale indeterminacy, we normalize it to unit norm: $\|\theta\| = 1$.

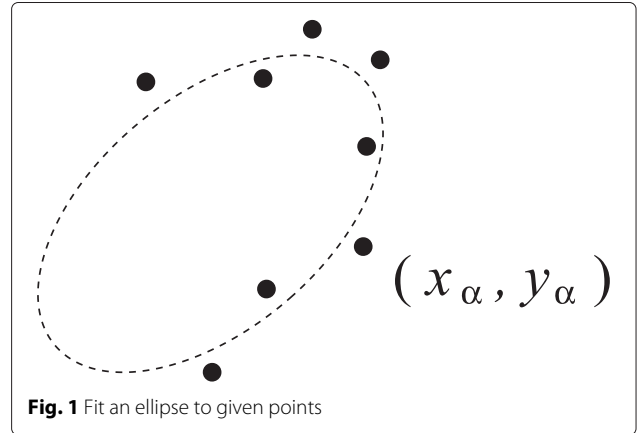


Fig. 1 Fit an ellipse to given points

Since Eq. (4) is not exactly satisfied in the presence of noise, we compute a θ such that $(\xi_\alpha, \theta) \approx 0, \alpha = 1, \dots, N$. For computing a θ that is close to its true value, we need to consider the statistical properties of noise. The standard model is to regard the noise in (x_α, y_α) as an independent Gaussian random variable of mean 0 and standard deviation σ . Then, the covariance matrix of the vector ξ_α has the form $\sigma^2 V_0[\xi_\alpha]$, where

$$V_0[\xi_\alpha] = 4 \begin{pmatrix} x_\alpha^2 & x_\alpha y_\alpha & 0 & f_0 x_\alpha & 0 & 0 \\ x_\alpha y_\alpha & x_\alpha^2 + y_\alpha^2 & x_\alpha y_\alpha & f_0 y_\alpha & f_0 x_\alpha & 0 \\ 0 & x_\alpha y_\alpha & y_\alpha^2 & 0 & f_0 y_\alpha & 0 \\ f_0 x_\alpha & f_0 y_\alpha & 0 & f_0^2 & 0 & 0 \\ 0 & f_0 x_\alpha & f_0 y_\alpha & 0 & f_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (5)$$

which we call the "normalized covariance matrix" [7]. We use the normalized covariance matrix to compute residuals of ellipse fitting.

3 Fitzgibbon's method

In order to segment an input point sequence at intersection points of a fitted ellipse, we fit an ellipse by Fitzgibbon's method. For fitting only an ellipse, Fitzgibbon et al. [3] proposed to minimize the *algebraic distance*

$$J_{LS} = \sum_{\alpha=1}^N (\xi_\alpha, \theta)^2, \quad (6)$$

subject to $AC - B^2 = 1$. This constraint is written as

$$(\theta, N_F \theta) = 1, \quad N_F \equiv \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (7)$$

The solution that minimizes Eq. (6) subject to this constraint is obtained by solving a generalized eigenvalue problem

$$N_F \theta = \mu M_{LS} \theta, \quad (8)$$

and computing the unit eigenvector θ for the largest eigenvalue μ , where the matrix M_{LS} is defined by

$$M_{LS} = \frac{1}{N} \sum_{\alpha=1}^N \xi_{\alpha} \xi_{\alpha}^{\top}. \quad (9)$$

4 Proposed method

We assume that input data is spatially connected points and that the fitted ellipse intersects the curve at multiple points (Fig. 2a). We automatically segment the input curve into partial arcs at these intersection points. We compute the variation of the tangent angle to the graph of the fitting residual, which we simply call “error curvature” in this paper, and judge if each arc is an inlier or an outlier based on this “error curvature.”

The principle of our inlier arc selection is nearly equivalent to the curvature-based method. However, our method is more efficient, because our method computes the curvature of the residual graph at only one point where the residual takes a maximum in each segmented arc.

In Fig. 2, the input point sequence is divided into five partial arcs by the fitted ellipse. The residual value of the arc PQ , which consists only of an elliptic arc, smoothly changes around the peak value. On the other hand, we can see that the residual graph has a peaky shape over the arcs consisting of non-elliptic arcs.

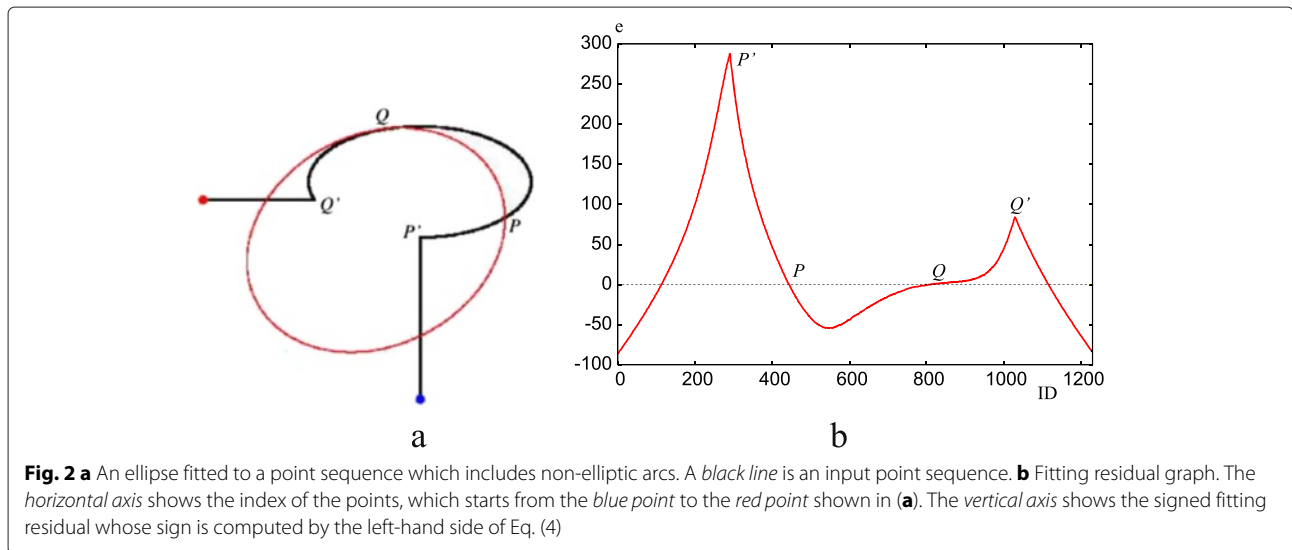
Moreover, we can see that the arcs PP' and QQ' , which are connected to the elliptic arc PQ , are also elliptic arcs. The points P' and Q' are the peak points of the partial arcs adjacent to the elliptic arc PQ . Therefore, if we use

not only the detected inlier arc but also the adjacent arcs like the arcs PP' and QQ' for ellipse fitting, we can effectively fit a correct ellipse. The algorithm of our method is summarized as follows:

1. Fit an ellipse to a point sequence by Fitzgibbon’s method [3].
2. Compute the sign of the left-hand side of Eq. (4) for all the points and segment the point sequence into partial arcs at the points across which the computed sign changes.
3. For each segmented arc, detect the point where the residual value takes a maximum and compute its error curvature ϕ at this point.
4. Go to step (a) if it is the first inlier selection, else go to step (b).
 - (a) Select an inlier arc which has the smallest value ϕ among those arcs whose arc lengths are longer than a threshold² and extend it to each peak point of adjacent arcs. Then, we fit an ellipse to the selected arc.
 - (b) Select the arcs whose error curvature ϕ s are smaller than a threshold $\hat{\phi}^3$ and fit an ellipse to them.
5. Repeat the procedures from step 2 to step 4 until the number of inliers does not change.

As discussed before, if a selected arc is an elliptic arc, the adjacent arcs are also elliptic arcs. Therefore, we can effectively fit a correct ellipse if we use those arcs. However, if we select a non-elliptic arc as an inlier and add adjacent outlier arcs to fit an ellipse, we cannot fit a correct ellipse.

For this reason, in the first iteration of our algorithm, we select among the arcs that are sufficiently long the one



whose error curvature ϕ at its peak point is the smallest. We regard it as a reliable inlier and extend it to the adjacent partial arcs. After the first iteration, we select all arcs whose error curvature ϕ s are smaller than a threshold $\hat{\phi}$. We do not extend those arcs, because the adjacent arcs are non-elliptic arcs if the selected inlier arcs approximately belong to the correct ellipse.

In the following sections, we describe the details of our method.

5 Segmentation of the point sequence

The left-hand side of Eq. (4) at the point p_α has a different sign outside and inside the ellipse θ . Using this fact, we can segment the input point sequence $\{p_\alpha\}$ into partial arcs at those p_α points which have different signs from their neighboring points.

However, the value (ξ_α, θ) may not be exactly zero even if the point p_α lies on the fitted ellipse θ ; the points lying on the fitted ellipse may irregularly change their signs, so these points may be segmented in very short arcs. To avoid this, we regard those partial arcs whose fitting residual values are close to zero as elliptic arcs and judge that they are inlier arcs without computing their curvatures. We call such arcs “tangent arcs.”

For fitted ellipse θ , we first compute a fitting residual e_α of a point p_α , $\alpha = 1, \dots, N$ by Eq. (10), which is a first approximation of the distance between the fitted ellipse and the point p_α . We also compute a sign of the residual by the value of the left-hand side of Eq. (4).

$$e_\alpha = \sqrt{\frac{(\xi_\alpha, \theta)^2}{(\theta, V_0[\xi_\alpha] \theta)}}. \quad (10)$$

Next, we segment the input point sequence into partial arcs at those p_α points which have different signs from their neighboring points. If the maximum residual in the segmented arc is smaller than a threshold E_{\min} ⁴, we regard the arc R_κ as a *tangent arc*.

6 Inlier arc selection

After segmenting the point sequence, we detect the point α^* where the fitting residual takes its maximum and compute its *error curvature* at this point by using its neighboring points β and γ (Fig. 3). If the computed curvature is larger than a threshold, we regard this arc as an outlier. Since the horizontal axis of the residual graph $\bar{Q}_\alpha = (\alpha, e_\alpha)^\top$ is an index of the input point sequence, the computed error curvature depends on the scale of the horizontal axis of the residual graph. For example, if two point sequences have the same shape and different scales, the curvatures of their sequences have different values.

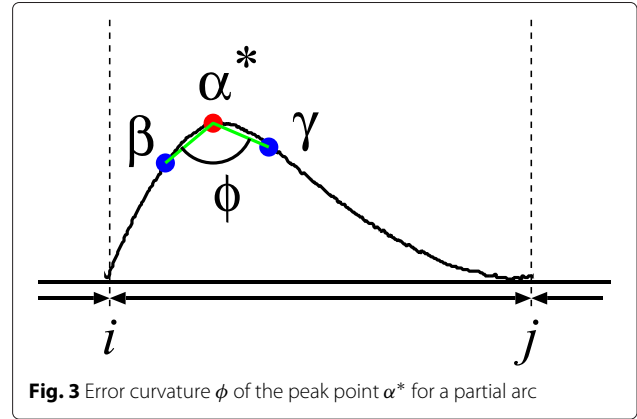


Fig. 3 Error curvature ϕ of the peak point α^* for a partial arc

Therefore, we normalize the scale of the horizontal axis of the residual graph in the form

$$Q_\alpha = \left(\frac{\lambda e_{\max} \alpha}{N}, e_\alpha \right)^\top, \quad (11)$$

where e_{\max} is the maximum value of all ellipse-fitting residuals and λ is a constant for normalization.⁵ If e_{\max} is extremely large, the normalization of Eq. (11) may not work well. So, if e_{\max} is larger than a threshold E_{\max} ⁶, we replacing the value e_{\max} with the threshold E_{\max} in the normalization computation.

In order to compute the error curvature, we need to select neighboring points Q_β and Q_γ . We first select indices β and γ such that $\beta = \alpha^* - d$ and $\gamma = \alpha^* + d$. Here, d is a constant for determining the distance from the point Q_{α^*} . If both of the points Q_β and Q_γ are not included in R_κ , we decrease d until either of them is included in R_κ . If Q_{α^*} is located near the start or the end point of the arc R_κ , we cannot select both of the neighboring points properly. In this case, we select at least one neighboring point and compute its symmetric point to α^* . More details are described in the following algorithm.

For M -segmented arcs $R_\kappa(i, j) = \{Q_\delta \mid \delta = i, \dots, j\}$, $\kappa = 1, \dots, M$, we compute error curvatures for selecting inlier arcs by the following algorithm.

1. Let α^* be the index of the point whose residual takes its maximum in the arc R_κ . Here, if a target arc is a tangent arc, finish this procedure.
2. Select two points whose indices β and γ are such that

$$\beta = \alpha^* - d, \gamma = \alpha^* + d, d = (j - i)/r, \quad (12)$$

where r ⁷ is a constant for determining the distance between the point α^* and its neighboring points β and γ . If both β and γ are out of R_κ , we update d to $d \leftarrow d - 1$ until either of the two points are in the arc R_κ .

3. Compute two vectors $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$ according to the following three rules. Here, $x_b^{(a)}$ denotes the b -th component of the vector $\mathbf{x}^{(a)}$.

case a: Points Q_β and Q_γ are both in the arc R_κ .

$$\mathbf{x}^{(1)} = Q_{\alpha^*} - Q_\beta, \mathbf{x}^{(2)} = Q_{\alpha^*} - Q_\gamma. \quad (13)$$

case b: Point Q_β is in the arc R_κ .

$$\mathbf{x}^{(1)} = Q_{\alpha^*} - Q_\beta, \mathbf{x}^{(2)} = (-x_1^{(1)}, x_2^{(1)})^\top. \quad (14)$$

case c: Point Q_γ is in the arc R_κ .

$$\mathbf{x}^{(2)} = Q_{\alpha^*} - Q_\gamma, \mathbf{x}^{(1)} = (-x_1^{(2)}, x_2^{(2)})^\top. \quad (15)$$

4. Compute the error curvature ϕ by

$$\phi = \pi - \cos^{-1} \left(\frac{(\mathbf{x}^{(1)}, \mathbf{x}^{(2)})}{\|\mathbf{x}^{(1)}\| \|\mathbf{x}^{(2)}\|} \right). \quad (16)$$

5. Regard the arc R_κ as an inlier arc if ϕ is smaller than the threshold $\hat{\phi}$.

If it is the first iteration of an inlier selection, we extend the selected inlier arc to each peak point of its adjacent arcs. If the adjacent arc is a *tangent arc*, we check the next adjacent arc and add the partial non-*tangent arc* as an inlier.

7 Experiment

7.1 Inlier selection process

In order to confirm how our proposed method selected inlier arcs, we checked the residual graph and the selected inlier arcs and fitted the ellipse step by step for Fig. 4(1)-(a) and Fig. 5(1)-(a).

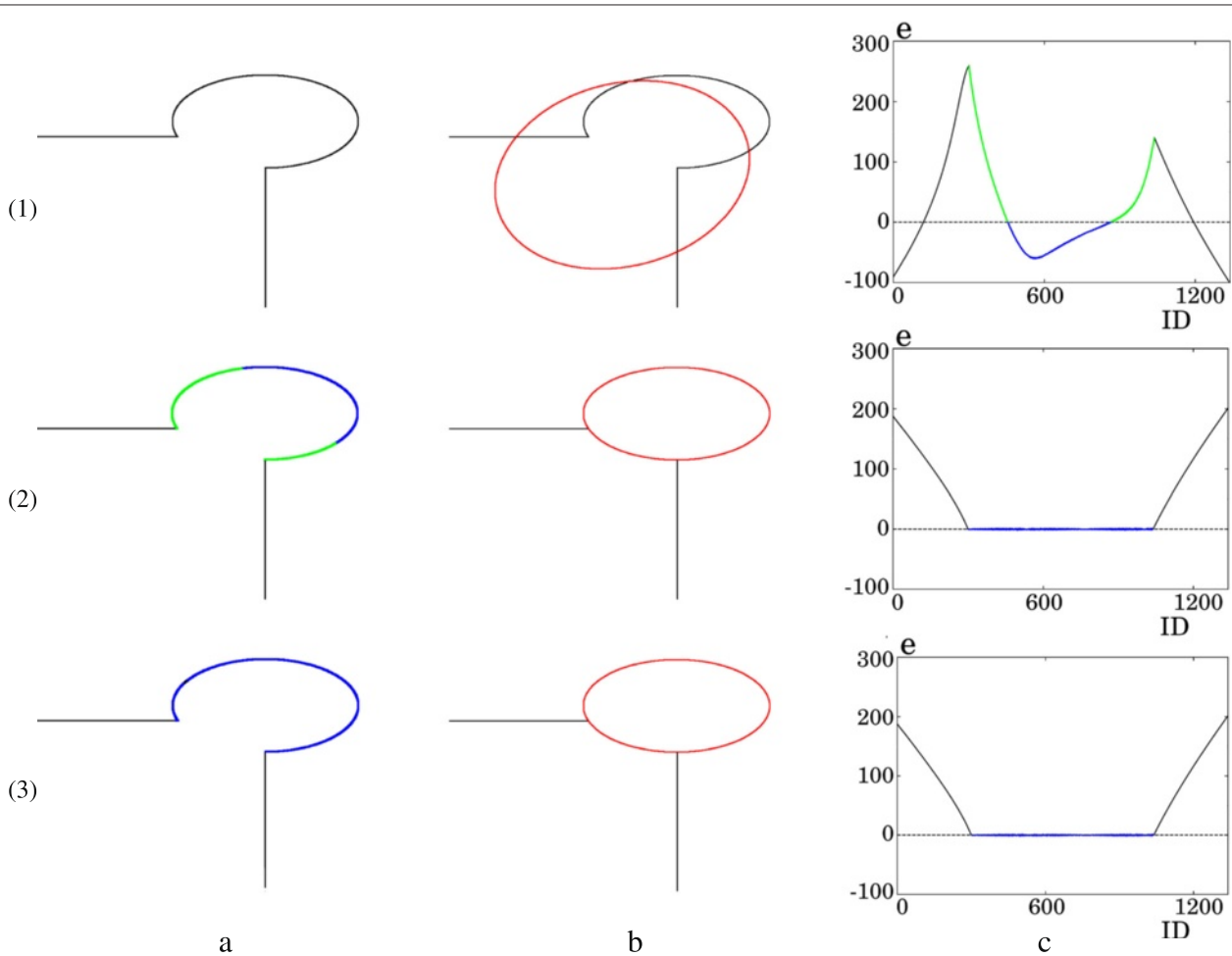


Fig. 4 Inlier arc selection process of our method. **a** Used points to fit the ellipse in **(b)**. The *blue arc* is the selected inlier arc. The *green arcs* are the extended arcs from the selected inlier arc. **b** Fitted ellipse. **c** Signed fitting residual graph. The *horizontal axis* shows the index of the points. The *vertical axis* shows the signed fitting residual. The *blue and green arcs* correspond to the arcs in **(a)** in the next row

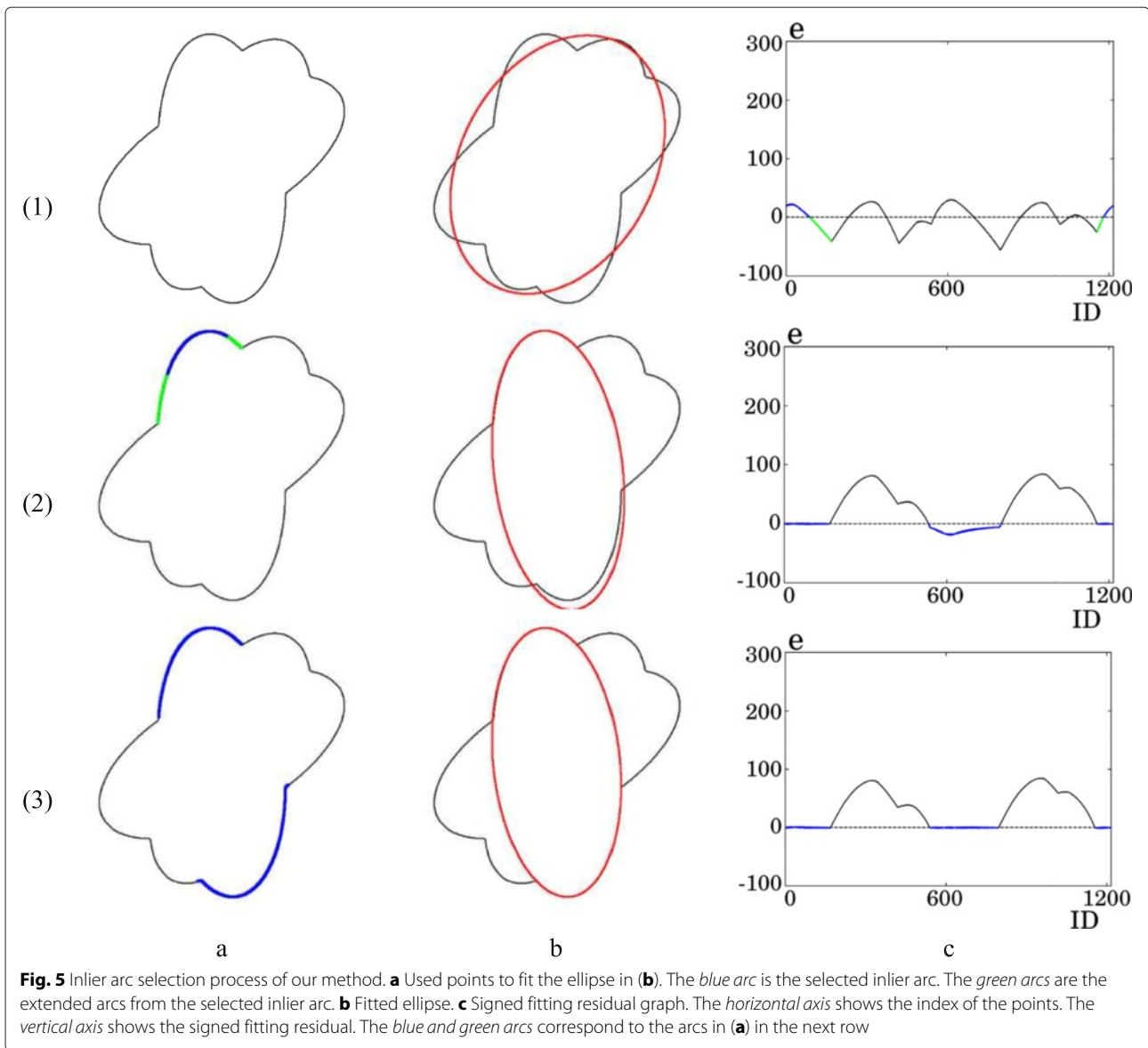
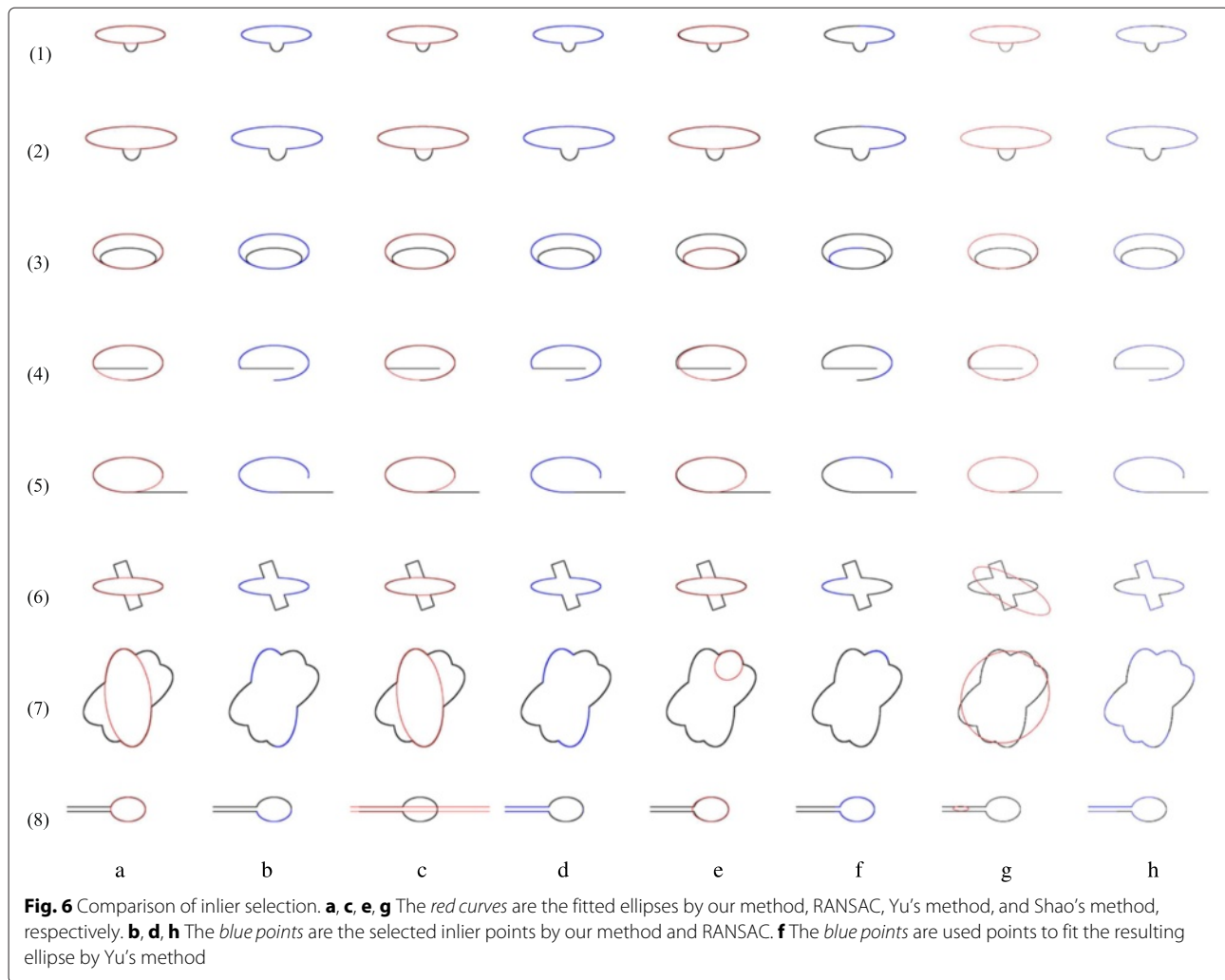


Figure 4(1)-(b) shows the fitted ellipse to all the input data by Fitzgibbon's method, and Fig. 4(1)-(c) is its signed residual graph. The horizontal and vertical axis of the graph indicate the index of the input points and a signed fitting residual, respectively.

In our first inlier selection, the blue arc was selected as an inlier arc and the green adjacent partial arcs were extended as inliers. These arcs correspond to the blue and green arcs in Fig. 4(2)-(a). In the second step, we fitted an ellipse to the selected inlier arcs (Fig. 4(2)-(a)). The fitted ellipse was shown in Fig. 4(2)-(b). As we can see, the segmented arcs could be obviously separated to inlier arcs (*tangent arcs*) and outlier arcs in Fig. 4(2)-(c). Our method stopped in the third step because the number of inlier points did not change and could fit a correct ellipse to all the inlier points.

The input data of Fig. 5(1)-(a) consists of three different elliptic arcs. Figure 5(1)-(b), (1)-(c) are the fitted ellipse to all the input data and its signed residual graph, respectively. This input point sequence is closed, so we consider that the start and the end points are connected in the residual graph. The blue arc is the selected inlier arc, and the green adjacent arcs are extended arcs as inliers. These arcs correspond to the blue and green arcs in Fig. 5(2)-(a). By checking the selected inlier arcs in Fig. 5(2)-(a), we can confirm that the selected arcs belong to one ellipse. Figure 5(2)-(b) shows the fitted ellipse to them. Since the selected arcs are short for the entire ellipse, the accuracy of the fitted ellipse is not good. However, our method selects the remaining elliptic arcs as inliers from the residual graph of Fig. 5(2)-(c) and fits a correct ellipse in the third step.



From these results, we could confirm that our method could select inlier arcs by repeating inlier selection based on an *error curvature* and ellipse fitting.

7.2 Simulations

In order to confirm the effectiveness of the our proposed method, we compared our method with RANSAC, Yu's

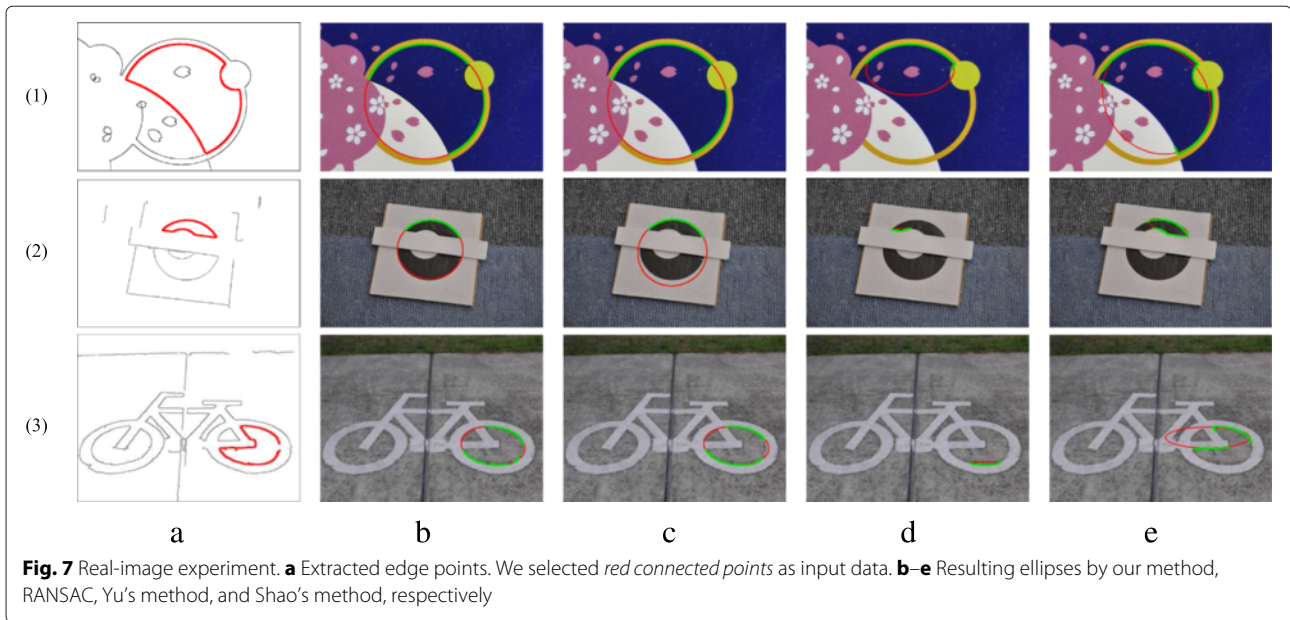
method, and Shao's method for many simulation data. Figure 6 shows some of the resulting ellipses. Figure 6(a), (c), (e), (g) are the fitted ellipses by our method, RANSAC, Yu's method, and Shao's method, respectively. The blue points in Fig. 6(b), (d), (h) are the selected inlier points

Table 1 Inlier selection ratios/outlier selection ratios

	Our method	RANSAC	Yu's method	Shao's method
(1)	0.99/0.00	0.99/0.00	0.49/0.01	0.80/0.00
(2)	0.99/0.00	1.00/0.01	0.49/0.02	0.81/0.00
(3)	0.99/0.04	1.00/0.07	0.40/0.04	0.93/0.09
(4)	1.00/0.00	1.00/0.01	0.36/0.00	0.90/0.05
(5)	0.99/0.00	1.00/0.04	0.47/0.00	0.93/0.04
(6)	0.99/0.00	1.00/0.01	0.48/0.01	0.61/0.45
(7)	0.90/0.00	1.00/0.02	0.45/0.06	0.54/0.68
(8)	0.50/0.00	0.02/0.98	1.00/0.10	0.51/0.63

Table 2 Comparison of computation time and number of iterations: computation time in milliseconds (number of iterations)

	Our method	RANSAC	Yu's method
(1)	4 (5)	41 (111)	80 (20)
(2)	8 (5)	61 (113)	148 (48)
(3)	12 (7)	79 (137)	208 (17)
(4)	4 (3)	49 (152)	288 (106)
(5)	8 (9)	42 (148)	600 (178)
(6)	4 (4)	59 (218)	216 (72)
(7)	8 (8)	68 (134)	48 (10)
(8)	4 (3)	64 (143)	424 (157)



by our method, RANSAC, and Shao's method, respectively. The blue points in Fig. 6(f) are used points to fit the resulting ellipse by Yu's method.

Input data of Fig. 6(1), (2) have the same shape, but their scales are different. For these data, all the methods fitted a correct ellipse. In Fig. 6(3), two ellipses are tangent to each other. Our proposed method, RANSAC, and Shao's method fitted an ellipse to the outer arc; however, Yu's method fitted an ellipse to the inner arc.

Figure 6(4) is an example which includes a line segment as outlier points. All methods fitted an almost correct ellipse. In Fig. 6(4), since Yu's method fitted an ellipse to a short arc, the accuracy of the resulting ellipse is not good compared with the other methods. Figure 6(5) is an example where an outlier arc is smoothly connected to an inlier arc. For such input data, outlier detection based on the curvature of a point sequence may not work well. However, our method can correctly detect inlier and outlier arcs because it iteratively selects inlier arcs and fits an ellipse. In Fig. 6(6), (7), input point sequences consist of elliptic arcs and contours of other shapes. Our method and RANSAC fitted a correct ellipse for each data. However, Yu's method fitted a small ellipse to a short arc for Fig. 6(7). Shao's method detected many outlier points. The inlier selection process of our method for Fig. 6(7) is shown in Fig. 5. The input data of Fig 6(8) includes many outlier points. Our method and Yu's method output an almost correct ellipse. Since the ratio of outliers was large, RANSAC did not work well.

Table 1 shows the inlier selection ratios for the simulation data shown in Fig. 6. We manually counted the number of inlier and outlier points in the input point sequences and computed inlier and outlier selection

ratios. From these results, the capabilities of the inlier arc selection of our method and RANSAC are almost the same, but our method is superior to RANSAC if an input point sequence includes many outlier points. In many cases, Yu's method detects small number of inliers. In Figs. 6(3), (7), since Yu's method detected elliptic arcs from the different ellipse from that of the other methods, we computed the inlier and outlier selection ratios for the corresponding ellipse.

Table 2 shows the number of iterations and computation times for the three methods. We used Intel Core 2 Duo 3.00 GHz $\times 2$ for the CPU with main memory 4 GB and Ubuntu 12.04 for the OS. For RANSAC, we stopped if the solution did not change after 50 consecutive iterations and counted the mean total number of iterations over 10 trials. Since our method, RANSAC, and Yu's method were implemented in C++, but Shao's method was implemented in Matlab, we did not show the computation time of Shao's method. From this result, the number of iterations and the computation time of our method is superior to RANSAC and Yu's method.

7.3 Real image experiment

Figure 7 shows real-image experiments. Figure 7(a) is the extracted edge points by canny operator. We removed

Table 3 Inlier selection ratios/outlier selection ratios

	Our method	RANSAC	Yu's method	Shao's method
(1)	1.00/0.00	1.00/0.02	0.49/0.08	0.95/0.23
(2)	1.00/0.00	1.00/0.02	0.00/1.00	0.71/0.46
(3)	0.83/0.02	0.83/0.05	0.27/0.00	0.36/0.48

Table 4 Comparison of computation time and number of iterations: computation time in milliseconds (number of iterations)

	Our method	RANSAC	Yu's method
(1)	3 (4)	70 (157)	228 (65)
(2)	1 (6)	16 (141)	3 (5)
(3)	4 (10)	49 (208)	56 (35)

successive edge points whose lengths were shorter than 50 pixels. The red points shown in Fig. 7(a) are manually selected edge points to fit an ellipse. Figures 7(b–e) are extracted elliptic points and the fitted ellipses by our method, RANSAC, Yu's method, and Shao's method, respectively. The green points in Figs. 7(b–e) indicate extracted the elliptic points, and the red ellipses are the fitted ellipses to those extracted points. Our method and RANSAC fitted almost correct ellipses. Yu's method fitted small ellipses for three data. We discuss

the reason that Yu's method tends to fit a small ellipse later. Shao's method extracted many outlier points in all data. This is because Shao's method could not separate inlier and outlier arcs in their point sequence segmentation step since they only consider the distance between adjacent points and the length of the segmented arcs.

Table 3 shows the inlier selection ratios for real images shown in Fig. 7. As we can see, our method and RANSAC have almost the same performance of inlier selection. However, our method has lower outlier selection ratio than that of RANSAC. Table 4 shows the number of iterations and computation times for Fig. 7.

Figure 8 shows the inlier selection process and corresponding residual graphs for Fig. 7(1). From the residual graph of the first ellipse fitting (Fig. 8(1)–(c)), our method selected the top-left arc of an ellipse as an inlier (the blue points in Fig. 8(2)–(a)) and extended the adjacent partial arcs as inliers (the green points in Fig. 8(2)–(a)). Since a correct ellipse was fitted to the selected arcs, our method

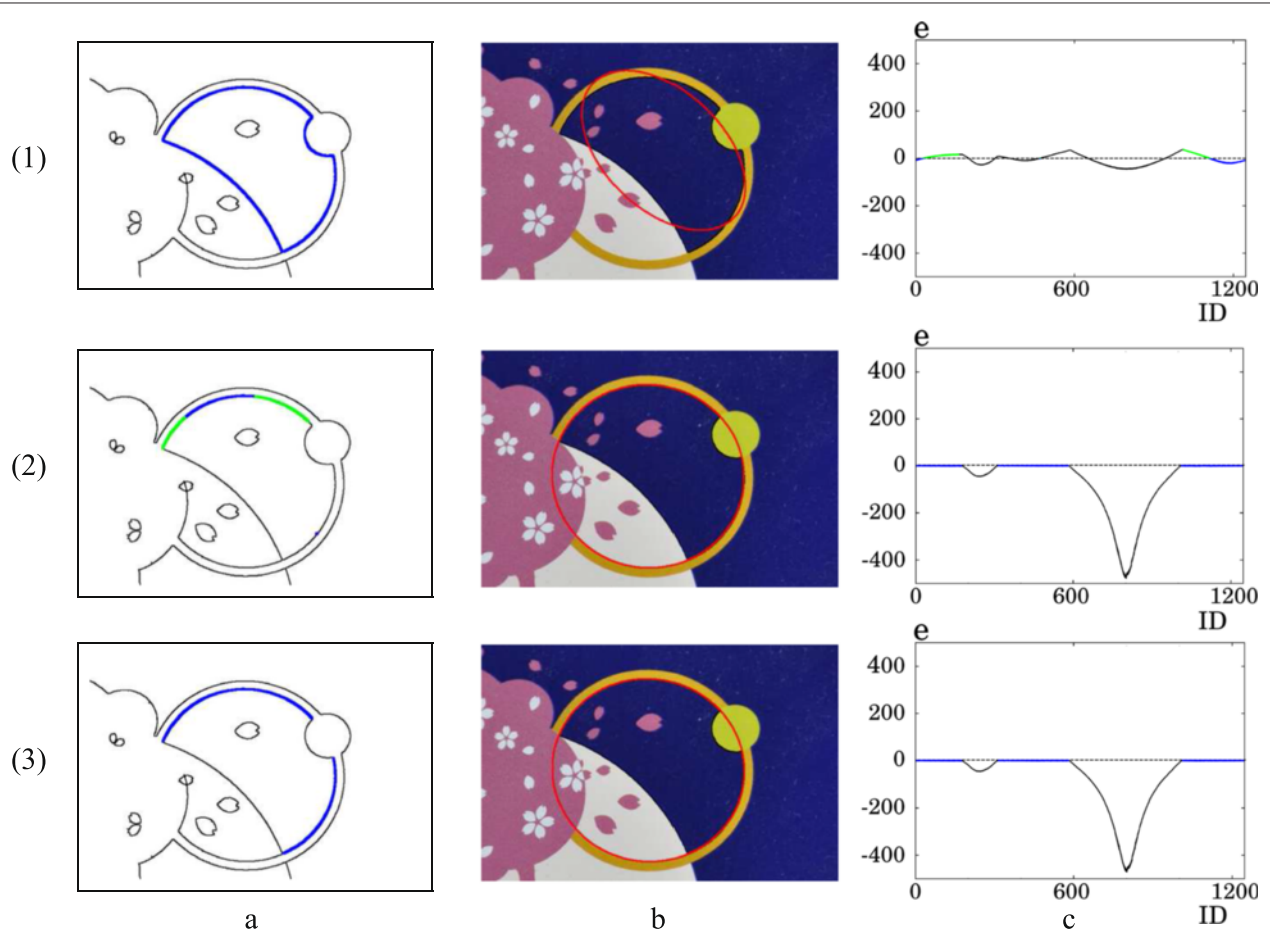


Fig. 8 Inlier arc selection process of our method. **a** Used points to fit the ellipse in **(b)**. The blue arc is the selected inlier arc. The green arcs are the extended arcs from the selected inlier arc. **b** Fitted ellipse. **c** Signed fitting residual graph. The horizontal axis shows the index of the points. The vertical axis shows the signed fitting residual. The blue and green arcs correspond to the arcs in **(a)** in the next row

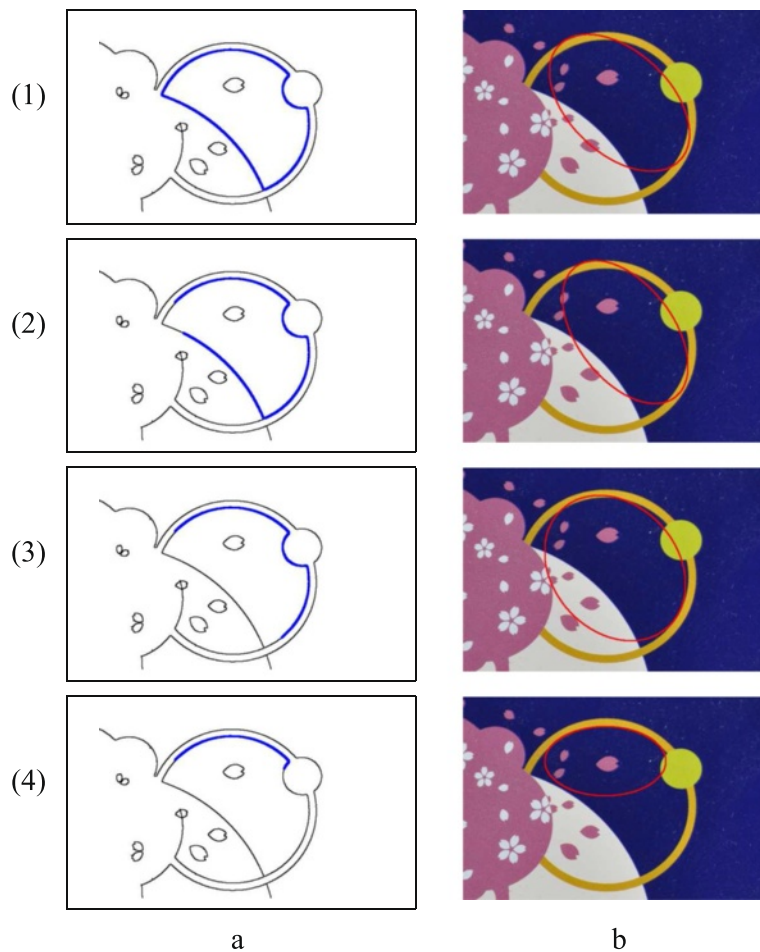


Fig. 9 Inlier selection process of Yu's method. **a** Used points to fit the ellipse in **(b)**. The *blue arc* is the selected inlier arc. **b** Fitted ellipse

also selected a right-bottom arc of the correct ellipse and converged in the next step.

We show the inlier selection process of Yu's method in Fig. 9. Figure 9(a) shows the remaining input points in each iteration step. Yu's method fits an ellipse to the input points and removes the points whose fitted residuals are larger than a threshold. So, the number of the input points gradually decreases in the course of the ellipse-fitting step.

For this reason, the number of the remaining inlier points becomes small and the resulting ellipse tends to be small compared with the correct shape.

Figure 10 shows a result where our method selects wrong point sequences. Our method cannot select point sequences on an ellipse if the rate of input points on the correct ellipse is low; hence, our method selects the wrong point sequence at the first iteration.

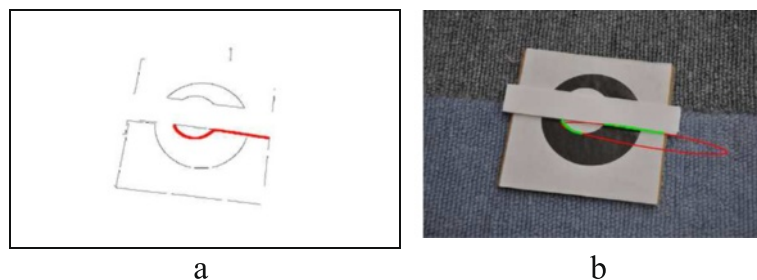


Fig. 10 Wrong selection of our method. **a** Extracted edge points. We selected the *red connected points* as the input data. **b** Fitted ellipse

8 Concluding remarks

We proposed a new method for extracting elliptic arcs from a spatially connected point sequence. Assuming that input points are a spatially connected sequence of edge points, we fit an ellipse to it and automatically segment it into partial arcs at the intersection points of the fitted ellipse. Then, we compute residuals of the fitted ellipse for all input points and select elliptic arcs among the segmented arcs by checking curvatures of the residual graph.

Our proposed method involves iterations, but the number of iterations is much less than those of RANSAC and Yu's method. Moreover, in contrast to Yu's method, our method has the possibility of fitting a more accurate ellipse because inliers are selected from all the input data in each iteration, meaning that the number of data to fit an ellipse does not decrease by iterations.

By using simulated data and real images, we compared the performance of our method with existing methods and showed that the performance of the inlier selection and computation time of the proposed method were superior to existing methods.

Endnotes

¹We set $f_0 = 600$.

²We set the threshold to be 5 % of the number of input edge points.

³We set $\hat{\phi} = 80^\circ$.

⁴We set $E_{\min} = 1.5$.

⁵We set $\lambda = 2.0$.

⁶We set E_{\max} to be (maximum image coordinates of the input points)/3.

⁷We set $r = 8$.

Acknowledgements

The authors thank Kenich Kanatani of Okayama University, Japan, for helpful discussions. The authors also thank Yoshihisa Iriji of OMRON Corporation, Japan, for allowing us to use the source code. This work was supported in part by JSPS Grant-in-Aid for Scientific Research(C) (26330192).

Authors' contributions

YS proposed a method to automatically segment a consecutive edge points and inlier arc selection method. TM tested the proposed method by using simulated and real data, and wrote the manuscript. YS also revised the manuscript written by TM. Both authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Received: 14 August 2015 Accepted: 21 April 2016

Published online: 06 September 2016

References

- Chojnacki W, Brooks MJ, van den Hengel A, Gawley D (2000) On the fitting of surfaces to data with covariances. *IEEE Trans Patt Anal Mach Intell* 22(11):1294–1303
- Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm ACM* 24(6):381–395
- Fitzgibbon A, Pilu M, Fisher RB (1999) Direct least squares fitting of ellipses. *IEEE Trans Patt Anal Mach Intell* 21(5):476–480
- Hartley R, Zisserman A (2004) Multiple view geometry in computer vision. 2nd ed.. Cambridge University Press, Cambridge, U.K
- Kanatani K (1993) Geometric computation for machine vision. Oxford University Press, Oxford, U.K
- Kanatani K, Ohta N (2004) Automatic detection of circular objects by ellipse growing. *Int J Image Graphics* 4(1):35–50
- Kanatani K, Sugaya Y (2007) Performance evaluation of iterative geometric fitting algorithms. *Comput Stat Data Anal* 52(2):1208–1222
- Kanatani K, Rangarajan P (2011) Hyper least squares fitting of circles and ellipses. *Comput Stat Data Anal* 55(6):2197–2208
- Kanatani K, Rangarajan P, Sugaya Y, Niitsuma H (2011) HyperLS and its applications. *IPSJ Trans Comput Vis Appl* 3:80–94
- Kanatani K, Al-Sharadqah A, Chernov N, Sugaya Y (2012) Renormalization returns: hyper-renormalization and its applications. *Proc 12th Euro Conf Comput Vis* 3:385–398
- Leedan Y, Meer P (2000) Heteroscedastic regression in computer vision: problems with bilinear constraint. *Int J Comput Vision* 37(2):127–150
- Masuzaki T, Sugaya Y, Kanatani K (2013) High accuracy ellipse-specific fitting. 6th Pacific-Rim Symposium on Image and Video Technology. Guanajuato, Mexico, p 314–324
- Matei J, Meer P (2006) Estimation of nonlinear errors-in-variables models for computer vision applications. *IEEE Trans Patt Anal Mach Intell* 28(10):1537–1552
- Shao M, Ijiri Y, Hattori K (2015) Grouped outlier removal for robust ellipse fitting. 14th IAPR International Conference on Machine Vision Applications. Tokyo, Japan, p 138–141
- Sugaya Y (2010) Ellipse detection by combining division and model selection based integration of edge points. 4th Pacific-Rim Symposium on Image and Video Technology, Siggapore, p 64–69
- Szpak ZL, Chojnacki W, van den Hengel A (2012) Guaranteed ellipse fitting with Sampson distance. *Proc 12th Euro Conf Comput Vis* 5:87–100
- Taubin G (1991) Estimation of planar curves, surfaces, and non-planar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Trans Patt Anal Mach Intell* 13(11):1115–1138
- Waibel P, Matthes J, Gröll L (2015) Constrained ellipse fitting with center on a line. *J Math Imaging Vis* 53(3):364–382
- Yu J, Kulkarni SR, Poor HV (2012) Robust ellipse and spheroid fitting. *Pattern Recognit Lett* 33(5):492–499
- Yu Q, Ong SH (2007) Arc-based evaluation and detection of ellipses. *J Pattern Recognit* 40(7):1990–2003

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com