ElliFit: An unconstrained, non-iterative, least squares based geometric Ellipse Fitting method

Dilip K. Prasad a,⁎, Maylor K.H. Leung b, Chai Quek a

a School of Computer Engineering, Nanyang Technological University, Singapore 639798, Singapore
b Faculty of Inform. & Comm. Tech., Universiti Tunku Abdul Rahman, Kampar, Malaysia

A R T I C L E   I N F O

Article history:
Received 14 December 2011
Received in revised form 24 July 2012
Accepted 7 November 2012
Available online 16 November 2012

Keywords:
Ellipse fitting
Shape analysis
Unconstrained optimization
Least squares method

A B S T R A C T

A novel ellipse fitting method which is selective for digital and noisy elliptic curves is proposed in this paper. The method aims at fitting an ellipse only when the data points are highly likely belong to an ellipse. This is achieved using the geometric distances of the ellipse from the data points. The proposed method models the non-linear problem of ellipse fitting as a combination of two operators, with one being linear, numerically stable, and easily invertible, while the other being non-linear but unique and easily invertible operator. As a consequence, the proposed ellipse fitting method has several salient properties like unconstrained, stable, non-iterative, and computationally inexpensive. The efficacy of the method is compared against six contemporary and recent algorithms based on the least squares formulation using five experiments of diverse practical challenges, like digitization, incomplete ellipses, and Gaussian noise (up to 30%). Three of the experiments comprise of a total of 44,400 ellipses (positive test data) while the other two are tested on 320,000 non-elliptic conics (negative test data). The results show that the proposed method is quite selective to elliptic shapes only and provides accurate fitting results, indicating potential application in medical, robotics, object detection, and other image processing industrial applications.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

In many image processing applications, shape based analysis plays an important role and it is desirable to detect specific geometrical shapes like polygons, ellipses, etc. Since ellipses are the most common non-linear shapes in natural and man-made objects, detecting ellipses from images is very useful. For example, detecting ellipses from images is desired in application like pupil tracking, biological cell segmentation, astronomical and geological shape segmentation, object detection, etc.

While there are some sophisticated algorithms for detecting ellipses from images, most of these algorithms cannot be used in real time applications. Further, there is generally a least squares method at the core of these algorithms [1–5] and the other pre-processing and post-processing steps are used to increase the selectivity of the algorithms to only elliptic shapes and reduce the false positives [6–11]. For instance, least squares methods typically need a fraction of a second, while extra processing steps used for improving selectivity may require a few seconds [9] to a several minutes [10]. Furthermore, the selectivity of the ellipses can be poor and ellipses may be fit in non-elliptic curves as well. For real time applications such as pupil tracking, where time is a critical parameter, a large number of processing steps are deemed undesirable and generally least squares method has to be used alone. Even for non-real time applications, it is desirable to have a least squares method that is inherently selective and good at reducing the false positives for non-elliptic curves, so that the burden on extra processing can be reduced.

The least squares methods for ellipse fitting currently in use are based on the fundamental work by Rosin [2–5,12] and Fitzgibbon [1]. They both employ the algebraic equation of general conics to define an error minimization problem and additional numeric constraints are introduced in order to restrict the solutions to the elliptic curves. In other works, Rosin [4,5,12,13] developed and tested several error metrics for quantifying the quality of fit. Fitzgibbon [1] solves the constrained minimization problem using generalized eigen value decomposition.

Although Fitzgibbon's method [1] is quite elegant and has several merits, some issues with Fitzgibbon's method [1] have been recognized. For example, Maini [14] recognized that Fitzgibbon's method [1] has the problem of numerical instability, ill-posedness of the scatter matrix, and the singularity of the constraint matrix. Further, if the data points lie exactly on the ellipse, all generalized eigenvalues are zero and do not result in any solution at all. Maini [14] suggests a modification of Fitzgibbon's method [1] which includes the
transformation and scaling of the data points. Next, Maini follows up by an iterative optimization where if Fitzgibbon's algorithm suffers from the zero eigen value problem, some random noise is added to the data points. Maini therefore improves Fitzgibbon's method of fitting ellipses for non-noisy as well as noisy set of elliptic data points. Liu [15] on the other hand aims at using Fitzgibbon's method as an initial guess and applies the gradient descent on one point at a time. As a consequence, the ellipse taken as the initial guess gets optimized for the data points in the sequence they occur. Thus, the final result of Liu is biased towards the last few data points in the sequence. Here, we are referring to only the least squares and gradient descent based elliptic parameter extraction done by Liu in [15] and not to other pre-processing and post-processing steps discussed therein.

Another approach taken by Harker [16] and Halir [17] (Harker extended the work of Halir), where the scatter matrix was partitioned into two smaller interdependent matrices and the constraint matrix was changed to a smaller non-singular constraint matrix. Harker addressed the low eccentricity bias in Fitzgibbon's method and proposed a bias correction technique for fitting ellipses of high eccentricity. Harker's approach indeed improved the method of Fitzgibbon. However, Harker demonstrated poor selectivity for elliptic shapes. The reason for this is the bias correction scheme in which Harker used a linear combination of an elliptic and a hyperbolic solution.

All the methods discussed above employ algebraic equation of the ellipse and the algebraic distance of the points on the ellipse as the cost function. As opposed to these, Ahn's method [18] uses the geometric distances of the data points from the ellipse as the central quantity for fitting the ellipse. Ahn's method uses two nested iterative loops. The outer loop considers the data of pixels as a whole and uses gradient descent approach to optimize the estimated geometric parameters (namely, coordinates of the center, angle made by the major axis with x-axis, and the lengths of semimajor and semiminor axes). The inner loop is executed for each pixel and the estimation of a point on the ellipse that is nearest to the considered pixel is optimized iteratively. Ahn’s method uses geometric parameters as the driving factors of the algorithm in both the loops. Hence, it is not incorrect to say that the central concept of the method is the distance of the pixels from the ellipse. Ahn's method also employs a bias correction technique which considers a linear combination of an elliptic and a hyperbolic solution that is further iteratively optimized. Ahn’s algorithm is expected to perform better than algebraic minimization methods due to the use of geometric distance as the main criterion. However, there are two issues with Ahn's method. Both the issues are related to the optimization process in Ahn’s algorithm. First, Ahn’s method is computationally expensive owing to the iterative optimization for each pixel within the outer loop. Second, the problem of local minima is present because non-linear optimizations are involved in both inner and outer loops. It is seen in the numerical simulations that Ahn’s algorithm has tendency to get stuck in the local minimum during optimization in both the inner and outer loops.

We also consider Chaudhri's method [19] as an ellipse fitting method that uses geometry. We would like to highlight that Chaudhri’s method fits an ellipse to a cluster of points which may be edge pixels or the pixels in the interior of an arbitrary shape. Its aim is to find representative ellipse for a given shape instead of attempting to fit ellipses selectively to elliptic curves. Hence, a very simple model for computing the parameters of representative ellipses is used. The centroid of the cluster of pixels is taken as the center of the representative ellipse and the angle of the major axis is found by determining the orientation of the cluster. The lengths of semimajor and semiminor axes of the representative ellipse are found using the simple geometric model of ellipse obtained after translation and rotation of the coordinate axes. As already highlighted, Chaudhri’s method is inherently non-selective for elliptic curves and hence is not appropriate for ellipse selective applications.

In this paper, we propose a least squares method that does not require constrained optimization because it uses a set of unconventional variables. These are related to the actual parameters of ellipses in a non-linear manner. Thus, the constraints are directly incorporated in the definition of the new variables. Due to this, the least squares method does not need additional constraints or non-linear optimization and still demonstrates high selectivity for elliptic curves. The main idea behind the proposed method is that since this method has to be applied on the digitized images (pixels), we can incorporate the effect of digitization in the development of the least squares formulation. Thus, rather than designing a least squares formulation using a general quadratic equation and satisfying certain constraints, we can use the geometric model of ellipse as the basic model and the distance of the pixels from the fitted ellipse as the criteria for designing the least squares formulation. Since the method is based on a geometry, the proposed method is called Geometric Ellipse Fitting (EllipFit) method. A preliminary work of this method has appeared in [20]. This article provides the mathematical foundations, the concept development, and the details of the proposed geometric Ellipse Fitting (EllipFit) method. In the context of the proposed method, it is of interest to note some important works on implicit fitting of polynomial curves [21–24]. While these works encompassed polynomial curves of upto a few orders, which included ellipses, some important differences from the current work should be highlighted. The focus of the current work is only on the problem of fitting ellipses on edge pixels or noisy cluster of points representing an elliptic boundary while [21–24] discussed a more general framework for polynomial curves. Second, the general motivation of these works was to represent complicated shapes with polynomial curves of necessary order, while the motivation of the current work is to fit the ellipses only on the curves with good elliptic nature. Thus, the focus of the current work is on selectivity or high true positive rate for ellipse fitting. Even so, the work presented in [23] is of interest since it considers one geometric shape at a time, for example an ellipse, and proposes a Bayesian framework for reducing the geometric distance between the data points and the shape that they represent. However, it assumes some knowledge about the noise model which may or may not be available in practice. Further, it is computation intensive. Even so, such an approach might be useful to understand the nature of error propagation of the proposed method and may comprise our future work.

The outline of the paper is as follows. Section 2 introduces the geometric distance used for defining the minimization problem, the initial minimization problem and its modification. Section 3 presents the mathematical model proposed for fitting the ellipses and the two operators used in the model. Section 4 is dedicated to the study of the injectivity of the non-linear operator introduced in section 3. Section 5 discusses the numerical stability of the linear operator introduced in Section 3. Section 6 discusses the computational complexity of the proposed method. Section 7 presents extensive numerical results. Sections 7.1 to 7.3 consider the elliptic data points (positive test data) while Sections 7.4 and 7.5 deal with the non-elliptic data points (negative test data). Section 7.6 presents some examples of real images and the ellipses detected by the proposed method. Section 8 concludes the finding of this research. The source code of EllipFit is available at the the weblink: https://sites.google.com/site/dilipprasad/source-codes/.

2. Geometric distance used for defining the minimization problem

We begin with the geometric equation of a generic ellipse with semi-major axis length a, semi-minor axis length b, angle of
orientation (angle made by the major axis of the ellipse with the x axis) \( \theta_c \) and center \((x_c, y_c)\). This is detailed in Eq. (1).

\[
\frac{(x-x_c)^2 \cos^2(\theta_c) - (y-y_c)^2 \sin^2(\theta_c)}{a^2} + \frac{((x-x_c) \sin \theta_c + (y-y_c) \cos \theta_c)^2}{b^2} = 1
\]

(1)

where, \(a, b, \theta_c, x_c, \) and \(y_c\) satisfy the following constraints:

\[
\begin{align*}
\text{C1:} & \quad a, b \in \mathbb{R}^+ \\
\text{C2:} & \quad b \leq a \\
\text{C3:} & \quad \theta_c \in [0, \pi) \\
\text{C4:} & \quad x_c, y_c \in \mathbb{R} \\
\end{align*}
\]

(2)

Eq. (1) can be simplified as Eqs. (3) and (4).

\[
\begin{align*}
\alpha &= \frac{a(x-x_c)^2 + b(y-y_c)^2 + \gamma[(x-x_c)(y-y_c)]}{a^2b^2} \\
\beta &= \frac{2a^2 \cos^2 \theta_c + b^2 \sin^2 \theta_c}{a^2b^2} \\
\gamma &= \frac{a \cos^2 \theta_c + b \sin^2 \theta_c}{a \cos^2 \theta_c + b \sin^2 \theta_c} \\
\end{align*}
\]

(3)

The slope of the tangent at the point \(P(x, y)\) is given by Eq. (5).

\[
\frac{dy}{dx} = \frac{-2a(x-x_c) + \gamma(y-y_c)}{2b(y-y_c) + \alpha(x-x_c)}
\]

(5)

and consequently, the equation of the tangent at a point \(P(x, y)\) on the ellipse is given by Eqs. (6)–(8).

\[
y = mx + c_i \\
m = \frac{dy}{dx} \bigg|_{P(x, y)} \\
c_i = \frac{2b(y-y_c) + 2a(x-x_c) - \gamma(y-x_c) - \alpha(x-y_c)}{2b(y-y_c) + \alpha(x-x_c)}
\]

(6)

(7)

(8)

Suppose, there is a pixel \(P_i(x_c, y_c)\), whose nearest point on the ellipse is \(P(x, y)\). Then the actual distance between the ellipse and the pixel \(P_i(x_c, y_c)\) is equal to the distance of \(P(x, y)\) from the tangent passing through \(P(x, y)\) and is given by Eq. (9).

\[
d_i = \sqrt{(y-y_c)^2 + (x-x_c)^2}
\]

(9)

For a sequence of pixels \(P_i(x_c, y_c): i = 1 \text{ to } N\), we intend to find the parameters \(a, b, \theta_c, x_c, \) and \(y_c\), such that the square of the above distance is minimized. Mathematically, this minimization is described by Eq. (10)

\[
\min \left( \frac{\left( y'_i - m_i x'_i - c_i \right)^2}{(1+m_i^2)} \right) \text{ subject to } \text{C1-C4}
\]

(10)

where, \(\text{C1-C4} \) are defined in Eq. (2). For the minimization problem above, the minima occurs when \(\partial (d_i^2) / \partial m_i = 0\) and \(\partial^2 (d_i^2) / \partial m_i^2 > 0\). The expressions of \(\partial (d_i^2) / \partial m_i\) and \(\partial^2 (d_i^2) / \partial m_i^2\) are shown in Eqs. (11) and (12).

\[
\frac{\partial (d_i^2)}{\partial m_i} = -2 \frac{\left( y'_i - m_i x'_i - c_i \right) \left( m_i x'_i + y'_i - m_i c_i \right)}{(1+m_i^2)^2}
\]

(11)

\[
\frac{\partial^2 (d_i^2)}{\partial m_i^2} = \frac{2}{(1+m_i^2)^2} \left[ \left( m_i x'_i + y'_i - m_i c_i \right)^2 + \left( y'_i - m_i x'_i - c_i \right) \right] \\
\times (2m_i x'_i - 2m_i^2 c_i + 3m_i x'_i - y'_i + c_i)]
\]

(12)

From the above, we see that \(\left( y'_i - m_i x'_i - c_i \right) = 0\) is the appropriate solution for minimizing \((d_i^2)\). However, since the point \(P(x_c, y_c)\) may not be actually on the ellipse, \(\left( y'_i - m_i x'_i - c_i \right) = 0\) cannot be satisfied. However, we can try to minimize \(|y'_i - m_i x'_i - c_i|\) such that it is as close to zero as possible. Thus, in effect, solving the minimization problem of Eq. (10) can be reformulated as attempting to determine the parameters \(a, b, \theta_c, x_c\), and \(y_c\), such that \(r_i\) in Eq. (13)

\[
r_i = |y'_i - m_i x'_i - c_i|
\]

(13)

is minimized subject to constraints \(\text{C1-C4}\) defined in Eq. (2). Assuming that the pixels \(P_i(x_c, y_c): i = 1 \text{ to } N\) were obtained by digitizing the points on the ellipses, that is only digitization noise is present, the point \(P(x_c, y_c)\) on ellipse nearest to the pixel \(P_i(x_c, y_c)\) satisfies the constraint defined by Eq. (14).

\[
|y'_i - y_c| \leq 0.5; |x'_i - x_c| \leq 0.5
\]

(14)

For the above condition, it can be proven that \(m_i / m_c \rightarrow 1\) and \(c_i / c_c \rightarrow 1\), where \(m_i\) and \(c_i\) are obtained by substitution of \(P_i(x_c, y_c)\) in the place of \(P(x_c, y_c)\) in Eqs. (7) and (8). Thus, \(r_i / r_0 \rightarrow 1\), where:

\[
r'_0 = |y'_0 - m_0 x'_0 - c_0|
\]

(15)

Thus, by minimizing the residue \(r'_0\) described by Eq. (15), we are indirectly minimizing the square of the geometric distance defined by Eq. (9).

3. Mathematical model of fitting

We propose to use the following metric spaces and maps:

\[
\begin{align*}
\mathbf{A} &= \begin{bmatrix} a & b & \theta_c & x_c & y_c \end{bmatrix}^T, \text{ subject to } \text{C1-C4} \\
\mathbf{V} &= \begin{bmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 \end{bmatrix}^T, \mathbf{V} \in \mathbb{R}^5 \\
\mathbf{Y} &= \begin{bmatrix} y'_1 & y'_2 & \cdots & y'_N \end{bmatrix}^T, \mathbf{Y} \in \mathbb{Z}^N \\
\mathbf{K}: \mathbf{A} \rightarrow \mathbf{V}; \mathbf{X}: \mathbf{V} \rightarrow \mathbf{Y}
\end{align*}
\]

(16)

(17)

(18)

(19)

The upright and non-bold letter ‘T’ in the superscript denotes the matrix/vector transpose of the matrices and vectors. The vector \(\mathbf{A}\) is a five dimensional vector that contains the parameters of the ellipse that we attempt to fit. This is the vector that we want to determine using the sequence of pixels \(P_i(x, y_c): i = 1 \text{ to } N\). The vector \(\mathbf{Y}\) contains the \(y\) coordinates of the pixels. We define a new five-dimensional vector \(\mathbf{V}\) containing new real valued variables \(\phi_i\) to \(\phi_5\), which acts as an intermediate stage defined for splitting the non-linear map \(\mathbf{M}: \mathbf{A} \rightarrow \mathbf{V}\). The non-linear map \(\mathbf{M}\) is split into two maps \(\mathbf{K}\) and \(\mathbf{X}\), as defined in Eq. (19). We explicitly design the variables \(\phi_i\) to \(\phi_5\) and the map \(\mathbf{X}\), such that the mapping \(\mathbf{X}: \mathbf{V} \rightarrow \mathbf{Y}\) is linear and can be written as the following matrix Eq. (20).

\[
\mathbf{Y} = \mathbf{X} \cdot \mathbf{V}
\]

(20)

Since \(\mathbf{M}: \mathbf{A} \rightarrow \mathbf{Y}\) is a non-linear map and \(\mathbf{X}: \mathbf{V} \rightarrow \mathbf{Y}\) is designed to be a linear map, hence the map \(\mathbf{K}: \mathbf{A} \rightarrow \mathbf{V}\) is non-linear.

In the subsequent sections, the properties of the operators \(\mathbf{K}\) and \(\mathbf{X}\) will be outlined and it will be shown that \(\mathbf{K}\) is non-linear but is a to-one mapping due to the constraints of \(\mathbf{A}\) and \(\mathbf{V}\). This implies that the parameters \(a, b, \theta_c, x_c\), and \(y_c\) can be obtained uniquely from \(\phi_i\) to \(\phi_5\), if \(\mathbf{V}\) can be determined by employing Eq. (20) to minimize the residue \(r'_0\) given by Eq. (15).

For designing the variables \(\phi_i\) to \(\phi_5\) and the map \(\mathbf{X}\), we define the following residual distance in Eq. (21).

\[
\sum_{i=1}^{N} r'_i = ||\mathbf{Y} - \mathbf{X} \cdot \mathbf{V}||^2
\]

(21)

where, \(\cdot\) denotes the Frobenius norm for matrices and Euclidean norm for vectors. The advantage in using the above definition is
that the Moore Penrose pseudoinverse can be used to determine the unique optimal solution of $\Psi$ in order to minimize the sum of squares of the residues $r_i$, $i = 1$ to $N$ as shown in Eq. (22)

$$\Psi = (X'X)^{-1}X'Y$$ (22)

In order to obtain a form similar to Eq. (21), the variables $\phi_1$ to $\phi_3$ are defined in Eqs. (23)–(27) and the map $X$ is defined in Eq. (28).

$$\phi_1 = x'/\beta$$ (23)

$$\phi_2 = y'/\beta$$ (24)

$$\phi_3 = 2x' + \gamma y' = 2\phi_1 x' + \phi_2 y'$$ (25)

$$\phi_4 = 2\beta x' + \gamma y' = 2\phi_2 x' + \phi_2 y'$$ (26)

$$\phi_5 = \frac{2x'^2 + 2\beta x'y' + \gamma x'y' - \alpha^2 y^2}{\beta} = \phi_1 x'^2 + \phi_2 x'y' - \alpha^2 y^2 / \beta$$ (27)

$$X = \left[ \begin{array}{c} -x_i^2/y_i - x_i/y_i - y_i^{-1} \\ \vdots \end{array} \right]$$ (28)

The Appendix presents the derivations of the expressions for the parameters $\phi_1$ to $\phi_3$ and the map $X$. For computing the actual parameters in $\overline{A}$, we may use the following inverse map from the parameters $\phi_1$ to $\phi_5$ to the parameters $a, b, \theta, c, x_c, y_c$ as shown in Eqs. (29)–(33).

$$a = 2 \frac{\phi_2 \phi_4 - \phi_4^2 \phi_1 - \phi_3^2 + \phi_3 \left( \phi_1^2 - 4 \phi_1 \right)}{\left( \phi_1^2 - 4 \phi_1 \right) \left( 1 + \phi_1 \right) - \sqrt{\phi_1^2 + 4 \phi_1}}$$ (29)

$$b = 2 \frac{\phi_2 \phi_4 - \phi_4^2 \phi_1 - \phi_3^2 + \phi_3 \left( \phi_1^2 - 4 \phi_1 \right)}{\left( \phi_1^2 - 4 \phi_1 \right) \left( 1 + \phi_1 \right) + \sqrt{\phi_1^2 + 4 \phi_1}}$$ (30)

$$\theta_c = -0.5 \tan^{-1} \left( -\phi_2 / (1 - \phi_1) \right)$$ (31)

$$x_c = (\phi_2 \phi_4 - 2 \phi_3) / \left( \phi_2^2 - 4 \phi_1 \right)$$ (32)

$$y_c = (\phi_2 \phi_3 - 2 \phi_1 \phi_4) / \left( \phi_2^2 - 4 \phi_1 \right)$$ (33)

We note that the inverse tangent function used in Eq. (31) is a four quadrant inverse tangent function. The proposed method is presented in algorithmic form below, so that the method can be applied with ease:

Step 1: Form the matrix $X$ using Eq. (28) and the vector $Y$ using Eq. (18).

Step 2: Compute $\Psi$ using Eq. (22).

Step 3: Compute $a, b, \theta, c, x_c, y_c$ and $y_c$ using Eqs. (29)–(33).

4. Injectivity of the map between $\Psi$ and $\overline{A}$, $K : \overline{A} \rightarrow \Psi$ and $K : \Psi \rightarrow \overline{A}$

In this section, we show that the map $K : \overline{A} \rightarrow \Psi$ is injective. This means that for a vector $\overline{A}$ defined by Eq. (16), there is one and only one vector $\Psi$ as computed by the map $K : \overline{A} \rightarrow \Psi$, and for a vector $\Psi$ given by Eq. (17) (subject to the conditions of existence of solution), there is one and only one vector $\overline{A}$ satisfying the constraints C1–C4 as described by Eq. (2).

4.1. Forward mapping $K : \overline{A} \rightarrow \Psi$

Due to the constraint C1 in Eq. (2), $a^2$ and $b^2$ are one to one functions of $a$ and $b$. Owing to this, $\phi_1$ and $\phi_2$ are both one to one functions of $a$ and $b$. Although $\phi_3$ and $\phi_4$ are not individually injective functions of $\theta_c$, when $\theta_c$ is subject to constraint C3, however, as a pair of functions $(\phi_3, \phi_4)$, the pair has a one to one relationship with $\theta_c$, as constrained by C3. This is similar to the fact that for a general angle $\theta \in [0, 2\pi)$, $\sin \theta$ and $\cos \theta$ are individually many to one, but a vector defined as $[\sin \theta \cos \theta]^T$ is a unique vector for any $\theta \in [0, 2\pi)$.

According to Eqs. (25) and (26), since $\phi_3$ and $\phi_4$ are simply linear combinations of $x_c$ and $y_c$, they are also one to one functions of $x_c$ and $y_c$. Further, $\phi_3$ and $\phi_4$ are linear functions of $\phi_1$ and $\phi_2$, which have a one to one relationship with $a, b, \theta, c$. Thus, as a pair, $\phi_3$ and $\phi_4$ also have a one to one relationship with $a, b, \theta, c$. Following the same logic, although $\phi_5$ is a many to one function of $\phi_1, x_c, y_c$, the vector $\Psi$ itself is a one to one map of the vector $\overline{A}$. This means that corresponding to a vector $\overline{A}$ in the five dimensional space, constrained by C1–C4 as described by Eq. (2), there is one and only one $\overline{A}$ in the five dimensional real space.

4.2. Inverse mapping $K : \Psi \rightarrow \overline{A}$

Now, we consider the inverse mapping, i.e., the mapping from $\Psi$ to $\overline{A}$, $K : \Psi \rightarrow \overline{A}$, as specified by Eqs. (29)–(33). Since we use the four quadrant inverse tangent in Eq. (31), $\theta_c$ is a one to one function of $\phi_1$ and $\phi_2$.

It can be shown that for a given $\Psi$, a solution $\overline{A}$ exists if and only if the following conditions of existence are satisfied as expressed by Eq. (34).

$$E1 : \phi_1 > 0$$

$$E2 : \{ \begin{array}{ll}
\text{Either } & \phi_5 > 0 \text{ AND } \phi_2 \phi_3 \phi_4 > \left( \phi_4^2 \phi_1 + \phi_3^2 \right) \\
\text{Or } & \phi_5 < 0 \text{ AND } \phi_2 \phi_3 \phi_4 < \left( \phi_4^2 \phi_1 + \phi_3^2 \right)
\end{array} \}$$ (34)

Now, we show that for a set of values for $\phi_1$ to $\phi_5$, the pair of $(x_c, y_c)$ is a unique pair. The expression $(\phi_5^2 - 4 \phi_4)$ can be written as Eq. (35).

$$\phi_5^2 - 4 \phi_4 = (1 - \phi_1 \sec 2 \theta_c)^2 - (1 + \phi_1)^2$$ (35)

Thus, given the condition of existence of solution E1 and the solution $\theta_c$, the denominator $(\phi_5^2 - 4 \phi_4)$ in Eqs. (32) and (33) is a one to one function of $\phi_1$ and $\theta_c$. Now, considering the numerators, $x_c$ is a many to one function of $\phi_2$ and $\phi_4$ (since $\phi_2 \phi_4 = (1 - \phi_2)(-\phi_4)$). Similarly, $y_c$ is a many to one function of $\phi_1$ and $\phi_4$. However, as a pair, $(x_c, y_c)$ together form one-to-one functions of $\phi_1$ to $\phi_4$.

Now, we consider the uniqueness of $a$ and $b$. Owing to the constraint C1 in Eq. (2), it is sufficient to prove the uniqueness of $a^2$ and $b^2$ for a given set of values of $\phi_1$ to $\phi_3$ that satisfy the existence conditions E1 and E2 in Eq. (34). Using Eqs. (29)–(33) and (35), we write $a^2$ and $b^2$ as Eqs. (36) and (37).

$$a^2 = 2 \frac{\phi_2 x_c + \phi_4 y_c - \phi_5}{(1 + \phi_1 \sec 2 \theta_c) - (1 + \phi_1)}$$ (36)

$$b^2 = 2 \frac{\phi_3 x_c + \phi_4 y_c - \phi_5}{(1 - \phi_1 \sec 2 \theta_c) + (1 + \phi_1)}$$ (37)

Using the arguments presented just after Eq. (35), the denominators in Eqs. (36) and (37) are one to one functions of $\phi_1$ and $\theta_c$. 
Since the pair $(x_i, y_i)$ is one to one function of $\phi_1$ to $\phi_4$, and the numerator in Eqs. (36) and (37) is simply a linear combination of $x_i$ and $y_i$, hence $a^2$ and $b^2$ are also one to one functions of $\phi_1$ to $\phi_5$ for a given set of $x_i$, $y_i$, and $\theta_i$. Thus, the vector $\bar{A}$ is a one to one function of the vector $\bar{B}$, subject to the existence conditions of $E1$ and $E2$, see Eq. (34).

Thus, in the framework of constraints $C1-C4$ and $E1-E2$, the mapping $K: \bar{A} \rightarrow \bar{B}$ and the inverse mapping $K: \bar{B} \rightarrow \bar{A}$ is injective. If any of the existence conditions $E1-E2$ are not satisfied while inverse mapping, either $a$ or $b$ or both will have complex values. This is an easy and direct way to filter off non-elliptic curves. This is the basis of the high selectivity demonstrated by the proposed method.

5. Modification based on the numerical stability of the map $X: \bar{B} \rightarrow \bar{Y}$

In this section, we consider the mathematical properties of the linear map $X: \bar{B} \rightarrow \bar{Y}$. Specifically, we address the properties concerning the computation of the Moore Penrose pseudo inverse of $X$. The Moore Penrose pseudo inverse $X: \bar{Y} \rightarrow \bar{B}$ of $X: \bar{B} \rightarrow \bar{Y}$ is given by Eq. (38).

$$\bar{X} = (X^T \cdot X)^{-1} \cdot X^T$$  \hspace{1cm} (38)

For convenience, we define $G$ in Eq. (39).

$$G = (X^T \cdot X)$$  \hspace{1cm} (39)

Thus, using Eq. (28), $G$ can be written as Eq. (40).

$$G = \begin{bmatrix}
g_1 & h_1 & j_1 & h_2 & g_2 \\
h_1 & g_2 & h_2 & j_2 & h_3 \\
j_1 & h_2 & g_3 & h_3 & j_3 \\
h_2 & j_2 & g_4 & h_4 & j_4 \\
g_3 & h_3 & j_3 & h_4 & g_4
\end{bmatrix}$$  \hspace{1cm} (40)

where:

$$g_1 = \sum_{i=1}^{N} x_i^4; \quad g_2 = \sum_{i=1}^{N} x_i^2; \quad g_3 = \sum_{i=1}^{N} x_i^2; \quad g_4 = \sum_{i=1}^{N} x_i^2; \quad g_5 = \sum_{i=1}^{N} x_i^2\cdot y_i$$

$$h_1 = \sum_{i=1}^{N} x_i^4; \quad h_2 = \sum_{i=1}^{N} x_i^2; \quad h_3 = \sum_{i=1}^{N} x_i^2; \quad h_4 = \sum_{i=1}^{N} x_i^2; \quad h_5 = \sum_{i=1}^{N} x_i^2\cdot y_i$$

$$j_1 = \sum_{i=1}^{N} x_i^4; \quad j_2 = \sum_{i=1}^{N} x_i^2; \quad j_3 = \sum_{i=1}^{N} x_i^2; \quad j_4 = \sum_{i=1}^{N} x_i^2$$  \hspace{1cm} (41)

The numerical stability of the matrix $G$ is integral to the accuracy and stability of the solution $\bar{B}$ and subsequently $\bar{A}$. The condition number of the matrix $G$ is an indication of the numerical stability of its inversion. Thus, we study the condition number of the matrix $G$.

We note that the matrix $G$ is a five-dimensional symmetric matrix containing only 12 distinct entries. Further, the perturbation in $G$ may occur only through the perturbations in the pixels $P_i(x_i,y_i); i = 1$ to $N$, which may be the result of digitization or distortion or other types of noise. Since the perturbations occur only through the pixels, the perturbations in $G$ are also symmetric. Thus, following [25], the Bauer Skekil condition number (infinity norm condition number) of the matrix $G$ is within a finite well defined ratio of the actual condition number of the matrix $G$, and can be considered as a direct indicator of the condition number. This theory allows us to study the condition number as related to the component with the maximum absolute value in the matrix $G$. It is evident from Eq. (41) that $G$ has the largest condition number when one of the pixels has zero $y$ coordinate, that is, $\exists i \in 1$to$N: y_i = 0$. When this happens, the determinant of $G$ becomes infinite, leading to a singular $G^{-1}$.

However, this numerical instability can be easily avoided by modifying $X$ and $\bar{Y}$ as follows in Eqs. (42) and (43).

$$X = \begin{bmatrix}
x_1^2 & -x_1 y_1 & y_1 & y_1 & -1 \\
x_2^2 & -x_2 y_2 & y_2 & y_2 & -1 \\
 \vdots & \vdots & \vdots & \vdots & \vdots \\
x_N^2 & -x_N y_N & y_N & y_N & -1
\end{bmatrix}$$  \hspace{1cm} (42)

$$\bar{Y} = \begin{bmatrix}
y_1^2 & y_2^2 & \cdots & y_N^2
\end{bmatrix}^T, \quad \bar{Y} \in \mathbb{R}^N$$  \hspace{1cm} (43)

As a result of this modification, instead of minimizing the residues $r_i^2$ through the cost function given in Eq. (21). The following cost function defined in Eq. (44) is used.

$$\sum_{i=1}^{N} (y_i^r)^2 = ||Y - X \cdot \bar{Y}||^2$$  \hspace{1cm} (44)

where, $X$ and $\bar{Y}$ are given by Eqs. (42) and (43), respectively. Although the modification improves the numerical stability of the inverse of matrix $G$ as defined in Eq. (39), it has an implication on the minimum number of pixels required for fitting the ellipse. From Eq. (44), if a pixel $P_i$ has zero $y$ coordinate, then $(y_i^r) = 0$, which implies that even though the residue $r_i$ may be non-zero, its contribution to the cost function is zero. Since any ellipse may intersect the $x$ axis at a maximum of two points, using $N \geq 7$ ensures that at least five points (same as the number of unknowns) contributes directly to the cost function.

Following the modifications given in Eqs. (42) and (43), the Bauer condition number of the matrix $G$ defined in Eq. (39) has the strongest contribution from either $\sum_{i=1}^{N} x_i^4$ or $\sum_{i=1}^{N} y_i^2$. We note that due to the modifications in Eqs. (42) and (43), all the components of the matrix $G$ are $N$ times the statistical moments $\psi_{m,n}(x_i,y_i)$ of the bivariate $(x_i,y_i)$ along $(0,0)$, where the $(n,m)$th moment around a point $(\bar{x},\bar{y})$ is given by Eq. (45).

$$\psi_{m,n}(x_i,y_i) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})^n (y_i - \bar{y})^m$$  \hspace{1cm} (45)

The statistical moments are minimum when $(\bar{x},\bar{y})$ are the averages (the first order moments) of the bivariate $(x_i,y_i)$, this is given in Eq. (46).

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i; \quad \bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i$$  \hspace{1cm} (46)

Thus, if the pixel space is translated at a new origin given by Eq. (46), the condition number of the matrix $G$ can be further reduced.

Based on the above arguments, the proposed ellipse fitting algorithm is modified as follows:

Step 1: Compute $\bar{x}_i = x_i - \bar{x}, \bar{y}_i = y_i - \bar{y}$ where $(\bar{x},\bar{y})$ are given by Eq. (46).

Step 2: Form the matrix $X$ and the vector $\bar{Y}$ as given in Eqs. (47) and (48). We highlight that Eqs. (47) and (48) are similar to Eqs. (42) and (43) and $x_i$ and $y_i$ in Eqs. (42) and (43) are replaced by $\bar{x}_i$ and $\bar{y}_i$ in Eqs. (47) and (48).

$$X = \begin{bmatrix}
x_1 - \bar{x}_i & \bar{y}_i & \bar{x}_i & \bar{y}_i & -1
-\bar{x}_i & \bar{y}_i & \bar{x}_i & \bar{y}_i & -1
\vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}$$  \hspace{1cm} (47)

$$\bar{Y} = \begin{bmatrix}
\bar{y}_1^2 & \bar{y}_2^2 & \cdots & \bar{y}_N^2
\end{bmatrix}^T, \quad \bar{Y} \in \mathbb{R}^N$$  \hspace{1cm} (48)

Step 3: Compute $\bar{A}$ using Eq. (22).

Step 4: Compute $a,b,c,d,e,f$, using Eqs. (29)-(33), where $x_i$ and $y_i$ in Eqs. (32) and (33) are replaced by $\bar{x}_i$ and $\bar{y}_i$.

Step 5: Compute $x_c = \bar{x}_i + \bar{x}$ and $y_c = \bar{y}_i + \bar{y}$. 
6. Computational complexity

The computational complexity of the proposed algorithm is determined as follows:

1. The computational complexity of Step 1 is $O(2N)$.
2. Assuming that the elements of matrix $G$ are computed directly, the complexity of forming $G$ is $O(12N)$.
3. The computational complexity of computing $G/C_0$ is $O(53)$.
4. Computation complexity of determining $F$ is $O(30N)$.
5. Computational complexity of determining $A$ is $O(5)$.

The computational complexity is determined mainly by the complexity of computing $F$: $O(30N)$. We see that the proposed method has a computational complexity of $O(N)$ only. Hence it is highly efficient technique to fit an ellipse.

7. Numerical experiments and comparison with other methods

We compare the performance of the proposed method (EllFit) with various methods based on the least squares fitting of ellipses. The methods used for comparison are Fitzgibbon [1], Chaudhri [19], Harker [16], Ahn [18], Maini [14], and Liu [15]. Fitzgibbon [1], Chaudhri [19], and Harker [16] are non-iterative ellipse fitting methods. Chaudhri [19] is basically an ellipse fitting algorithm for any cluster of data. Ahn [18], Maini [14], and Liu [15] are iterative methods. Ahn [18] is the only method based on the geometry of the ellipse. Maini [14] and Liu [15] are based on Fitzgibbon [1] and aimed at improving the performance of ellipse detection for special cases.

7.1. Digital incomplete/complete elliptic curves

We consider a family of elliptic curves given by $b \in [20, 150]$, $a \in [b, 150]$, $x \in (0, \pi)$, $x_c = 150$, and $y_c = 150$. In the range $[\theta_1, \theta_1 + \Delta \theta]$, where $\theta_1$ is randomly selected, a digital curve corresponding to the analytical ellipse is generated. For various values of $\Delta \theta$, 1000 such random curves are generated, corresponding to randomly chosen values of $a, b, x$, and $\theta_1$. For these curves, the ellipses are fitted using the proposed method and the six methods used for comparison. For the fitted ellipses, ellipses that satisfy the following conditions are retained for further considerations:

- The semi-major axis of a fitted ellipse should be less than 200.
- The ratio of the semi-minor to the semi-major axis (aspect ratio) should be more than 0.1 (that is the eccentricity of the ellipse should be less than 0.995).

The bias correction measures used in Harker [16] and Ahn [18] sometimes result in highly elliptic ellipses with huge values of semi-major axis. This unnecessarily tilts the performance metrics in favor of other methods since it can be easily concluded in practice that the ellipses in the images may not be so large (of the

![Fig. 1. Plot of error metrics for Experiment 7.1 (digital incomplete elliptic curves). (a) Plot of $(E13-E13_p)$ for various values of $\Delta \theta$, (b) Plot of $(E14-E14_p)$ for various values of $\Delta \theta$ and (c) Plot of $(d-d_p)$ for various values of $\Delta \theta$.](image-url)
order of several powers of 10). Thus, condition (i) above was included to compensate for this effect. The same bias corrections may alternatively keep the semi-major axis within reasonable value but push the value of the semi-minor axis to be close to zero. Condition (ii) above compensates for this effect.

The following performance parameters are plotted for each of the methods for the various values of $\Delta \theta$:

1. The mean of the error metric $E_{13}$ as proposed by Rosin in [4].
2. The mean of the error metric $E_{14}$ as proposed by Rosin in [12].
3. The mean of the distances $d = \sum_{i=1}^{N} d_i/N$ of the pixels from the fitted ellipse.
4. Total number of detected ellipses $E_{total}$.
5. Recall: $\text{Recall} = \frac{E_{O \geq 0.8}}{E_{total}}$, where $E_{O \geq 0.8}$ is the number of detected ellipses that have overlap ratios $O$ (given by Jaccard index [26]) more than or equal to 0.8 with the corresponding actual ellipses.
6. Precision: $\text{Precision} = \frac{E_{O \geq 0.8}}{E_{total}}$.

In the above, recall and precision are defined in Eqs. (49) and (50) below.

\[
\text{Precision} = \frac{\text{number of true positive elliptic hypotheses}}{\text{total number of elliptic hypotheses}} \quad (49)
\]

\[
\text{Recall} = \frac{\text{number of true positive elliptic hypotheses}}{\text{number of actual ellipses}} \quad (50)
\]

The true positive ellipses in Eqs. (49) and (50) are defined using the overlap ratio $O$ for ellipses as defined in [27]. We note that the first three are error metrics and the lower values of these parameters are indicator of better fitting. On the other hand, the remaining three are related to the ellipse detection characteristics. The results are plotted in Fig. 1 (error metrics) and Fig. 2 (ellipse detection characteristics). For a better representation of the data in Fig. 1, we plotted the parameters $E_{13}$, $E_{14}$, and $d$ with respect to the proposed data, i.e., we plot $(E_{13} - E_{13p})$, $(E_{14} - E_{14p})$, and $(d - dp)$, where the subscript $P$ denotes the proposed method. The values of $E_{13p}$, $E_{14p}$, and $dp$ are listed in the parts (a), (b), (c) of Fig. 1, respectively. The results in Fig. 1 show that the proposed method has the lowest values of the error metrics $E_{13}$, $E_{14}$, and $d$ for all the values of $\Delta \theta$, while Ahn [18] and Harker [16] are the closest competitors.

Fig. 2 shows the total detected ellipses (Fig. 2(a)), recall (Fig. 2(b)), and precision (Fig. 2(c)) for various values of $\Delta \theta$. It is seen that the proposed method has successfully detected all the ellipses for all values of $\Delta \theta$ except for $\Delta \theta = 135^\circ$, for which the proposed method detected 982 ellipses out of 1000. While Chaudhri [19], Maini [14], and Liu [15] detected close to 1000 ellipses for each value of $\Delta \theta$, their recall and precision values are very poor. This is understandable because while Chaudhri [19] is an ellipse fitting algorithm that fits ellipses on any given cluster of points, it is not selective for elliptic shapes.

On the other hand, Liu [15] and Maini [14] are marginal improvements of Fitzgibbon [1] which in most cases perform very similar to Fitzgibbon. Although Harker [16] detected close to 1000 ellipses for each value of $\Delta \theta$, it has poor recall and precision as compared to the proposed method and Ahn [18]. Although Ahn [18] has good precision (very close to the proposed method and the maximum value 1), it performed poorer than the proposed method for recall and significantly poorer than the proposed method in terms of total detections. The total number of detected ellipses is low for Ahn [18] because it uses two nested non-linear iterative optimization processes and it is easy for Ahn [18] to either misconverge to a local minimum or to become non-convergent. On the other hand, the use of non-linear iterative optimization

---

**Fig. 2.** Ellipse detection characteristics for Experiment 7.1 (digital incomplete elliptic curves). (a) Plot of total number of detected ellipses for various values of $\Delta \theta$, (b) Plot of recall for various values of $\Delta \theta$ and (c) Plot of precision for various values of $\Delta \theta$. 

---
based on geometric principles helped Ahn [18] to be very precise if it converges to the global minimum.

For illustration, we present an example of a digitized ellipse with partial data corresponding to $\Delta \theta = 140^\circ$ in Fig. 3. We note that Ahn [18] does not detect the ellipse due to misconvergence, Fitzgibbon [1] and Liu [15] does not generate an ellipse that satisfies the conditions (i) and (ii) in Section 7.1. Maini [14] fits an ellipse that is quite different from the actual ellipse, Harker [16] generates an ellipse similar (but not exactly matching) to the actual ellipse, Chaudhri [19] fitted an inaccurate ellipse. Only the proposed method managed to successfully detect the ellipse that is very close to the actual ellipse from the partial digital ellipse data given as its input.

The results demonstrate good applicability of the proposed method for detecting elliptic shapes from digital images. Thus, the proposed method should be effective for applications that require detection of elliptic shapes from digital images. Since digital images are ubiquitous in today’s world and several natural and man-made structures are elliptic, a wide range of applications may benefit from the proposed method.

7.2. Noisy cluster of points around an ellipse

We consider a family of ellipses given by $b \in [20,150]$, $a \in [b,150]$, $x \in [0,\pi]$, $x_c = 150$, and $y_c = 150$. We first computed the points on the ellipse corresponding to various values of $\theta$ at an interval of 10 degrees. Let the set of points be denoted as $\{P\}$. Subsequently, we add zero mean Gaussian noise to the value of the coordinates, such that the standard deviations of the noise for the $x$ and $y$ coordinates are $\sigma_x = \kappa \max \{|x-x_c|/100\}$ and $\sigma_y = \kappa \max \{|y-y_c|/100\}$, respectively, where $\kappa$ is the noise percentage. We computed the same error metrics as in Section 7.1 for the various values of $\kappa$. The results are plotted in Fig. 4 (error metrics) and Fig. 5 (ellipse detection characteristics).

![Fig. 3. An example of digital incomplete elliptic curve (Experiment 7.1) and the result of various algorithms. The actual ellipse is shown using thin (black) curve while the fitted ellipses are shown in thick (red) curves. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)](image)

![Fig. 4. Plot of error metrics for Experiment 7.2 (noisy cluster of points around an ellipse). (a) Plot of ($E_{13,13P}$) for various values of noise percent $\kappa$, (b) Plot of ($E_{14,14P}$) for various values of noise percent $\kappa$, and (c) Plot of ($d_d-d_P$) for various values of noise percent $\kappa$.](image)
data points. Here, we show the results for up to 20% noise percentage.

From Fig. 4, it is evident that Ahn [18] and the proposed method have the lowest values of E13 and $d$. For high noise levels ($\kappa > 14$), Ahn [18] performs slightly better than the proposed method for $d$. However, the proposed method clearly outperformed against all the other methods with respect to the parameter E14. For the current experiment, Chaudhri [19] demonstrated the third best performance. This is as expected because Chaudhri [19] benefitted in this experiment by the uniform availability of data points across the complete curvature of the ellipse.

The ellipse detection characteristics are shown in Fig. 5. It is seen that the proposed method detected the maximum number of ellipses and the values of the recall and precision for the proposed method are close to one. In comparison, Ahn [18] has slightly poorer recall and better or same precision as the proposed method.

An example is illustrated in Fig. 6. The data points are quite deviated from the actual ellipse as seen in Fig. 6. It is seen that the proposed method and Ahn [18] have detected ellipses that are quite similar and have good overlap with the actual ellipse. Chaudhri [19] also detected an ellipse that has good overlap with the actual ellipse, but the detected ellipse has lower eccentricity as compared to the actual ellipse. Harker [16] and Maini [14] detected similar but smaller ellipses than the actual ellipse. Fitzgibbon [1] detects a similar but larger ellipse and Liu [15] misconverges to an inaccurate ellipse with different orientation.

We make a note that for the noise percent $\kappa = 18, 20$, Ahn [18] misconverges for a few ellipses, resulting in very high value of $(d - d_0)/d_0$ (close to 50) and $(E_{14} - E_{14f})/E_{14}$ (close to 25).

For the ellipse detection characteristics shown in Fig. 8, we note that the proposed method detected the maximum number of ellipses. Though Ahn [18] detected a lesser number of ellipses, its recall and precision is very close to the proposed method. For higher values of noise, if Ahn [18] detects ellipses, they are detected with greater precision and recall as compared to the proposed method. This is as expected because the proposed method employs a single step of linear least squares retrieval of parameters, Ahn [18] recoursed to several iterations in which it gets the chance to correct and improve on the fitting.
Fig. 7. Plot of error metrics for Experiment 7.2 (noisy cluster of points around an ellipse) for data from 270° angular sector. (a) Plot of \((E_{13} - E_{13P})\) for various values of noise percent \(\kappa\), (b) Plot of \((E_{14} - E_{14P})\) for various values of noise percent \(\kappa\) and (c) Plot of \((d - d_P)\) for various values of noise percent \(\kappa\).

Fig. 8. Ellipse detection characteristics for Experiment 7.2 (noisy cluster of points around an ellipse for data from 270° angular sector). (a) Plot of total number of detected ellipses for various values of \(\kappa\), (b) Plot of recall for various values of \(\kappa\) and (c) Plot of precision for various values of \(\kappa\).
An example with 270° angular sector data and $\kappa=20$ is illustrated in Fig. 9. It is noted that the ellipse fitted by the proposed method is closest to the actual ellipse, followed by Ahn [18], Fitzgibbon [1] fitted to a very small ellipse, Chaudhri [19], Harker [16], Maini [14], and Liu [15] detected the ellipse with greater overlap. Furthermore, these are not close to the actual ellipse.

Noise is often present in most practical scenarios. The effect is more prominently seen in astronomical and medical data where the images often have clusters of points that are close to elliptic shapes but with high variance. Further, such applications are quite critical to the accurate detection of elliptic patterns. Robust performance of the proposed method makes it a good candidate for such applications involving significant amount of noise.

7.3. Multiple incomplete ellipses within an image

In this section, we consider a more challenging scenario in which an image may contain several digital ellipses and these...
may or may not be complete. To generate such images, the setup
described as follows is used. To generate the images, consider an
image size of $300 \times 300$ pixels and randomly generate $N_E \in \{4, 8, 12, 16, 20, 24\}$ ellipses within the region of the image. The parameters
of the ellipses are generated randomly as follows: the center points
of the ellipses are arbitrarily located within the image, the lengths of
semi-major and semi-minor axes are randomly assigned values in the
range $[10, \frac{300}{\sqrt{2}}]$, and the orientations of the ellipses are also
chosen randomly. The only constraint is that each ellipse must be
completely contained in the image and overlap with at least one
ellipse. For each value of $N_E$, we generate 100 images containing edge
curves of the incomplete ellipses. From the edge map of the image,
continuous edge curves with smooth edges (removal of sharp
changes in curvature and inflexion points) are obtained. Each edge
curve is given as one input to the ellipse detection algorithm. Thus,
the number of edge curves may be larger than the number of ellipses
in the image and some of the edge curves may be very small. Some
example images and the edge curves used to test the ellipse detection
are shown in Fig. 10.

As evident from Fig. 10, the scenario considered in this
section is closer to real images in which multiple elliptic objects
may be present simultaneously and possibly occlude each other.
Further, in comparison with Sections 7.1 and 7.2, this section
allows simultaneous visualization and assessment of a variety

![Fig. 12. Ellipse detection characteristics for the experiment in Section 7.3. In figure (a), the black line denotes the actual number of ellipses corresponding to different values of $N_E$. (a) Total number of detected ellipses $E_{\text{total}}$, (b) Recall and (c) Precision.](image)

![Fig. 13. Examples of multiple ellipses in the same image region and detection of ellipses by various methods (Section 7.3).](image)
of scenarios—very small fragments to large fragments, large ellipses to small ellipses, ellipses with high and low eccentricity, etc, which is also encountered in real world scenario.

It is highlighted that none of the methods, including the proposed method, group the incomplete edges potentially belonging to the same ellipse. Each method, including the proposed method, has exactly the same input—the coordinates of the pixels corresponding to one edge curve only, i.e., one continuous curve represented by a single color in Figs. 10 and 13 (second column). Thus, in one execution, the methods are in fact unaware of the existence of other partial curves.

Since there are several ellipses and edge curves in each image, the standard deviation of the error metrics is also considered. Thus, for this experiment, we plot the mean values of the error metrics $E_{13}$, $E_{14}$, and $d$ (Fig. 11(a)–(c)), as well as their standard deviations Fig. 11(d)–(f) in Fig. 11.

It is evident that the proposed method outperformed the rest of the methods in terms of $E_{13}$ and $d$, while it performed similar to or better than most methods in terms of $E_{14}$. It is notable that Ahn [18] has significantly high mean values for all three error metrics. This is because the misconvergence for one of the edge curves in the image results in very high mean value of the error metrics although the error metrics may have small value for the other edge curves. This is also evident from the standard deviation data plotted in Fig. 11(d)–(f), where it is seen that Ahn [18] has very high value for standard deviation as well. Similar effect happens with less frequency for Harker [16] as well, occasionally resulting in high mean and standard deviations (for example, see the data of Harker [16] for $N_e=12$ and $N_e=24$ in Fig. 11(b), (e)).

The ellipse detection characteristics are plotted in Fig. 12. The number of ellipses detected by the proposed method is always close to the actual number of ellipses. The recall of the proposed method is the highest among all the methods and is close to one for all values of $N_e$. However, the precision of the proposed method is slightly poorer than the Ahn [18]. This is because the total number of ellipses detected by the proposed method is slightly higher than the actual number of ellipses for all values of $N_e$ as the number of input curves to the methods might be greater than or equal to the total number of ellipses. Ahn [18] has a slightly poor recall ratio since the number of ellipses detected by Ahn [18] is less than the number of actual ellipses. However, Ahn’s method detected the ellipses with slightly better precision than the proposed method. After the proposed method and Ahn’s method, the next best performance is demonstrated by Harker [16]. We provide some examples in Fig. 13 to illustrate the ellipse detection results in actual images.

![Fig. 14](image)

**Fig. 14.** Performance of ellipse fitting methods for analytical non-elliptic conics of Section 7.4 (the curves in black thin line denote the actual conics). (a) total number of detected ellipses and (b) an example and detected ellipses.

![Fig. 15](image)

**Fig. 15.** Performance of ellipse fitting methods for digital non-elliptic conics of Section 7.4 (the curves in black thin line denote the actual conics). (a) total number of detected ellipses and (b) an example and detected ellipses.
The scenarios of multiple occluded elliptic shapes are often encountered in biological cell segmentation problems [28] and object detection problems [29]. Most ellipse detection methods for practical applications, including Bai [28] and Chia [29], use Fitzgibbon’s method [1] as the ellipse fitting approach. The results in this section demonstrate that such applications will be substantially benefitted by the proposed method.

7.4. Non-elliptic conics—Analytical and digital

In this section, we consider conics that are non-elliptic and inspect if the ellipse fitting methods are able to detect ellipses in non-elliptic data. It is desirable in several applications that the ellipses are fitted only for elliptic data and the non-elliptic data is not fitted falsely to an ellipse. This characteristic of selectivity of the ellipse fitting methods to elliptic data is studied in this experiment.

Consider the mathematical model of conics given in Eq. (51).

$$\begin{align*}
    x &= \frac{l \cos \theta}{1 - \varepsilon \cos \theta} + x_0; \\
    y &= \frac{l \sin \theta}{1 - \varepsilon \cos \theta} + y_0
\end{align*}$$

(51)

where \( l \) is the semi-latus rectum of the conic and \( \varepsilon \) is the eccentricity of the conic. The following family of conics given by \( x_0=150, y_0=150, l \in [20,150], \varepsilon \in [1,2] \) (i.e., parabolae and hyperbolae, but no ellipses) are considered in this analysis. A conic is randomly chosen from this family and is used to generate the corresponding curves using \( \theta \in [180-\Delta \theta, 180 + \Delta \theta] \) in the image region \( 300 \times 300 \) pixels. 1000 such images for each value of \( \Delta \theta \) are generated, where \( \Delta \theta \) is chosen from 90° to 180° at an interval of 10°. These digital curves are tested by the ellipse fitting methods.

For this experiment, the number of ellipses detected for each value of \( \Delta \theta \) is considered. Ideally no non-elliptic conic should be detected as elliptic by the ellipse fitting methods. The total numbers of detected ellipses for various values of \( \Delta \theta \) are plotted in Fig. 14(a). It is noted that the proposed method detected no curves as elliptic, while Ahn [18] detected only a few ellipses, and the remaining methods performed poorly by detecting numerous ellipses with false positive rate ranging from \( \sim 60\% \) to 100\% (about 600 to 1000 ellipses out of 1000, as shown in Fig. 14(a)). This can be attributed to the fact that the proposed method and Ahn’s method, both use geometric models for the least squares fitting of ellipses, hence would have higher selectivity towards the elliptic curves.

The digital curves for the same family of conics are considered. The experimental setup is the same as based on Eq. (51) except that the curves are now digitized. The total number of detected ellipses for the digital non-elliptic conics is plotted in Fig. 15(a). The proposed method and Ahn [18] are the only methods detecting no ellipses for all the curves, while the others detected numerous ellipses with a false positive rate ranging from \( \sim 65\% \) to 100\%. This is in agreement with the discussion presented in the previous paragraph.

Two examples, one for the analytical non-elliptic conic and the other for the digital non-elliptic conics, are presented in Figs. 14(b) and 15(b), respectively. Ahn’s method and the proposed method do not detect the ellipses while the remaining methods fitted ellipses on the non-elliptic analytical and digital curves.

7.5. Non-elliptic noisy conics

In this section, the case of non-elliptic noisy conics is considered. For the mathematical model of conics in Eq. (51) and the family of non-elliptic conics as described in Section 7.4, the data points on the conics for \( \theta \in [180-\Delta \theta, 180 + \Delta \theta] \) are generated. The data points are restricted in the image region of \( 300 \times 300 \) pixels. Let the set of points be denoted as \( P(x,y) \). Zero mean Gaussian noise is added to the value of coordinates, such that the standard deviations of the noise for \( x \) and \( y \) coordinates are \( \sigma_x = k \max(|x-x_c|)/100 \) and \( \sigma_y = k \max(|y-y_c|)/100 \), respectively, where \( k \in [1,30] \); \( k \) is a natural; is the noise percentage. 1000 such images for each value of \( k \) and \( \Delta \theta \) were generated, where \( \Delta \theta \) is chosen from 90° to 180° at an interval of 10°. Thus, for each value of \( k \), there are 10,000 images.

The total number of detected ellipses for each value of \( k \) is plotted in Fig. 16. Even in the case of very high noise \( k = 30 \), the proposed method has a low false positive rate of 3.61%, that is only a 361 ellipses were detected out of 10,000 images from the conic family.

The proposed method has the least number of detected ellipses for every value of \( k \). Ahn [18] performed the second best, though the number of ellipses detected by Ahn’s method rises rapidly as the amount of noise increases. In order to study the impact of the availability of curvature, the total number of detected ellipses for three specific values of \( \Delta \theta \), i.e., \( \Delta \theta = 90°, 135°, 180° \) in Fig. 17(a)(b)(c) are analysed, respectively.

For \( \Delta \theta = 90° \), the proposed method did not detect any ellipse while Ahn [18] detected a few ellipses for high level of noise \( \kappa \geq 25 \).
Even though the numbers of ellipses detected by the proposed method increase when the value of $D_y$ is increased, the numbers are significantly less as compared against other methods. Some examples for illustration of the detected ellipses by various methods are shown in Fig. 18. In Fig. 18(a) and (c), only Ahn [18] and proposed method did not detect the non-elliptic conic as ellipse. Fig. 18(b) is more challenging because the cluster of data points are quite similar to an ellipse. In this case, even Ahn [18] fitted an ellipse and the proposed method is the only method that did not detect the non-elliptic conic as ellipse.

The results in Sections 7.4 and 7.5 demonstrate that the proposed method shows good selectivity of elliptic shapes and low false positive rates for non-elliptic shapes. This is a greatly desirable property for medical and robotic applications that require ellipse detection with low false positive rates and decision making is critically dependent on the correct detection of elliptic shapes.

7.6. Real examples

This section presents some examples of real images. The images are taken from Caltech 256 dataset [30]. The following pre-processing is done on the images:

Step 1: Image is converted to gray scale.
Step 2: Adaptive histogram equalization is applied on the gray scale image.
Step 3: Range filtering is applied on the histogram equalized image.
Step 4: Edgemap is extracted from the filtered image using Canny edge detector with lower and upper thresholds of 0.1 and 0.2 (this is according to [9]).
Step 5: Connected edge contours are obtained using Kovesi’s codes [31]. The curves are approximated as polygons using RDP-mod method presented in [32].
Step 6: Edge curvature corrections are done (splitting edge curves at sharp turns and inflexion points, according to Section 2 of [27]).

Standard Matlab routines are used for steps 1–4. Then the edge curves are passed individually as the inputs to the proposed method. The original images, the edge curves passed as inputs, and the ellipses detected by the proposed method are shown in Fig. 19. A close look at the second column in Fig. 19 reveals that the edge curves corresponding to several ellipses are incomplete and represent only a small portion of the ellipse. In addition, several edge curves are afflicted by noise and hence appear quibbled. Nevertheless, it is seen that the proposed method performs well for a variety of images. Application of simple filtering techniques on the result of the proposed method such as used in [11,27,28] shall decrease the false positive rate. The true positive rate can also be enhanced using grouping schemes for grouping edges potentially belonging to the same ellipses [9,11,27].

8. Conclusion

This paper presents a novel method of ellipse fitting based on the geometric distance between a data point and an ellipse. The proposed method splits the mathematical problem of ellipse fitting into two operators such that the overall algorithm is non-iterative, does not involve constrained optimization, and is numerically stable. Since the model is based upon the geometric distance and not the algebraic equation of ellipse, the selectivity of the method for a set of elliptic data points is higher than most contemporary methods even if the data points are quite noisy (up to 20% Gaussian noise for positive test data and up to 30% Gaussian noise for the negative test data). The proposed method shows superior performance in terms of several performance parameters like E13 (proposed by Rosin in [4]) and E14 (proposed by Rosin in [12]), mean distance of the set of data points from the fitted ellipse, total number of detections, recall, and precision of fitted ellipses. Empirically, it is noted that only Ahn [18] is the closest technique, which is expected since Ahn’s iterative non-linear optimization method suffers from the problem of local minima and high time complexity. Thus, the proposed method has a significant advantage over Ahn [18]. In the next step of this research, we intend to apply it for practical real time ellipse detection problems of pupil tracking, biological cell detection, and robotic applications, although this algorithm should find application in several image processing applications like object detection.

Fig. 18. Examples of noisy non-elliptic conics of Section 7.5 and the ellipses detected by various methods (the curves in black thin line denote the actual conics). (a) $D_y=90\degree, \kappa=30$, (b) $D_y=135\degree, \kappa=30$ and (c) $D_y=180\degree, \kappa=30$. 
astronomical and geological data analysis. It might also be of interest to analyze the error of the proposed method using a known noise model in a manner similar to [23].

**Appendix**

This section provides the derivation that serves as the motivation to choose the variables $\phi_1$ to $\phi_5$. Using Eq. (1) and the definitions in Eq. (4), the equation of a general ellipse is given by Eq. (3). Eq. (3) can be expanded as shown in Eq. (52) below.

$$ax^2+2bxy+cy^2+\gamma xy=\alpha^2b^2-\gamma y^2-\gamma xy$$  \hspace{1cm} (52)

Eq. (52) can be rewritten as Eq. (53) after applying some algebraic manipulations.

$$\frac{\alpha^2}{\beta^2}+\frac{\gamma}{\beta}y\left(\frac{(2\gamma x+\gamma y)}{\beta}\right)x-\frac{(2\beta y_0-\gamma k)}{\beta}y+\left(\frac{(x_0^2+\beta y_0^2+\gamma xy_0-a^2b^2)}{\beta}\right)$$  \hspace{1cm} (53)

---

Fig. 19. Example of real images and the ellipses detected by the proposed method.
Eq. (53) can be written in a matrix formulation after dividing both the sides by \((-y\)) as shown in Eq. (54) below.
\[
\begin{bmatrix}
\frac{x}{y}
\end{bmatrix}
= \begin{bmatrix}
\frac{a/b}{\gamma/b}
\end{bmatrix}
\begin{bmatrix}
\left(2x_0 + \gamma y_0\right)/\beta \\
\left(2\beta y_0 - \gamma x_0\right)/\beta \\
\gamma x_0 - \alpha^2 y_0^2/\beta
\end{bmatrix}
\end{bmatrix} = \begin{bmatrix} y \end{bmatrix}
\] (54)

As seen from Eq. (54), the definitions of the variables \(\phi_1\) to \(\phi_5\) should be given by Eqs. (23–27) and the matrix operator \(X\) should be given by (28).

References

Dilip Kumar Prasad received the B.Tech degree in Computer Science and Engineering from Indian School of Mines, Dhanbad, India in 2003. He did his PhD in Computer Engineering from Nanyang Technological University, Singapore. Currently, he is a research fellow at National University of Singapore, Singapore. He has 5 years of industrial experience with IBM, Mediatek, and Philips in the field of embedded systems. He has published over 25 internationally peer-reviewed research articles. His current research interests include image processing, pattern recognition and discrete geometry.

Dr. Maylor K.H. Leung received the BSc degree in physics from the National Taiwan University in 1979, and the BSc, MSc and PhD degrees in computer science from the University of Saskatchewan, Canada, in 1983, 1985 and 1992, respectively. Currently, Dr. Leung is a Professor with Faculty of Inform. & Comm. Tech., Universiti Tunku Abdul Rahman (Kampar), Malaysia. His research interest is in the area of Computer Vision, Pattern Recognition and Image Processing.

Dr. Chai Quek is currently in the School of Computer Engineering, Nanyang Technological University, Singapore since 1990. He received his Bachelor degree in Science and Ph.D. degrees from the Heriot-Watt University (Edinburgh). His research interests include Neurocognitive informatics, Biomedical Engineering and Computational Finance. He has done significant research work in his research areas and published over 200 top quality international conference and journal papers. He has been often invited as a program committee member and reviewer for a number of premier conferences and journals, including IEEE TNN, TCV etc. Dr. Quek is a senior member of IEEE. He is also a member of the IEEE Technical Committee on Computational Finance and Economics. He has constantly and successfully groomed several high caliber researchers who are awarded prestigious Singapore Millennium Foundation Scholarship and Fellowship, Lee Kuan Yew Fellowship and A*Star Scholarship.