Characteristic Mapping for Ellipse Detection Acceleration

Qi Jia, Xin Fan, *Senior Member, IEEE*, Yang Yang, Xuxu Liu, Zhongxuan Luo, Qian Wang, Xinchen Zhou, and Longin Jan Latecki, *Senior Member, IEEE*

***Abstract*— It is challenging to characterize the intrinsic geometry of high-degree algebraic curves with lower-degree algebraic curves. The reduction in the curve's degree implies lower computation costs, which is crucial for various practical computer vision systems. In this paper, we develop a characteristic mapping (CM) to recursively degenerate 3n points on a planar curve of *n*th order to 3(n − 1) points on a curve of (n − 1)th order. The proposed characteristic mapping enables curve grouping on a line, a curve of the lowest order, that preserves the intrinsic geometric properties of a higher-order curve (ellipse). We prove a necessary condition and derive an efficient arc grouping module that finds valid elliptical arc segments by determining whether the mapped three points are colinear, invoking minimal computation. We embed the module into two latest arc-based ellipse detection methods, which reduces their running time by 25% and 50% on average over five widely used data sets. This yields faster detection than the state-of-the-art algorithms while keeping their precision comparable or even higher. Two CM embedded methods also significantly surpass a deep learning method on all evaluation metrics.**

***Index Terms*— Fast ellipse detection, order reduction, arc segment pruning.**

I. Introduction

BOTH natural and man-made curves contain intrinsic geometry. It is crucial to keep the geometric relations when mapping complex high-degree curves to lower-degree curves. The mapping implies simplified representation and lower computation costs, which is widely used in cartoon design, text spotting [1], and efficient object detection [2] in computer vision.

The representation of curves and surfaces by parameterized polynomials is one of the most commonly known curve degree reduction methods. A given space curve or Bézier curve can be subdivided and approximated by a number of lower degree curves [3], [4]. The parameterized representation is used to simplify expression and calculation in computer-aided geometric design (CAGD) and computer-aided design (CAD) [5], [6]. However, in computer vision, natural images lack explicit parameters of curves as priors. Hence, we need to detect and preserve the geometric nature while mapping them to lower-degree curves.

In computer vision, the geometric nature of an object can highly promote the accuracy and efficiency of detecting the object. Fan et al. detect accurate facial landmarks by developing the geometric relation of facial landmarks by a projective invariance [7]. Sun et al. detect 3D objects via the guidance of object shape prior [8]. Ellipse detection is such a fundamental task in computer vision that it finds wide applications in various practical scenarios including camera calibration [9], unmanned aerial vehicle (UAV) landing [10], and robotic manipulation [11]. These applications demand not only detecting rough locations of elliptical objects, but also accurately measuring the five parameters of every ellipse (center coordinates, semi-major and semi-minor axes, and rotation angle) in a natural scene. Moreover, fast detection running on limited resources is crucial for practical systems as they have to spare resources for other real-time actions such as disparity calculation, landing control, and object grasping. Therefore, accurate and efficient ellipse detection still remains challenging even when the community has been witnessing the great success of deep neural networks in generic object detection [12]. Among all the methods, arc-based detectors output the fastest detection for real-world natural images [2]. They link edge points into arc segments and then fit these arcs to ellipse parameters. Their performance highly depends on the quality of grouping arc segments belonging to one common ellipse. Incorporating geometric properties of ellipses can facilitate this grouping process so as to obtain high-quality detection results. However, there is still room to improve, because the arc-grouping step is generally a bottleneck that cannot achieve real-time detection.

This study investigates the intrinsic geometry of planar curves that plays a critical role in both Hough transformation (HT) and arc-based methods to achieve accurate and fast detection. Inspired by the order reduction of Béizer curves in the field of CAGD and CAD [5], [6], we develop a characteristic mapping (CM) to perform arc grouping on a line, a curve of the lowest order, that preserves intrinsic geometric properties of a higher order curve (ellipse). Thus, this key

Manuscript received 28 October 2021; revised 31 May 2022 and 18 October 2022; accepted 29 March 2023. Date of publication 24 April 2023; date of current version 4 May 2023. This work was supported in part by the Natural Science Foundation of China under Grant 62272083 and Grant 61876030, in part by the Liaoning Provincial Natural Science Foundation under Grant 2022-MS-128, and in part by the U.S. National Science Foundation under Grant IIS-1814745. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Guo-Jun Qi. *(Corresponding author: Xin Fan.)*

Qi Jia, Xin Fan, Yang Yang, Xuxu Liu, and Zhongxuan Luo are with the International School of Information Science and Engineering, Dalian University of Technology, Dalian 116620, China (e-mail: xin.fan@dlut.edu.cn).

Qian Wang is with the School of Mathematics, Liaoning Normal University, Dalian 116029, China.

Xinchen Zhou is with the School of Mathematical Sciences, Jiangsu University, Zhenjiang 212013, China.

Longin Jan Latecki is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122 USA.

This article has supplementary downloadable material available at https://doi.org/10.1109/TIP.2023.3268563, provided by the authors.

Digital Object Identifier 10.1109/TIP.2023.3268563

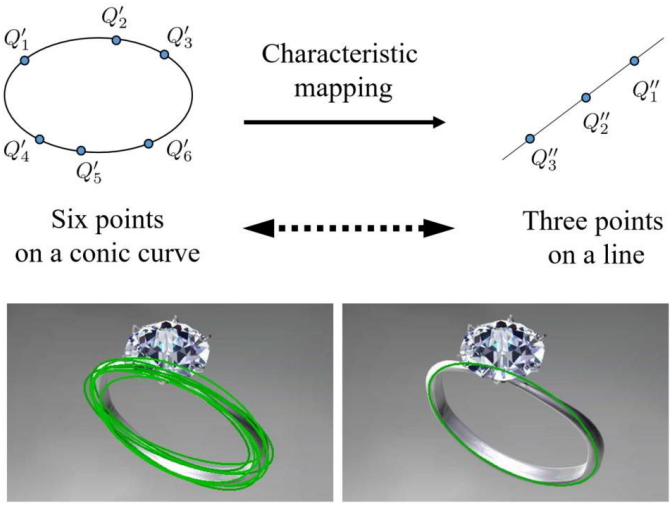


Fig. 1. The upper image sketches the key idea of Characteristic Mapping (CM), which degenerates six points on an ellipse (curve of the second degree) to three points on a line. The figure below demonstrates a pair of ellipse detection instances without (left) and with (right) CM constraints, rendering less execution time and fewer false positive detections.

step for arc-based algorithms invokes minimum calculations to validate arc segments for fitting, resulting in extremely efficient detection. Additionally, we are able to substitute the arc grouping module of other methods boosting their performance. Figure 1 sketches the key idea of mapping six points on an ellipse to three collinear points and its application to ellipse detection, which removes false positive detection and reduces execution time. Our main contributions are as follows:

- We develop a characteristic mapping (CM) upon the characteristic number [13] that recursively degenerate $3n$ points on a planar curve of nth order to $3(n-1)$ points on a curve of $(n-1)$th order, and prove the theorem that the $3(n-1)$ points lying on the lower order curve are necessary for the $3n$ points on the higher order curve. This mapping along with the theorem paves the possibility to detect curves with lower computation costs.
- We apply the characteristic mapping to the simplest case of a conic curve (second order) mapped to a line segment, and develop an efficient arc grouping module that finds valid elliptical arc segments by determining whether the mapped three points lie on one line.
- We embed this module into two latest arc-based ellipse detection methods and accelerate them to produce state-of-the-art performance. Experiments on five real-world data sets demonstrate that our computation module derived from solid geometry reduces the running time by 25% and 50% on average of two latest algorithms [2] and [14] with comparable or higher accuracy.

II. Related Work

This paper brings the theory that connects a mapping of a higher-order curve to a lower-order one and applies the theory to practical ellipse fitting and detection. Hence, this section reviews previous works related to three aspects.

A. Curve Order Reduction

The order/degree reduction problem has a very important application in CAGD and CAD to approximate complex curves and surfaces [6] with simpler ones. The representation of curves and surfaces by parameterized polynomials is one of the most commonly known methods. Rababah et al. demonstrated an approximation of space curves by a piecewise polynomial [3]. Wozny et al. proposed an approach to multi-degree reduction of Bézier curves by using the dual-constrained Bernstein basis polynomials associated with the Jacobi scalar product [15]. Chen et al. proposed a tangent method for achieving a higher approximation order, which derived from a linear equation system on the unknown control points of the resultant approximation Bézier curve [4]. We follow the philosophy of curve order reduction that simplifies the representation of complex curves by low-order ones and hence reduces computation costs. Specially, we focus on the mapping of points from ellipses to lines, motivated by the acceleration request of real-time ellipse detection.

B. Ellipse Fitting

Existing ellipse fitting methods can be generally classified into least-square (LS) principle-based methods and voting/sampling (VS) based methods.

Least-square based methods estimate ellipses by minimizing the sum of the squared orthogonal distances from the observed points to the ellipse. Kanatani et al. [16] systematically introduced the application of the LS method in ellipse fitting. However, the fitting results of LS are strongly influenced by the outliers in the data.

Hough transform (HT) was widely used to estimate five parameters using a voting scheme [17]. Most HT-based methods are devoted to overcoming the problems of huge execution time and high memory usage. Xie and Ji [18] proposed a fast method where only one parameter (short-axis) was voted and the other four were estimated by geometric symmetry. Basca et al. sped up Xie's method by clustering the detection results [19]. Suyog et al. improved Xie's method with the major-axis-based voting, more suitable for detecting ellipses with high aspect ratios [20]. Mulleti et al. leveraged the finite rate of innovation sampling principle to fit noisy or partial ellipses [21]. However, HT-based methods are still not efficient enough, and also vulnerable to image noise and model hyper-parameters, *e.g.*, peak threshold and bin size.

Apart from the above representative methods, the generalized Gaussian mixture models (GMM) [22] were introduced for ellipse parameter estimation [23]. Zhao et al. [24] proposed hierarchical Gaussian mixture models for ellipse fitting in noisy, outliers-contained, and occluded settings. Unfortunately, ellipse fitting methods heavily rely on the detected point set, which is vulnerable in realistic scenarios.

C. Ellipse Detection

Recent studies introduce deep neural networks that have gained great success in object detection into ellipse detection. Shi et al. formulated the ellipse fitting task as a non-smooth constrained optimization problem and designed an approach based on two analog neural network models [25]. Li et al. detected lesion bounding ellipses with Gaussian Proposal Networks (GP) [26]. Dong et al. designed a convolutional neural network (CNN) as a region detector to infer oval

objects [27], performing well on occluded ellipses. Deep neural networks largely rely on numerous training examples and demand tremendous computational resources. Therefore, these approaches may be inapplicable to lightweight practical scenarios such as calibration, UAV landing, and robotic manipulation.

Arc-based methods have become mainstream in the past decade owing to their superior detection performance in terms of accuracy and efficiency. ELSDc proposed by Patraucean et al. links small line segments detected by a line detector to form arcs [28]. Kim et al. employed the arc fitting algorithm to merge short lines so as to reduce the number of candidate arcs [29]. Libuda [30] and Prasad [31] further reduced the computational requirements of Kim's method by applying geometric constraints on ellipses. Fornaciari et al. used the convexity and geometric position of arcs to divide them into four quadrant combinations to avoid fitting ellipse within the same combination [32]. Chen et al. [33] and Claudia et al. [23] combined the advantages of HT-based and arc-based methods to detect ellipses in industrial images. Lu et al. proposed an edge connecting strategy named supports arcs and achieved high-quality detection results [14]. Jia et al. selected arcs upon the conic nature of ellipses [2]. Jin et al. [10] cascaded Jia's [2] method with a CNN. Shen et al. [34] proposed a novel method based on the fast computation of convex hulls and directed graphs. It achieved promising results in both accuracy and efficiency. Meng et al. [35] proposed an arc adjacency matrix-based ellipse detection (AAMED) method, rendering fast detection.

The aforementioned methods start the grouping and estimation from points or arcs with positional and/or gradient constraints. Their core acceleration step lies in grouping the arcs belonging to the same ellipse. In this paper, we introduce characteristic mapping to convert the grouping on a conic curve to the calculation on a line segment. Our method is a general accelerator that can be embedded in various arc-based methods.

III. Characteristic Mapping

In this section, we first derive the theorem of characteristic mapping from the property of Characteristic Number [13] for curve degree reduction on the plane. We also prove the necessity of the theorem in 2-dimensional conditions, which provides a theoretical guarantee for the following application of ellipse detection. Then we demonstrate an instance from a conic to a line.

A. Characteristic Mapping Theorem

We give the definition of Characteristic Mapping (CM) in Definition 1, which provides the relationship between collinear points. Then Theorem 1 illustrates how to map points on curves of distinct degrees in a projective space [13]. We provide the proof of Theorem 1 in Appendix A in the Supplementary Material.

Definition 1: Let P_1, P_2, Q, R be collinear points in $\mathbb{P}^2(K)$ (K: the field of complex numbers), where

$$\begin{cases} Q = \lambda_1 P_1 + \lambda_2 P_2 \\ R = \mu_1 P_1 + \mu_2 P_2 \end{cases} \tag{1}$$

Fig. 2. Illustration of CM definition. R is the mapping point of Q. They are symmetrical with respect to the mid-point of P_1 and P_2 in euclidean space.

if

$$\frac{\lambda_1}{\lambda_2} \cdot \frac{\mu_1}{\mu_2} = 1, \tag{2}$$

then the mapping $\chi_{(P_1,P_2)} : Q \mapsto R$ is called the ***characteristic mapping*** (CM) of Q with respect to P_1 and P_2, denoted as $R = \chi_{(P_1,P_2)}(Q)$.

Definition 1 gives the proportion relation of Q and R with respect to P_1 and P_2, which is applicable to points on high dimensional curves. The points Q, P_1 and P_2 are three collinear points, and Q can be linear represented by P_1 and P_2 according to Eq. (1). In the Euclidean space, Eq. (2) reveals R and Q are symmetrical with respect to the mid-point of P_1 and P_2, as demonstrated in Fig. 2. The third value of the homogeneous coordinate of points should be 1, deriving another two constraints in the Euclidean space: $\lambda_1+\lambda_2 = 1$ and $\mu_1 + \mu_2 = 1$. The symmetrical property in the Euclidean space invokes less computation cost than that in the projective space. Consequently, it is unnecessary to solve Eq. (1) as we can easily obtain the mid-point of P_1 and P_2 and obtain the mapping point R according to Q. We provide the proof of the symmetrical property of CM in Appendix B in the Supplementary Material.

Then, we give the characteristic mapping theorem for 2-dimensional (plane) curves of higher degree, which is derived from the higher-dimensional property of Characteristic Number [13]. The theorem embeds the mapping in Definition 1. If Q is constructed by points on a curve of nth order, the mapping point R lies on a curve of $(n-1)$th order.

Theorem 1: P_1, P_2, P_3 are 3 linear independent points in $\mathbb{P}^2(K)$ (K: the field of complex numbers). Suppose there are n points $Q_i^{(1)}, Q_i^{(2)}, \cdots, Q_i^{(n)}$ different from P_1, P_2, P_3 on each line P_iP_{i+1} (i $= 1, 2, 3$, $n \in \mathbb{Z}_+$, $n \geq 2$, and $P_4 := P_1$), where multiple points on the same line have distinct superscripts. For each j ($j = 1, 2, 3$), the line H_j through $\{Q_i^{(j)}\}_{i\neq j}$ intersects the lines P_jP_{j+1} in R_j, and the characteristic mapping points of R_j is $S_j = \chi_{(P_j,P_{j+1})}(R_j)$. Then the $3n$ points $\{Q_i^{(j)}\}_{i=1,2,3}^{j=1,2,\cdots,n}$ lie on a curve of degree n not through any P_i if and only if the $3(n-1)$ points

$$\{S_j\}_{j=1}^{3} \cup \{Q_i^{(j)}\}_{i=1,2,3}^{j=i,4,5,\cdots,n} \tag{3}$$

lie on a curve of degree $n-1$ not through any P_i. There is a special case of $n = 2$. The degree of the curve is $(n-1) = 1$, and the curve degenerates into a line. Also, we set $Q_i^{(3)} := Q_i^{(i)}$ and $\{Q_i^{(j)}\}_{i=1,2,3}^{j=i,4,5,\cdots,n} := \emptyset$. Thus, in this case, R_j ($j = 1, 2, 3$) are three collinear points. The necessity proof for Theorem 1 is illustrated in the Appendix in the Supplementary Material, providing a theoretical guarantee for the application to ellipse detection.

B. Instances of Characteristic Mapping

In order to demonstrate how to map points from a high-degree curve to a low-degree one, we demonstrate an

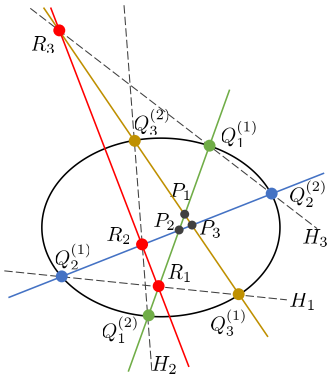


Fig. 3. An example of CM with $n = 2$. Six points $\{Q_i^{(j)}\}_{i=1,2,3}^{j=1,2}$ on the ellipse are mapped to three collinear red points $\{R_i\}_{i=1}^3$.

example of $n = 2$, which is the foundation of the following ellipse detection acceleration.

Figure 3 illustrates a black ellipse (n = 2) and three non-collinear points P_1, P_2 and P_3 (black points) lying outside the curve. They can form three lines P_1P_2, P_2P_3 and P_1P_3 labeled as green, blue, and yellow, respectively. Three lines intersect the curve at $n \times 3 = 6$ points $\{Q_i^{(j)}\}_{i=1,2,3}^{j=1,2}$, where i is the label of lines and j is the label of intersection points on line i. Then, we take 6 points to form three lines H_j, $j = 1, 2, 3$ labeled as black dotted lines, where H_j through $\{Q_i^{(j)}\}_{i\neq j}$. That is, the line H_1 through points $Q_2^{(1)}$ and $Q_3^{(1)}$ intersects P_1P_2 at point R_1. The line H_2 through points $Q_1^{(2)}$ and $Q_3^{(2)}$ intersects line P_2P_3 at point R_2. According to the instruction below Eq. (3), $Q_1^{(3)} = Q_1^{(1)}$ and $Q_2^{(3)} = Q_2^{(2)}$. Line H_3 through points $Q_1^{(1)}$ and $Q_2^{(2)}$ intersects line P_3P_1 at point R_3. Based on the characteristic mapping definition, the mapping points of $\{R_i\}_{i=1}^3$ are $\{S_i\}_{i=1}^3$, which should be collinear. It is worth noting that it is unnecessary to map points R_i to points S_i in this special case. The reason is that if $\{R_i\}_{i=1}^3$ are collinear, the mapping points $\{S_i\}_{i=1}^3$ are collinear, as R_i and S_i are symmetrical with respect to the mid-point of P_i and P_{i+1}.

For $3n$ points on a curve of degree n, the sufficient and necessary condition is their corresponding $3(n-1)$ mapping points lie on curves of degree $n-1$. As it is a recursive relation, we can get the mapping points on a lower-degree curve, until three collinear points. Thus, based on the necessary condition of Theorem 1, we can determine whether the $3n$ points lie on the curve of degree n by judging whether their corresponding mapping points lie on a curve of degree $n-1, n-2, \ldots$, or a line. More specifically, we can determine the collinearity of three mapping points to verify the corresponding six points are lying on a conic.

The CM facilitates using fewer points on curves of a low degree to resolve problems on curves of a high degree, which implies a lower computational cost. In order to convert collinearity to numerical representation, we employ the area of the triangle formed by three mapping points, as it is easily calculated by determinant instead of calculating six points on a conic. Moreover, the calculation on a lower degree curve introduces less error than that on a higher degree one.

IV. Accelerating Ellipse Detection With CM

Ellipse detectors should balance precision and efficiency to meet the requirements of real-time applications. In this section, we first illustrate the standard procedure of arc-based ellipse detection methods and analyze the bottleneck that affects the precision and efficiency of existing methods. Finally, we take advantage of the low computation cost of CM, and design two ways to embed it in ellipse detection algorithms. We make it possible to convert the time-consuming arc grouping problem into a collinear problem, which speeds up the ellipse detection. The proposed CM is generally applicable to most arc-based ellipse detectors.

A. Arc-Based Ellipse Detection

As shown in Fig. 4, an arc-based ellipse detection method can be roughly divided into four steps: pre-processing, arc grouping, ellipse fitting, and ellipse clustering and validation.

In the step of pre-processing, edge points are first extracted. Canny detector is commonly used to calculate the edge points and the gradient information of the image [36]. Then, an eight-neighbor algorithm is used to connect points into arc segments. The arc growth algorithm can be further used to obtain long and high-quality arc segments [2]. Figure 4(a) is the original image, and Fig. 4(b) shows the extracted arcs labeled in different colors.

Arc grouping is crucial to reduce the detection time and false positive detection results. As the following ellipse fitting step spends plenty of time to calculate the parameters of ellipses, fewer arc groups for fitting can save much running time. Most methods take any two arcs as a group for fitting. If there are l detected arcs, there are C_l^2 arc groups at most for fitting [2]. Figure 4(c) demonstrates the fitting ellipses to all possible arc groups, while most of them are invalid combinations, such as $\widehat{arc}_3$ and $\widehat{arc}_4$ in Fig. 4(d). Thus, non-elliptical arcs should be removed, and valid arc groups should include arcs that belong to the same ellipse.

In the arc grouping step, geometric and algebraic constraints are widely used to obtain valid candidate arc combinations, such as curvature, convexity, the direction of arcs, position, conic constraint, and so on [2], [14]. The proposed CM can combine other basic constraints to remove invalid arc groups. As shown in Fig. 4(d), the blue arcs $\widehat{arc}_1$, and $\widehat{arc}_2$ belong to the same ellipse, which can be validated by the fact that three mapping, blue points on the blue dotted line are collinear. On the contrary, the mapping points for $\widehat{arc}_3$ and $\widehat{arc}_4$ are three red non-collinear points, indicating the corresponding arcs belong to different ellipses.

The ellipse fitting step leverages the candidate arc groups produced by the grouping step to calculate the parameters of possible ellipses. Figure 4(e) demonstrates the fitting results after grouping, rendering less fitting loads than Fig. 4(c). In Fig. 4(e), a small number of invalid, proposal ellipses exist, resulting from invalid arc groups that were not pruned by CM, since we work with real and hence imperfect data. In particular, errors introduced by arc detection, low resolution, and noises cause the six points on arcs to deviate from their real locations. (More on this when we discuss threshold Th_{ED}

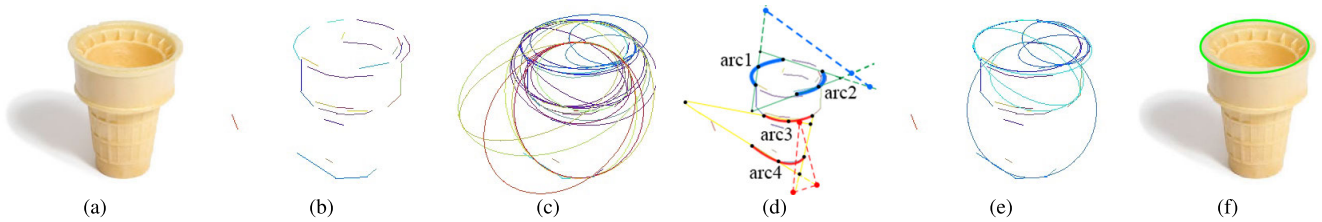


Fig. 4. Ellipse detection flowchart with CM constraints. (a) Original image (b) Extracted arc-support line segments are labeled with different colors. (c) Ellipse candidates without CM constraints on arcs grouping. (d) Two arc group instances are labeled in blue and red, respectively. Three collinear blue points indicate the corresponding blue arcs belong to the same ellipse, while non-collinear red points indicate the red arcs form an invalid group. (e) After filtering by CM module, the number of invalid candidate ellipses is largely reduced. (f) The final detected ellipse.

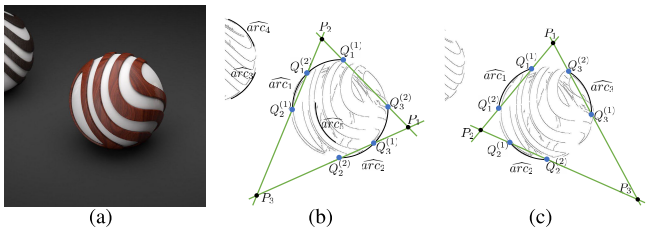


Fig. 5. CM grouping strategy on two or three arcs of each ellipse. (a) The original image. (b) Arc grouping on two arcs, and three points on each arc. (c) Arc grouping on three arcs, and two points on each arc.

in the text below.) However, most of the invalid ellipses will be removed by the ratio of the total arc length of the arc combination to the circumference [32]. Finally, the ellipse clustering and validation steps combine some of the output ellipses into single ellipses, as they may be fitted by different arcs of the same ellipse. Figure 4(f) shows the final detection result.

B. Ellipse Detection Based on CM

In arc-based ellipse detection methods, fitting arc groups is the most time-consuming step, while the arc grouping step determines how many arc groups are fitted. Wrong arc groups can also induce wrong fitting results. Thus, arc grouping is crucial to the precision and speed of an ellipse detection method. A good arc grouping method should preserve valid arcs and prune as many as possible invalid arc groups. Meanwhile, the grouping method itself should be efficient. Thus, it should balance time and performance.

In this paper, we propose a new arc grouping method based on characteristic mapping (CM). We group the arcs by points on arcs. If the points on the arcs belong to the same ellipse, so do the arcs. According to Theorem 1 and Fig. 3, each side of triangle $\triangle P_1P_2P_3$ intersects the ellipse at two points, and there are six points $\{Q_i^{(j)}\}_{i=1,2,3}^{j=1,2}$ in total. We can estimate whether the six points are on the same ellipse by calculating whether their mapping points $\{R_i\}_{i=1}^{3}$ are collinear. Thus, if the six points lie on different arcs, we can judge whether the arcs belong to the same ellipse.

In order to meet the requirements of Theorem 1, we need six points on the ellipse, and three lines through each two of them can form a triangle. We propose two alternative methods to construct characteristic mapping with $n = 2$, which are demonstrated in Fig. 5(b) and Fig. 5(c) respectively. As shown in Fig. 5(b), we can validate two arcs each time by taking three points on each arc, two points on the ends of the arc and the third in the middle of the arc. Figure 5(c) demonstrates the way that validates three arcs each time by taking two points on the ends of each arc. Both situations satisfy the theorem, including the green triangle $\triangle P_1P_2P_3$ and six intersections $\{Q_i^{(j)}\}_{i=1,2,3}^{j=1,2}$ with arcs. In both methods, we take at least two points on each arc, as one point cannot determine the position of an arc, and it is more sensitive to noise. Two methods are different in accuracy and efficiency. Three points on each arc are more accurate than two points on each arc, as more points can reflect the condition of the arc better. However, the situation with two points on each arc estimates three arcs each time, which is more efficient than the strategy on two arcs. In our paper, we take the strategy of validating two arcs, as most existing methods also apply their constraints on two arcs. We further map six points to three points by characteristic mapping. Finally, we can judge whether arcs with six points belong to the same ellipse by validating the collinearity of the three mapped points.

The proposed CM can be embedded in any arc-based method, and it can be used separately or combined with other existing constraints. CM is a strict constraint, which requires arcs belong to the same ellipse. It can be used after some loose constraints, such as position and quadrant constraints. For example, the quadrants constraint requires that arcs in a group are from different quadrants. However, arcs in different quadrants are likely from different ellipses. As shown in Fig. 5(b), we can easily prune the combination of $\widehat{arc_2}$ and $\widehat{arc_3}$ in the fourth quadrant by quadrant constraint, but CM can further prune $\widehat{arc_2}$ and $\widehat{arc_5}$. Algorithm 1 details the arc combination picking process with CM on two arcs.

The threshold Th_{ED} (line 6 of algorithm 1) is used to evaluate the collinearity of three mapped points. The determinant of 3 points indicates the area of the triangle formed by them [13]. When it is too small, some correct arc combinations are removed; when it is too large, some wrong arc combinations are included. We set $Th_{ED} = 2$ to get the best performance in our experiments. We also test the distance of the third point from the line defined by the other two. However, the determinant computation takes less time to execute.

Remarks: (1) The difference between CM and Pascal's theorem. As one of the classical theorems in projective geometry, Pascal's theorem asserts that *if a hexagon is made out of six points on a conic, the intersections of opposite sides of this hexagon must be collinear* [37]. It also implies a mapping from 6 points on conic to 3 collinear points.

Algorithm 1 Arc Grouping Algorithm With CM on Two Arcs

Input: Arc set Set_{arc}, Collinear determination threshold Th_{ED}

Output: Arc combination set $Set_{combination}$

for $\widehat{arc_i} \in Set_{arc}$ **do**
  for $\widehat{arc_j} \in Set_{arc}$ and $j \neq i$ **do**
    if $(\widehat{arc_i}, \widehat{arc_j})$ do not meet position or quadrants constraints **then**
      Continue;
    end if
    if CM-VALIDATION$(\widehat{arc_i}, \widehat{arc_j}) > Th_{ED}$ **then**
      Continue;
    end if
    Add $\{\widehat{arc_i}, \widehat{arc_j}\}$ to arc combination set $Set_{combination}$.
  end for
end for
return $Set_{combination}$;
function CM-VALIDATION $(\widehat{arc_i}, \widehat{arc_j})$
$\{Q_x^{(y)}\}_{x=1,2,3}^{y=1,2} \leftarrow$ Take two ends and the midpoint of $\{\widehat{arc_i}, \widehat{arc_j}\}$;
$\{S_x\}_{x=1}^{3} \leftarrow$ Obtain mapping points by characteristic mapping;
$Value_{real} \leftarrow$ Calculate the collinearity of $\{S_x\}$;
return $Value_{real}$;
end function

However, the collinear estimation may fail in arc grouping, as the opposite sides of the hexagon have a high chance to be parallel for adjacent arcs in Algorithm 1. As demonstrated in Fig. 6(a), side A_3B_3 is almost parallel to side A_1B_1, and the corresponding intersection point is close to point at infinity, yielding a significant error in estimating the collinearity with M_1 and M_2. Our characteristic mapping (CM) facilitates arc grouping in more extensive practical scenarios.

(2) The difference between CM and characteristic number (CN) [2]. They are different in basic theory, computational efficiency, and accuracy. The definition and theorem of CM fundamentally differ from CN. CM characterizes the intrinsic connections between algebraic curves of different degrees and does *not* rely on the parametric forms of curves, while CN reflects the intrinsic geometry of an underlying planar curve of points. The proposed method groups curve segments by solving the problem on a curve of a lower degree, which implies a lower computation cost. CN only employs the properties of conics itself, giving the value of $+1$ on any six points on an ellipse.

Further, CM has higher computational efficiency than CN. As the quadratic curve is a special case in CM, the symmetrical property of CM enables to measure the collinearity of $\{R_i\}_{i=1}^{3}$ instead of their mapping points $\{S_i\}_{i=1}^{3}$. Consequently, it is unnecessary to solve Eq. (1), and the collinearity of three points is easily calculated by the determinant. In contrast, calculating CN values of the six points on an ellipse involves solving a set of six linear equations, which makes CN unable to maintain linear computational complexity.

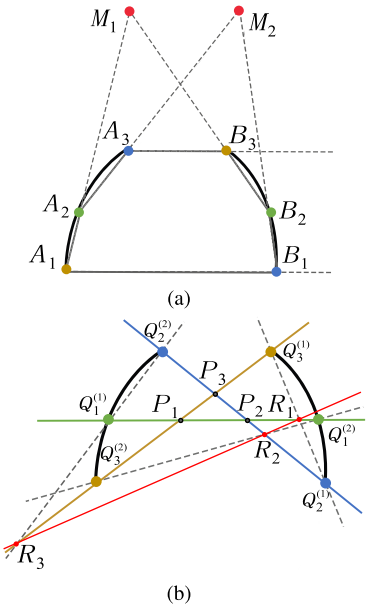


Fig. 6. Comparison of the mapping strategies of (a) the classic Pascal's theorem and (b) CM.

CM has higher accuracy than CN. CM leverages the collinearity of three mapping points to group arcs. The error induced by noise and arc detection can be easily estimated. In contrast, CN relies on the product of a series of coefficient ratios, and the error in any multiplier is heavily magnified in the product.

V. Experimental Analysis and Results

This section includes experimental settings, a comparison with the state-of-the-art, and an analysis of methods with and without the proposed CM accelerator.

A. Experimental Setting

In this section, we introduce the datasets, the employed methods for comparison, and general evaluation metrics.

1) Datasets: To evaluate the performance of the proposed CM accelerator, we use five public, challenging real-world datasets. Traffic data set [38] consists of 273 images of various traffic signs from frames of several videos captured by smartphones [32]. Bicycle data set contains 356 images of bicycles with different backgrounds and shooting angles. These images are also selected from the frames of the video mentioned above. Prasad data set [39] is a classic data set consisting of 198 complex real-world images, where objects of oval shapes like human faces are regarded as ellipses. The original Prasad Dataset has 400 images in total. However, there are only 198 images available online. Shen et al. [34] collected the remaining images of the original Prasad dataset and named it Prasad+, which includes 193 images. The varying image size with a cluttered background is the major challenge. Dataset#1 [32] selects 400 images from the MIRFlickr and LabelMe repositories, and randomly collects high-quality images that contain a single ellipse or noisy images that contain multiple objects.

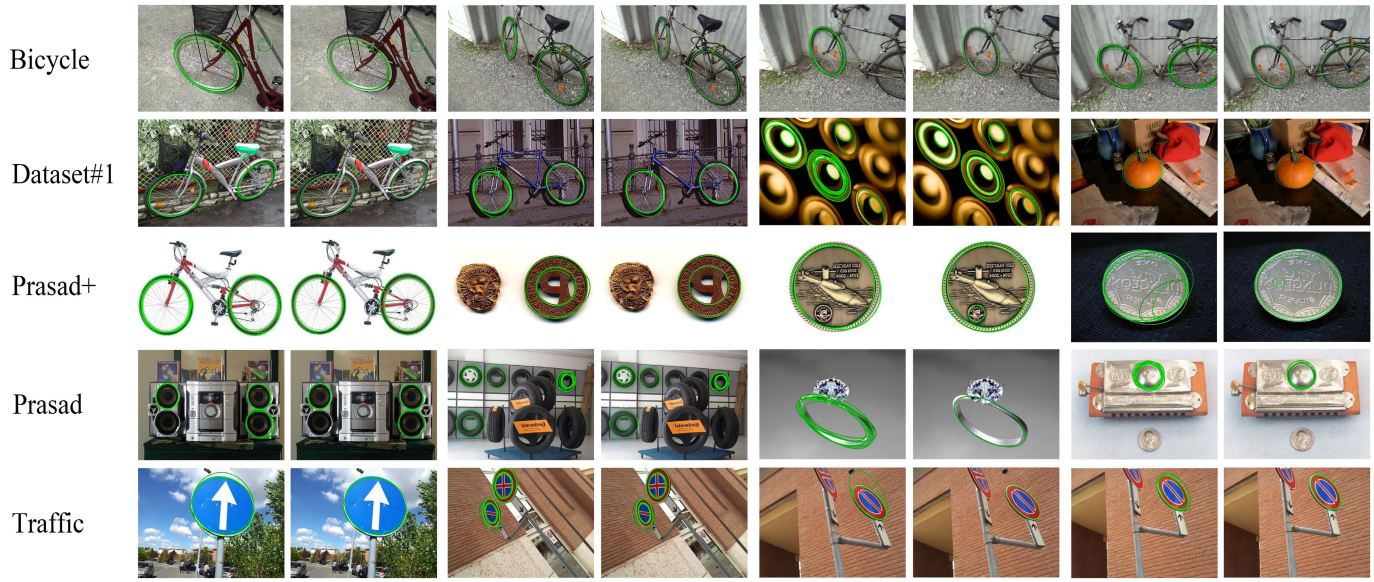


Fig. 7. Four pairs of detection instances produced by Jia's method without (left) and with CM (right) on each data set. The left picture in each pair (produced by the original Jia's method) includes more false positives and redundant detections.

2) Compared Methods: We select six traditional and a deep learning based methods for quantitative and qualitative comparisons, including Fornaciari method [32], ELSDc [28], AAMED [35], Shen [34], Jia [2], Lu's method [14], and Gaussian Proposal Networks (GPN) [26]. For a fair comparison, we adopt the source codes provided by their authors. Further, we evaluate the proposed CM accelerator by embedding it in two representative ellipse detectors: Jia's method [2] and Lu's method [14]. Jia et al. employ a projective invariant, named characteristic number to pick valid arc segments, which is the fastest one for real-world natural images reported in the latest work. Lu et al. construct support arcs upon the gradient information that removes outliers, yielding robust and accurate detection [14].

3) Evaluation Metrics: All experiments are executed on a laptop with Intel Core i7-6700HQ CPU 2.60GHz, Nvidia GeForce GTX 1060 6GB, 16GB RAM, and Windows 10 64bit operating system. The software versions are MATLAB 2018a, Visual Studio 2017, and OpenCV-3.4.7_vc15. The memory utilization during execution is $23\% \sim 35\%$.

We follow the same protocol as comparable methods. The performance of ellipse detectors is evaluated in terms of running time, Precision (Pre), Recall (Rec), and F-measure (F-m). F-measure is defined as F-measure $= 2$ / (Precision^{-1} + Recall^{-1}), where Precision $=$ TP / (TP + FP) $=$ TP / Ω, and Recall $=$ TP / (TP + FN) $=$ TP / Φ. The symbol Ω denotes the number of detected ellipses, and Φ indicates the number of ground-truth ellipses. The overlapping ratio of a detected ellipse E_d to the ground truth E_g is defined as:

$$M(E_d, E_g) = \frac{area(E_d) \cap area(E_d)}{area(E_d) \cup area(E_d)}, \tag{4}$$

where area(E) is the number of pixels inside the ellipse E. The detected ellipse E_d is considered as TP (true positive) if $M(E_d, E_g)$ is larger than a threshold $Th_{correct}$. The threshold $Th_{correct}$ is set to 0.8 throughout our experiments. FP and FN are false positive and false negative, respectively. We accumulate TP, FP and FN of each picture to obtain F-m value and execution time of the entire dataset.

B. Performance Analysis of Arc Selection

We use the collinear constraint of three mapping points derived from six points on arcs to pick up arc segments belonging to one ellipse. Theoretically, the area of a triangle formed by three mapping points equals 0, but various imaging conditions (e.g., thermal noise and lens distortions) in practical applications and errors introduced by the arc detection step may cause the value to deviate from 0. As discussed in Section IV-B, we relax this hard constraint to a range in the vicinity of 0 determined by Th_{ED}. Herein, we perform an experimental analysis of the relationship between point coordinates on arcs and the area of a triangle formed by three mapping points. This analysis does not only give rise to an appropriate threshold but also validates the effectiveness of the arc selection based on CM.

We analyze the relationship between the accuracy of the proposed method and the distribution of the mapping points. Assuming an ellipse centered at the coordinate origin, we fix five distinct points on the ellipse, and vary the sixth point around the ellipse to show the distribution of the collinearity of mapping points in Fig. 8. We use the area of the triangle formed by mapping points to measure the collinearity, which can be easily calculated by the determinant of the three points. Different colors indicate various area values given by the color bar in Fig. 8. All the area values higher than 2 are colored in red-brown, while all those lower than -2 are in blue. Fig. 8 illustrates that most areas formed by mapping points with the sixth point close to the ellipse fall within the range from -0.5 to 0.5. In contrast, CN is sensitive to other conic curves, such as hyperbola and parabola. The sixth point lying inside or outside of the ellipse can also generate the desired value of $+1$ [2].

TABLE I
COMPARISONS OF ORIGINAL AND ACCELERATED METHODS ON FIVE DATA SETS

Dataset		Fornaciari [32]	ELSDc [28]	AAMED [35]	Shen [34]	Jia [2]	Jia's with CM	Lu [14]	Lu's with CM
Traffic	Pre	0.55	0.05	0.84	0.70	0.53	0.62	0.92	**0.92**
	Rec	0.70	0.76	0.67	0.84	0.73	0.73	**0.88**	0.87
	F-m	0.62	0.09	0.75	0.76	0.61	0.67	0.90	**0.90**
	Time(ms)	531.7	1149.7	291.4	962.8	200.5	**180.5**	346.0	181.0
Prasad	Pre	0.72	0.09	0.77	0.62	0.70	0.75	0.76	**0.77**
	Rec	0.19	0.34	0.32	0.33	0.25	0.24	0.35	**0.35**
	F-m	0.30	0.14	0.45	0.43	0.37	0.37	0.48	**0.48**
	Time(ms)	184.3	347.1	67.3	326.8	71.0	**60.1**	105.4	67.7
Prasad+	Pre	0.65	–	0.74	0.56	0.67	0.74	0.78	**0.78**
	Rec	0.41	–	0.44	0.52	0.47	0.45	**0.57**	0.55
	F-m	0.51	–	0.55	0.54	0.55	0.56	**0.66**	0.65
	Time(ms)	1011.0	–	126.3	619.8	202.2	**123.1**	368.5	164.2
Dataset#1	Pre	0.64	–	**0.71**	0.54	0.62	0.67	0.69	0.69
	Rec	0.40	–	0.38	0.47	0.43	0.42	**0.48**	0.46
	F-m	0.49	–	0.49	0.50	0.51	0.51	**0.56**	0.55
	Time(ms)	766.2	–	186.7	745.8	188.1	**145.2**	462.3	199.5
Bicycle	Pre	0.50	–	0.61	0.40	0.50	0.56	0.75	**0.76**
	Rec	0.47	–	0.23	0.56	0.44	0.41	**0.59**	0.54
	F-m	0.49	–	0.33	0.46	0.47	0.47	**0.66**	0.63
	Time(ms)	1671.0	–	437.4	1288.6	335.1	**243.7**	1028.8	476.2

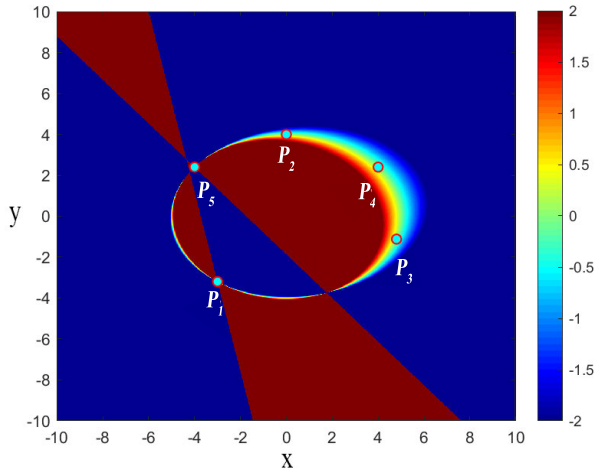


Fig. 8. Relationship between point coordinates and collinearity of mapping points. Different colors indicate the area of the triangle formed by mapping points given by the color bar. Five distinct points on the ellipse are fixed, and the position of the sixth point is varied around the ellipse to show the distribution of the collinearity of mapping points.

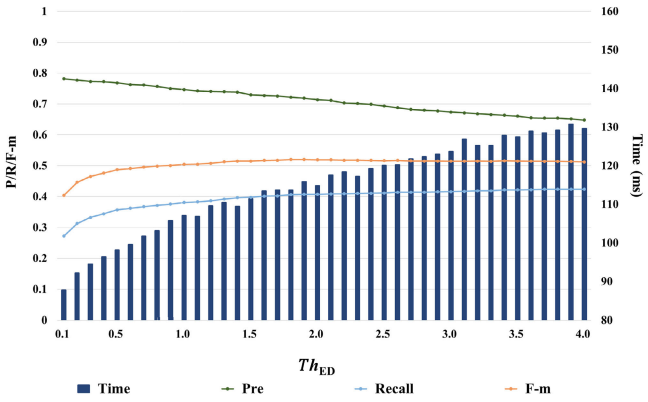


Fig. 9. Effects of the threshold Th_{ED} ranging from 0.1 to 4.0.

To obtain an appropriate value for threshold Th_{ED}, we test Jia's method with CM on Dataset#1, as it is the largest dataset including various situations in practical scenarios. We vary the threshold Th_{ED} between 0.1 and 4.0, giving 40 Th_{ED} values in total. Figure. 9 demonstrates the values of running time, precision, recall, and F-m upon various thresholds. The hard constraint $Th_{ED} = 0.1$ (the arcs have to rigorously form an ellipse) only keeps a small number of arc segments for parameter fitting, demanding the least computation. Unfortunately, significant false negatives exist since this choice excludes many arc segments slightly deviating from an ellipse. Accordingly, the value of precision is high, while those of recall and F-m are notably low. The increase of Th_{ED} includes more arcs for fitting, resulting in a higher computational cost. In these cases, the selection step has little impact on lowering false positives. Hence, the values of precision, recall, and F-m remain stable. We fix the threshold as $Th_{ED} = 2.0$. It generates the best performance for the rest of the experiments. Actually, the hyper-parameter Th_{ED} indicates the tolerance to deviations of a group of arc candidates from a right ellipse on which any six points map to the three collinear points giving a zero determinant. As a byproduct, the choice of Th_{ED} affects the number of candidate arcs that directly determines the time saved for subsequent arc grouping stages. However, the original number of candidate arcs determines how much CM can accelerate.

C. Comparisons With the State-of-the-Art Methods

1) Comparisons With Traditional Methods: We embed the proposed CM accelerator into two representative methods, Jia's method [2] and Lu's method [14]. Both of them have their own arc grouping strategy. We replace or add our accelerator to the original methods directly, and keep the other parts

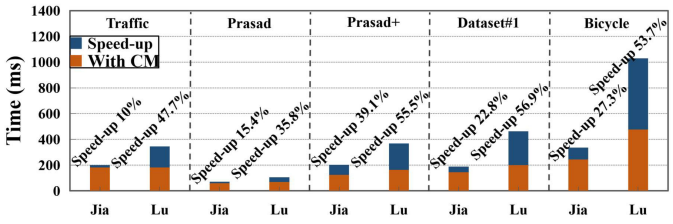


Fig. 10. The acceleration ratio of two representative ellipse detection methods Jia [2] and Lu [14] when the proposed CM is used. The horizontal axis shows the name of two methods, and performances on different datasets are divided by dotted lines. The value of each bar is labeled by the vertical axis. The orange bar shows the execution time of the corresponding method with CM, while the blue part shows how much time CM saves. The speed-up ratio is shown on the top of each bar.

unchanged. Then, we compare the accelerated methods with the original Jia's [2], and Lu's method [14], and the other four traditional methods of Fornaciari [32], ELSDc [28], AAMED [35], and Shen [34].

As shown in Tab. I, we compare the Precision, Recall, F-measure, and execution time of all the methods over five datasets. Generally speaking, almost all the best performances are produced by the methods with CM accelerator. Lu's method with and without CM accelerator researches the highest F-m over five datasets, which is as high as 0.9 on Traffic dataset. Jia's method with CM accelerator has the smallest execution time over five datasets. The smallest execution time of 60.1 ms per image is obtained on Prasad dataset. CM slightly improves F-m value for Jia's method, while speeding up the execution time of the original method 25% on average over five datasets. The speed-up ratio for Lu's method is more significant than for Jia's method. Lu's method with CM only uses about half of the original execution time with almost the same F-m.

In general, CM improves Jia's method on all the indexes. CM also obtains comparative or even higher F-measure with about 50% of Lu's original running time on three datasets. There is only about 0.02 decrease on average in recall of Lu's method, as CM may remove a few true positive arc groups due to error and noise of sample points on arcs. There are only 0.01 and 0.03 decreases on F-measure on dataset #1 and dataset Bicycle. However, it is worth noting that we only use about 44% running time of the original method on these two datasets.

In order to illustrate the acceleration performance of CM module, we use a bar plot to represent the execution time with and without CM in Fig. 10. Two methods combined with our accelerator are sped-up to different degrees over five datasets. We can see two methods are sped-up from 10% to 57%. The speed-up ratio of Jia's method on the Prasad+ dataset is 39.1% but down to 27.3% on the Bicycle dataset, given the same value of Th_{ED}. The root of this discrepancy attributes to different counts of candidate arcs in these two datasets. We obtain the highest speed-up ratio of 57% on Dataset#1, as noise in this data set produces a vast amount of arc segments for grouping.

2) Comparison With a Deep Learning Based Method: We compare the proposed method with Gaussian Proposal Networks (GPN) [26]. GPN is designed to detect lesion-bounding ellipses in the medical image processing area, in which accuracy is highly demanded. We train the model on the Traffic dataset, as only one or two elliptic objects exist in most of the pictures, and almost all the objects have clear boundaries, which is similar to the lesion detection scenario. We obtain 1076 augmented images by flipping and rotation. All the 1076 augmented images are used as the training set, while the testing set is the original 296 images to keep the same protocol as our method. Table. II demonstrates that the proposed method outperforms GPN in almost all aspects. The performance of Lu's method with CM outperforms GPN on all indexes. The F-measure of Lu's method with CM is over 1.5 times GPN's F-measure, while the running time of Lu's method with CM is only 11.8% of GPN's running time. Similarly, Jia's method with CM also surpasses GPN in F-measure and running time.

TABLE II
COMPARISON OF ACCELERATED METHODS AND THE DEEP LEARNING BASED METHOD GPN [26] ON TRAFFIC DATASET

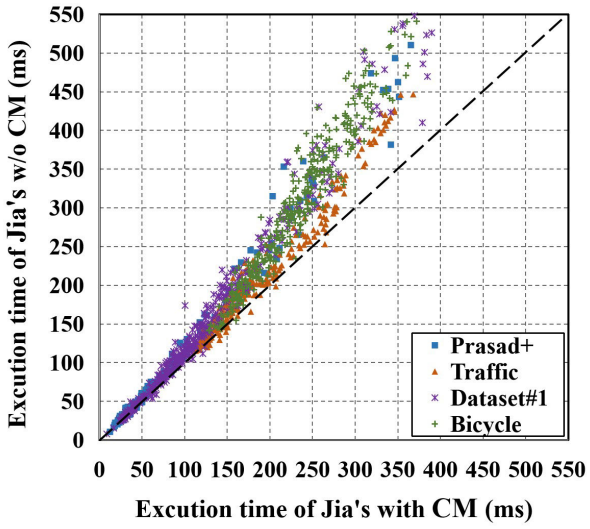
Methods	Jia's with CM	Lu's with CM	GPN [26]
Pre	0.62	**0.92**	0.68
Rec	0.73	**0.87**	0.53
F-m	0.67	**0.90**	0.59
Time(ms)	**180.5**	181.0	1520

D. Effectiveness Analysis of CM Accelerator

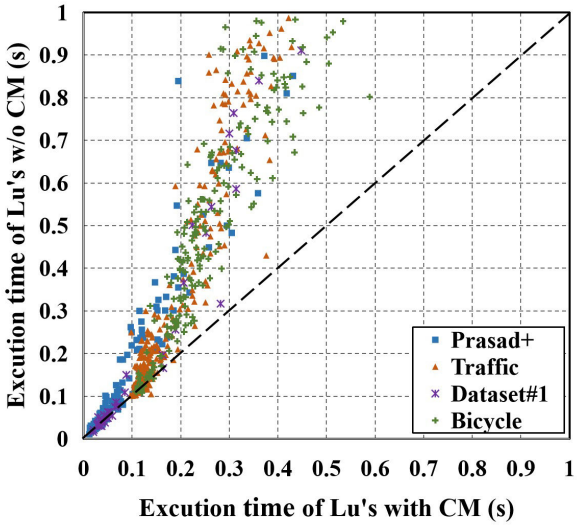
We illustrate how we embed our accelerator into two methods, and analyze why the methods combined with CM accelerators can reach the best performance.

Jia's method is the fastest one for real-world natural images reported in the latest work [35]. Our accelerator can further reduce the running time by 25% on average with higher accuracy. Jia's method employs a projective invariant, named characteristic number (CN), to pick valid arc combinations [2]. Six points on two elliptical arcs are used to determine whether the arcs belong to the same ellipse. As this grouping strategy is also a conic constraint, we replace it with our accelerator with other parts unchanged. Although both grouping strategies are conic constraints, our CM accelerator only works on three points, and the collinearity of three points is easily calculated by the determinant. In contrast, Jia's method demands more computations due to the calculation of the product of the characteristic number of six points. Meanwhile, the product value is sensitive to errors. Thus, we can improve the value of F-m while reducing the execution time with the accelerator. Some detection instances of Jia's method without and with CM on five datasets are shown in Fig. 7. Our accelerator evaluates the arc combinations more precisely, yielding fewer false positive or redundant detections.

Lu's method reaches the highest F-m among the test methods by a complicated arc detection strategy, that's why it spends more running time than Jia's method. As most invalid arcs are removed after arc extraction, the number of arcs is limited. They only employ simple geometric constraints to obtain candidate arc combinations. We add our accelerator



(a)



(b)

Fig. 11. Running time comparison between methods with/without CM on each image. The X-axis is the running time of methods with CM, while Y-axis is that without CM. The images in different data sets are labeled in different colors.

after the geometric constraints to prune more invalid combinations. Thus, our accelerator can keep the F-m while speeding up the execution time significantly.

We further plot the execution time of each image with and without CM accelerator to illustrate the effectiveness of the proposed module. As shown in Fig. 11, almost all the points are above the diagonal, which indicates the accelerator can speed up ellipse detection on the vast majority of images. The most obvious acceleration occurs on Dataset#1 and Prasad+ data set, as there are more arc group candidates for grouping. Thus, our accelerator can reduce execution time significantly on multiple candidate arcs.

E. Stepwise Execution Time Analysis and Robustness to Noise

In order to validate the proposed CM accelerator can reduce the execution time while keeping the performance of the original methods, we illustrate the execution time of each step

TABLE III
EXECUTION TIME (MS) FOR EACH STEP WITH AND WITHOUT CM MODULE

method	Arc grouping selection	Candidate ellipse estimation	Validation
Lu	17.81	3.28	79.75
Lu with CM	18.26	2.20	42.67

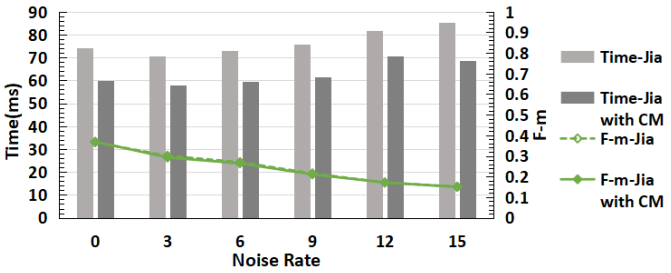


Fig. 12. Robustness of Jia's method w or w/o CM to noise. The horizontal axis indicates the ratios of noise to an image, the left vertical axis indicates the execution time, and the right vertical axis shows the value of F-m. The execution time of Jia's method w or w/o CM is shown by dark grey and grey bars, respectively. F-m values w or w/o CM are illustrated by green Solid and dotted lines, respectively.

with or without CM, and demonstrate the robustness of CM to noise.

In order to demonstrate how the embedded CM accelerates arc-based methods, we compare the execution time of Lu's method with or without CM on Prasad dataset by dividing Lu's method into three steps: arc grouping, candidate ellipse estimation, and validation. In the arc grouping step, the input image is pre-processed, arcs are formed and candidate arc pairs are selected. As shown in Tab. III, the original Lu's method use 17.81 ms on average to obtain valid arc pairs, while Lu's method with CM cost 18.26 ms. Removing invalid arc groups save more time than it takes, as lots of invalid arc groups are removed. We only use 0.45 ms more to embed CM in the arc pairs selection step, but we save 1.08 ms in the candidate ellipse estimation and save 37.08 ms in the validation step.

Our accelerating module has no impact on the stability of the original ellipse detection method. We add salt-and-pepper noise to images of Prasad data set with the ratios of noise to image set as 0%, 3%, 6%, 9%, 12%, and 15%. As shown in Fig. 12, the F-measure for Jia's method with or without CM is almost the same with the noise increasing, indicating the decrease of F-m is introduced by the original method other than CM. Thus, CM speeds up the execution time while having no impact on the performance.

VI. CONCLUSION

In this paper, we develop a characteristic mapping and prove a necessary condition for its application. An efficient arc grouping module is proposed to find valid elliptical arc segments by determining whether the mapped three points lie on one line. The developed module can be embedded into existing arc-based ellipse detection methods to reduce their run time. Experiments with the two latest methods show that their running time can be reduced by 25% and 50%, respectively. It is worth noticing that the computation on three points is more precise than that on six elliptical points of the original methods, as it is less sensitive to errors in point locations.

This yields faster than the state-of-the-art algorithms while keeping their precision comparable or even higher. Moreover, CM can be applied to curves of higher dimensions and any higher degrees. We will consider this application in man-made CAD graphics in our feature work.

REFERENCES

[1] Y. Liu, H. Chen, C. Shen, T. He, L. Jin, and L. Wang, "ABCNet: Real-time scene text spotting with adaptive Bezier-curve network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9809–9818.

[2] Q. Jia, X. Fan, Z. Luo, L. Song, and T. Qiu, "A fast ellipse detector using projective invariant pruning," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3665–3679, Aug. 2017.

[3] A. Rababah, "High accuracy Hermite approximation for space curves in rd," *J. Math. Anal. Appl.*, vol. 325, no. 2, pp. 920–931, Jan. 2007.

[4] X.-D. Chen, W. Ma, and Y. Ye, "Multi-degree reduction of Bézier curves with higher approximation order," in *Proc. Int. Conf. Computer-Aided Design Comput. Graph.*, 2013, pp. 427–428.

[5] P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, *Model Reduction and Approximation: Theory and Algorithms*, vol. 15. Philadelphia, PA, USA: SIAM, 2017.

[6] M. Eck, "Degree reduction of Bézier curves," *CAGD*, vol. 10, nos. 3–4, pp. 237–251, 1993.

[7] X. Fan, R. Liu, Z. Luo, Y. Li, and Y. Feng, "Explicit shape regression with characteristic number for facial landmark localization," *IEEE Trans. Multimedia*, vol. 20, no. 3, pp. 567–579, Mar. 2018.

[8] J. Sun et al., "Disp R-CNN: Stereo 3D object detection via shape prior guided instance disparity estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10545–10554.

[9] L. Hajder, T. Tóth, and Z. Pusztai, "Automatic estimation of sphere centers from images of calibrated cameras," in *Proc. 15th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2020, pp. 1–13.

[10] R. Jin, H. M. Owais, D. Lin, T. Song, and Y. Yuan, "Ellipse proposal and convolutional neural network discriminant for autonomous landing marker detection," *J. Field Robot.*, vol. 36, no. 1, pp. 6–16, Jan. 2019.

[11] H. Dong, E. Asadi, G. Sun, D. K. Prasad, and I.-M. Chen, "Real-time robotic manipulation of cylindrical objects in dynamic scenarios through elliptic shape primitives," *IEEE Trans. Robot.*, vol. 35, no. 1, pp. 95–113, Feb. 2018.

[12] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[13] Z. Luo, X. Zhou, and D. X. Gu, "From a projective invariant to some new properties of algebraic hypersurfaces," *Sci. China Math.*, vol. 57, no. 11, pp. 2273–2284, Nov. 2014.

[14] C. Lu, S. Xia, M. Shao, and Y. Fu, "Arc-support line segments revisited: An efficient high-quality ellipse detection," *IEEE Trans. Image Process.*, vol. 29, pp. 768–781, 2019.

[15] P. Woźny and S. Lewanowicz, "Multi-degree reduction of Bézier curves with constraints, using dual Bernstein basis polynomials," *Comput. Aided Geometric Design*, vol. 26, no. 5, pp. 566–579, Jun. 2009.

[16] K. Kanatani, Y. Sugaya, and Y. Kanazawa, "Ellipse fitting for computer vision: Implementation and applications," *Synth. Lectures Comput. Vis.*, vol. 6, no. 1, pp. 1–141, Apr. 2016.

[17] P. V. Hough, "Method and means for recognizing complex patterns," Dec. 18, 1962, U.S. Patent 3 069 654.

[18] Y. Xie and Q. Ji, "A new efficient ellipse detection method," in *Proc. ICPR*, 2002, pp. 957–960.

[19] C. A. Basca, M. Talos, and R. Brad, "Randomized Hough transform for ellipse detection with result clustering," in *Proc. Int. Conf. Comput. Tool*, vol. 2, 2005, pp. 1397–1400.

[20] S. Sawala, S. Ragothaman, S. Narasimhan, and M. G. Basavaraj, "A versatile major axis voted method for efficient ellipse detection," *Pattern Recognit. Lett.*, vol. 104, pp. 45–52, Mar. 2018.

[21] S. Mulleti and C. S. Seelamantula, "Ellipse fitting using the finite rate of innovation sampling principle," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1451–1464, Mar. 2016.

[22] B. Jian and B. C. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1633–1645, Aug. 2011.

[23] C. Arellano and R. Dahyot, "Robust ellipse detection with Gaussian mixture models," *Pattern Recognit.*, vol. 58, pp. 12–26, Oct. 2016.

[24] M. Zhao, X. Jia, L. Fan, Y. Liang, and D.-M. Yan, "Robust ellipse fitting using hierarchical Gaussian mixture models," *IEEE Trans. Image Process.*, vol. 30, pp. 3828–3843, 2021.

[25] Z. Shi et al., "Robust ellipse fitting based on Lagrange programming neural network and locally competitive algorithm," *Neurocomputing*, vol. 399, pp. 399–413, Jul. 2020.

[26] Y. Li, "Detecting lesion bounding ellipses with Gaussian proposal networks," in *Proc. MLMI*. Cham, Switzerland: Springer, 2019, pp. 337–344.

[27] W. Dong, P. Roy, C. Peng, and V. Isler, "Ellipse R-CNN: Learning to infer elliptical object from clustering and occlusion," *IEEE Trans. Image Process.*, vol. 30, pp. 2193–2206, 2021.

[28] V. Pătrăucean, P. Gurdjos, and R. G. Von Gioi, "Joint a contrario ellipse and line detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 788–802, Apr. 2017.

[29] E. Kim, M. Haseyama, and H. Kitajima, "Fast and robust ellipse extraction from complicated images," in *Proc. IEEE Inf. Technol. Appl.*, Nov. 2002, pp. 138–145.

[30] L. Libuda, I. Grothues, and K.-F. Kraiss, "Ellipse detection in digital image data using geometric features," in *Proc. Adv. Comput. Graph. Comput. Vis., Int. Conf.* Cham, Switzerland: Springer, 2007, pp. 229–239.

[31] D. K. Prasad, M. K. H. Leung, and C. Quek, "ElliFit: An unconstrained, non-iterative, least squares based geometric ellipse fitting method," *Pattern Recognit.*, vol. 46, no. 5, pp. 1449–1465, May 2013.

[32] M. Fornaciari, A. Prati, and R. Cucchiara, "A fast and effective ellipse detector for embedded vision applications," *Pattern Recognit.*, vol. 47, no. 11, pp. 3693–3708, Nov. 2014.

[33] S. Chen, R. Xia, Y. Chen, M. Hu, and J. Zhao, "A hybrid method for ellipse detection in industrial images," *Pattern Recognit.*, vol. 68, pp. 82–98, Aug. 2017.

[34] Z. Shen, M. Zhao, X. Jia, Y. Liang, L. Fan, and D.-M. Yan, "Combining convex hull and directed graph for fast and accurate ellipse detection," *Graph. Models*, vol. 116, Jul. 2021, Art. no. 101110.

[35] C. Meng, Z. Li, X. Bai, and F. Zhou, "Arc adjacency matrix-based fast ellipse detection," *IEEE Trans. Image Process.*, vol. 29, pp. 4406–4420, 2020.

[36] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[37] J. G. Semple and G. T. Kneebone, *Algebraic Projective Geometry*. New York, NY, USA: Oxford Oxford Univ. Press, 1998.

[38] C. Lu, S. Xia, W. Huang, M. Shao, and Y. Fu, "Circle detection by arc-support line segments," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 76–80.

[39] D. K. Prasad, M. K. Leung, and S.-Y. Cho, "Edge curvature and convexity based ellipse detection method," *Pattern Recognit.*, vol. 45, no. 9, pp. 3204–3221, Sep. 2012.

Qi Jia received the B.E. and Ph.D. degrees in computer science and technology from the Dalian University of Technology, Dalian, China, in 2005 and 2014, respectively. She joined the Dalian University of Technology in 2008, where she is currently an Associate Professor. Her current research interests include computational geometry, image processing, and computer vision. She serves as an Associate Editor for the *Pattern Recognition* journal and the Area Chair of OpenCV.

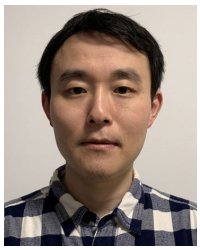
Xin Fan (Senior Member, IEEE) was born in 1977. He received the B.E. and Ph.D. degrees in information and communication engineering from Xi'an Jiaotong University, Xi'an, China, in 1998 and 2004, respectively. He was with Oklahoma State University, Stillwater, from 2006 to 2007, as a Postdoctoral Research Fellow. In 2008, he was with the Southwestern Medical Center, The University of Texas at Dallas, Dallas, for the second postdoctoral training. He joined the School of Software, Dalian University of China, Dalian, China, in 2009, where he is currently a Professor and the Dean of the School of International Information and Software. His current research interests include computational geometry and machine learning and their applications to target tracking, image processing, and DTI-MR image analysis.

Yang Yang received the B.E. degree in digital media technology from the Dalian University of Technology, Dalian, China, in 2020, where he is currently pursuing the M.E. degree with the Laboratory of Geometric Computing and Digital Media Technology. His current research interests include computational geometry and computer vision.

Xuxu Liu received the B.E. degree in software engineering from Zhengzhou University in 2019 and the M.E. degree in software engineering from the Dalian University of Technology in 2021. His main research directions are computer vision, shape analysis, ellipse detection, and acceleration processing.

Zhongxuan Luo received the B.S. and M.S. degrees in computational mathematics from Jilin University, China, in 1985 and 1988, respectively, and the Ph.D. degree in computational mathematics from the Dalian University of Technology, China, in 1991. He has been a Full Professor with the School of Mathematical Sciences, Dalian University of Technology, since 1997, where he is currently the Vice President. His research interests include computational geometry and computer vision.

Qian Wang was born in 1982. She received the B.S. and Ph.D. degrees in computational mathematics from the Dalian University of Technology, China, in 2005 and 2014, respectively. She joined the School of Mathematics, Liaoning Normal University, in September 2014. Her research interests include computational geometry and computer-aided geometric design.



Xinchen Zhou received the B.S. and Ph.D. degrees in mathematics from the Dalian University of Technology, Dalian, China, in 2011 and 2017, respectively. He is currently working with the School of Mathematical Sciences, Jiangsu University, Zhenjiang, China. His research interests include computational geometry and numerical methods for PDEs.

Longin Jan Latecki (Senior Member, IEEE) is currently a Professor with Temple University. He has published over 300 research papers and books. His main research interests include computer vision and machine learning. He received the Annual Pattern Recognition Society Award together with Azriel Rosenfeld for the best article published in the journal *Pattern Recognition* in 1998. He was a recipient of the 2018 Amazon Research Awards. He is also the Associate Editor-in-Chief of *Pattern Recognition*, an Editorial Board Member of *Computer Vision and Image Understanding*, and on the Advisory Board of *Journal of Imaging*.