ROBUST FITTING OF MULTIPLE CURVES

QIAO YU

(B. Eng., Shanghai Jiaotong University)

(M. Eng. Shanghai Jiaotong University)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2004

Acknowledgments

I would like to express my deepest gratitude to my supervisor, A/Prof. Ong Sim-Heng, who has guided me through my Ph.D. study. Without his gracious encouragement and generous guidance, I would not be able to finish the work here. His unwavering confidence and patience have aided me tremendously. His wealth of knowledge and accurate foresight have greatly impressed and benefited me. I am indebted to him for his constructive criticisms and countless patient hours spent in reviewing this thesis.

I am grateful to Mr. Francis Hoon, lab technician in the Vision and Image Processing Laboratory for providing me with technical support.

I also wish to thank Mr. Wang Yi-Zhou, Mr. Chen Kui-Ran, who have in one way or another given me their kind help. Not forgetting my friends and colleagues, I would like to express my thanks to many others in Vision and Image Processing Laboratory for making the everyday work so enjoyable. I enjoyed very much the time spent with them. I am also grateful to the National University of Singapore for the research scholarship.

Finally, this thesis would not have been possible without the love, patience and support from my family. The encouragement from my parents and my brother has

been invaluable. I would like to dedicate this thesis to them and hope that they will find joy in this humble achievement.

Qiao Yu

October, 2003

Contents

List of Figures

List of Tables

Summary

The detection of multiple curves in a noisy image is an important task in computer vision. Linearly parameterizable models such as straight lines, circles and ellipses have attracted a lot of attention because they efficiently describe large parts of man-made objects. Robust estimation techniques are needed to successfully fit linearly parameterizable functions to edge data while ignoring outliers in the data. A model-fitting algorithm should be robust against an arbitrarily high percentage of outliers, capable of handling an unknown number of models, insensitive to noise, and most importantly, not miss meaningful models while avoiding false detections.

A variety of techniques have been developed to extract multiple linearly parameterizable models. The Hough Transform is a well-established shape detection method. However, the high computational cost and low accuracy due to the high dimensionality and quantization of the parameter space degrade the performance of this approach. Another serious problem is that it often yields many wrong detections in complex and noisy images. Robust regression techniques, e.g., the least median of squares, are other important statistical tools frequently employed in computer vision for fitting a model to noisy data. They work well when the number of outliers does not exceed 50%. The LS method, while achieving opti-

mum results when the noise distribution is Gaussian, is unreliable for the fitting of multiple models because outliers may severely affect performance.

The objective of this thesis is to develop a robust approach for fitting multiple curves, which is robust against more than 50% outliers and able to extract *a priori* unknown number of occluded or touching models.

Firstly, after a brief mathematical description of the linearly parameterizable curves estimation problem, we develop a novel *shulidu*-based data cloud center estimator. A general framework for fitting multiple touching or occluded linearly parameterizable curves is presented. A robust single-model extractor is used for detecting single models by employing a region trimming scheme that is robust against an arbitrarily high percentage of outliers. The underlying models can then be extracted sequentially by repeatedly applying this extractor to the updated edge data set. A model verifier is used to evaluate the validity of the models detected by single-model extractor such that false models are effectively excluded.

Secondly, a connectivity-based multiple curve estimator is devised by introducing pixel connectivity into the general framework. The curve fitting turns to be a task of the search for meaningful arcs. The criteria for inliers and valid arcs are discussed in detail. A search strategy based on both pixel connectivity and region trimming scheme is devised for detecting outliers. False detection is effectively avoided due to the intra-connectivity feature of valid arcs. Restricting the search of curves within the intra-connected data subsets significantly reduces the computational complexity.

Thirdly, circle fitting is accomplished by using the connectivity-based multiple

curve fitting algorithm. The appropriate error function for circle fitting as well as the criteria for valid circular arcs are proposed. The LS estimator for circular model fitting is discussed. A new efficient method is developed for finding the endpoints of circular arcs and the angles subtended by circular arcs. Multiple occluded circles in noisy synthetic and real images are well reconstructed without false detections.

Fourthly, the connectivity-based multiple curve fitting algorithm is applied to ellipse fitting. Error function for ellipse extraction is chosen. The search of the endpoints of elliptic arcs and computing the angles subtended by elliptic arcs are achieved with an efficient method. The estimator successfully extracts *a priori* unknown number of multiple ellipses from synthetic and real examples in the presence of more than 50% outliers and severe salt and pepper noise.

Chapter 1

Introduction

1.1 Motivation and research objectives

The detection of curves in a noisy image is an important task in computer vision. For example, the accurate identification of spherical objects is crucial in many areas of image analysis [11]. It is especially important in industrial applications such as automatic inspection and assembly. In the semiconductor device inspection, the circular balls on the package need to be located and measured with high precision such that good devices and bad devices are discriminated. In the automotive industry many circular components are used. These manufactured parts also need to be checked. In many situations round objects are viewed obliquely and ellipse detection is thus required.

Linearly parameterizable models such as straight lines, circles and ellipses are of great importance because they efficiently describe large parts of man-made objects [15], [34], [46], [37]. Robust estimation techniques are needed to successfully

fit linearly parameterizable functions to edge data while ignoring outliers in the data [35], [43]. Three requirements for a desired parametric model-fitting algorithm have been proposed [8]:

1. The fitting algorithm should achieve the desired results, i.e., it should be able to extract all meaningful curves while avoiding any false detection.

2. The approach must be robust against an arbitrarily high percentage of outliers in the input data.

3. The algorithm must be capable of handling an unknown number of curves.

The Hough transform (HT) is a well-established circle detection method [21]. It is an efficient tool for the detection of multiple curves and is robust with respect to outliers. However, the high computational cost and low accuracy due to the high dimensionality and quantization of the parameter space [20] degrade the performance of this approach. Another serious problem is that it often yields many wrong detections in complex and noisy images.

In order to alleviate these problems, improved HT schemes have been proposed, for example, the randomized Hough transform (RHT), the probabilistic Hough transform (ProbHT), the dynamic combinatorial Hough transform (DCHT) [23], and their variants. However, almost all of them focus on easing computational complexity by random sampling or by reducing the search area in the image space, but are less effective in reducing wrong detections. Furthermore, unlike the least squares (LS) method, the HT transform and its variants cannot offer an optimal fitting when the inliers of the model are corrupted by Gaussian noise.

Robust regression techniques, e.g., the least median of squares (LMedS) [35], are other important statistical tools frequently employed in computer vision for fitting a model to noisy data. They work well when the number of outliers does not exceed 50% [35]. However, robust statistical methods usually have to make a tradeoff between accuracy and robustness, and are of low computational efficiency. The second requirement above implies that the breakdown point of a model-fitting technique should be more than 50%. Unfortunately, most robust methods have a breakdown point below this [35]. Stewart's MINPRAN estimator is able to handle more than 50 percent outliers but requires a fixed outlier distribution, which is not feasible in practice [45].

The least squares (LS) method has been widely adopted in computer vision and pattern recognition. It achieves optimum results when the noise distribution is Gaussian. However, the method becomes unreliable for the fitting of multiple models because outliers may severely affect performance. Therefore the key issue is to find a modified LS approach that is robust against outliers. M-estimators can be regarded as a re-weighted LS methods [16], [19], [1]. The major weakness of the re-weighted LS method is that the fitting results depend to a large extent on the quality of the initial guess [35]. To avoid the initialization problem of the M-estimator, Chen [7] applied a statistical windowing technique to obtain an initial estimate. However, a meaningful initial guess requires that the data come from the same model, which is difficult to guarantee. A parametric model fitting framework based on inlier characterization has been proposed for the fitting of multiple parametric models [8]. Its major shortcoming is that inlier characterization of the

input data, e.g., the covariance matrix, is required. Furthermore, a poor initial guess will lead to a non-meaningful estimate.

Based on the above analysis, there is almost no estimation method that can satisfy all three requirements for a desired estimator [8]. Our research goal in this thesis is to develop an approach for multiple model estimation that satisfies all three requirements and is practical in real applications.

It is worth pointing out that in many real machine vision applications, the underlying curves to be detected are characterized by arcs, which are actually intra-connected subsets that fit the correct curve models. Since most model estimation methods do not consider the geometric relationship among the candidate points, a common failing is the production of a false model. Edge points that fit the false model in this kind of images are considered to satisfy the criterion for curve validity even though they may actually belong to other geometric shapes.

Our proposed estimator is motivated by some observations of how the human visual system can easily classify the data and ignore the outliers while fitting multiple models. First, approximate estimations are made by examining data taken over the entire image, i.e., it is a global process. Second, closely connected edge points are likely assumed to come from the same model. Third, the full set of inliers is not needed to extract meaningful models, only the inlier subset consists of meaningful arcs are necessary.

This thesis presents a general multiple-model estimation framework that is robust against an arbitrarily high percentage of outliers and is able to extract an unknown number of touching or occluded models. In our framework, an image

pre-partitioning technique is used to classify edge data into groups based on some *a priori* application dependent assumptions. A single-model extractor, which is actually an iterative weighted LS approach, is then applied to each group of data for the detection of underlying curves formed by edge points in this group. It starts from an initial global model estimate based on the data in each group. The most qualified outliers that have the largest deviation are first kept out of the following estimation. The most significant subset from the remaining edge data is taken as the updated data set for the next round of model estimation. In this way, this region trimming scheme drives the estimate from a initial global model to converge to a final local desired model by iterative elimination of detected outliers. All underlying models in the image can be reconstructed sequentially by repeatedly applying the single-model extractor to the edge data that were excluded in the previous fitting. A model verifier is applied to all extracted models to determine whether they are desired models or pseudo models. The only condition that assures the success of our approach in handling outliers is that there should be enough inliers to construct the meaningful models.

The connectivity-based multiple curve estimator, an example of the general multiple-model estimation framework, is realized by using pixel connectivity as one of the most important features of meaningful arcs. The connectivity-based single-model extractor and model verifier are well developed. The appropriate criteria for curve inliers and valid arcs are proposed by considering the connectivity among edge points from the same meaningful curves.

The application of our connectivity-based multiple curve estimator into multiple

circle and ellipse fitting shows the following advantages:

1. It can extract all meaningful curves while producing no false detection.
2. It is robust against an arbitrarily high percentage of outliers in the input data.
3. It is capable of handling an unknown number of same kind of curve models.
4. It works well with the edge data heavily corrupted by noise.
5. It effectively fits occluded or touching curves.

1.2 Thesis Contributions

The major contributions of this thesis are summarized below:

- **Multiple model estimation:** A novel general framework for fitting multiple touching or occluded linearly parameterizable curves is developed. This framework presents a new approach for handling cases with a high percentage of outliers. A robust single-model extractor employing a region trimming scheme is used for outlier detection. The underlying models can be extracted sequentially by repeatedly applying this extractor to the updated edge data set. The detected models are then evaluated with a model verifier to determine their validity. The robustness against an arbitrarily high percentage of outliers is achieved by region trimming scheme. This framework facilitates the handling of an *a priori* unknown number of meaningful models by updating data set during the iterative estimation process. Effective fitting of

multiple occluded or touching curves is obtained by the initial estimate starting from the entire data set as well as the treatment of other model inliers as outliers in the single-model extractor. False models are effectively excluded by the model verifier.

- **Connectivity-based multiple curve fitting:** The introduction of pixel connectivity in the general estimation framework generates an efficient multiple curve fitting approach, which is an application of the general framework and inherits all of its advantages. The estimation of curve models is performed by a search of meaningful arcs instead of a set of single inliers. The intra-connectivity feature of valid arcs effectively prevent from false curve detection comprised by unrelated data. It improves the robustness against noise by excluding unconnected noisy data. The computational complexity is significantly reduced by limiting the search of curves within the intra-connected data subsets.

- **Fitting circles:** The connectivity-based multiple curve fitting algorithm is applied to circle fitting [40]. The appropriate error function for circle fitting is proposed. The criteria for valid circular arcs are discussed. The LS estimator used in the single-model extractor for circular model fitting is presented. A new efficient connectivity-based method for finding endpoints of circular arcs and calculating the angles subtended by circular arcs is developed. The performances of this estimator in both synthetic and real applications of fitting multiple touching or occluded circles in the presence of heavily corrupted

noisy data and much more than 50% outliers are demonstrated.

- **Multiple ellipse estimation:** The problem of ellipse fitting is more complicated that of circle fitting. The circle is rotation invariant while the ellipse has major and minor functions and thus generates different conic models rotated by different angles. The translation of different ellipse models is required in the procedure of ellipse fitting. The estimation of multiple occluded or touching ellipses is achieved by the application of the connectivity-based multiple curve estimator. Error function for ellipse extraction is selected. A new efficient method based on pixel connectivity for the search of endpoints of elliptic arcs and computing the angles subtended by elliptic arcs is devised using location-dependent threshold values. The results of applying this estimator to fitting multiple occluded or touching ellipses in synthetic and real images are analyzed.

1.3 Organization of the thesis

The subject of this thesis focuses on the robust algorithm for fitting multiple touching or occluded curves in the presence of noise and more than 50% outliers. It is organized in six chapters followed by appendices and a bibliography.

In Chapter 2, we give a brief review of some previous works for multiple curve fitting, such as Hough transform (HT) and its variants, Robust regression methods and Least squares (LS) based algorithms. The relationships between these approaches and our work are highlighted.

Chapter 3 describes the problem of fitting multiple linearly parameterizable curve model. The criteria for testing the validity of extracted models are then discussed. A new concept, *shulidu*, which describes the relationship between a single element and the entire set, is developed and applied to the data cloud center estimation. We then present a general framework for effective estimation of multiple-model that is robust against an arbitrarily high percentage of outliers and is able to extract an unknown number of touching or intersecting models. In our framework, an image pre-partitioning technique is used to classify edge data into groups based on some application dependent assumptions. The region trimming scheme is proposed, which is an effective way to deal with outliers. A single-model extractor, which is actually an iterative weighted LS approach, is then applied to each group of data for the detection of underlying curves formed by edge points in this group. All underlying models in the image can be reconstructed sequentially by repeatedly applying the single-model extractor to the edge data that were excluded in the previous fitting. A model verifier is applied to all extracted models to determine whether they are desired models or pseudo models. The advantages of this framework are discussed.

In Chapter 4, a connectivity-based method for curve fitting is developed. The motivation of introducing pixel connectivity into the general framework for curve fitting (Chapter 3) is discussed. The criteria for intra-connected subsets and valid arcs are presented in details. A search strategy based on both pixel connectivity and region trimming scheme is devised for excluding outliers. The connectivity-based single-model extractor is then developed for fitting meaningful arcs. The

model verifier is designed to test the meaningfulness of models extracted by single-model extractor using the intra-connectivity feature of arcs.

Chapter 5 presents the application of connectivity-based multiple curve estimator to circle fitting. We develop the appropriate error function for circle fitting and criteria for valid circular arcs. The mathematical model of the LS estimator for circle extraction is presented. A new effective method for detecting endpoints of circular arc and the angles subtended by circular arcs is proposed. The advantages of our estimator for circle estimation are analyzed in detail. The results of using our estimators for fitting circles in real examples are explained.

In Chapter 6, the performance of using connectivity-based multiple curve estimator for ellipse estimation is evaluated. The translation between both nominal and general elliptic models is discussed. The criteria for valid elliptic arcs are described. We then develop an efficient approach to search for endpoints of elliptic arcs and calculate the angles subtended by the elliptic arc. A comparison between our estimator and other methods is presented based on the results of our estimator on some synthetic images. The results of applying our estimator to real multiple ellipse detection are given.

Chapter 7 summarizes our work in this thesis and discusses future research. Supplementary materials, including the derivation of an approach for finding endpoints of both circular and elliptic arcs as well as the calculation of angles subtended by circular and elliptic arcs, are included in the appendices.

Chapter 2

Literature Review

In this chapter, we give a brief review of previous work in multiple curve fitting. The relationship with our work is highlighted to emphasize our motivations and contributions.

2.1 Robust fitting of multiple curves

The detection of multiple curves in a noisy image is an important task in computer vision. Linearly parameterizable models such as straight lines, circles and ellipses are of great importance because they efficiently describe large parts of man-made objects [15], [34], [46], [37]. Robust estimation techniques are needed to successfully fit linearly parameterizable functions to edge data while ignoring outliers in the data [35], [43]. Three requirements for a desired parametric model-fitting algorithm have been proposed [8]:

1. The fitting algorithm should achieve the desired results, i.e., no false models

will be extracted.

2. The fitting algorithm must be robust against an arbitrarily high percentage of outliers in the input data.

3. The algorithm must be capable of handling an unknown number of curve models.

Since edge data classification and parameter estimation strongly influence each other, it is difficult to satisfy the above mentioned requirements for multiple curve fitting [8]. On the one hand, if the data is correctly classified to be the inliers of a certain model, it would be easier to extract the accurate curve model because it avoids the difficult problem of outliers. On the other hand, if the parameters of the curve model are determined, then it would be straightforward to evaluate the data with the available curve model. The difficulty of the co-dependence of inlier classification and parameter estimation requires a robust multiple curve fitting algorithm to be a "classify-while-fit" approach [8].

Various curve fitting methods have been proposed to solve this problem, such as the Hough Transform (HT) and its variants, Rough Regression methods and least squares (LS) based approaches etc.. However, none of them meets all the three requirements of a desired multiple curve estimation algorithm discussed above.

2.2 The Hough transform

The Hough transform is a well accepted shape detection method [21]. It is robust against outliers and it can handle multiple models. In this method, each edge point

in the image space is first transformed into a parametric curve of the parameter space. Then an accumulator corresponding to a cell of the parameter space is used to count the number of curves that pass through this cell. Finally, the parameter coordinates of those cells with a local maximum of scores are selected as the estimation of model parameters in the image space. In this way, it attempts to solve a continuous problem with a discrete method via a voting scheme [8]. As a consequence, it cannot achieve the optimally accurate estimation of model parameters due to the finite quantization of the parameter space [20]. What is more, model estimation with the Hough transform is often of high computational complexity and storage requirement because of the high dimensionality of parameter space.

Many modified Hough transforms have been studied to alleviate some of the shortcomings of the standard Hough transform (SHT). They can be classified into two major categories: probabilistic and non-probabilistic Hough transforms [23]. The probabilistic HT method is characterized by a many-to-one mapping and selecting only a small subset of data using random sampling for parameter-space cell voting.

The SHT is a typical non-probabilistic HT, which is the most accurate method, but has the disadvantages of low computational speed and large storage requirements [21]. The hierarchical Hough transform (HHT) was introduced by Illingworth *et al.* for line fitting [38]. In the HHT, the image is separated into small subimages and the HT is applied to each subimage. The curves are extracted from subimages. The accumulation procedure of the HT stops when the maximum value of the accumulators reach a predefined threshold. As a consequence, a group of

models is detected for each subimage. The line segments detected from subimages are grouped hierarchically level by level with the HT. The size of the accumulator array can be kept small due to the introduction of subimages. It thus reduces significantly the storage requirement.

The combinatorial Hough transform (CHT) was proposed by Ben-Tzvi and Sandler [3]. It is a fast HT algorithm for line fitting. In this algorithm, two edge data of the image are used to calculate the line parameters. One vote in a (ρ, θ) cell in the parameter space is determined by each pair of two image edge data. The selection of image pixel pair is limited in an image segment. This greatly reduces the number of pixel pair combination. The CHT is faster than the SHT. However, this method may miss some lines if the segment size is too small. If the segment size is enlarged, the computation speed is lower. Another disadvantage of the CHT is that it depends more upon the distribution of the image edge data than other HT methods.

The curve fitting Hough transform (CFHT) was suggested by Liang [31]. In this method, only a small neighborhood of edge points in the image space is used in the voting procedure in the parameter space to fit the curve parameters. If the edge points in the neighborhood fit the model within a given tolerance, the model is mapped to a single point in the parameter space. The process of fitting is performed to each $M \times M$ window in the edge image. Then the parameter space is examined to extract the curve model. It is actually a many-to-one mapping from a small neighborhood of edge points in the image to a single point in the parameter space. Therefore the CFHT is the fastest non-probabilistic HT [23]. However, the

CFHT seems to be one of the most inaccurate methods of HT. It fails to find some obvious lines due to the selection of edge points in a small neighborhood [32].

Recently, several probabilistic HT methods have been developed. The basic idea of these methods are the random sampling of edge data in the image and the mapping of a group of data samples in the image space to a point in parameter space. This mechanism significantly reduce the computation time and memory storage. The RANSAC approach proposed by Fischler and Firschein is one of the earliest random methods [13]. The basic idea of the RANSAC method is that n pixels are randomly selected for obtaining an estimate of curve parameters, and then other pixels are used to test the estimate.

Kiryati *et al.* proposed the probabilistic Hough transform (ProbHT) [27], in which only a small randomly selected data subset instead of entire data set is used for the HT. Because the size of the subset is quite small, the time complexity of the HT is reduced considerably. However, it is still a one-to-many mapping. Each data in the subset is mapped to a curve in the parameter space by the SHT.

The randomized Hough transform (RHT) proposed by Xu *et al.* is a basic probabilistic HT for curve fitting [49], [28]. In this method for fitting curve models with n parameters, n pixels are randomly selected in the image space and mapped into one point of the parameter space. It is quite different from the SHT in which one pixel of image space is transformed into an $n-1$ dimensional hypersurface of parameter space. In other words, the mapping is a many-to-one in the RHT in contrast to one-to-many in the SHT. Compared with SHT, it has the advantages of small storage, high speed, infinite accumulator space and arbitrarily high

resolution [48].

The dynamic RHT (DRHT) method is an iterative process of two RHTs for line fitting [23]. In the first iteration stage the original RHT is performed until the value of some accumulator cell reaches the accumulator threshold. During the second iteration, the feature points in the parameter space is determined by collecting the edge data that are close to the line extracted from the first iteration. The RHT is performed on these data near the line with higher accumulator resolution and higher threshold in the zeroed accumulator space. When the accumulator threshold has been exceeded by the maximum value of accumulators, the line is found. The time complexity is reduced due to the lower resolution and lower threshold in the first iteration and the small number of edge data investigated in the second iteration.

A novel windowing version of the RHT, the window RHT (WRHT), was proposed by Kälviäinen *et al.* [24]. In the WRHT, one edge point is selected randomly, which represents the location of window. A fixed size window is determined, which is actually a neighborhood of the selected edge point. Then certain model estimation techniques are employed in this window to extract curves. Only the models supported by enough data samples in the window are accepted to update the accumulator space. The procedure for random window selection and model extraction in the window is repeated until the maximum score in the accumulator space reaches a predefined threshold. The model parameters represented by the cell with maximum score is then extracted. The most important constraint of the WRHT is window size. Desired detection accuracy require a large enough window

size while the average separation distance of adjacent curves limits the window size.

In order to solve the problem of window size constraint, Kälviäinen *et al.* [24] presented the improved version of the WRHT, the random window RHT (RWRHT). The window size of the RWRHT is changed randomly. Like the WRHT, an edge point representing the location of window is first chosen randomly. Then the size of the window located here is also randomized. The RHT procedure is applied to this window for model extraction. The maximum amount of the random sampling in this window is a randomly selected number R, where R could be a function of window size. During the RHT procedure in this window, if the curve is fitted when the number of random sampling is less than R, the extracted model is stored and start the search for other curves by forming a new random window; otherwise, this window is deserted and a new random window is selected for model estimation. Compared with the basic RHT, the random sampling is limited to a smaller random window and the number of random sampling is also restricted by a random value in the RWRHT.

Ben-Tzvi proposed the dynamic combinatorial Hough transform (DCHT) for line fitting [2], [29]. In the DCHT, the line parameter is (ρ, θ). At first, a seed point is randomly selected among feature points. Every pair of edge data with the seed point is accumulated into a single value in a θ-histogram. The process continues until the height of some seed points in θ-histogram reaches a predefined threshold. In the next step, a detected line together with edge data lying on it is removed. The whole process is repeated until all points have been removed. The

speed of this method is obviously faster than the SHT or even the CHT.

2.3 Robust regression techniques

Regression analysis is a basic statistical tool frequently employed in computer vision for fitting a model to noisy data. Three criteria are usually used to evaluate a regression method [35]:

1. Relative efficiency: the ratio between lowest achievable variance of the estimated parameters and the actual variance of the method.

2. Breakdown point: the ratio between the largest amount of outlier that make the estimate meaningful and the total number of edge data.

3. Time complexity: the computation time of the method for model estimation.

Based on these criteria, three requirements that a good robust regression method should satisfy are stated [35]:

1. Reliability when edge data are contaminated by various types of noise.

2. Large breakdown point, i.e., robustness against a high percentage of outliers.

3. A time complexity not much greater than that of the LS method ($O(n^2)$).

Many statistical regression methods have been proposed that satisfy some of the above requirements. They can be classified into M-estimators, R-estimators and L-estimators [19].

The M-estimators are the most popular robust regression methods for model estimation. They estimate model parameters by minimizing the sum of a symmetric, positive-definite function $l(r_i)$ of the residual r_i, i.e.,

$$\min \sum_i l(r_i). \tag{2.1}$$

The minimization problem is usually converted into a weighted LS problem. The weights of the LS method depend on the function $l(.)$ The reliability of estimates depends heavily on the quality of the initial guess.

The R-estimators use the idea of ordering the set of residuals. A R-estimator proposed by Jaeckel [22] estimates the curve parameters by solving the minimization problem

$$\min \sum_i a_n(R_i) r_i, \tag{2.2}$$

where r_i is the residual; $a_n(R_i)$ is a score function of the rank R_i of the residual r_i in the ordered list. An important advantage of R-estimators over M-estimators is that the result of R-estimators is independent from the variance of the noise.

The L-estimators are based on the linear combinations of order statistics [19]. The median and α-trimmed-mean based methods are popular L-estimators. Various simulations and experiments show that L-estimators produce less satisfactory estimates than the M- and R-estimators [18].

Although these three kinds of estimators (M-, R- and L-estimators) are robust against various distributions, their breakdown points are less than $1/(p+1)$, where p is the number of model parameters estimated in regression methods [16].

The least median of squares (LMeds) robust regression method proposed by Rousseeuw [41] is a popular regression method that achieves a breakdown point of 0.5. The model parameters are estimated by solving the nonlinear minimization problem

$$\min \underset{i}{\text{med}}\, r_i^2. \tag{2.3}$$

The estimate of LMeds is actually the model with the minimum value for the median of squared residuals r_i^2 for all data samples. The LMeds estimate is achieved by a nonlinear search in the space of the median of the squared residuals of all possible models generated by the data samples. The projection pursuit technique is employed to reduce a multidimensional regression problem to one dimension. Mode-estimation is used in the search of the median. The time complexity is, not surprisingly, very high. The time complexity required for the basic LMeds method is $O(n^{p+1} \log n)$ for the estimation of model with p-tuple parameters [35].

To alleviate the high computational complexity, Monte Carlo type speed-up techniques are employed in the case that a probability of error $Q \ll 1$ is allowed [35]. Then the probability that all m different p-tuple data samples selected randomly contain at least one or more outliers is

$$P = [1 - (1 - \epsilon)^p]^m, \tag{2.4}$$

where ϵ is the ratio of outliers in the sample data and p is the number of parameters of the model. Then $1 - P$ is the probability that at least one p-tuple data sample from the chosen m data groups has all uncorrupted samples and the correct parameter values can be recovered. The minimum acceptable value for m is the

integer rounded upward by the solution of $P = Q$. It is independent of the size of data n. The time complexity is thus significantly reduced to $O(mn \log n)$ [35].

Stewart proposed a new robust estimator, minimize the probability of randomness (MINPRAN), which is able to handle more than 50 percent outliers [45]. Unlike other techniques relying on the a known error bound for the good data, MINPRAN assumes that the bad data are uniformly distributed. Based on this assumption, MINPRAN extracts an accurate estimate of the model from random sampling data and finds the inliers to this estimate that are least likely to occur randomly. The time complexity for MINPRAN is $O(n^2 + mn \log n)$, where n is the amount of data sample and m is the number of random samples.

Boyer and Mirza have developed a robust sequential estimator(RSE) that accommodates the assumption that the noise model follows a t distribution, which is a more realistic model than the Gaussian distribution [6], [36]. It can efficiently reconstruct the 3D surfaces in the noisy image from a fixed library of models whose number and complexity are application dependent.

2.4 Fuzzy Clustering Technique-based methods

The fuzzy c-means algorithm and various fuzzy c-shells clustering techniques are also good choices for extracting circular, elliptic and straight lines in images [9], [10], [50]. The data samples are clustered into different groups based on fuzzy membership with each group representing a valid curve model. Curve model parameters are then estimated for the clustered data,.

Fuzzy c-means algorithms (FCM) is a square-error clustering method, where the sum of weighted distances of the points from the cluster prototype is minimized [10]. The weight is calculated with the fuzzy membership of the sample to the cluster. The use of different norms make it available to detect various shapes such as spherical or elongated clusters. However, it can only be applied to the detection of clusters whose prototype is a point and would fail to detect curves which have hollow interiors [9].

Fuzzy c-shells (FCS) algorithms utilize shells, which are p-dimensional hyper-spherical surfaces, as the cluster prototype [9]. The distance of a sample from the hyper-spherical surface is defined as the distance measurement of a sample from the cluster. The FCS can be used to detect circular shapes in two-dimensional digital images. The hyper-spherical shell prototype, S_I, is the set

$$S_I(\vec{v}, r) = \{\vec{x} \in R^p \mid (\vec{x} - \vec{v})^T I(\vec{x} - \vec{v}) = r^2\}, \tag{2.5}$$

where p is the dimension of the sample and curve spaces, $\vec{v} \in R^p$ is the cluster center, $r \in R^+$ is the radius, $\vec{x} \in R^p$ is the sample and I is a $p \times p$ identity matrix [10]. The distance of a point $\vec{x}_k$ from the ith shell $S_I(\vec{v}_i, r_i)$, $(D_{ik})^2$, is

$$(D_{ik})^2 = ([(\vec{x}_k - \vec{v}_i)^T I(\vec{x}_k - \vec{v}_i)]^{1/2} - r_i)^2. \tag{2.6}$$

The FCS seeks the best shell clustering by minimizing the functional

$$J_s(U, V, R) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m (D_{ik})^2, \tag{2.7}$$

where c is the number of clusters, n is the amount of samples, u_{ik} is the fuzzy membership of the kth sample in the ith cluster and exponent $m \in [1, \infty)$. The fixed point iteration scheme is used for searching the minimum of functional $J_s(U, V, R)$.

The adaptive fuzzy c-shell (AFCS) algorithm is proposed to detect ellipsoidal shell clustering in two-dimensional digital images [9], [10]. The shell prototype is generalized by introducing the matrix A, which is a $p \times p$ symmetric positive definite matrix that stores the information about the eccentricity and orientation of the ellipsoid. The prototype used by AFCS is

$$S_I(\vec{v}, r, A) = \{\vec{x} \in R^p \mid (\vec{x} - \vec{v})^T A(\vec{x} - \vec{v}) = r^2\}. \tag{2.8}$$

The distance of a point $\vec{x}_k$ from the ith shell $S_I(\vec{v}, r, A)$, $(D_{ik})^2$, is then defined as

$$(D_{ik})^2 = ([(\vec{x}_k - \vec{v}_i)^T A_i(\vec{x}_k - \vec{v}_i)]^{1/2} - r_i)^2. \tag{2.9}$$

The performance functional of the AFCS also changes to

$$J_s(U, V, R, A) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m (D_{ik})^2, \tag{2.10}$$

The AFCS seeks the minimum of $J_s(U, V, R, A)$ to optimize the cluster parameters U, V, R and A [9], [10]. The fixed point iteration scheme is applied to finish the task. At each iteration, Newton's methods is used to search the solution of nonlinear equations.

FCS and AFCS work well for circle and ellipse fitting in two-dimensional images which includes only circular or elliptic shapes. However, they have some drawbacks that limit their usefulness in the extraction of circles and ellipses from more complex images.

- FCS and AFCS cannot avoid the false detections that occur in HT-based methods, which are comprised by unrelated samples. The reason for false fitting is that the connectivity of samples is not considered in FCS and AFCS.

- The search strategy of FCS and AFCS is to select as inlier the data with small deviation from current estimated model [9], [10]. Hence they are sensitive to outliers since the estimated model may be pulled off significantly from the expected models by the outliers. In that case, the data with small deviation from current models may not necessarily be inliers of expected models.

- FCS and AFCS depend heavily on the quality of the initial guess. The data set is divided into several subsets during initialization. These algorithms require good initializations for the memberships as well as model parameters. If the initial partition is not close enough to the expected partition, the solutions may not converge [9], [10].

- The number of clusters is fixed in FCS and AFCS [9], [10]. They cannot detect an *a priori* unknown number of models.

- FCS and AFCS have difficulty in extracting partial shapes [9], [10]. They cannot extract circles and ellipses in the digital images that include different geometric shape due to the constraint of the fuzzy membership $\sum_{i=1}^{c} u_{ik} = 1$.

- They may not work well in the presence of considerable overlap or inclusions [9], [10].

- They are not robust against noise [9], [10].

- The Newton methods for solving the nonlinear equation will lead to local minima in AFCS. They may not converge to a global optimum if the initializations are not good [9], [10].

- The computational burden is heavy due to the search of nonlinear equations solutions.

The fuzzy curve tracing (FCT) algorithm extracts a smooth curve from unordered noisy data that describe the local and global features [50]. It focuses more on edge data thinning techniques and describes the local features with smooth curves. The algorithm consists of three steps: data thinning, region ordering and curve parameter estimation. In the first stage, the input data are clustered into different regions using the FCM algorithm. Then the region is ordered by determining the closeness of one region to its neighboring regions using the averages of the class membership values. A graph is formed by linking the region to its two closest regions. The initial representation of the curve is obtained by removing small loops in the graph. In the parameter estimation step, the initial representation converges to a local optimum smooth curve with a smooth constraint. FCT is powerful in dealing with unordered data, compared with polygon approximation [50]. However, FCT is a local shape description technique. One single circular or elliptic models may be represented by piecewise smooth different nonlinear curves with FCT. The circular or elliptic representation cannot be guaranteed. FCT represents the local data distribution with smooth curves. Therefore it is not robust against noise. The presence of outliers will cause the local smooth curve to drift away from the true model though both true and false extracted curves are smooth.

Principal curves are smooth one-dimensional curve passing through the middle of a p-dimensional data cloud and provide a nonlinear summary of the data. They are nonparametric, and their shape is suggested by the data [17] [25]. The principal

curve does not intersect itself and has finite length inside any bounded subset. It is a self consistent curve, which means that each point of the curve is the average of all the data that project there. It is obvious that the principal curve is a concise summary of local data distribution. The characteristics of self consistency makes it sensitive to the outliers because the existence of outliers may bias the average of the data on a single projection line from the expected place. Naturally, the shape of the principal curve also changes. The principal curve works well only for some special data distribution density, like radially symmetric densities and uniform density. The summary of local data distribution may cause the phenomenon that a global circular shape will be represented by several smooth non-circular principal curves. Thus it may fail to extract the global data distribution shape.

2.5 LS-based methods

The least squares (LS) method has been widely adopted in computer vision and pattern recognition. It achieves optimum results when the noise distribution is Gaussian. Apart from the optimality in dealing with Gaussian noise, the LS algorithm also has the advantage of significantly lower computational complexity ($O(n^2)$) compared with those of above mentioned HT-based approaches and robust regression methods [8].

However, the LS method becomes unreliable when fitting multiple models because outliers may severely affect performance. Therefore the key issue is to find a modified LS approach that is robust against outliers. Re-weighted LS methods are

proposed to solve the problem of outliers [16], [19], [1]. The trimmed LS method is robust against outliers only if the number of outliers in the data sample is known *a priori* [42]. This condition is not possible for most applications. The robust regression methods, M-estimators, can also be regarded as re-weighted LS methods because the weight of data in M-estimators are determined by a symmetric, positive-definite function of residuals [35]. The major weakness of the re-weighted LS method is that the fitting results depend to a large extent on the quality of the initial guess [35]. If the initial guess is biased, the estimators have a large probability to extract a non-meaningful model. Hence, these re-weighted LS estimators can only be used in applications in which a good initial guess is available.

To avoid the initialization problem of the M-estimator, Chen [7] proposed a data-driven intermediate level feature extraction algorithm. In this method, he applied a statistical windowing technique to obtain an initial estimate. A search region (which grows during the iterative estimation) is then determined. All data in the search region are verified for their goodness-of-fit to the current estimated model. The current model is then re-estimated with the sample of favorable goodness-of-fit value. Other data with poor goodness-of-fit are rejected as outliers. In the iterative process, the algorithm drives the estimate to converge to a local optimum model. Multiple models can be extracted by repeatedly applying the algorithm to updated data sample set that exclude the inliers of previous estimates. However, a meaningful initial guess of this method requires that the data come from the same model, which is difficult to guarantee in the applications of multiple model estimation.

A parametric model fitting framework based on inlier characterization has been proposed for the fitting of multiple parametric models [8]. It consists of a parameter estimation module based on a general LS method for model extraction and of a error propagation procedure. In the error propagation procedure, a data snooping technique is employed to exclude outliers, in which all residuals of data sample are analyzed based on the statistical output of the LS method. Based on the the result from error propagation, the model parameters are complemented with a precision estimate, inliers are assigned to the fitted model and the *a priori* unknown number of models are extracted. A major advantage of this method is that no assumption on the outlier distribution is required. Its major shortcoming is that inlier characterization of the input data, e.g., the covariance matrix, is required. Furthermore, a poor initial guess may still lead to a non-meaningful estimate.

2.6 Conclusion

The Hough transform (HT) is a well-established curve detection method [21]. It is an efficient tool for the detection of multiple curves and is robust with respect to outliers. However, the high computational cost and low accuracy due to the high dimensionality and quantization of the parameter space [20] degrade the performance of this approach. Another serious problem is that it often yields many wrong detections in complex and noisy images.

In order to alleviate these problems, improved HT schemes have been proposed, like RHT, ProbHT, DCHT [23], and their variants. However, almost all of them

focus on easing computational complexity by random sampling or by reducing the search area in the image space, but are less effective in reducing wrong detections. Furthermore, unlike the least squares (LS) method, the HT transform and its variants cannot offer an optimal fitting when the inliers of the model are corrupted by Gaussian noise.

Robust regression is another important statistical tool frequently employed in computer vision for fitting a model to noisy data. However, robust statistical methods usually have to make a tradeoff between accuracy and robustness, and are of low computational efficiency. The second requirement above for a good robust regression approach implies that the breakdown point of a model-fitting technique should be more than 50 percent. Unfortunately, most robust methods have a breakdown point below this [35]. Stewart's MINPRAN estimator is able to handle more than 50 percent outliers but requires a fixed outlier distribution, which is not feasible in practice [45].

FCS and AFCS are curve fitting algorithms based on fuzzy clustering technique. The ignorance of geometric relationship among samples may produce wrong detections with FCS and AFCS. The inlier search strategy makes them sensitive to the outliers and noises. FAC and AFCS depend crucially on the quality of initial guess and will not converge if the initialization is not good enough. They may work poorly in the presence of considerable overlap or inclusions of models. FCT and principal curve-based curve fitting algorithms provide piecewise smooth curves passing through the middle of the data cloud for the summary of local data features. They are sensitive to the local data distribution and may miss the global

shape feature due to the occurrence of outliers and noise.

The LS method, while achieving optimum results when the noise distribution is Gaussian, is unreliable for the fitting of multiple models because only one outlier can severely deviate the LS estimates. Re-weighted LS methods [19] are popular methods to avoid the influence of outliers, but a major weakness is that the fitting results depend crucially on the quality of the initial guess [35]. Several approaches have been proposed for solving the initialization problem of the M-estimator [8], [7]. The quality of initialization is improved with these methods, but a poor initial guess may still lead to an undesired estimate.

It is worth pointing out that in many real machine vision applications, the underlying curves to be detected are characterized by arcs, which are actually intra-connected subsets that fit the correct curve models. Since most model estimation methods do not consider the geometric relationship among the candidate points, a common failing is the production of false models. Edge points that fit the false model in this kind of images are considered to satisfy the criterion for curve validity even though they may actually belong to other geometric shapes.

The estimator proposed in this thesis is motivated by some observations of how the human visual system can easily classify the data and ignore the outliers while fitting multiple models. First, approximate estimations are made by examining data taken over the entire image, i.e., it is a global process. Second, closely connected edge points are likely assumed to come from the same model. Third, the full set of inliers is not needed to extract meaningful models.

Chapter 3

Mathematical Modelling and Shulidu-based Estimation

3.1 Introduction

In this chapter a new concept called *shulidu* is developed, which describe the relationship between a single element and the entire set. A robust method for estimating data cloud center is proposed by incorporating the concept of *shulidu*. We then describe a novel estimator for fitting multiple intersecting linearly parameterizable curves. This estimator presents a new approach for handling cases with a high percentage of outliers. A robust single-model extractor employing a region trimming scheme is used for outlier detection. This extractor drives the initial global estimate to the desired model by eliminating outliers iteratively. The underlying models can be extracted sequentially by repeatedly applying this extractor to the updated edge data set. The detected models are then evaluated with a model

verifier to determine their validity.

Three major distinguishing features of this estimator are:

1. It is robust against an arbitrarily high percentage of outliers.

2. It is able to handle an *a priori* unknown number of meaningful models in the edge data without any statistical information of the inlier and outlier distribution.

3. It is effective fitting of multiple intersecting or touching curves.

3.2 Linearly parameterizable curve model

In computer vision, conic and polynomial curves are of great importance because they efficiently describe large parts of man-made objects. They are linearly parameterizable models that can be used to approximate most other complex models in a piecewise fashion. Therefore we devote our attention to linearly parameterizable models in the Euclidean plane.

Let the vector $\vec{x}= (x, y)$ be an edge point in image space I and the vector $\vec{\alpha}$ denote a point in parameter space P. A linearly parameterizable curve $C(\vec{\alpha})$ can be represented by

$$C(\vec{\alpha}) = \{\vec{x}\in I \mid f(\vec{\alpha}, \vec{x}) = 0\}, \tag{3.1}$$

where $\vec{\alpha}= (\alpha_0, \alpha_1, \cdots, \alpha_{p-1})^T$ is the parameter vector and $f(\vec{\alpha}, \vec{x})$, given by

$$f(\vec{\alpha}, \vec{x}) = \sum_{i=0}^{p-1} \alpha_i f_i(\vec{x}) + f_p(\vec{x}), \tag{3.2}$$

is the linearly parameterizable model of the curve. We let $f_i(\vec{x}),\ \ i = 0, 1, \cdots, p$, be polynomial functions on I,

$$f_i(\vec{x}) = x^{n_i} y^{m_i},\ \ i = 0, 1, \cdots, p, \tag{3.3}$$

where n_i and m_i are non-negative integers. Then the polynomial surface of order N has the form of (3.1), and

$$f(\vec{\alpha}, \vec{x}) = \sum_{n=0}^{N} \sum_{k=0}^{n} \alpha_{nk} x^{n-k} y^k, \tag{3.4}$$

where α_{nk} are polynomial coefficients. Obviously, a polynomial surface of the Nth order is linearly parameterizable. Besl and Jain [4] demonstrated that many common surfaces in range images can be well approximated by polynomial surfaces of up to order 4.

3.3 Problem formulation

We now present a mathematical formulation of the parameter estimation problem for the class of multiple curves described by the linearly parameterizable model. Generally, there is more than one curve in an image. For any one of these curves, the edge points of other curves can be viewed as outliers; more often than not, the outliers outnumber the edge points and may make the fitting results far from the true value. The detection and removal of outliers from the set of edge points, which is a classification problem, is the key to multiple-curve estimation.

For each underlying curve in the image, the entire edge point set S can be separated into two subsets, Ω and $\overline{\Omega}$. Ω denotes the subset of the edge points that

may lie on the fitted curve or slightly deviate from it and is the inlier subset of the curve $C(\vec{\alpha})$. $\overline{\Omega}$, on the other hand, is the outlier set of this underlying curve. For a valid curve, its inlier subset Ω should satisfy some requirements that would ensure that the curve is meaningful.

We now construct a performance function that appropriately formulates the multiple-curve fitting problem by taking the inlier-outlier classification into consideration:

$$L(\vec{\alpha}, \Omega) = \sum_{i=1}^{N} w_i^2 l(\vec{\alpha}, \vec{x}_i), \tag{3.5}$$

where

$$\Omega \text{ and } \vec{\alpha} \quad \text{satisfy} \quad q(\vec{\alpha}, \Omega), \tag{3.6}$$

$$w_i = \begin{cases} w(\vec{x}_i, \Omega), & \text{if} \quad \vec{x}_i \in \Omega \\ 0, & \text{if} \quad \vec{x}_i \in \overline{\Omega} \end{cases}, \tag{3.7}$$

and N is the total number of edge points. $l(\vec{\alpha}, \vec{x}_i)$ is a loss function that quantifies the loss when the edge data $\vec{x}_i$ are not exactly on the estimated curve $C(\vec{\alpha})$, $q(\alpha, \Omega)$ are the inlier criteria that a valid inlier subset Ω of the underlying curve $C(\vec{\alpha})$ should satisfy, and $w(\vec{x}_i, \Omega)$ is a weight function with respect to the inlier subset Ω that assigns weights to each of the edge data in Ω.

Thus the problem of multiple-model estimation is to find a group of parameter vectors, $\vec{\alpha}_k$, and the corresponding inlier set Ω_k such that they comprise the local minima of the performance function (3.5) while the valid inlier requirements (3.6) are concurrently satisfied. It is clear that the selection of the inlier set Ω_k greatly influences the estimation of parameter vector $\vec{\alpha}_k$. It is also worth noting that the weights w are closely related to the final estimate $C(\vec{\alpha})$. Therefore the linearly

parameterizable multiple-model estimation problem requires not only the estimation of the curve model but also the detection of its inlier set. It is essentially a "classify and fit" problem.

3.4 Inlier criteria and weight function

The linearly parameterizable multiple-model estimation problem that has been formulated is actually a problem involving both data classification and parameter estimation. Parameter estimation is strongly related to the weight of each edge point, which in turn is obviously determined by the inlier set Ω as well as the weight function. A good set of inlier criteria will be helpful in preventing unwanted inlier subsets as well as pseudo curves.

Generally, the inliers of a curve are those edge data that lie exactly on or, because of random errors, slightly away from it. Hence we propose the following inlier criterion to determine whether an edge point is an inlier:

Criterion 3.1 (Inlier Criterion).

$$g(\vec{\alpha}, \vec{x}_i) \leq T_g, \tag{3.8}$$

where T_g is a predefined threshold and $g(\vec{\alpha}, \vec{x}_i)$ is a distance function that measures the closeness of edge data $\vec{x}_i$ to the curve $C(\vec{\alpha})$.

The edge data that satisfy Criterion 3.1 consist of the inlier subset

$$\Omega(\vec{\alpha}) = \{\vec{x}_i \,|g(\vec{\alpha}, \vec{x}_i) \leq T_g, \quad i = 1, \cdots, \mathcal{N}(\vec{\alpha})\}, \tag{3.9}$$

where $\mathcal{N}(\vec{\alpha})$ is the number of edge data in $\Omega(\vec{\alpha})$, i.e., the number of inliers of the curve $C(\vec{\alpha})$. It is a reasonable requirement that a meaningful curve should have enough inliers. Therefore the second inlier criterion for the number of inliers(NOI) is

Criterion 3.2 (NOI Criterion).

$$N(\vec{\alpha}, \mathcal{N}(\vec{\alpha})) \geq T_N, \tag{3.10}$$

where T_N is a threshold value. $N(\vec{\alpha}, \mathcal{N}(\vec{\alpha}))$ represents in some sense the arc of the curve formed by the inlier set. It is a function of number of edge data in the inlier set.

In most real images, some of the inliers in $\Omega(\vec{\alpha})$ should form at least one meaningful part of the curve that distinguishes this curve from others. This group, which includes those edge data that form the most significant part of the curve, is referred to as the *Significant Inlier Subset*(SIS). In fact the SIS can be in some sense regarded as the feature subset of the curve.

Pseudo curves are a serious problem in curve estimation. In the detection of multiple curves, pseudo curves whose inlier subsets comprise many unrelated edge points or the inliers of other curves will be frequently regarded as valid according to NOI Criterion (Criterion 3.2). Therefore only NOI Criterion (Criterion 3.2) is not a robust criterion against pseudo inlier subsets. Instead, we focus on the most salient arc of the underlying curve and then extend this meaningful arc to reconstruct the entire curve, which leads to the following robust inlier criterion:

Criterion 3.3 (SIS Criterion).

$$\phi(\vec{\alpha}, \Omega_s(\vec{\alpha})) \geq T_\phi, \tag{3.11}$$

where $\Omega_s(\vec{\alpha})$ is the SIS of curve $C(\vec{\alpha})$, $\phi(\vec{\alpha}, \Omega_s(\vec{\alpha}))$ is a feature measurement of curve $\vec{\alpha}$ and its SIS $\Omega_s(\vec{\alpha})$, T_ϕ is a threshold value. This criterion requires that a meaningful curve should have at least one significant arc that confirms its validity.

A reasonable assumption for SIS is that all edge data in the SIS are connected according to some connection criteria. The inlier set can be divided into several intra-connected inlier subsets. In most applications, the SIS is the largest intra-connected subset. However the definition of the SIS is not limited to connectivity but can be determined with any appropriate criteria that describe the features of the model better and, in fact, it can be application dependent.

The SIS Criterion is a robust criterion for arc because it depends on the meaningful inlier subset. What is more, it can be curve size independent if SIS feature is selected appropriately, which means it will represent the arc feature no matter whether the curve is large or small. For example, the angle subtended by arc is a good choice for a curve size independent arc feature. On the contrary, the NOI Criterion is severely influenced by the curve size. The number of inliers in an arc of large circle with small subtended angle may be much more than that of a complete circle with small radius. Therefore it does not represent complete feature of arc. However, it is very useful to prevent a small crowd of edge data from being mistakenly treated as a valid arc. Because with only SIS Criterion, this small block of edge data caused by noise or the intersection of different curves may be verified

as a complete small circle or ellipse, which is obviously an incorrect decision. The NOI Criterion require a valid arc to have enough inliers , which will avoid these pseudo curve models.

When the SIS, Ω_s, is available, the weight function can be defined. Since only the SIS is employed in multiple-curve detection, we assign non-zero weights to edge data in the SIS instead of those in the entire inlier set. Then edge points are inliers and can be assigned positive weight:

$$w_i = \begin{cases} v_i, & \text{if} \quad \vec{x}_i \in \Omega_s \\ 0, & \text{elsewhere} \end{cases}. \tag{3.12}$$

where $v_i \geq 0$ and $\sum_{i=0}^{\mathcal{N}_s} v_i = 1$. The basic idea behind the weight function (3.12) and Criterion 3.3 is that only the inlier points in the SIS are employed in data classification and parameter estimation. The weight function, Inlier Criterion (Criterion 3.1), NOI Criterion (Criterion 3.2) and SIS Criterion (Criterion 3.3) together provide an efficient and robust formulation for the problem of multiple-curve detection.

3.5 Using *shulidu* for sound estimation

Measurement is a fundamental part of science. Like multiple curve fitting, it is also adversely affected by outlier corruption. An effective estimation method was proposed to remove *a priori* unknown outliers in the observation set when it follows a unimodal distribution [39]. Since the measurement problem can be regarded as

the model estimation problem with curve model

$$f(x) = x - a, \tag{3.13}$$

this method is in fact a powerful tool to deal with outliers in this kind of estimation problem. Its basic idea together with its key concept, *shulidu*, form the basis of our new multiple-model estimator for linearly parameterizable curve fitting.

3.5.1 Basic ideas

All measurements are invariably subject to error and uncertainty. If the values of a set of raw observations vary due to random errors, it would be natural to take the mean of the set as the best estimate. Sometimes, however, observations are corrupted by systematic errors or gross errors that bias the mean significantly. Many techniques have been developed to remove the influence of outliers, most of which are based on statistical assumptions about observation distributions, such as Chauvenets's criterion [47]. However, it is a difficult if not an impossible task to design an effective statistical model that fits the observed data.

To avoid this problem, we can identify outliers based on the mutual displacement of the entire observation set. It is also pertinent to note that in the observations of a certain measured quantity, the true value may generally be located within the largest dense area of the observation set, called the *Significant Dense Area* (SDA), and only the observations here should be used to make the estimation.

3.5.2 *Shulidu*

In order to reveal the mutual displacement relation among the observation set in a mathematical way, Qiao [39] developed a new concept, *shulidu*, a Chinese term that means the metric that is used to represent the closeness of a single element in the set to the entire set. The *shulidu* of each observation is defined as the sum of pre-defined distances between this observation and any other observation in the set [39]:

$$s(x_i) = \sum_{k=1}^{n} d(x_i, x_k), \tag{3.14}$$

where $d(x_i, x_k)$ is a pre-defined distance function between x_i and x_k. With this definition of *shulidu*, an observation that deviates greatly from other observations has a very large *shulidu* value, while the observation located in the dense area has a relatively small value. It is worth pointing out that distance does describe the similarity between two single observations, and *shulidu* does reveal the proximity of a single observation to the entire data set. It can also be proved that the *shulidu* values contain the complete distribution information of the set, and hence the entire data set can be exactly reconstructed with only the *shulidu* of each point in the set.

With the available *shulidu*, we can assign to each observation a weight factor that can be regarded as the strength of the influence of the observation on the weighted mean. The weight is calculated by [39]

$$w(x_i) = \frac{\frac{1}{s(x_i)}}{\sum_{j=1}^{n} \frac{1}{s(x_j)}}. \tag{3.15}$$

It is not difficult to see that when the observation has a large *shulidu* value it

is far from the entire set and hence its influence upon the mean should be greatly reduced, which indicates a small weight assigned for this observation. The weight can be expressed in a more general form of the *shulidu* function:

$$w(x_i) = W(s(x_i)|s(x_1), \cdots, s(x_n)) \quad \text{for } i = 1, \cdots, n, \tag{3.16}$$

where $W(s(x_i)|s(x_1), \cdots, s(x_n))$ should satisfy the following three conditions:

Condition 3.1.

$$W(s(x_i)|s(x_1), \cdots, s(x_n)) \geq 0 \quad \textit{for } i = 1, \cdots, n. \tag{3.17}$$

Condition 3.2.

$$\sum_{i=1}^{n} W(s(x_i)|s(x_1), \cdots, s(x_n)) = 1. \tag{3.18}$$

Condition 3.3.

$$W(s(x_i)|s(x_1), \cdots, s(x_n)) \leq W(s(x_k)|s(x_1), \cdots, s(x_n)), \quad \textit{if } s(x_i) > s(x_k), \tag{3.19}$$

$$W(s(x_i)|s(x_1), \cdots, s(x_n)) \geq W(s(x_k)|s(x_1), \cdots, s(x_n)), \quad \textit{if } s(x_i) < s(x_k), \tag{3.20}$$

$$W(s(x_i)|s(x_1), \cdots, s(x_n)) = W(s(x_k)|s(x_1), \cdots, s(x_n)), \quad \textit{if } s(x_i) = s(x_k). \tag{3.21}$$

Conditions 3.1 and 3.2 ensure that the weights are non-negative and normalized. Condition 3.3 reveals the influence of the *shulidu* function on weights of individual observations. According to Condition 3.3, the observation that has a large *shulidu* value will be assigned a small weight and *vice versa*.

With the acquired weights, the weighted estimate is

$$\overline{x} = \sum_{k=1}^{n} w(x_k) x_k. \tag{3.22}$$

Table 3.1. Observations, *shulidu* and weights

i	x_i	$s(x_i)$	$w(x_i)$	i	x_i	$s(x_i)$	$w(x_i)$	i	x_i	$s(x_i)$	$w(x_i)$
1	0.9	36.9	0.0778	6	1.2	36.8	0.0778	11	0.8	38	0.0758
2	0.8	38	0.0758	7	1.1	36.3	0.079	12	1.1	36.3	0.079
3	1.0	36.2	0.0795	8	0.9	36.9	0.0778	13	1.2	36.8	0.0778
4	1.0	36.2	0.0795	9	13.0	149	0.019	14	15.0	175	0.016
5	10.0	116	0.0252	10	1.0	36.2	0.0795	15	1.0	36.2	0.0795

Clearly, from (3.15), we see that the greater the *shulidu*, the lower the weight. Therefore, the contribution of significantly deviated observations that have large *shulidu* is sharply reduced while the normal observations exert more influence on the estimate. For example, in Table 3.1, which includes 15 observations of a measurement with true value 1, x_{14} has the largest *shulidu* value and hence is regarded currently as the most qualified outlier. Further, the biasing influence of x_{14} is heavily suppressed because of the relatively large value of its *shulidu* $s(x_{14})$. The *shulidu*-based weight of x_{14}, $w(x_{14})$ is only about one-fifth of that of x_3, whose *shulidu* is minimum. Compared with the conventional algorithmic mean of 3.3, the *shulidu*-weighted estimate, 1.6, is much less biased from the true value caused by the outliers.

3.5.3 A robust method for sound estimation

The *shulidu*-weighted mean is still heavily corrupted by outliers when there is more than one outlier in the observation set even though the undesired influence is sig-

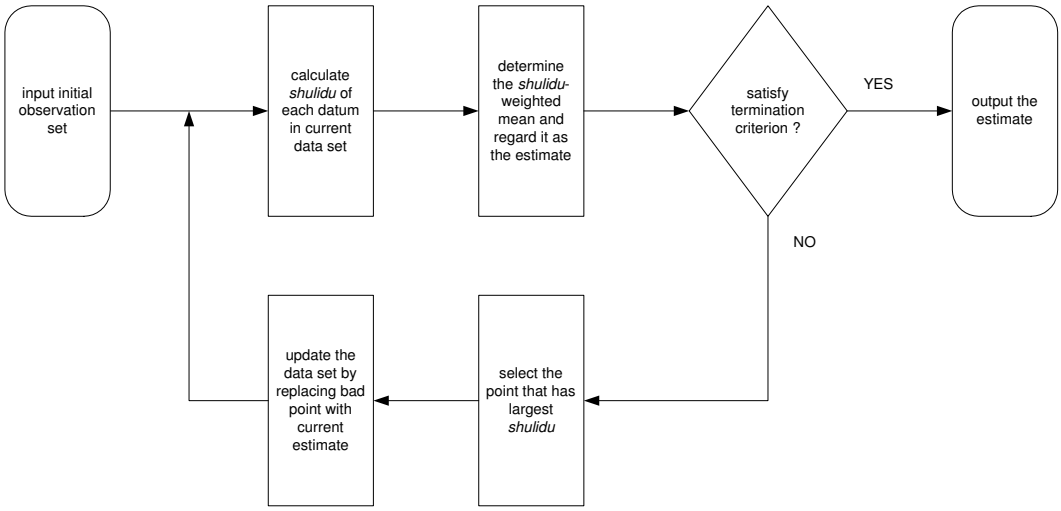


Figure 3.1. Scheme of *shulidu*-based estimation

nificantly suppressed. An improved estimate is obtained by an iterative method (Fig. 3.1) in which a series of estimates converges to the SDA of the observation set [39]. The current estimate contains the information of all observations and, compared with the arithmetic mean, is closer to the true value, while the observations with the largest *shulidu* are most likely to be outliers (provided the outliers exist). Hence the current estimated mean is used to replace the observation with the largest *shulidu* in the current iteration and the observation data set is then updated. In the next iteration, the influence of the discarded data is further weakened and, naturally, the new estimate is closer to the desired value. The iterative process terminates when the difference between two successive estimates is sufficiently small. With the weakening influence of the outliers, the *shulidu*-weighted estimates would converge to the SDA, which is within the desired neighborhood of the true values.

The iterative estimation method consists of three major steps:

(1) Calculate the *shulidu* of each observation.

(2) Determine the *shulidu*-weighted estimate.

(3) Examine the result and update the observation set. If the result is desired, e.g., the difference of the last two *shulidu*-weighted estimate is small enough, stop; otherwise, go back to step (1).

The algorithm achieves satisfactory results when the observation set is unimodally distributed. In the example of Table 3.1, there are three outliers, $x_5 = 10.0$, $x_9 = 13.0$, $x_{14} = 15.0$. They are replaced by the temporary estimates during the iteration. The final estimate is 1.02, which is very close to the true value. Clearly, the biasing influence of the three outliers has been almost completely removed.

3.5.4 Analysis

In clustering approaches, the similarity measurements of a sample to the cluster is a critical criterion when classifying samples into clusters. Usually the sample will be assigned to a cluster according to its similarity measurement. In conventional clustering methods, the similarity measurement of a sample to a cluster is generally defined as the the distance between the sample and the mean vector of this cluster [12],

$$J(x_i) = d(x_i, m), \tag{3.23}$$

where m is the mean vector of the samples in the cluster,

$$m = \frac{1}{n}\sum_{i=1}^{n} x_i, \tag{3.24}$$

and n is the number of samples in this cluster. A simple interpretation is that the mean vector of a given cluster best represent samples in the cluster in the sense that it minimizes the sum of the squared errors of all samples of the cluster to the cluster mean. The above similarity measurement is actually the deviation of a sample from the mean vector of the cluster. Therefore the performance of this similarity measurement (3.23) depends critically on the mean vector.

The deviation $d(x_i, m)$ in (3.23) can be various forms of distance. The Euclidean distance is most commonly used in the deviation with the similarity measurement of the form

$$J_e(x_i) = |x_i - m|. \tag{3.25}$$

The Mahalanobis distance is another popular selection for deviation. The similarity measurement using the Mahalanobis distance is

$$J_m(x_i) = (x_i - m)A^{-1}(x_i - m), \tag{3.26}$$

where A is the estimation of the covariance matrix

$$A = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - m)(x_i - m)', \tag{3.27}$$

These mean-based similarity measurements ($J_e(x_i)$ and $J_m(x_i)$) are appropriate criteria for distinguishing outliers and inliers if the cluster forms a compact cloud. The mean vector is located near the center of the cluster in this situation. However, if there are some outliers in the cluster, the mean vector may be significantly biased away from the cluster center. Then this mean-based similarity measurement (3.23) may provide the wrong information about the closeness of a sample to the cluster.

Table 3.2. Observations, Euclidean distance, Mahalanobis distance and *shulidu*

i	x_i	$J_e(x_i)$	$J_m(x_i)$	$s(x_i)$
1	0.941	3.983	0.241	59.824
2	0.940	3.984	0.241	59.831
3	1.021	3.903	0.231	59.514
4	0.954	3.969	0.239	59.727
5	0.940	3.984	0.241	59.831
6	0.903	4.021	0.245	60.308
7	1.049	3.874	0.228	59.542
8	0.989	3.935	0.235	59.554
9	1.086	3.837	0.223	59.653
10	0.993	3.930	0.234	59.541
11	4.919	0.005	0.000	78.815
12	5.346	0.423	0.003	82.450
13	5.025	0.102	0.000	79.560
14	22.026	17.103	4.435	265.933
15	26.721	21.798	7.205	326.967

An example in Table 3.2 shows that some outliers cannot be recognized with the mean-based similarity measurement. There are 15 observations of a measurement with true value 1. The data $x(1)$ - $x(10)$ are inliers while the data $x(11)$ - $x(15)$ are outliers. The mean of all observations $m = 4.92$ is significantly biased from the true value and close to the outlier data $x(11)$ - $x(13)$. Consequently, the mean-based similarity measurement of these three outliers (both $J_e(x_{11})$ - $J_e(x_{13})$ and $J_m(x_{11})$ - $J_m(x_{13})$) are much smaller than those of inliers (both $J_e(x_1)$ - $J_e(x_{10})$ and $J_m(x_1)$ - $J_m(x_{10})$). Suppose we have *a priori* information that there are five outliers, we cannot identify the outliers $x(11)$ - $x(13)$ if we use the mean-based similarity measurements ($J_e(x_i)$ and $J_m(x_i)$). Actually, due to the closeness between the mean and outliers $x(11)$ - $x(13)$, the mean-based similarity measurement will fail to recognize the outliers $x(11)$ - $x(13)$ no matter what kind of distance is used. On the contrary, the *shulidu* is not influenced by the emergence of outliers. From Table 3.2, the *shulidu* of all inliers ($s(1)$ - $s(10)$) are smaller than those of the outliers ($s(11)$ - $s(15)$). With the *shulidu*, all five outliers will be correctly recognized. The initial *shulidu*-weighted mean is 2.454063, which is also sharply biased from the true value. However, it is much closer to the true value than the mean value defined in (E:Mean). Actually, the deviations of all outliers to the initial *shulidu*-weighted mean are larger than those of inliers. Therefore the mean-based similarity measurement can work for outlier recognition with the initial *shulidu*-weighted mean. The *shulidu*-weighted mean converges to 1.004920 with the method proposed in Section 3.5.3, which is close enough to the true value.

Example in Table 3.2 illustrates that the *shulidu* is much more reliable than the

mean-based similarity measurement in the presence of outliers. The *shulidu* employs the sum of distance of this sample to each member of a cluster to describe the similarity of a sample to this cluster. Thus the *shulidu* contains more information of the cluster distribution than the mean-based similarity measurement (3.23). It ensures the robustness of the *shulidu* against outliers and avoids the sensitiveness of the mean to outliers.

3.6 Extracting a single model

Before we investigate the problem of fitting multiple linearly parameterizable curves, we need an effective method for extracting a single model. LS optimization is a popular choice, but there are generally some *a priori* unknown outliers in the image that will severely distort the estimation results of the LS method. We use the iterative weighted LS approach to circumvent the outlier problem. We then present an effective strategy, the region trimming scheme, that defines weight functions to handle outliers. The above techniques form an effective single-model extractor for the fitting of multiple linearly parameterizable curves.

3.6.1 Iterative Weighted LS Approach

The iterative weighted LS approach is an efficient mathematical framework for solving the curve fitting problem (Section 3.3). In this iterative method, weights of the edge data and the curve model converge to the correct values by adjusting the weights and making an optimal model estimation at every cycle.

Let $\Omega = \{\vec{x}_i,\ i = 1, 2, \cdots, Q\}$ denote the edge data set where Q is the number of edge points in Ω. The parametric form of the underlying curve is described by (3.1). For simplicity, we use the matrix notation of (3.2),

$$Y = X\vec{\alpha}, \tag{3.28}$$

where

$$Y = \begin{bmatrix} f_p(\vec{x}_1) \\ f_p(\vec{x}_2) \\ \vdots \\ f_p(\vec{x}_q) \end{bmatrix}_{q\times 1}, \tag{3.29}$$

$$\vec{\alpha} = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{p-1} \end{bmatrix}_{p\times 1}, \tag{3.30}$$

$$X = \begin{bmatrix} -f_0(\vec{x}_1) & -f_1(\vec{x}_1) & \cdots & -f_{p-1}(\vec{x}_1) \\ -f_0(\vec{x}_2) & -f_1(\vec{x}_2) & \cdots & -f_{p-1}(\vec{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ -f_0(\vec{x}_q) & -f_1(\vec{x}_q) & \cdots & -f_{p-1}(\vec{x}_q) \end{bmatrix}_{q\times p}. \tag{3.31}$$

For a certain curve $C(\vec{\alpha})$, the error is

$$e_i = f(\vec{\alpha}, \vec{x}_i) = \sum_{j=0}^{p-1} \alpha_j f_j(\vec{x}_i) + f_p(\vec{x}_i). \tag{3.32}$$

The matrix form of (3.32) is

$$Y = X\vec{\alpha} + \vec{e}, \tag{3.33}$$

where $\vec{e} = [e_1, e_2, \cdots, e_q]^T$.

The initial estimates of the curve parameters, $\vec{\alpha}^{(0)}$, are known either *a priori* or by applying the conventional LS method. The current weight $w_i^{(n)}$ for $\vec{x}_i \in \Omega$ is obtained by different inlier search strategy. We define the total weighted deviation of the points in Ω from the current estimated curve $C(\vec{\alpha}^{(n)})$ by

$$E_w(\vec{\alpha}^{(n)}) = \sum_{\vec{x}_i \in \Omega} [w_i^{(n)} f(\vec{\alpha}^{(n)}, \vec{x}_i)]^2 = \sum_{\vec{x}_i \in \Omega} [w_i^{(n)} e_i]^2. \tag{3.34}$$

The subproblem of reconstructing a curve from edge data based on the previous estimate $\vec{\alpha}^{(n-1)}$ is to find the estimated curve parameters $\vec{\alpha}^{(n)}$ such that the total weighted deviation over Ω is minimized, i.e.,

$$E_w(\hat{\vec{\alpha}}^{(n)}) = \min_{\vec{\alpha}^{(n)}} E(\vec{\alpha}^{(n)}). \tag{3.35}$$

This is exactly a weighted LS optimization problem. Let W be a diagonal weight matrix, i.e.,

$$W^{(n)} = \begin{bmatrix} w_1^{(n)} & 0 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \cdots & 0 \\ 0 & 0 & w_i^{(n)} & 0 & 0 \\ \vdots & 0 & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & w_Q^{(n)} \end{bmatrix}_{Q \times Q}. \tag{3.36}$$

We then obtain the matrix form of (3.34):

$$E_w(\vec{\alpha}^{(n)}) = \|W^{(n)} Y - W^{(n)} X \vec{\alpha}^{(n)}\|_2^2. \tag{3.37}$$

The solution to

$$E_w(\hat{\vec{\alpha}}^{(n)}) = \min_{\vec{\alpha}^{(n)}} \|W^{(n)} Y - W^{(n)} X \vec{\alpha}^{(n)}\|_2^2 \tag{3.38}$$

is equivalent to solving the equations

$$X^T(W^{(n)})^T W^{(n)} X\vec{\alpha}^{(n)} = X^T(W^{(n)})^T W^{(n)} Y, \tag{3.39}$$

where $(\cdot)^T$ denotes the matrix transpose. The solution $\vec{\alpha}^{(n)}$ is used to evaluate the new weights $W^{(n+1)}$ for further estimation of $\vec{\alpha}^{(n+1)}$ and so on. The iterative estimation procedure will not be terminated until the curve models meet the termination criteria.

3.6.2 Region Trimming Scheme

In an image with multiple curves, most of the outliers are edge points that support other curves. Hence they generally do not scatter in the form of a Gaussian distribution with respect to the fixed curve. Due to the outliers' diverse statistical distribution characterization, conventional methods based on statistical assumptions often offer no solution to the elimination of outliers and may even yield non-meaningful results. A further difficulty is that such methods require a good initial estimate of the model.

To solve the initialization problem, we propose a novel global-based estimation strategy. It detects a globally favorable curve of the image by identifying and discarding outliers from the current available data set in an iterative process.

The original data set is grouped into several subsets called the initial inlier candidate sets. The strategy uses all the data in the initial inlier candidate set to make an initial estimate of the underlying curve models. Since the initial set may be the union of the inliers of all the intersecting or touching curves, the initial

estimate may not necessarily be a meaningful one for any of these underlying curves. However, it would be a useful model for detecting the outliers of the current favorable curve.

The current most qualified outlier candidates should be the data that have the maximum deviation from the current estimated model and also contribute most to make the present estimate far from the underlying favorable curve. Therefore, the detected outlier data are removed from the current inlier candidate set. Moreover, in order to overcome noise and pseudo curves, the most significant subset of the new reduced set, which is obtained according to some application-dependent criteria, is selected as the new updated inlier candidate set for next iteration. Hence the new estimate is expected to be improved with the updated inlier candidate set. With outliers being discarded, not only do the estimates converge to the correct favorable curve model, the updated inlier candidate set converges to the SIS of the favorable curve as well. When the residuals of all the inlier candidates satisfy the fitting requirements, the present estimate will fit the desired underlying curve model. The remaining inlier candidates are the true inliers of the fitted curve.

There is obviously no initialization problem in the above global-based estimation strategy. The major feature that distinguishes this scheme from conventional methods is that the detection of outliers is based on the global inlier candidate set instead of being dependent on an accurate local guess about the model inliers. The proposed strategy is very similar to the process of tree trimming in which unwanted branches are removed; hence we call it global-based region trimming.

Fig. 3.2 shows the flowchart of the single-model extractor. The iterative weighted

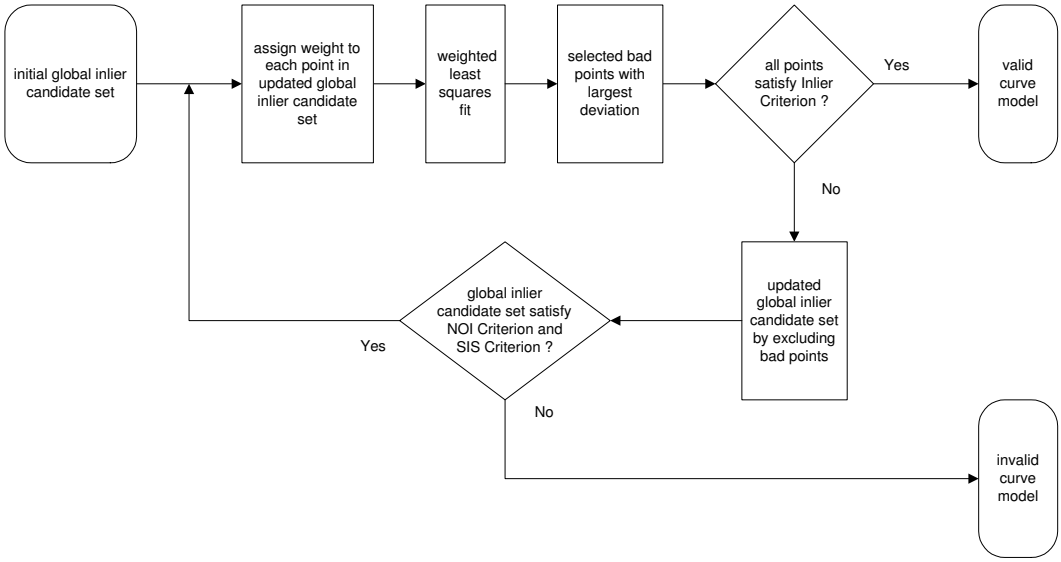


Figure 3.2. Single-model extraction

LS approach discussed in Section 3.6.1 provides the general mathematical framework for the single-model extractor. The region trimming strategy can be well formulated by the iterative weighted LS approach by setting the weight of edge points excluded at each iteration as 0.

3.7 Fitting multiple curves

Fig. 3.3 describes the multiple-model estimator, which can handle multiple touching or intersecting linearly parameterizable curves in the presence of an arbitrarily high percentage of *a priori* unknown outliers. The estimator consists of three major steps: (1) preprocessing the original data, (2) fitting meaningful curves, and (3) verifying the fitted curves.

Step 1: Preprocessing the original data.

We first examine the edge data in the entire image $\mathcal{I}$ and divide them into

different groups, G_j, according to some criterion based on some *a priori* knowledge, i.e., $\mathcal{I} = \cup\, G_j, \quad$ where $\; G_j \cap G_i = \emptyset \;$ for any $\; j \neq i$. The data subsets G_j are the initial inlier candidates subsets.

Step 2: Fitting meaningful curves.

Due to possible intersection of different curves, a single subset G_j may still contain multiple curves. The global-based region trimming scheme is used in G_j to find the outliers as well as estimate the model parameters. When the most favorable model $C(\overset{\wedge}{\vec{\alpha}})$ is reconstructed with the single-model extractor, the current inlier candidate set $R_j^{(k)}$ that satisfies both NOI Criterion and SIS Criterion is in fact its SIS. Since the inliers of $C(\overset{\wedge}{\vec{\alpha}})$ will become the outliers of other underlying curves, these inliers are discarded for subsequent model estimation. If the reconstruction fails this time, the remaining subset $R_j^{(k)}$ that meets neither of NOI Criterion and SIS Criterion is still discarded such that the estimates of other models is available. The initial edge data set G_j is then updated by $\widetilde{G}_j = G_j - R_j^{(k)}$, where $\widetilde{G}_j$ is the updated initial inlier candidate set. All inliers of curves in their respective SIS's that have already been extracted are excluded from this updated subset. Data set $\widetilde{G}_j$ can be separated into subsets that can be used for other curve fitting. The other curves of the same parametric form are reconstructed by repeating step (2) with the original or updated initial inlier candidate sets until no such kind of subset is available for curve fitting.

Step 3: Verifying fitted curves.

When all initial inlier candidate sets have been investigated for curve reconstruction, we obtain a set of estimated models. However, the edge points of a

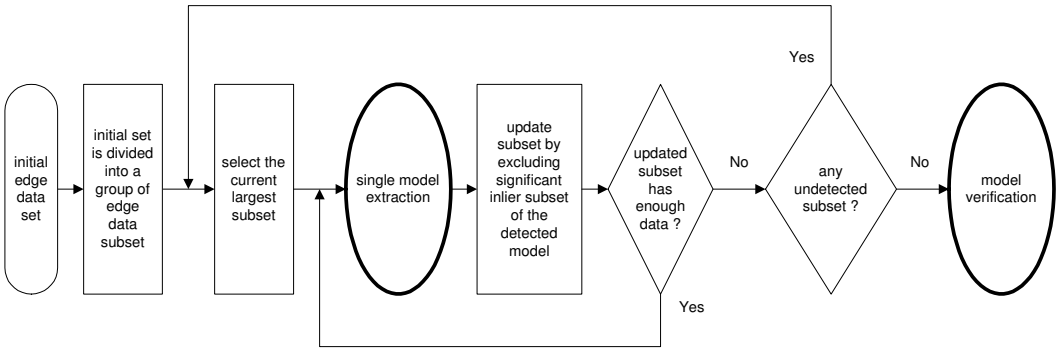


Figure 3.3. Multiple-model estimation

single curve may override several initial inlier candidate sets, and edge data from different curves in an initial inlier candidate set may be mistakenly used to reconstruct pseudo curves (curves that do not have enough non-overlapping inliers). In order to detect these pseudo curves, we must obtain the true non-overlapping inliers of each curve. The model verifier (Section 4.7) is used to discriminate between true curves and pseudo curves.

The iterative multiple-model estimation approach to reconstruct 2-D curves from edge data can be easily extended to multi-dimensional model reconstruction since the edge data $\vec{x}$ are treated as a multi-dimensional vector in the linearly parameterizable curve models.

3.8 Model verification

Since a single curve may be separated into several parts in different initial inlier candidate sets, the fitting results may be repeated. Moreover, pseudo curves are likely to occur when the edge data in SIS are heavily corrupted by noise. The

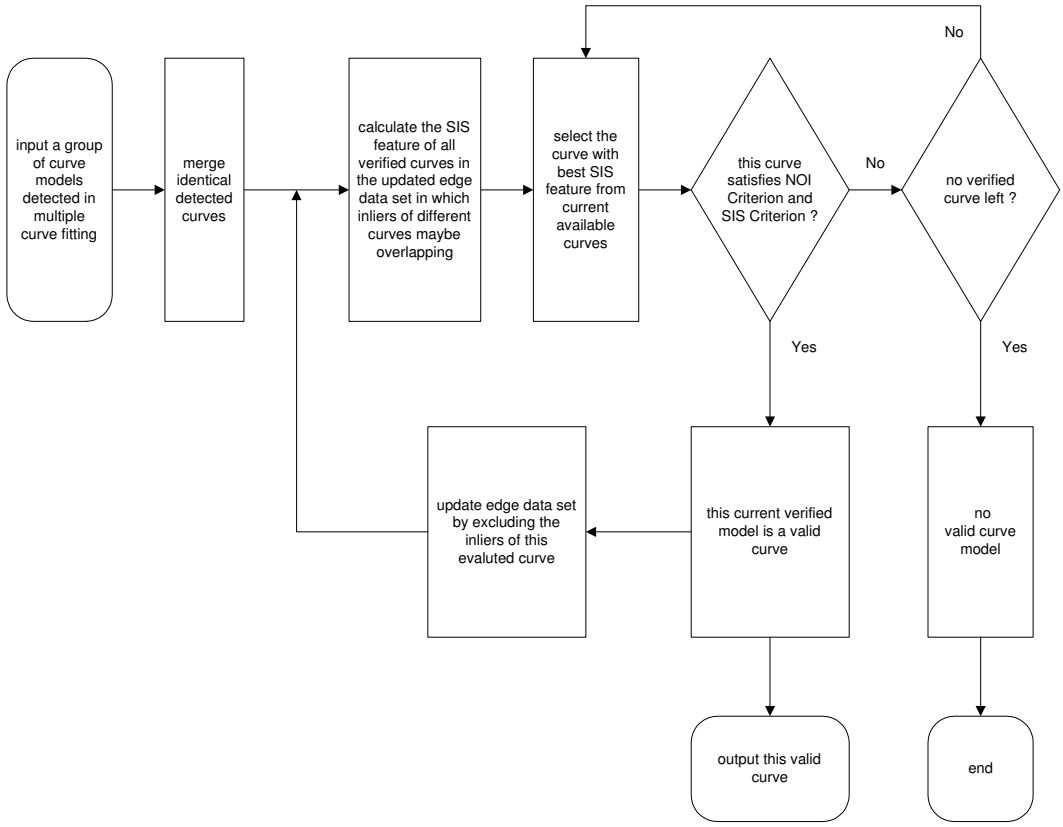


Figure 3.4. Model verification

model verifier (Fig. 3.4) is employed to evaluate the validity of fitted curves.

The verification process comprises three major steps: (1) merging identical curves, (2) evaluating the current most favorable curve, and (3) updating the original edge data set $\mathcal{I}$.

Step 1: Merging identical curves.

We merge curves that are extracted from different initial inlier candidate sets when the difference between the curve parameters is sufficiently small. If $C(\vec{\alpha}_i)$ and $C(\vec{\alpha}_j)$ are regarded as identical curves, a new curve $C(\vec{\alpha}'_i)$ is obtained by taking the mean of the two extracted curve parameters. Then both $C(\vec{\alpha}_i)$ and $C(\vec{\alpha}_j)$ are removed from the fitted curve set while the new model $C(\vec{\alpha}'_i)$ is added

to the curve set to represent them. After all identical curves are detected and merged, we obtain a set of distinct estimated models.

Step 2: Evaluating the current most favorable curve.

Using the current updated edge data $\mathcal{I}_k$, we sort all unverified curves according to their favorable function. Here we take the feature measurement of the SIS, $\phi(\vec{\alpha}, \mathcal{N}_s(\vec{\alpha}))$, as the favorable function. The curves in the current evaluated curve set Θ_k are sorted according to their favorable function:

$$\Theta_k = \{C(\vec{\alpha}_i) | \phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i)) \geq \phi(\vec{\alpha}_j, \Omega_s(\vec{\alpha}_j)) \text{ for any } j \geq i \geq k\}. \tag{3.40}$$

The top ranked curve $C(\vec{\alpha}_k)$ in Θ_k is actually the most favorable curve of the current unverified models. The inlier set of $C(\vec{\alpha}_k)$ in the entire image, $\Gamma(\vec{\alpha}_k)$, includes all current available edge data that fulfill Inlier Criterion. If $C(\vec{\alpha}_k)$ satisfies both NOI Criterion and SIS Criterion, then it is a valid model, go to step 3 for updating edge data set; otherwise, it will be treated as a pseudo curve and discarded. Its inliers will not be excluded and still available for other verified curves. We then evaluate the second most favorable model, and so on. This step is not terminated until we either extract a valid model or find that none of them is valid, in which case the procedure of model verification stops.

Step 3: Updating the original edge data set after curve $C(\vec{\alpha}_k)$ has been verified.

The original edge data set $\mathcal{I}_k$ is updated by discarding $\Gamma(\vec{\alpha}_k)$, i.e., $\mathcal{I}_{k+1} = \mathcal{I}_k - \Gamma(\vec{\alpha}_k)$, where $\mathcal{I}_1 = \mathcal{I}$. We then return to Step 2 to select and evaluate the next most favorable curve $C(\vec{\alpha}_{k+1})$ with the updated edge data set $\mathcal{I}_{k+1}$. When the above iterative model verification procedure is completed, we obtain the non-overlapping inlier set of each curve. The pseudo curves are those that cannot satisfy

both NOI Criterion and SIS Criterion are discarded.

In this iterative fashion, the curve with the largest SIS feature $\phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i))$ is given priority in selecting its non-overlapping inliers. This scheme is efficient in preventing pseudo curves from surviving because they are reasonably supposed to have smaller SIS features than genuine curves and are hence verified after those genuine models. This model verifier not only detects the pseudo curves but also gives the non-overlapping inlier set of the genuine curves.

3.9 Conclusion

In this chapter we propose a new *shulidu*-based method for the estimation problem of the model $f(x) = x - a = 0$ for a unimodal sample distribution. *Shulidu* is a new similarity measurement of a sample from the cluster, which is actually the sum of the distances of the sample to each member of the cluster. Compared with the conventional similarity measurement - the distance from a sample to the mean vector of the cluster - *shulidu* is more robust against outliers and contains more information about the cluster distribution. Examples show that *shulidu* is a reliable criterion in the presence of considerable outliers while conventional similarity measurement fail to provide the correct similarity measurements of both inliers and outliers due to the significant bias of mean vector. The *shulidu*-weighted mean appears to be closer to the center of compact data clouds than the conventional mean, which is calculated by assigning each sample of the cluster the same weight.

We present in this chapter a multiple-model estimator that is robust against an

arbitrarily high percentage of outliers and is able to extract an unknown number of touching or intersecting models. In our estimator, an image pre-partitioning technique is used to classify edge data into groups based on some application dependent assumptions. A single-model extractor, which is actually an iterative weighted LS approach, is then applied to each group of data for the detection of underlying curves formed by edge points in this group. It starts from an initial global model estimate based on the data in this group. The most qualified outlier candidates that have the largest deviations are first kept out of the following estimation. The most significant subset from the remaining edge data is taken as the updated data set for the next round of model estimation. In this way, this scheme drives the estimate from a initial global model to converge to a final local desired model by iterative elimination of detected outliers. All underlying models in the image can be reconstructed sequentially by repeatedly applying the single-model extractor to the edge data that were excluded in the previous fitting. A model verifier is applied to all extracted models to determine whether they are desired models or pseudo models.

This multiple-model estimator has several novel features and compares favorably with parametric model fitting algorithms proposed previously:

- It does not require an accurate initial guess of curves. Due to the global outlier search style of the region trimming scheme, the single-model extractor starts outlier detection from the entire edge data set rather than from a subset.

- It is able to handle an arbitrarily high percentage of outliers. The strategy of removing only the current most qualified outliers enables the single-model extractor to drive the global initial estimate towards the local desired solution.

- The use of the most significant subset in the updating of inlier candidate set dramatically reduces the possibility of producing pseudo curves. The introduction of the SIS in inlier criterion, updating initial inlier subset and ranking of verified curves enables our estimator to extract and evaluate curves based on their significant features.

These effective strategies protect our estimator from producing pseudo models. The distinctive abilities of successfully dealing with a high percentage of outliers and pseudo curves results in an estimator capable of extracting meaningful models from touching or intersecting curves.

Chapter 4

Connectivity-based Multiple Curve Fitting

4.1 Introduction

This chapter proposes a connectivity-based method for curve fitting. The use of pixel connectivity in the iterative LS estimation algorithm using region trimming technique (Chapter 3) effectively avoids false curve detection, improves the robustness against noise and significantly reduces the computational load. The desired curve models are extracted by searching for meaningful arcs. The algorithm does not require a good initial guess and is effective for extracting an *a priori* unknown number of curves even when the number of outliers exceeds 50%.

4.2 Basic idea

Model reconstruction from a complex image is an important and difficult task in computer vision. One of the difficulties often encountered is the fitting of false models. HT-based methods are one-to-many maps from image space to parameter space that employ a voting scheme in the parameter space to extract model parameters [21]. Robust regression and LS-based methods merely use the error of edge points with respect to the verified model as the inlier criterion [35] [19]. All pixels are treated equally in these methods regardless of the geometric relationship among them. Thus, individual pixels that are far away from each other may be thought to constitute a meaningful model that actually does not exist. If the number of inliers is used as the criterion, it would be difficult to discriminate these false models from meaningful ones.

Valid curves in real applications consist not only of individual inliers but also of meaningful arcs. These arcs are actually intra-connected pixel subsets with sufficiently large extent. Consequently, by introducing pixel connectivity, the task of curve fitting is not one of finding enough individual inliers but of searching for meaningful arcs. With this criterion for curve validity, false models whose inliers are unconnected pixels or are made up of invalid arcs are easily recognized since they are not formed from meaningful arcs.

The iterative weighted LS approach using region trimming searching technique discussed in Section 3.6.1 provides a good mathematical framework for curve fitting, which are robustness against an arbitrarily high percentage of outliers and able to handle an *a priori* unknown number of meaningful models in the edge

data without any statistical information of both inlier and outlier distribution. In this chapter, we develop an efficient multiple model fitting algorithm based on the concept of pixel connectivity and valid arcs that can handle false curve fitting and greatly reduce the computational complexity.

4.3 Connectivity and valid curve criterion

4.3.1 Connectivity

We use the Euclidean distance $d(\cdot)$ to measure the connectivity between two points $p_1(x_1, y_1)$ and $p_2(x_2, y_2)$:

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \tag{4.1}$$

Then two edge points $p_1(x_1, y_1)$ and $p_2(x_2, y_2)$ are connected to each other if the distance satisfy

Criterion 4.1 (Point Connection Criterion).

$$d(p_1, p_2) < T_c, \tag{4.2}$$

where T_c is a threshold for connectivity. Based on the connectivity between two edge points, an intra-connected data set can be defined. The data set $\Omega = \{(x_i, y_i),\ i = 1, 2, \cdots, q\}$ is an intra-connected data set if, for any two points $p_i(x_i, y_i)$ and $p_j(x_j, y_j)$ in Ω, there exists a sequence of points $\{p_{k_n} \in \Omega,\ n = 1, 2, \cdots, m\}$ such that they satisfy

Criterion 4.2 (Set Connection Criterion).

$$\max_{n=1,2,\cdots,m-1}(d(p_i, p_{k_1}), d(p_{k_m}, p_j), d(p_{k_n}, p_{k_{n+1}})) < T_c. \tag{4.3}$$

The inequality (4.3) indicates that in the intra-connected set there is a path between all pairs of points in which the two endpoints of each step are connected.

4.3.2 Valid arcs

An important step in our curve fitting algorithm is the detection of meaningful arcs. An arc is an intra-connected edge-point subset whose elements are inliers of curves. For an arc to be considered valid (i.e., meaningful), it should be of significant extent. Therefore, the geometrical residuals of points in the data set and the angle subtended by the corresponding arc are the two evaluation criteria.

The closeness of an edge point $P(x, y)$ to the curve $C(\vec{\alpha})$ is measured by the point error function $g(\vec{\alpha}, P)$. A point $P(x, y)$ is an inlier if its point error function $g(\vec{\alpha}, P)$ satisfies the Inlier Criterion (Criterion 3.1), which means that its error is sufficiently small. Here the error of a point with respect to the curve $g(\vec{\alpha}, P)$ is regarded as its deviation with respect to the edge point set. A set error function $G(\vec{\alpha}, \Omega)$, which describes the fitness of the data subset Ω to the curve model $C(\vec{\alpha})$, is defined as

$$G(\vec{\alpha}, \Omega) = \max_{P_i \in \Omega} g(\vec{\alpha}, P_i). \tag{4.4}$$

The data set Ω is an inlier set of circle $C(\vec{\alpha})$ if the set error function also satisfies the Inlier Criterion, i.e.

$$G(\vec{\alpha}, \Omega) < T_g. \tag{4.5}$$

To summarize, the data set Ω forms a valid arc of curve $C(\vec{\alpha})$ if

(1) Ω is an intra-connected data set.

(2) Ω is an inlier set of curve $C(\vec{\alpha})$ (4.5).

(3) the feature measurement of extent of arc formed by Ω, $\phi(\vec{\alpha}, \Omega)$, is large enough (NOI Criterion and SIS Criterion).

4.4 Search strategy

A meaningful arc is an intra-connected edge-point subset in which the points fit the curve model within a pre-defined error tolerance. To identify potential arcs, it is natural to first partition the data set into intra-connected subgroups. Each subgroup may consist of intersecting arcs from different curve models or even other kind of curve models. The search for meaningful arcs may be complicated by a significantly larger population of outliers, but the scope of investigation is now narrowed to the intra-connected subgroups, thus greatly simplifying the computational load.

A simple iterative technique to deal with outliers is described in [44]. Points that are located far from the model extracted by LS fitting are regarded as outliers. They are removed and LS fitting is repeated. However, the accurate classification of outliers is a difficult problem and requires *a priori* knowledge about the noise and outlier statistics.

If no such *a priori* information is available, only the point with the largest error — the extreme point — can be treated as the outlier. Other points with

large errors may be inliers, while points with small errors may not necessarily be inliers. Since the current estimate may be pulled off significantly by the extreme point and other unknown outliers, a safe strategy is to remove only the extreme point at each iteration of LS fitting. This cautious approach is robust against a high percentage of outliers but the computational complexity is a serious problem because only one bad point is deleted at each iteration.

In order to reduce the amount of computation, we employ pixel connectivity in the search for potential arcs. We first define the expanded neighborhood of an extreme point as the edge set satisfying these two properties:

(i) the error of any point in the neighborhood is larger than the error of any point not in the neighborhood;

(ii) all points in the neighborhood are connected to each other, that is, it is an intra-connected set.

Therefore, with respect to a curve model $C(\vec{\alpha})$, Δ, the expanded neighborhood of an extreme point of a given data set Ω, is an intra-connected data subset such that

$$\min_{p_i \in \Delta} g(\vec{\alpha}, p_i) > G(\vec{\alpha}, \Gamma), \tag{4.6}$$

where $\Gamma = \Omega - \Delta$. According to this definition, there may exist several expanded neighborhoods corresponding to an extreme point. All these expanded neighborhoods can be regarded as outlier subsets. Since the extreme point is an outlier, the points in its neighborhood that have errors larger than the errors of points outside the neighborhood are also very likely outliers. In our proposed strategy,

the largest expanded neighborhood of the extreme point, which is regarded as an outlier subset, is removed from the data set at each iteration.

After an outlier subset is removed from the current data set, the points left behind may not necessarily be inter-connected. We partition the remaining points into intra-connected subgroups. The largest subgroup is taken as the potential SIS of the underlying curve and thus treated as the updated data set in the subsequent search for search of desired arcs. Other subgroups are also treated as outliers and rejected from the following fitting, thus significantly reducing the search space.

When a valid or invalid arc is finally detected, the points belonging to this arc are eliminated from the original data set since they are the outliers of other valid curves. Outliers that have been removed in the previous fitting process are used in seeking further potential arcs. In this way, multiple curve models in this subgroup are sequentially extracted.

4.5 Connectivity-based weight function and LS estimator

The iterative weighted LS approach discussed in Section 3.6.1 provides a good mathematical framework for multiple curve fitting. We describe connectivity-based weight function and the LS estimator for multiple curve fitting using the idea of pixel connectivity and valid arcs.

From the search strategy described in Section 4.4, we learn that at each iteration of curve fitting, the original data set is updated by removing the outlier subset.

Then the updated set is further segmented into intra-connected subsets. Only the edge data in the largest subset Ω will be used in the next round estimation. Other subsets are also regarded as outliers. Therefore the connectivity-based weight function is defined as

$$w(P_i, g(\vec{\alpha}, P_i)) = \begin{cases} 1, & \text{if} \quad P_i \in \Omega \\ 0, & \text{if} \quad P_i \notin \Omega \end{cases}, \tag{4.7}$$

In this search strategy, the LS estimator is applied only to the edge data in updated edge data subset Ω. Therefore the matrix form of (4.7) for subset Ω is

$$W = I = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \vdots & 0 & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix}_{Q \times Q}, \tag{4.8}$$

where Q is the number of edge data in subset Ω.

Based on the search strategy, together with weight function 4.8, the LS estimator of Eq. (3.39) simplifies to

$$X^T(\vec{\alpha}^{(n)})X(\vec{\alpha}^{(n)}) = X^T(\vec{\alpha}^{(n)})Y. \tag{4.9}$$

4.6 Single-model extractor

In an image with multiple curves, most outliers are edge points that support other curves. The outliers do not follow Gaussian distributions; hence conventional methods based on statistical assumptions usually offer no solution to their elimination

and may even yield non-meaningful results. To solve the outlier problem, we employ an LS-based single model extractor that follows the search strategy outlined in Section 4.4. It detects a meaningful arc by identifying and discarding outliers iteratively.

Consider an intra-connected data set $S = \{p_i,\ i = 1, 2, \cdots, q\}$, in which p_i is an edge point with coordinate (x_i, y_i). The detailed algorithm is as follows.

1. Initialize edge data set $\Omega^{(1)} = S$. Let $k = 1$.

2. Estimate the curve model $C(\vec{\alpha}^{(k)})$ of current data set $\Omega^{(k)}$ with Eq. (4.9).

3. Evaluate the quality of the detected arc formed by $\Omega^{(k)}$. If $\Omega^{(k)}$ forms a valid arc, go to step 7; otherwise proceed to step 4.

4. Update the current data set $\Omega^{(k)}$.

   4.1. Seek the outlier subset $\Delta^{(k)}$ such that $\min\limits_{p_i \in \Delta^{(k)}} g(\vec{\alpha}^{(k)}, p_i) > G(\vec{\alpha}^{(k)}, \Gamma^{(k)})$, where $\Gamma^{(k)} = \Omega^{(k)} - \Delta^{(k)}$.

   4.2. Remove outlier subset $\Delta^{(k)}$ and divide the remaining subset $\Gamma^{(k)}$ into intra-connected subsets. Select the largest intra-connected subset as the updated data set $\Omega^{(k+1)}$.

5. If $\Omega^{(k+1)}$ does not include enough points, go to step 6. If $\Omega^{(k+1)}$ has enough points, let $k = k + 1$, and go to step 2 for the next iteration.

6. Determine the failure of estimation, output the remaining data set, $\Omega^{(k+1)}$, and go to step 8.

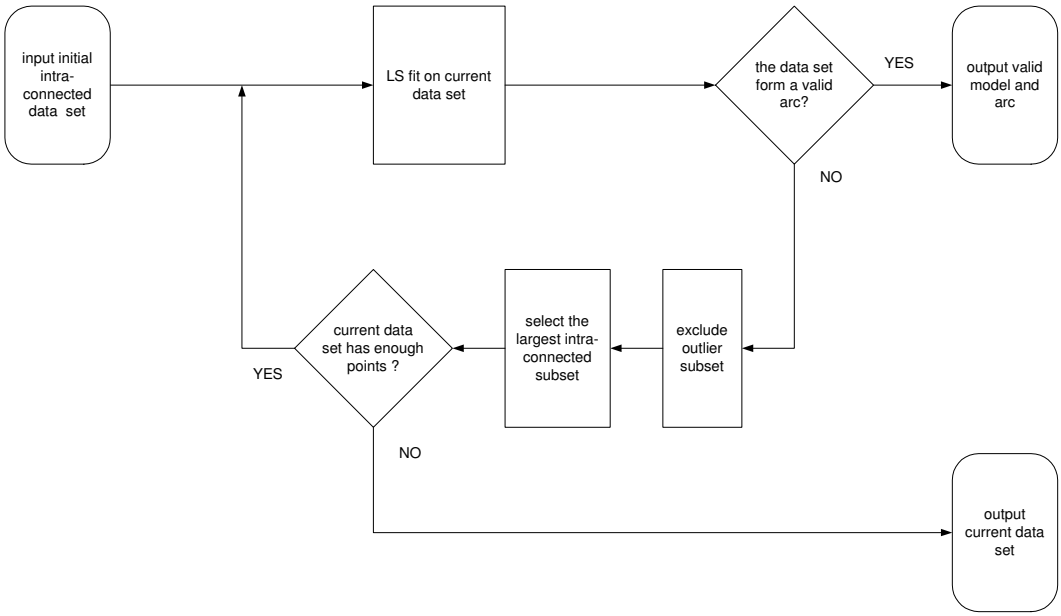


Figure 4.1. Flowchart describing single-model extraction.

7. Determine the success of estimation, output circle model $C(\vec{\alpha}^{(k)})$ and its inlier data set $\Omega^{(k)}$, and go to step 8.

8. End.

Fig. 4.1 shows the flowchart of the single-model extractor. In this extractor, an intra-connected outlier subset $\Delta^{(k)}$ is first removed from the current data set $\Omega^{(k)}$ after a curve model $C(\vec{\alpha}^{(k)})$ is extracted with LS fitting on $\Omega^{(k)}$. The largest intra-connected data subset is extracted from the remaining data set $\Gamma^{(k)}$ and is taken as the updated subset $\Omega^{(k+1)}$ for the following fitting. This process continues until a valid arc is found or there are not enough data left for curve estimation.

4.7 Model evaluation

A group of curve models are extracted by single-model extractor. However, not all detected models are valid. Some valid models have several disconnected arcs in the image. Due to the introduction of intra-connected inlier subset, more than one model with same parameters may be extracted from different arcs of the same curve. On other hand, there may be pseudo curve models estimated with inlier subsets of other valid models because the competition between these false models and valid models on these inlier subsets are determined not only by the fitting these valid and false models but also by the distribution of other edge data connected to these inlier subsets. Moreover, pseudo curves are likely to occur when the edge data in SIS are heavily corrupted by noise. Therefore we applied model evaluation (Fig. 4.2) to all detected curve models to search for valid models and discriminate false models.

Let $\mathcal{I} = \{p_i,\ i = 1, 2, \cdots, n\}$, in which p_i is an edge point with coordinate (x_i, y_i) and n is the total number of edge points in this image. Following is the model verification process.

1. Initialize current edge data set $\mathcal{I}^{(1)} = \mathcal{I}$. Let $k = 1$.

2. Merge curves that are extracted from different initial inlier candidate sets when the difference between the curve parameters is sufficiently small such that curve models candidates $C(\vec{\alpha}_i)$ are non-overlapping curves.

3. In the current updated edge data set $\mathcal{I}^{(k)}$, search for the SIS arc $\Omega_s(\vec{\alpha}_i)$ of all available curve model candidates $C(\vec{\alpha}_i)$, then calculate their SIS feature

$\phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i))$.

4. Rank the current available curve model candidates $C(\vec{\alpha}_i)$ according to the value of their SIS feature $\phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i))$, select the curve model $C(\overset{\wedge}{\vec{\alpha}})$ with best SIS feature, i.e.,

$$\phi(\overset{\wedge}{\vec{\alpha}}, \Omega_s(\overset{\wedge}{\vec{\alpha}}) = \max_{\vec{\alpha}^{(i)}} \phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i)). \tag{4.10}$$

5. If the SIS arc $\Omega_s(\overset{\wedge}{\vec{\alpha}})$ satisfies NOI Criterion and SIS Criterion, go to step 7; Otherwise, go to step 6.

6. Remove the selected model $C(\overset{\wedge}{\vec{\alpha}})$ from the list of current available curve model candidates $C(\vec{\alpha}_i)$ because it is not valid. If there are available model candidates on the list to be evaluated, go to step 4; otherwise, go to step 9.

7. The evaluated model $C(\overset{\wedge}{\vec{\alpha}})$ is a valid model; output this model and remove it from list of current available curve model candidates $C(\vec{\alpha}_i)$, go to step 8.

8. Update edge data set $\mathcal{I}^{(k+1)}$ by

$$\mathcal{I}^{(k+1)} = \mathcal{I}^{(k)} - \Omega(\overset{\wedge}{\vec{\alpha}}), \tag{4.11}$$

where $\Omega(\overset{\wedge}{\vec{\alpha}})$ is the inlier subset of curve model $C(\overset{\wedge}{\vec{\alpha}})$. Let $k = k + 1$, go to step 3.

9. End

During the iterative model evaluation procedure, the curve with the better SIS feature $\phi(\vec{\alpha}_i, \Omega_s(\vec{\alpha}_i))$ has the opportunity to choose its non-overlapping inliers

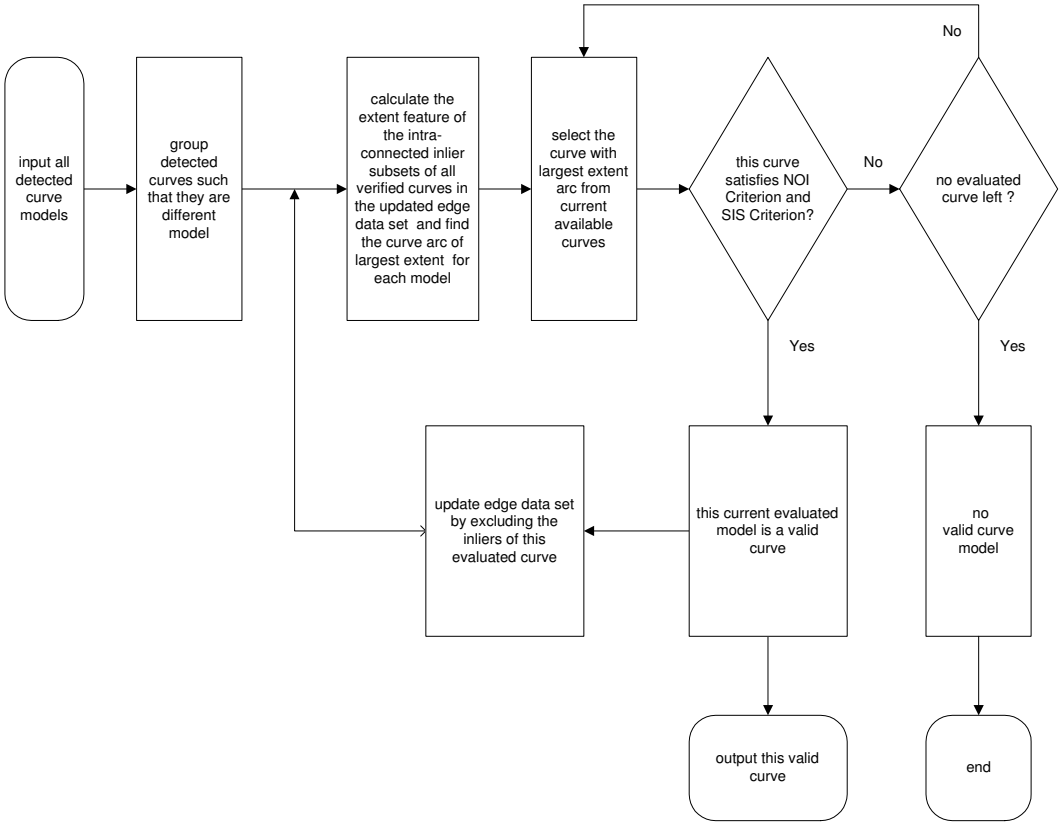


Figure 4.2. Flowchart describing model evaluation

before others. It is an efficient mechanism to prevent false curves from being misclassified as valid because they are usually formed by inliers of other valid models and thus their SIS features are smaller than those of valid curves. In this way those pseudo curves are verified after those genuine models. Hence the edge data that support them have been evaluated as the inliers of valid models verified before them and they cannot benefit from those edge data. This model evaluation scheme not only discriminates against the pseudo curves but also find the corresponding non-overlapping inlier set of the valid models.

4.8 Fitting multiple curves

The flowchart of Fig. 4.3 describes the multiple-model estimator that can detect unknown number of multiple touching or intersecting linearly parameterizable curves even there are an arbitrarily high percentage of *a priori* unknown outliers. The estimator consists of three major steps: the segmentation of the original data into intra-connected sets, extracting valid curves with single-model extractor, and evaluating the detected curves.

Let $\mathcal{I} = \{p_i,\ i = 1, 2, \cdots, n\}$, in which p_i is an edge point with coordinate (x_i, y_i) and n is the total number of edge points in this image. The multiple-model fitting algorithm is presented in the following.

1. Examine the connectivity of all edge data in the image and divide them into intra-connected subsets, G_j, i.e.,

$$\mathcal{I} = \cup\, G_j, \quad \text{where } G_j \cap G_i = \emptyset \text{ for any } j \neq i. \tag{4.12}$$

These subsets G_j are listed into current intra-connected subsets.

2. Choose the largest subset from the list of current intra-connected subsets as the initial inlier candidates subset S.

3. Apply the single-model extractor in the initial inlier candidates subset S.

4. If a valid model $C(\vec{\alpha})$ is extracted from subset S, record the model and update the subset S by removing the curve arc Ω of model $C(\vec{\alpha})$, i.e. $S = S - \Omega$. Separate S into intra-connected subsets, Z_j,

$$S = \cup\, Z_j, \quad \text{where } Z_j \cap Z_i = \emptyset \text{ for any } j \neq i. \tag{4.13}$$

Add these intra-connected subsets, Z_j into the list of current intra-connected subsets.

If no valid curve model is extracted in S, update the subset S by removing subset Ω' output by the single-model extractor, i.e. $S = S - \Omega'$. Separate S into intra-connected subsets, Z_j,

$$S = \cup\, Z_j, \quad \text{where } Z_j \cap Z_i = \emptyset \text{ for any } j \neq i. \tag{4.14}$$

Add these intra-connected subsets, Z_j, into the list of current intra-connected subsets.

5. If there is a subset in current intra-connected subsets list, go to step 2. If none of them in the list is large enough for model extraction, go to step 6.

6. Verify all extracted models with model evaluation module.

7. Output valid models.

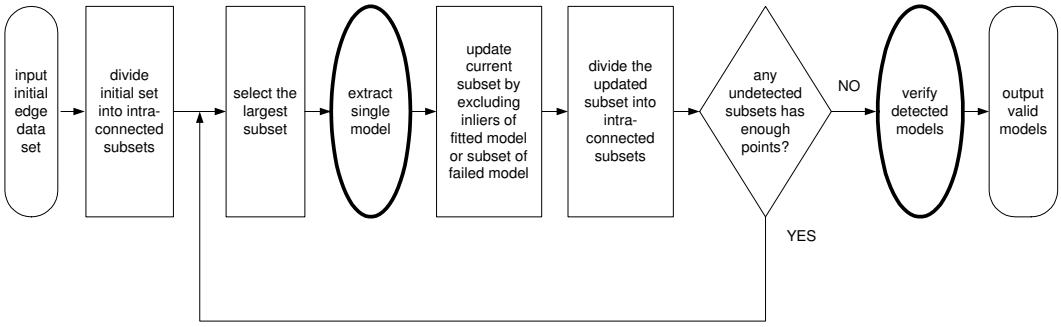


Figure 4.3. Flowchart describing multiple-model estimation.

In the process of the multiple-model estimation, the current edge data set is updated by excluding the inliers of the models previously extracted by the single-model extractor. Thus, all the underlying models in the image can be reconstructed sequentially by repeatedly applying the single-model extractor to the updated edge data set. Subsequently, all the extracted circle candidates are verified to be either desired models or false models.

4.9 Conclusion

In this chapter, a connectivity-based algorithm for fitting multiple curves in machine vision applications is presented. The problem of false-curve detection is effectively solved by using valid arcs, which are intra-connected data subsets that agree with the curve models within a specified error, as the criterion for valid curve. We also propose an efficient outlier search strategy, again based on pixel connectivity, that greatly speeds up the fitting process. In the proposed algorithm, a single-model extractor that seeks meaningful arcs is applied to the current edge points of the image. It does not require a good initial guess but, by deleting

outliers iteratively, drives the estimate to a good curve model candidate, namely, one that has an arc formed by an intra-connected edge-point set selected by the iterative estimation process. In the multiple-model estimator, the current edge data set is updated by excluding the inliers of the models previously extracted by the single-model extractor. Thus, all the underlying models in the image can be reconstructed sequentially by repeatedly applying the single-model extractor to the updated edge data set. Subsequently, all the extracted model candidates are verified to be either desired models or false models.

Four major distinguishing features of this multiple-model estimator are: (1) robustness against an arbitrarily high percentage of outliers; (2) ability to handle an *a priori* unknown number of meaningful models in the edge data without any statistical information of the inlier and outlier distribution; and (3) effective fitting of multiple intersecting or touching curves; (4) effective prevention of false models from being extracted.

Chapter 5

Multiple Circle Fitting

5.1 Introduction

The detection of multiple intersecting or occluded geometric shapes such as circles in a complex image is an important task in computer vision. In this chapter, we apply the connectivity-based multiple curve estimator discussed in Chapter 4 to the fitting of unknown number of occluded or intersecting multiple circles [40]. The advantages of this estimator for circle fitting are analyzed. Experimental results with real images are also presented.

5.2 LS estimator and valid circular arc for circle fitting

Before the application of the iterative weighted LS approach (Section 3.6.1) for multiple circle fitting, we need to define connectivity-based LS estimator, error

and circular arc validity.

Let the circle be represented as

$$(x-a)^2+(y-b)^2=r^2, \tag{5.1}$$

where (a,b) is the center of circle and r is the radius. To facilitate the use of the LS method, we describe the circle in the form

$$f(\vec{\alpha},x,y)=-\alpha_1 x-\alpha_2 y-\alpha_3+x^2+y^2=0, \tag{5.2}$$

where

$$\vec{\alpha}=[\alpha_1\ \ \alpha_2\ \ \alpha_3]^T,\ a=\frac{1}{2}\alpha_1,\ b=\frac{1}{2}\alpha_2,\ r^2=\alpha_3+a^2+b^2 \tag{5.3}$$

are the unknown parameters that need to be estimated.

Let $\Omega=\{(x_i,y_i),\ \ i=1,2,\cdots,q\}$ denote the edge data set for LS fitting with q being the number of edge points in Ω. For simplicity, we use the matrix representation of (5.2),

$$\vec{Y}=\mathbf{X}\vec{\alpha}, \tag{5.4}$$

where

$$\vec{Y}=\begin{bmatrix} x_1^2+y_1^2 \\ x_2^2+y_2^2 \\ \vdots \\ x_q^2+y_q^2 \end{bmatrix}, \tag{5.5}$$

$$\vec{\alpha}=\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}, \tag{5.6}$$

$$\mathbf{X} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_q & y_q & 1 \end{bmatrix}. \tag{5.7}$$

The algebraic residual of edge point (x_i, y_i) is

$$\begin{aligned} e_i &= f(\vec{\alpha}, x_i, y_i) \\ &= x_i^2 + y_i^2 - \alpha_1 x_i - \alpha_2 y_i - \alpha_3. \end{aligned} \tag{5.8}$$

Then the matrix form of (5.8) is

$$\vec{Y} = \mathbf{X}\vec{\alpha} + \vec{e}, \tag{5.9}$$

where

$$\vec{e} = [e_1, e_2, \cdots, e_q]^T. \tag{5.10}$$

The total deviation of the edge points in Ω is defined as

$$\begin{aligned} E(\vec{\alpha}, \Omega) &= \sum_{(x_i, y_i) \in \Omega} f^2(\vec{\alpha}, x_i, y_i) \\ &= \sum_{(x_i, y_i) \in \Omega} e_i^2 = \vec{e}^T \vec{e}, \end{aligned} \tag{5.11}$$

which is actually the sum of the squared residuals. The problem of finding a circular model that best fits the edge data set Ω in the sense of minimizing the total deviation $E(\vec{\alpha}, \Omega)$ is a typical LS fitting problem, which is equivalent to solving the equations

$$\mathbf{X}^T \mathbf{X} \vec{\alpha} = \mathbf{X}^T \vec{Y}, \tag{5.12}$$

where $(\cdot)^T$ denotes the matrix transpose.

For the estimation of circles, the point error $g(\vec{\alpha}, P)$, which describe the closeness of an edge point $P(x, y)$ to the circle $C(\vec{\alpha})$, is defined as,

$$g(\vec{\alpha}, P) = \frac{1}{r}\left|\sqrt{(x-a)^2 + (y-b)^2} - r\right|, \tag{5.13}$$

where (a, b) is the circle center and r is the radius of the circular model $C(\vec{\alpha})$. A point $P(x, y)$ is an inlier if its error is sufficiently small, i.e.,

$$g(\vec{\alpha}, P) < T_r, \tag{5.14}$$

where T_r is an error threshold. A set error function $G(\vec{\alpha}, \Omega)$, which describes the fitness of the data subset Ω to the circular model $C(\vec{\alpha})$, is defined as

$$G(\vec{\alpha}, \Omega) = \max_{P_i \in \Omega} g(\vec{\alpha}, P_i). \tag{5.15}$$

The data set Ω is an inlier set of circle $C(\vec{\alpha})$ if

$$G(\vec{\alpha}, \Omega) < T_r. \tag{5.16}$$

With respect to a circular model $C(\vec{\alpha})$, Δ, the expanded neighborhood of an extreme point of a given data set Ω, is an intra-connected data subset such that

$$\min_{p_i \in \Delta} g(\vec{\alpha}, p_i) > G(\vec{\alpha}, \Gamma), \tag{5.17}$$

where $\Gamma = \Omega - \Delta$. The largest expanded neighborhood of the extreme point is chosen as the outlier subset.

The subtended angle is a measure of the significant extent of a circular arc. It is independent of the size of the circle and is used as a criterion for determining the validity of a circular arc.

To summarize, the data set Ω forms a valid arc of circle $C(\vec{\alpha})$, if

1. Ω is an intra-connected data set that satisfies the Set Connection Criterion.

2. Ω is an inlier set of circle $C(\vec{\alpha})$, i.e., all edge data in Ω satisfy the Inlier Criterion.

3. The subtended angle of arc formed by Ω, $\phi(\vec{\alpha}, \Omega)$, satisfies the SIS Criterion.

4. The number of inliers in set Ω, $N(\vec{\alpha}, \mathcal{N}(\vec{\alpha}, \Omega)) = \mathcal{N}(\vec{\alpha}, \Omega)$, satisfies the NIO Criterion.

5.3 The angle subtended by a circular arc

The subtended angle is an important feature of a circular arc. It is independent of the size of circle and thus is taken as the SIS feature in the SIS Criterion to measure the validity of a circular arc.

For convenience, we assume in this section, that the circle discussed is centered at the origin of the coordinate system, i.e. $a = 0$ and $b = 0$. If the circle center is not at the origin, we translate the coordinate system by

$$x_{new} = x_{old} - a, \tag{5.18}$$

$$y_{new} = y_{old} - b. \tag{5.19}$$

Suppose the point $P(x, y)$ is an edge data in Fig. 5.1(a). Let θ be a positive polar angle measured in a counterclockwise direction from $\overline{OX}$ to $\overline{OP}$ where $\theta \in [0, 2\pi)$.

clearly,

$$\theta = \begin{cases} \arctan\frac{y}{x}, & \text{if } P \text{ is in quadrant } \mathbf{I} \\ \frac{\pi}{2}, & \text{if } x = 0,\ y > 0 \\ \pi + \arctan\frac{y}{x}, & \text{if } P \text{ is in quadrant } \mathbf{II} \text{ or } \mathbf{III} \\ \frac{3}{2}\pi, & \text{if } x = 0,\ y < 0 \\ 2\pi + \arctan\frac{y}{x}, & \text{if } P \text{ is in quadrant } \mathbf{IV} \end{cases} \cdot \tag{5.20}$$

Fig. 5.1(b) shows a circle centered at O and its two arcs $\widehat{BC}$ and $\widehat{DA}$. The arc $\widehat{BC}$ does not cross the axis $\overline{OX}$, hence its subtended angle is $\phi_2 = \theta_B - \theta_C$. On the contrary, the arc $\widehat{DA}$ crosses the axis $\overline{OX}$, and therefore its subtended angle is $\phi_1 = 2\pi + \theta_A - \theta_D$.

The recognition of endpoints of the arc is a key to the computation of subtended angle of the arc. However, it is not an easy task to find endpoints. The coordinates of two endpoints may not necessarily be the extreme values among that of interior points of the arc. What is more, the polar angles of arc endpoints may not necessarily be the largest angle or the smallest one. For example, in arc $\widehat{DA}$, both the values of θ_D and θ_A are in the middle compared to those of edge points close to the axis $\overline{OX}$ in both directions. Fig. 5.1(c) shows a typical case in which both coordinates and polar angles of the arc endpoints are neither the largest nor the smallest among those of the edge points of the arc. Points M and N are the endpoints of arc $\widehat{NM}$, while Points E, F, G, H and K are the edge points of arc $\widehat{NM}$. For the coordinates of these edge points along the X axis, we have

$$x_G < x_N < x_M < x_E, \tag{5.21}$$

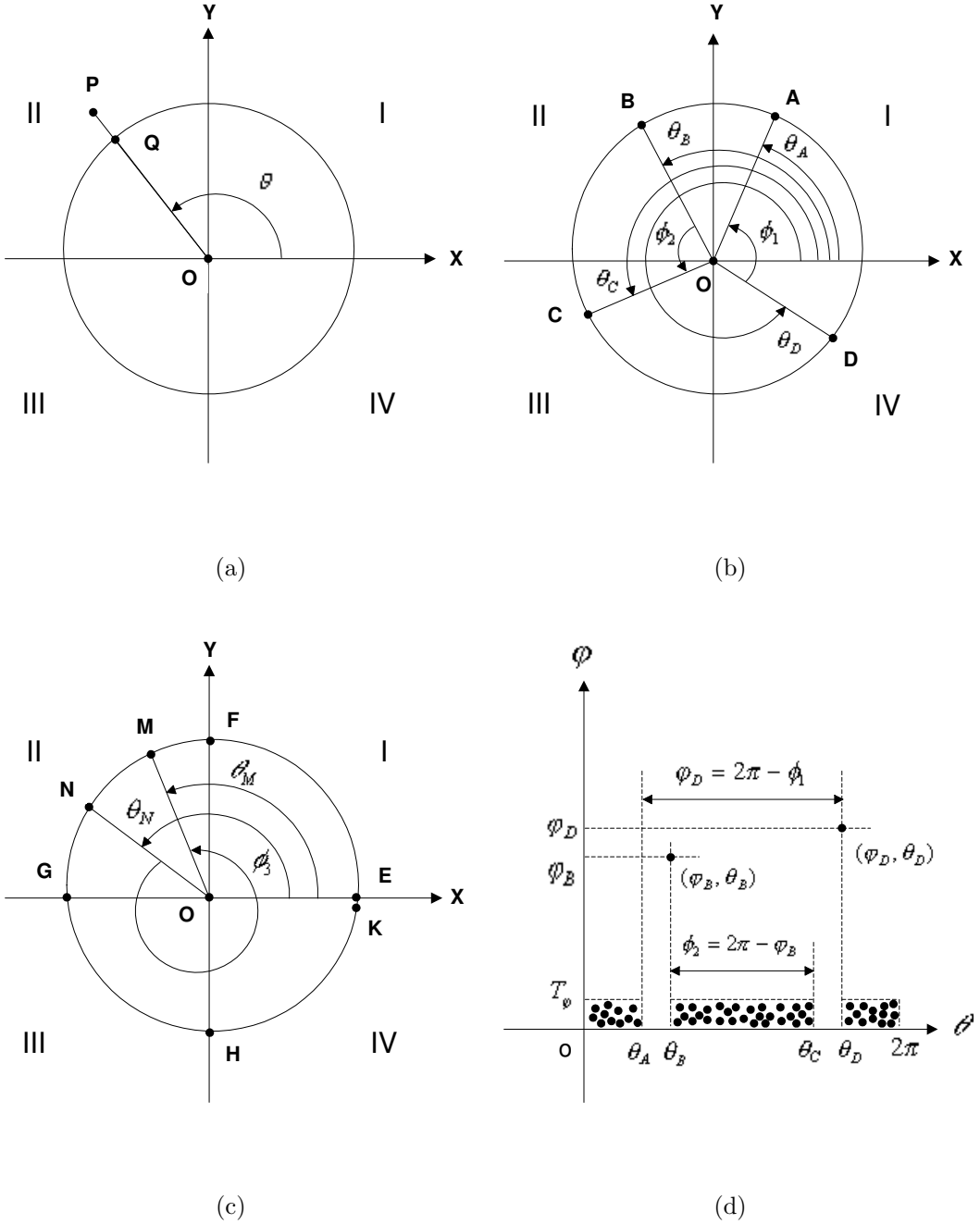


Figure 5.1. (a) The polar angle of edge point, θ; (b) The subtended angles of circular arcs $\widehat{BC}$ and $\widehat{DA}$; (c) The subtended angle of circular arc $\widehat{NM}$; (d) Polar angles and their incremental angles of edge data in arcs $\widehat{BC}$ and $\widehat{DA}$ (black points represent edge points with (θ_i,φ_i)).

for those Y coordinates we have

$$y_H < y_N < y_M < y_F. \tag{5.22}$$

The polar angles of endpoints M and N are in the middle of those of edge points E and K, i.e.,

$$\theta_E < \theta_M < \theta_N < \theta_K. \tag{5.23}$$

Obviously, we can not directly extract the endpoints of arc $\widehat{NM}$ from data set Ω by simply comparing coordinates and polar angles of edge points.

In order to calculate the subtended angle of an arc, we introduce the incremental angle of edge point polar angles. Suppose that data set Ω represents an arc such as $\widehat{DA}$ in Fig. 5.1(b); we first sort the polar angles of all points in Ω in an increasing order,

$$\Theta(\vec{\alpha},\ \Omega) = \{\theta_i | 0 \le \theta_i \le \theta_j \le 2\pi,\ \text{ for all } 1 \le i < j \le n\}. \tag{5.24}$$

Since some points have the same polar angle, the number of polar angles may be less than the number of edge points, i.e., $1 \le n \le q$. For convenience, let $\theta_0 = \theta_n - 2\pi$. Obviously, edge points with polar angle θ_n are actually those with θ_0.

The incremental angle of sorted θ_i, φ_i, is then defined as:

$$\varphi_i = \theta_i - \theta_{i-1},\ \text{ for } i = 1, \cdots, n \tag{5.25}$$

$\varphi_1 = \theta_1 - \theta_0 = 2\pi + \theta_1 - \theta_n$ indicates the angle in a counterclockwise direction from polar angle θ_n to polar angle θ_1. Obviously, each incremental angle is subtended by a piece of circular arc.

With the introduction of φ_i, the subtended angle of the detected arc can therefore be obtained using the following theorem. Its basic idea is that the angle constructed by two connected point and the ellipse center should be below a threshold. Therefore the method of finding circular arc endpoints is connectivity-based.

Theorem 5.1. *Suppose data set Ω is an intra-connected inlier subset of circle $C(\vec{\alpha})$, which contains a circular arc $\upsilon(\Omega)$. Let φ_i be the incremental angle of the sorted polar angle θ_i of edge points in Ω and $T_\varphi = \arccos \frac{2[(1-T_r)r]^2 - T_c^2}{2[(1-T_r)r]^2}$.*

(1) There is at most one angle φ_k such that

$$\varphi_k = \theta_k - \theta_{k-1} = \max_i \varphi_i > T_\varphi, \tag{5.26}$$

$$0 \leq \varphi_i \leq T_\varphi, \quad \textit{for all } i \neq k. \tag{5.27}$$

The edge points, which have polar angles θ_k and θ_{k-1}, are the endpoints of the arc $\upsilon(\Omega)$. Hence, the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = 2\pi - \varphi_k. \tag{5.28}$$

(2) If no incremental angles is larger than T_φ, i.e.,

$$0 \leq \varphi_i \leq T_\varphi, \quad i = 1, \cdots, n, \tag{5.29}$$

then the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) \geq 2\pi - T_\varphi. \tag{5.30}$$

(For proof, see Appendix A.)

Fig. 5.1(d), which reveals the relation between θ and φ, illustrates Theorem 5.1. Arcs $\widehat{BC}$ and $\widehat{DA}$ correspond to case (1) in this theorem. $\varphi_D = \theta_D - \theta_A$, which is

the only angle larger than T_φ, is of key importance for the recognition of points A and D, the endpoints of arc $\overset{\frown}{DA}$. On the other hand, $\varphi_B = 2\pi + \theta_B - \theta_C > T_\varphi$, helps to find points B and C, the endpoints of arc $\overset{\frown}{BC}$. Therefore, the subtended angles of arc $\overset{\frown}{BC}$ and $\overset{\frown}{DA}$ are $\phi(\overset{\frown}{BC}) = 2\pi - \varphi_B$ and $\phi(\overset{\frown}{DA}) = 2\pi - \varphi_D$ respectively.

Based on this theorem, we then develop an algorithm for detecting the subtended angle of Ω an intra-connected inlier subset of circle $C(\vec{\alpha})$:

1. Compute the polar angle θ_i of edge points $(x_i,\ y_i)$ in Ω.

2. Sort the polar angle θ_i in an increasing order.

3. Calculate φ_i, the incremental angle of the sorted polar angle θ_i, for each edge point in Ω.

4. Search for the pair of edge points $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ with largest φ_i, i.e., $\varphi_k = \max_i \varphi_i$

5. if $\varphi_k > T_\varphi$, then the subtended angle of data set Ω, $\phi(\vec{\alpha}, \Omega) = 2\pi - \varphi_k$, and the edge points pair $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ are the endpoints of arc Ω; if $\varphi_k \leq T_\varphi$, then $\phi(\vec{\alpha}, \Omega) \geq 2\pi - T_\varphi$, which indicates that no endpoints of this arc is found.

5.4 Analysis

In this section, we discuss the performance of the proposed algorithm in the presence of outliers and noise and demonstrate the importance of using pixel connectivity in circle fitting. Comparisons between the algorithm and other curve-fitting

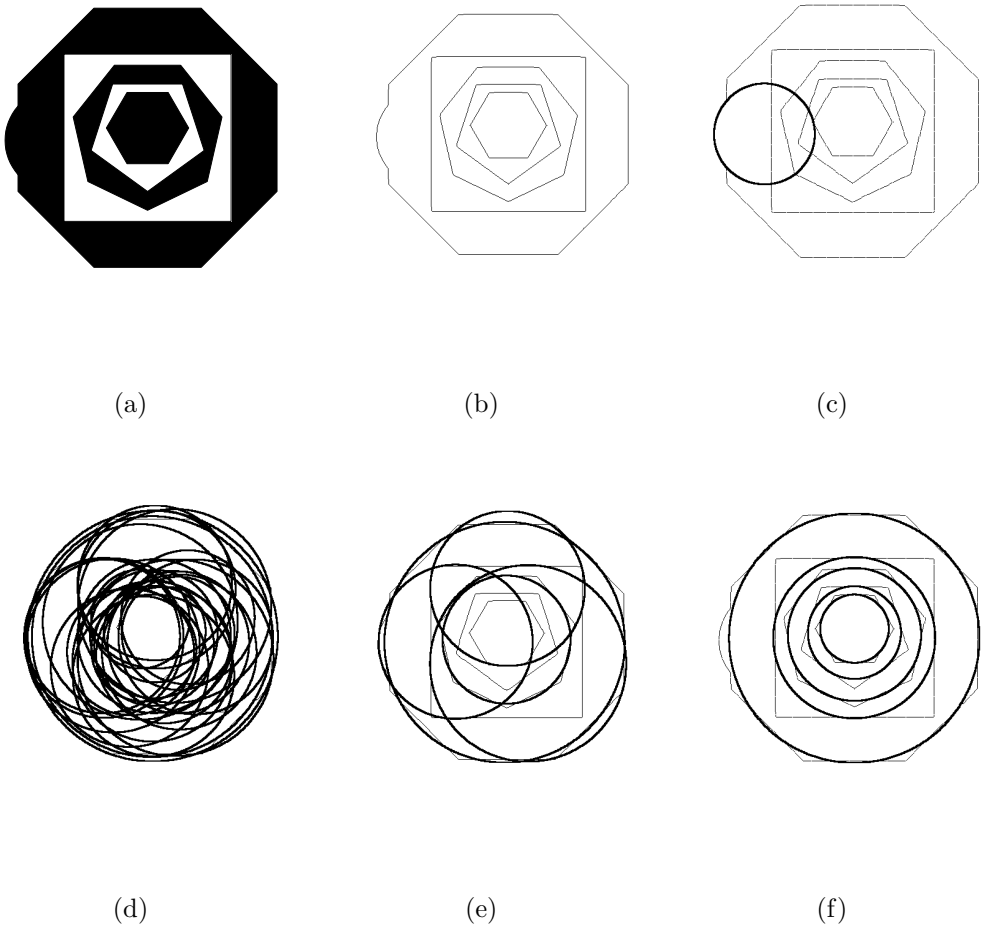
methods are also presented.

5.4.1 Outliers

The presence of outliers, especially if they are found in overwhelming numbers, is one of the major reasons for false detection in most curve-fitting algorithms. To evaluate the ability of our method to deal with outliers, we employ the synthetic test image of Fig. 5.2(a), which comprises one partially occluded circle and several polygons. The edge image (Fig. 5.2(b)) is obtained by applying the Canny edge detector. The outliers, comprising 96.7% of the total edge data, greatly outnumber the inliers. Fig. 5.2(c) shows that the circle is correctly extracted by our algorithm and no false circles are mistakenly fitted.

If pixel connectivity is disregarded, a large number of spurious circles are detected (Fig. 5.2(d)). The detected circular models tend to touch as many sides of the polygons as possible due to the least-squares characteristic and the influence of all the edge data from the polygons. A very different result is obtained if pixel connectivity is taken into account; the task of detecting meaningful circles then becomes one of seeking intra-connected subsets that form valid circular arcs.

It is clear that robust regression methods will not succeed with this example since they require outliers to number less than 50%. The likelihood of extracting the true circle by LS-based methods that start circle fitting from an initial guess is small [8], [7], [30]. This is due to the fact that the probability of randomly selecting an initial edge data subset that is located exactly on the true circle is below 3.3%. In contrast, our algorithm does not involve an initial guess and commences circle



(a) (b) (c)

(d) (e) (f)

Figure 5.2. Example 1: (a) original image of size 512×512; (b) edge data; (c) circles detected by the proposed method; (d) circles detected if pixel connectivity is not considered; (e) circles detected in (d) that have more inliers than the true circle; (f) subset of the false circles detected by the Hough transform whose inliers outnumber that of the true circle.

fitting with the full set of edge data. The result of every experiment with our approach is identical and only the true circle in the test image is successfully extracted.

The Hough transform uses the scores accumulated in the cells of the parameter space to evaluate the validity of circles represented by the cells. The score of a cell is the number of edge points lying on the circle whose parameters correspond to the cell location. The ten circles in Figs. 5.2(e) and (f) have more inliers than the true circle and are a subset of all the false circles detected by the HT. These ten circles will be chosen ahead of the true circle. Furthermore, if we know *a priori* there is only one circle in the image, the HT is most unlikely to detect it.

The random Hough transform randomly samples three edge data each time from the image space and maps this triplet to a cell in parameter space. The probability of choosing three edge points from the true circle is less than $0.033^3 = 3.6 \times 10^{-5}$, i.e., in a random selection of $100,000$ triplets only four triplets are from the true circle. It seems that extracting the true circle here is almost an impossible task for the RHT. It should also be noted that the probability of sampling three points from the largest false circle of Fig. 5.2(f) is more than 100 times that of the true circle in Fig. 5.2(c); thus the RHT is much more likely to detect the former than the latter.

5.4.2 Criterion for determining circle validity

The criterion for evaluating the validity of detected circles is an important component of curve fitting algorithms. It plays a key role in preventing false detections.

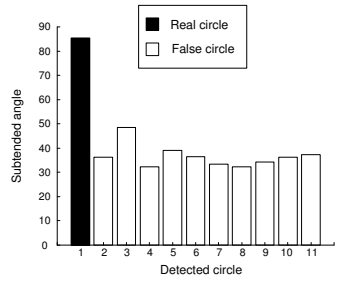
The validity criterion frequently used in HT-based methods is the cell score in parameter space, i.e., the total number of inliers of the evaluated circle. However, this criterion is significantly influenced by the circle size and is thus not reliable. To negate the effect of circle size, the ratio of the total inlier number to the circumference can be considered. However, when the false circles are supported by polygons (Fig. 5.2(e)), this ratio fails to extract the true circle and may even become more unreliable in the regions of high inlier density.

Our proposed algorithm uses the subtended angle of the largest arc of the detected circle to determine circle validity. It is clearly effective in preventing the detection of false circles that are large or touch the sides of polygons. It is also unaffected by the uneven distribution of edge data.

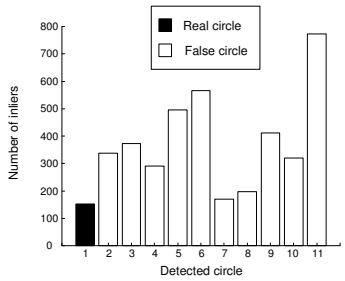
Fig. 5.3 compares the performance of the three validity criteria in evaluating the detected circles of the first test image. The graphs show the criterion values for each of the detected circles, 1 to 11, where circle 1 is the true circle. It is seen in Fig. 5.3(a) that the subtended angle of the largest arc of the true circle is above 75° while those of the false circles (circles 2-11) are less than 50°. Hence the true circle is easily identified. From Figs. 5.3(b) and (c), it is clear that both the number of inliers and the ratio of the total inlier number to the circumference are ineffective in identifying the true circle.

5.4.3 Overlapping Circles

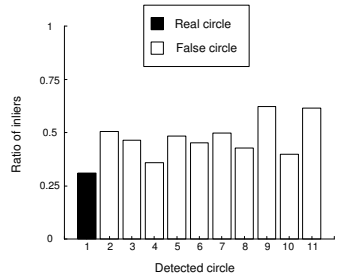
The presence of overlapping circles is a special form of the outlier problem. For every circle, the inliers of other overlapping circles are outliers. The test image of



(a)

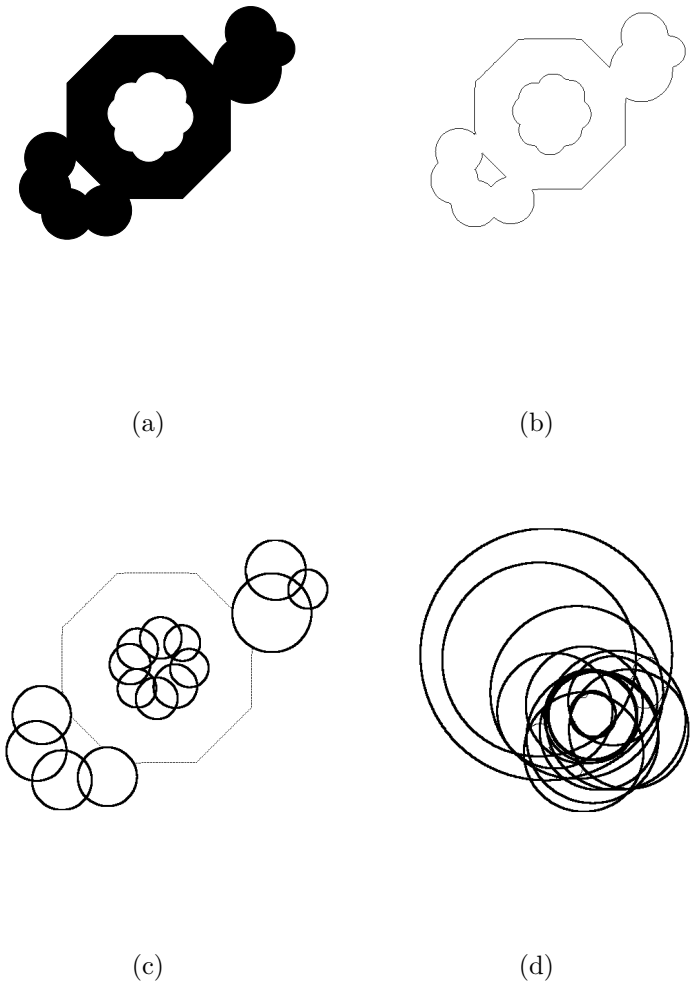


(b)



(c)

Figure 5.3. Comparison of different validity criteria: (a) angle subtended by the largest arc; (b) total number of inliers; (c) ratio of the total number of inliers to the circumference. Circle 1 is the detected circle in Fig. 5.2(c), circles 2-6 are those in Fig. 5.2(e) with increasing radii, circles 7-11 are those in Fig. 5.2(f) also with increasing radii.



(a) (b)

(c) (d)

Figure 5.4. Example 2: (a) original image of size 512×512; (b) edge data; (c) circles detected by the proposed method; (d) circles detected without considering pixel connectivity and whose inliers exceed 36 (the number of inliers of the smallest true circle).

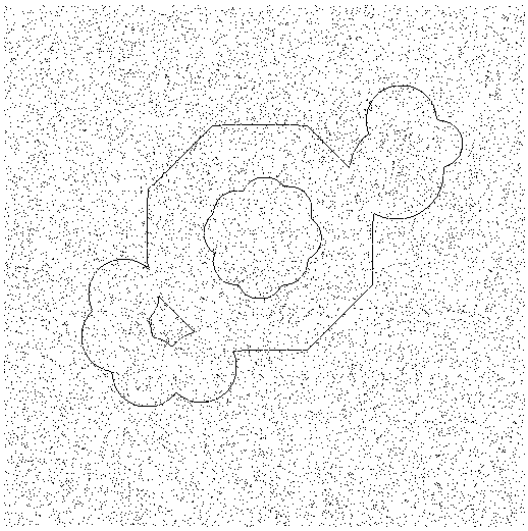
Fig. 5.4(a) is used to demonstrate the effectiveness of the proposed algorithm in such a situation. It contains overlapping circles and circles occluded by an octagon. The edge image obtained with the Canny edge detector is shown in Fig. 5.4(b).

In Fig. 5.4(c), the result obtained with our algorithm is presented. All 15 overlapping incomplete circles are found with the smallest circle containing 36 inliers. If pixel connectivity is ignored, a large number of false circles would be the outcome. Fig. 5.4(d) shows the detected circles that consist of at least 36 inliers. None of the true circles is extracted. It is worth pointing out that the 16 false circles are very likely to be selected if the HT is used. Unlike HT-based methods, our algorithm successfully solves the problem of detecting multiple overlapping circles.

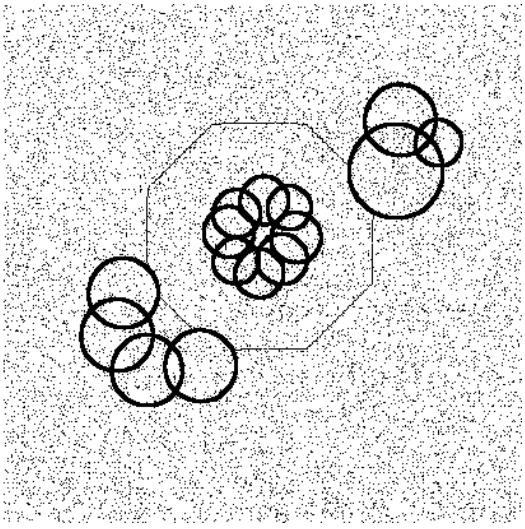
5.4.4 Noise

Noise is another important cause of unsuccessful circle detection. Since our algorithm is based on LS estimation, it inherits the robustness of the LS method against Gaussian noise. Therefore we focus here on salt-and-pepper noise, which severely distorts the detection in HT-based algorithms.

Fig. 5.5(a) shows the edge image of the second example corrupted by 10% salt-and-pepper noise. In this noisy image, none of the inlier percentages (the number of inliers as a percentage of the total number of edge points used in the fitting) of the 15 true circles exceeds 1.5%, and the sum of the inlier percentages of all the 15 circles is less than 8%. Our algorithm manages to extract all 15 true circles without any false detection (Fig. 5.5(b)) despite an outlier percentage of 98.5%.



(a)



(b)

Figure 5.5. (a) Edge data of Fig. 5.4(b) corrupted by salt-and-pepper noise; (b) circles detected by the proposed method.

Furthermore, we note that for LS-based methods that commence fitting from an initial guess, the probability of poor initialization is more than 92%, which will very likely result in their failure.

The probability of sampling three edge points from a single model of all 15 true circles is below 1.2×10^{-6}. Such a low probability indicates that the RHT is very likely to fail in this example.

It is clear that our algorithm deals very effectively with salt and pepper noise, the reason being that scattered noisy edge data are grouped into small intra-connected subsets that contain very few points. These subsets are rejected during circle fitting as there are too few points to form a valid circular arc.

5.4.5 Computation Time

Computation time is an important topic in curve estimation, especially when the number of edge points in the image is large. In most parameter estimation methods, the computation time depends on the number of edge points of the image, as in the HT method. Since our method is based on connectivity, computation time is influenced not only by the number of edge points but also by many other factors such as the inlier ratio, the number of fitted circles, and the distribution of edge points with regard to connectivity, e.g., the number of intra-connected subsets.

Table 5.1 presents the computation time for multiple circle fitting using the images of Fig. 5.2(c)(image I_A), Fig. 5.4(c)(I_B) and Fig. 5.5(b)(I_C) (MATLAB implementation on a Pentium IV 1.7 GHz PC). It can be seen that the computation time for I_A is less than that for I_B, although there are many more edge points in

Table 5.1. Computation Time of Circle Fitting.

Test Image	Number of Edge Points	Number Of Intra-connected Subsets	Minimum Inlier Percentages	Canny Detection Time (s)	Circle Fitting Time (s)
I_A(Fig. 5.2(c))	3548	5	10.6%	5.2	113
I_B(Fig. 5.4(c))	1554	3	5.9%	7.1	131
I_C(Fig. 5.5(b))	14473	6470	4.3%	7.1	491

the former. One reason is that the number of underlying circles in I_B (15) is much more than that in I_A (1). Another reason is that the minimum inlier percentage of the fitted circles in I_B is only half of that in I_A; therefore the detection of the circle with the minimum inlier percentage in I_B may require more computation time compared with that of the circle in I_A.

When the image is corrupted by noise as in I_C, the number of edge points is 14473, almost ten times that of I_B, but the computation time in I_C is only four times that of I_B. The noise points in I_C are grouped into more than 6000 intra-connected subsets, and hence most of the latter are too small to form valid circle arcs and will be discarded. Most of the added computation time is spent in searching for the connectivity of the new noise points.

The clustering of all edge points into intra-connected subsets before circle estimation requires the detection of connectivity between each pair of edge data. The computational complexity for connectivity detection ranges from $O(n)$ to the worst

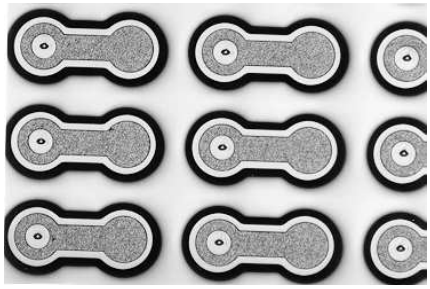
case of $O(n^2)$. The computation burden for outlier search depends on the number of data in the intra-connected subset and the inlier percentage of the valid models. The worst case is that all data in the image are connected, then the complexity for iterative weighted LS methods is $O(n^2)$. In a word, the computational complexity with our method is no more than $O(n^2)$. It is better than the computational complexity of the HT method for circle detection, which is $O(n^3)$ [26].

Generally speaking, computation time is application dependent in our method due to the use of connectivity. The number of edge points in the image is no longer the most important factor that determines computation time. Therefore we may develop application-specific improvements to speed up circle fitting.

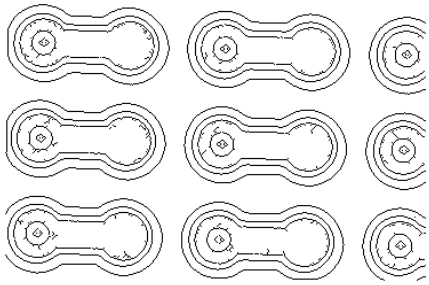
5.5 Experimental results

The proposed multiple-model estimator has been implemented for the reconstruction of occluded and intersecting circles from real binary edge images. A variety of images is used to test the performance and versatility of the algorithm. In the experiments, the edge images are obtained by applying the Canny detector.

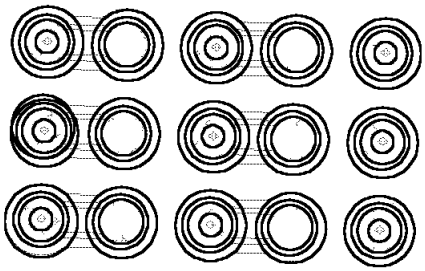
Fig. 5.6(a) shows a real image in which most real circles are intersected by straight lines and hence the number of outliers exceed 50%. The edge image by Canny detector is shown in Fig. 5.6(b). Fig. 5.6(c) presents the estimation results of meaningful circles. Here the connectivity threshold is $T_c = 2.5$, the error threshold $T_r = 0.02$ and the subtended angle threshold $T_\phi = 72°$. All real circles in this image are successfully detected without producing any false circles.



(a)



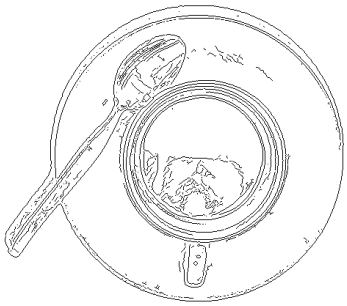
(b)



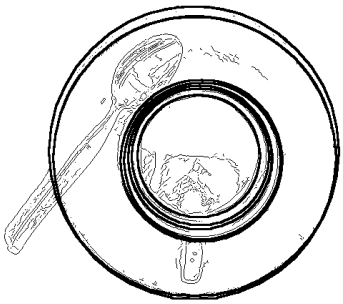
(c)

Figure 5.6. (a) Original image of size 370 × 246; (b) edges detected by Canny detector; (c) extracted circles.

(a)



(b)



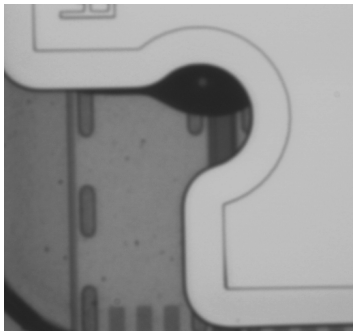
(c)

Figure 5.7. (a) Original image of size 512×480; (b) edges detected by Canny detector; (c) extracted circles.

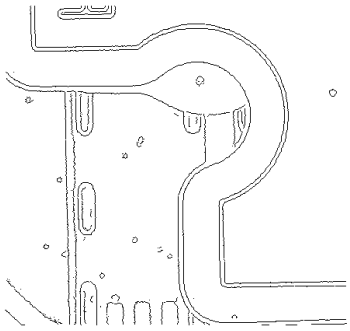
Fig. 5.7(a) shows a real image containing a spoon, a bowl, a cup and a plate and Fig. 5.7(b) the edge image. The desired estimation of circles is achieved by applying the proposed algorithm (Fig. 5.7(c)). With the connectivity threshold $T_c = 2.5$, the error threshold $T_r = 0.02$ and the subtended angle threshold $T_\phi = 180°$, the boundary of the plate and the circle generated by the shadow of the plate are successfully extracted. The concentric circles at the center of the image are due to the edges generated as a consequence of the illumination pattern on the circular objects.

A part of a circuit pattern is presented in Fig. 5.8(a) and its edge image in Fig. 5.8(b). Obviously, there are significant outliers of each real circles. Fig. 5.8(c) shows the successful extraction of circles with the connectivity threshold $T_c = 2.5$, the error threshold $T_r = 0.02$, the subtended angle threshold $T_\phi = 180°$ and the minimum number of inliers threshold $T_N = 150$. All three real circles, whose arcs are larger than semi-circles, are well extracted.

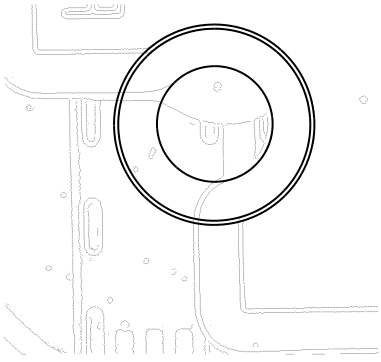
Fig. 5.9 shows different circle fitting results with different threshold selections. Fig. 5.9(a) presents extraction results with $T_N = 0$, i.e., there is no requirement for the minimum number of inliers in a valid circular arc. Hence quite a lot of small circles formed by small groups of data are identified as valid models. In Fig. 5.9(b), the threshold for the angle subtended by a valid circular arc, T_ϕ, is set to 33°, which leads to the wrong detection of the circle with the largest radius in the image. The error threshold T_r is relaxed to 0.1 in Fig. 5.9(c). The consequence is that the inner one of two valid circular arcs in Fig. 5.8(c) is mistakenly classified as the inliers of the exterior valid circular arc. Actually, for the exterior circle model, the inner



(a)

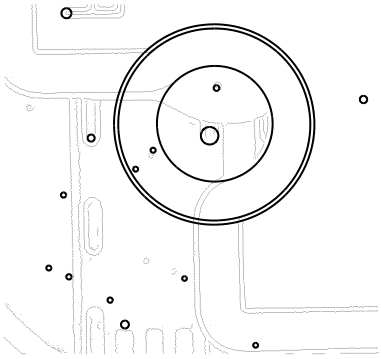


(b)

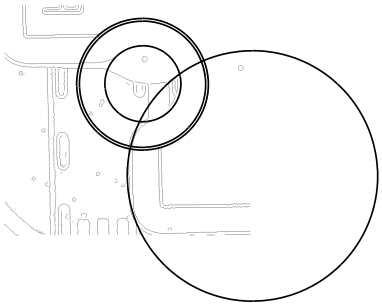


(c)

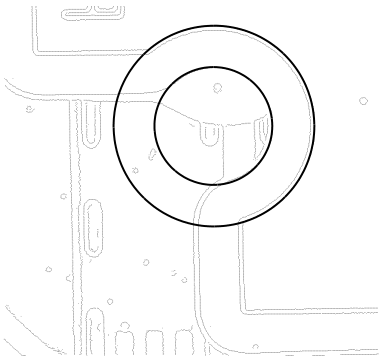
Figure 5.8. (a) Original image of size 480 × 426; (b) edges detected by Canny detector; (c) extracted circles.



(a)



(b)



(c)

Figure 5.9. (a) Extracted circles with $T_N = 0$; (b) extracted circles with $T_\phi = 33°$; (c) extracted circles with $T_r = 0.1$.

arc satisfies the Inlier Criterion (Criterion 3.1) with $T_r = 0.1$. Thus the inner valid circular arc cannot be extracted. Fig. 5.9 illustrates the fact that the selection of thresholds is very important to the correctness of extraction results. Inappropriate thresholds may lead to the false circle detections.

In the three examples, the outlier percentage is greater than 50% for the circles that intersect other curves. Therefore robust regression methods will not work well. The presence of a variety of curves in these examples will cause the HT and its variants to generate false circles formed by the edges of straight lines and other curves. Our proposed algorithm, however, successfully extracts real circles while avoiding false circles by searching for valid circular arcs.

5.6 Conclusion

In this chapter, we apply the connectivity-based algorithm for fitting multiple touching or intersecting circles with appropriately selected LS estimator, error and feature of valid circular arc. An efficient method to search for the endpoints of circular arc is presented based on the connectivity feature of the circular arc.

Analysis of algorithm performance and experimental results show five major advantages of this multiple-model estimator for circle fitting: (1) it can effectively deal with outliers more than 50% of all edge data; (2) it can extract an *a priori* unknown number of meaningful circles in the image; and (3) it can handle fitting of multiple intersecting or touching circles; (4) it can prevent false circles from being extracted; (5) it reduces the search field for model fitting using pixel connectivity,

thus reduce computational complexity significantly.

Chapter 6

Multiple Ellipse Fitting

6.1 Introduction

Ellipse fitting in a noisy image is of great importance for many industrial applications [15], [37]. Examples include cell counting in the detection of breast cancer and particle classification in an industrial setting.

Most methods perform ellipse estimation by LS fitting to a general conic model and rejecting non-elliptical fits. Their performance is unsatisfactory when the edge data are not strictly elliptical and corrupted by noise. Fitzgibbon *et al.* [14] proposed a direct LS ellipse-specific fitting method with the equality constraint $4ac - b^2 = 1$, which yields only one elliptic solution. It is robust, efficient and non-iterative. However, it cannot handle multiple occluded or touching ellipses.

An ellipse is a typical kind of linearly parameterizable curve. Therefore the connectivity-based multiple curve estimator discussed in Chapter 4 is applied to extract an unknown number of occluded or touching multiple ellipses. In the

single-model extractor, we take the direct LS ellipse-specific fitting method as the LS estimator for ellipse model fitting. In this chapter we analyze the advantages of this estimator for ellipse fitting. Experimental results with real images shows that our connectivity-based estimator is efficient in detection of multiple occluded or touching ellipses and reliable in situation when outliers exceed 50%.

6.2 Elliptic model

Let (x, y) be an edge point in image space I shown in Fig. 6.1. The nominal model of ellipse centered at $(x_0,\ y_0)$ with both major and minor axes parallel to the axes of coordinate system can be described by

$$\frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} = 1, \tag{6.1}$$

where a and b are length of semimajor and semiminor axes along the X and Y axes of the coordinate system.

The ellipse described by the nominal ellipse model (6.1) can also be represented in polar coordinates

$$x = x_0 + \rho(\theta)\ \cos\theta, \tag{6.2}$$

$$y = y_0 + \rho(\theta)\ \sin\theta, \tag{6.3}$$

where θ is the point angle of edge data $P(x, y)$ in Fig. 6.1 and

$$\rho(\theta) = \frac{ab}{\sqrt{b^2\ \cos^2\theta + a^2\ \sin^2\theta}}, \tag{6.4}$$

is a function of θ. When the ellipse is a circle, $\rho(\theta) = r$ is the radius of the circle. From the polar nominal model (6.2) and (6.3), the ellipse can be regarded as a

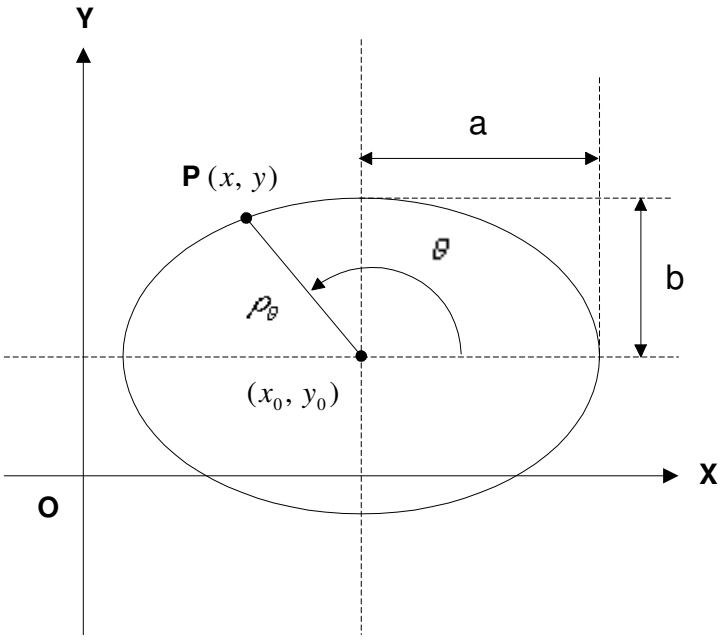


Figure 6.1. Nominal model of an ellipse centered at (x_0, y_0).

generalized circle with the radius varying at different point angles.

Suppose the point $P(x, y)$ is an edge datum and θ be its point angle in Fig. 6.1. Clearly,

$$\theta = \begin{cases} \arctan \frac{y-y_0}{x-x_0}, & \text{if } x > x_0,\ y > y_0 \\ \frac{\pi}{2}, & \text{if } x = x_0,\ y > y_0 \\ \pi + \arctan \frac{y-y_0}{x-x_0}, & \text{if } x < x_0 \\ \frac{3}{2}\pi, & \text{if } x = x_0,\ y < y_0 \\ 2\pi + \arctan \frac{y-y_0}{x-x_0}, & \text{if } x > x_0,\ y < y_0 \end{cases} . \tag{6.5}$$

In real applications, the semimajor and semiminor axes of an ellipse may not necessary be parallel to the axes of coordinate system, i.e., the ellipse is rotated. Therefore an implicit second order linearly parameterizable curve model is taken

as the general conic model of the ellipse

$$\begin{aligned} f(\vec{\alpha}, x, y) &= \alpha_1 x^2 + \alpha_2 xy + \alpha_3 y^2 + \alpha_4 x + \alpha_5 y + \alpha_6 \\ &= 0, \end{aligned} \tag{6.6}$$

where $\vec{\alpha} = [\alpha_1 \ \ \alpha_2 \ \ \alpha_3 \ \ \alpha_4 \ \ \alpha_5 \ \ \alpha_6]$. We call $f(\vec{\alpha}, x, y)$ the algebraic distance of an edge point (x, y) to the curve $f(\vec{\alpha}, x, y) = 0$.

6.3 Translation of the elliptic models

In most applications, due to the rotation of the ellipse, the general elliptic model (6.6) is the only available model in the original coordinate system, and this general model is convenient for LS estimation. However, it is difficult to obtain the features of the ellipse from it, such as the lengths of the semimajor and semiminor axes and the center of the ellipse. On the other hand, the nominal model (6.1) presents the features of the ellipse but cannot be directly used in the LS estimation. Hence there is a need to translate the general model to the nominal model or *vice versa*.

6.3.1 Translation of the elliptic models without rotation

In some applications, the ellipses are just shifted but not rotated, i.e., their semimajor and semiminor axes are still parallel to the axes of coordinate system. Therefore they can be represented by the nominal elliptic model (6.1) without rotation of the axes of coordinate system. The translation of coefficients from the nominal elliptic

model (6.1) to the general model (6.6) is specified by

$$\alpha_1 = \frac{1}{a^2}, \tag{6.7}$$

$$\alpha_2 = 0, \tag{6.8}$$

$$\alpha_3 = \frac{1}{b^2}, \tag{6.9}$$

$$\alpha_4 = \frac{-2x_0}{a^2}, \tag{6.10}$$

$$\alpha_5 = \frac{-2y_0}{b^2}, \tag{6.11}$$

$$\alpha_6 = \frac{x_0^2}{a^2} + \frac{y_0^2}{b^2} - 1. \tag{6.12}$$

The translation of coefficients from the general model (6.6) to the nominal elliptic model (6.1) is achieved by

$$a = \frac{1}{\sqrt{\alpha_1}}, \tag{6.13}$$

$$b = \frac{1}{\sqrt{\alpha_3}}, \tag{6.14}$$

$$x_0 = -\frac{\alpha_4}{2\alpha_1}, \tag{6.15}$$

$$y_0 = -\frac{\alpha_5}{2\alpha_3}. \tag{6.16}$$

It is worthy noting that $\alpha_2 = 0$ is the condition for the translation of general elliptic models to nominal models with Eq. (6.13)-(6.16).

6.3.2 Translation of the elliptic models by the rotation of the coordinate system

In many applications for ellipse fitting, the axes of the ellipses are not parallel to the coordinate axes. In these cases, the coefficient of the term xy is not zero, i.e.,

$\alpha_2 \neq 0$. In order to obtain the features of the ellipse from the general model, we need to derive the nominal model in a different coordinate system for the same ellipse.

Suppose the general model of an ellipse (6.6) is known and $\alpha_2 \neq 0$, we then calculate the rotated angle between axes of both the ellipse and the coordinate system by

$$\psi = \begin{cases} \arctan Q_1, & \text{if } Q_1 > 0 \\ \arctan Q_2, & \text{if } Q_2 > 0 \end{cases}, \tag{6.17}$$

where

$$Q_1 = \frac{2(\alpha_3 - \alpha_1) + \sqrt{4(\alpha_3 - \alpha_1)^2 + 4\alpha_2^2}}{2\alpha_2}, \tag{6.18}$$

$$Q_2 = \frac{2(\alpha_3 - \alpha_1) - \sqrt{4(\alpha_3 - \alpha_1)^2 + 4\alpha_2^2}}{2\alpha_2}. \tag{6.19}$$

A new coordinate system $X'Y'$ is then set up by rotating the axes of current coordinate system XY with angle ψ of (6.17). Fig. 6.2 shows the ellipse in two coordinate systems XY and $X'Y'$. The translation of point coordinates between different coordinate systems is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi \\ -\sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{6.20}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix}, \tag{6.21}$$

where (x, y) and (x', y') are the corresponding coordinates of the same point in coordinate system XY and $X'Y'$.

By substituting Eq. (6.21) into the general model (6.6), the general model (6.6) in XY is turned into a new general model in $X^{'}Y^{'}$

$$f(\vec{\alpha}^{'}, x^{'}, y^{'}) = \alpha_1^{'} x^{'2} + \alpha_2^{'} x^{'} y^{'} + \alpha_3^{'} y^{'2} + \alpha_4^{'} x^{'} + \alpha_5^{'} y^{'} + \alpha_6^{'} = 0, \tag{6.22}$$

where $\vec{\alpha}^{'} = \begin{bmatrix} \alpha_1^{'} & \alpha_2^{'} & \alpha_3^{'} & \alpha_4^{'} & \alpha_5^{'} & \alpha_6^{'} \end{bmatrix}$. The translation of coefficients from the general elliptic model (6.6) in XY plane to the general model (6.22) in $X^{'}Y^{'}$ plane is

$$\alpha_1^{'} = \alpha_1 \cos^2\psi + \alpha_2 \sin\psi\cos\psi + \alpha_3 \sin^2\psi \tag{6.23}$$

$$\begin{aligned} \alpha_2^{'} &= -2\alpha_1 \sin\psi\cos\psi + \alpha_2 \cos^2\psi - \alpha_2 \sin^2\psi + 2\alpha_3 \sin\psi\cos\psi \\ &= 0, \end{aligned} \tag{6.24}$$

$$\alpha_3^{'} = \alpha_1 \sin^2\psi - \alpha_2 \sin\psi\cos\psi + \alpha_3 \cos^2\psi, \tag{6.25}$$

$$\alpha_4^{'} = \alpha_4 \cos\psi + \alpha_5 \sin\psi, \tag{6.26}$$

$$\alpha_5^{'} = -\alpha_4 \sin\psi + \alpha_5 \cos\psi, \tag{6.27}$$

$$\alpha_6^{'} = \alpha_6. \tag{6.28}$$

The nominal model of this ellipse can be obtained by Eq. 6.13-6.16 with the rotated general model $f(\vec{\alpha}^{'}, x^{'}, y^{'})$.

The process to obtain the nominal model of a rotated and translated ellipse from a general conic model can be summarized in the following:

1. Calculate ψ, the rotated angle of the ellipse, with Eq. (6.17).

2. Set up a new coordinate system $X^{'}Y^{'}$ by rotating XY with the angle ψ.

3. Translate the general conic model $f(\vec{\alpha}, x, y)$ in XY plane into a new model $f(\vec{\alpha}^{'}, x^{'}, y^{'})$ in $X^{'}Y^{'}$ such that $\alpha_2^{'} = 0$.

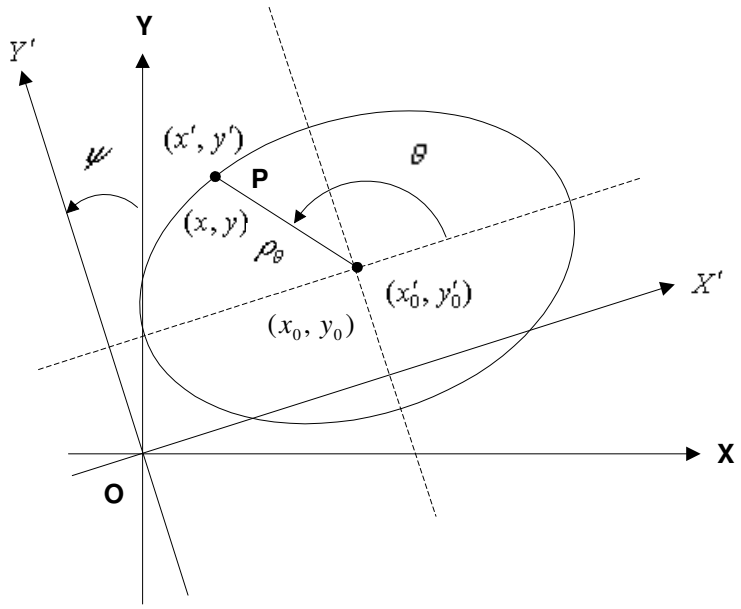


Figure 6.2. The ellipse model in two coordinate system XY and $X'Y'$.

4. Obtain the nominal model of (6.1) in $X'Y'$ plane from general model $f(\vec{\alpha}', x', y')$ by Eq. 6.13-6.16. Obtain the ellipse center and the lengths of the semimajor and semiminor axes.

6.4 Direct LS ellipse-specific fitting

Ellipse fitting methods are divided into two broad categories: clustering (such as the Hough transform) and LS fitting. LS-based methods find the ellipse parameters by minimizing the distance between the edge points and the ellipse. The most commonly used metric is the algebraic distance $f(\vec{\alpha}, x, y)$ defined in (6.6). In this case, the problem of ellipse fitting turns to be an optimization problem by minimizing the sum of squared algebraic distances

$$E_\Omega(\vec{\alpha}) = \sum_{i=1}^{q} f(\vec{\alpha}, x_i, y_i)^2, \tag{6.29}$$

where $\Omega = \{(x_i, y_i),\ i = 1, 2, \cdots, q\}$ denotes the edge data set and where q is the number of edge points in Ω.

The general elliptic model of (6.6) can be represented by

$$\begin{aligned} f(\vec{\alpha}, x_i, y_i) &= \alpha_1 x_i^2 + \alpha_2 x_i y_i + \alpha_3 y_i^2 + \alpha_4 x_i + \alpha_5 y_i + \alpha_6 \\ &= f(\vec{\alpha}, \vec{\mathbf{x}}_i) = \vec{\alpha} \cdot \vec{\mathbf{x}}_i = 0, \end{aligned} \tag{6.30}$$

where

$$\vec{\alpha} = [\alpha_1 \ \ \alpha_2 \ \ \alpha_3 \ \ \alpha_4 \ \ \alpha_5 \ \ \alpha_6]^T, \tag{6.31}$$

$$\vec{\mathbf{x}}_i = \left[x_i^2 \ \ x_i y_i \ \ y_i^2 \ \ x_i \ \ y_i \ \ 1\right]^T. \tag{6.32}$$

In order to avoid the trivial solution $\vec{\alpha} = [0 \ \ 0 \ \ 0 \ \ 0 \ \ 0 \ \ 0]^T$, some constraints for the parameter vector $\vec{\alpha}$ are necessary. The quadratic constraints can be represented by

$$\vec{\alpha}^T \ \mathbf{C} \ \vec{\alpha} = 1 \tag{6.33}$$

where $\mathbf{C}$ is a 6×6 constraint matrix. Then the performance function turns to be

$$\begin{aligned} E_\Omega(\vec{\alpha}) &= \sum_{i=1}^{q} f(\vec{\alpha}, \vec{\mathbf{x}}_i)^2 + \lambda(1 - \vec{\alpha}^T \ \mathbf{C} \ \vec{\alpha}) \\ &= \sum_{i=1}^{q} (\vec{\alpha} \cdot \vec{\mathbf{x}}_i)^2 + \lambda(1 - \vec{\alpha}^T \ \mathbf{C} \ \vec{\alpha}) \\ &= (\mathbf{H} \ \vec{\alpha})^T \mathbf{H} \ \vec{\alpha} + \lambda(1 - \vec{\alpha}^T \ \mathbf{C} \ \vec{\alpha}), \end{aligned} \tag{6.34}$$

where λ is a Lagrange multiplier and $\mathbf{H} = \left[\vec{\mathbf{x}}_1 \ \ \vec{\mathbf{x}}_2 \ \ \cdots \ \ \vec{\mathbf{x}}_q\right]^T$.

Bookstein [5] has shown that if the constraint is quadratic the optimization problem of (6.34) can be solved by seeking the solution of the generalized eigenvalue problem

$$\vec{\alpha}^T \ \mathbf{C} \ \vec{\alpha} = 1 \tag{6.35}$$

$$\mathbf{H}^T\mathbf{H}\,\vec{\alpha} = \lambda\mathbf{C}\,\vec{\alpha}, \tag{6.36}$$

Actually Eq. (6.35) and Eq. (6.36) are obtained by differentiating (6.34) with respect to $\vec{\alpha}$ and λ.

Various constraints have been proposed. Fitzgibbon *et al.* [14] used the equality constraint

$$4ac - b^2 = 1. \tag{6.37}$$

This constraint can be expressed in the form of (6.35) with the constraint matrix

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{6.38}$$

Then the ellipse estimation problem is reduced to finding $\vec{\alpha}$, which minimizes $\|\mathbf{H}\,\vec{\alpha}\,\|^2$ subject to the constraint (6.35). It is in fact the solution of (6.35) and (6.36). Fitzgibbon *et al.* [14] proved that (6.35) and (6.36) with the constraint $\mathbf{C}$ defined in (6.38) has exactly one elliptic solution, which corresponds to the single positive generalized eigenvalue of (6.36).

The direct LS ellipse-specific fitting method is a robust, efficient and non-iterative ellipse fitting approach due to the fact that its constraint $4ac - b^2 = 1$ enables this method to yield a unique elliptical solution. However, it is only valid for extracting a single ellipse and does not work well when there are multiple touching or occluded ellipses.

6.5 Valid elliptic arcs for ellipse fitting

Before the application of the iterative weighted LS approach (Section 3.6.1) for multiple ellipse fitting, we need to define error and elliptic arc validity.

For the estimation of ellipses, the point error $g(\vec{\alpha}, P)$, which describe the closeness of an edge point $P(x, y)$ to the ellipse $C(\vec{\alpha})$ represented by the nominal model (6.1), is defined as,

$$g(\vec{\alpha}, P) = \frac{1}{\rho(\theta)} \left| \sqrt{(x - x_0)^2 + (y - y_0)^2} - \rho(\theta) \right|, \tag{6.39}$$

where (x_0, y_0) is the ellipse center and θ is the point angle of $P(x, y)$ using (6.5). $\rho(\theta)$ in Fig. 6.1 can be obtained with Eq. 6.4. A point $P(x, y)$ is an inlier if its error is sufficiently small, i.e.,

$$g(\vec{\alpha}, P) < T_\rho, \tag{6.40}$$

where T_ρ is an error threshold. A set error function $G(\vec{\alpha}, \Omega)$, which describes the fitness of the data subset Ω to the elliptic model $C(\vec{\alpha})$, is defined as

$$G(\vec{\alpha}, \Omega) = \max_{P_i \in \Omega} g(\vec{\alpha}, P_i). \tag{6.41}$$

The data set Ω is an inlier set of ellipse $C(\vec{\alpha})$ if

$$G(\vec{\alpha}, \Omega) < T_\rho. \tag{6.42}$$

With respect to a elliptic model $C(\vec{\alpha})$, Δ, the expanded neighborhood of an extreme point of a given data set Ω, is an intra-connected data subset such that

$$\min_{p_i \in \Delta} g(\vec{\alpha}, p_i) > G(\vec{\alpha}, \Gamma), \tag{6.43}$$

where $\Gamma = \Omega - \Delta$. The largest expanded neighborhood of the extreme point is chosen as the outlier subset.

The subtended angle is a measure of the significant extent of an elliptic arc. It is independent of the ellipse size and is used as a criterion for determining the validity of an elliptic arc.

To summarize, the data set Ω forms a valid arc of ellipse $C(\vec{\alpha})$, if

1. Ω is an intra-connected data set that satisfies the Set Connection Criterion.

2. Ω is an inlier set of ellipse $C(\vec{\alpha})$, i.e., all edge data in Ω satisfy the Inlier Criterion.

3. The subtended angle of arc formed by Ω, $\phi(\vec{\alpha}, \Omega)$, satisfies the SIS Criterion.

4. The number of inliers in set Ω, $N(\vec{\alpha}, \mathcal{N}(\vec{\alpha}, \Omega)) = \mathcal{N}(\vec{\alpha}, \Omega)$, satisfies the NIO Criterion.

6.6 Angles subtended by elliptic arcs

The subtended angle is an important feature of an elliptic arc. It is independent of the ellipse size and thus is taken as the SIS feature in the SIS Criterion to measure the validity of an elliptic arc.

For convenience, we assume in this section, that the ellipse discussed is centered at the origin of the coordinate system, i.e., $x_0 = 0$ and $y_0 = 0$. Its semimajor and semiminor axes are parallel to the coordinate axes. This indicates that the ellipse can be represented by the nominal model (6.1) with $x_0 = 0$ and $y_0 = 0$. If the

ellipse center is not at the origin, we translate the coordinate system by

$$x_{new} = x_{old} - x_0, \tag{6.44}$$

$$y_{new} = y_{old} - y_0. \tag{6.45}$$

Fig. 6.3(a) shows an ellipse centered at O and its two arcs $\widehat{BC}$ and $\widehat{DA}$. The arc $\widehat{BC}$ does not cross the the axis $\overline{OX}$, hence its subtended angle is $\phi_2 = \theta_B - \theta_C$. On the contrary, the arc $\widehat{DA}$ crosses the axis $\overline{OX}$ and therefore its subtended angle is $\phi_1 = 2\pi + \theta_A - \theta_D$.

The recognition of the endpoints of an arc is a key to the computation of subtended angle of the arc. However, the endpoints of the arc cannot be directly detected by simply comparing coordinates and polar angles of edge points. Fig. 6.3(b) shows a typical case in which both coordinates and polar angles of the arc endpoints are neither the largest nor the smallest among those of the edge points of the arc. Points M and N are the endpoints of arc $\widehat{NM}$, while Points E, F, G, H and K are the edge points of arc $\widehat{NM}$. For the coordinates of these edge points along the X axis, we have

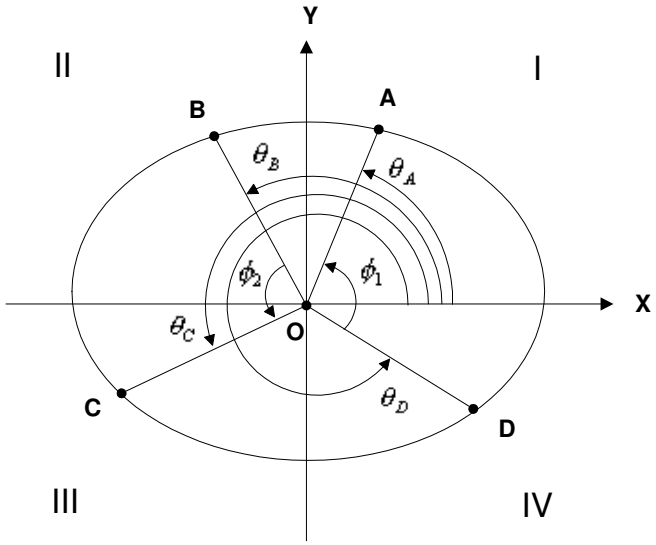
$$x_G < x_N < x_M < x_E, \tag{6.46}$$

and for the Y coordinates we have

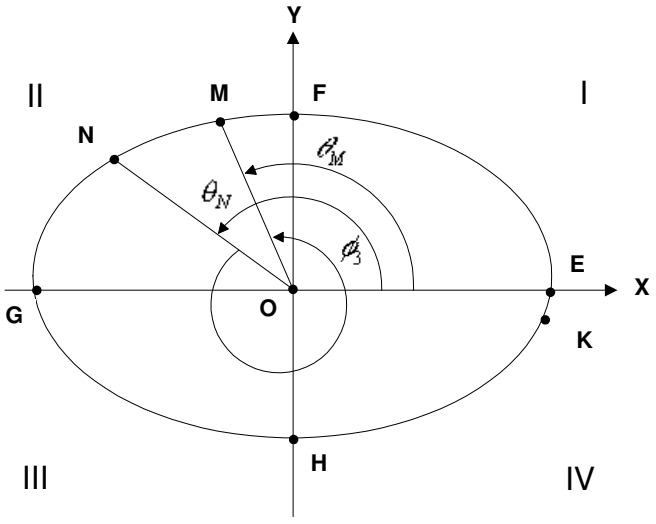
$$y_H < y_N < y_M < y_F. \tag{6.47}$$

The polar angles of endpoints M and N are in the middle of those of edge points E and K, i.e.,

$$\theta_E < \theta_M < \theta_N < \theta_K. \tag{6.48}$$



(a)



(b)

Figure 6.3. (a) The subtended angles of elliptic arcs $\widehat{BC}$ and $\widehat{DA}$; (b) The subtended angle of elliptic arc $\widehat{NM}$.

Obviously, we cannot directly extract the endpoints of arc $\widehat{NM}$ from data set Ω by simply comparing coordinates and polar angles of edge points.

In order to calculate the subtended angle of an arc, we introduce the incremental angle of edge point polar angles. Suppose that data set Ω represents an arc such as $\widehat{DA}$ in Fig. 6.3(a); we first sort the polar angles of all points in Ω in increasing order,

$$\Theta(\vec{\alpha},\ \Omega) = \{\theta_i | 0 \leq \theta_i \leq \theta_j \leq 2\pi,\ \text{ for all } 1 \leq i < j \leq n\}. \tag{6.49}$$

Since some points have the same polar angle, the number of polar angles may be less than the number of edge points, i.e., $1 \leq n \leq q$. For convenience, let $\theta_0 = \theta_n - 2\pi$. Obviously, edge points with polar angle θ_n are actually those with θ_0.

The incremental angle of sorted θ_i, φ_i, is then defined as:

$$\varphi_i = \theta_i - \theta_{i-1},\ \text{ for } i = 1, \cdots, n \tag{6.50}$$

$\varphi_1 = \theta_1 - \theta_0 = 2\pi + \theta_1 - \theta_n$ indicates the angle in a counterclockwise direction from polar angle θ_n to polar angle θ_1. Obviously, each incremental angle is subtended by a piece of elliptic arc.

With the introduction of φ_i, the subtended angle of the detected arc can be obtained using the following theorem. It is based on the fact that the angle constructed by two connected point and the ellipse center should be below a threshold. Compared with the recognition of circular arc endpoints, the problem for elliptic arc is more complicated. The threshold value for elliptic arc endpoints is point angle-variant while the threshold for a circular arc is constant. Therefore the ap-

proach for finding circular arc endpoints is connectivity-based.

Theorem 6.1. *Suppose data set Ω is an intra-connected inlier subset of ellipse $C(\vec{\alpha})$, which contains an elliptic arc $\upsilon(\Omega)$. Let φ_i be the incremental angle of the sorted polar angle θ_i of edge points in Ω. Let $T_\varphi(\theta_i) = \arccos \frac{2[(1-T_\rho)\rho(\theta_i)]^2 - T_c^2}{2[(1-T_\rho)\rho(\theta_i)]^2}$.*

(1) There is at most one angle φ_k such that

$$\varphi_k = \theta_k - \theta_{k-1} > \max(T_\varphi(\theta_{k-1}), T_\varphi(\theta_k)), \tag{6.51}$$

$$0 \le \varphi_i \le \max(T_\varphi(\theta_{i-1}), T_\varphi(\theta_i)), \quad \textit{for all } i \ne k. \tag{6.52}$$

The edge points, which have polar angles θ_k and θ_{k-1}, are the endpoints of the arc $\upsilon(\Omega)$. Hence, the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = 2\pi - \varphi_k. \tag{6.53}$$

(2) If no incremental angles is larger than $\max(T_\varphi(\theta_i), T_\varphi(\theta_{i-1}))$, i.e.,

$$0 \le \varphi_i \le \max(T_\varphi(\theta_{i-1}), T_\varphi(\theta_i)), \quad i = 1, \cdots, n, \tag{6.54}$$

then the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) \ge 2\pi - \max_i T_\varphi(\theta_i). \tag{6.55}$$

(For the proof, see Appendix B.)

Fig. 6.3(a), which reveals the relation between θ and φ, illustrates Theorem 6.1. Arcs $\overset{\frown}{BC}$ and $\overset{\frown}{DA}$ correspond to case (1) in this theorem. $\varphi_D = \theta_D - \theta_A$, which is the only angle larger than $\max(T_\varphi(\theta_A), T_\varphi(\theta_D))$, is of key importance for the recognition of points A and D, the endpoints of arc $\overset{\frown}{DA}$. On the other hand,

$\varphi_B = 2\pi + \theta_B - \theta_C > \max(T_\varphi(\theta_B), T_\varphi(\theta_C))$, helps to find points B and C, the endpoints of arc $\overset{\frown}{BC}$. Therefore, the subtended angles of arc $\overset{\frown}{BC}$ and $\overset{\frown}{DA}$ are $\phi(\overset{\frown}{BC}) = 2\pi - \varphi_B$ and $\phi(\overset{\frown}{DA}) = 2\pi - \varphi_D$, respectively.

It is worth pointing out that the threshold $T_\varphi(\theta_i)$ for the incremental angle φ_i is not a constant like that in circle fitting but a function of θ_i. It indicates that the $T_\varphi(\theta_i)$ varies at different point angle θ_i. This is due to the fact that the distance between the ellipse center and the ellipse edge point, $\rho(\theta)$, changes when θ changes (Eq. 6.4). For the circle, the radius r is constant for any point angle θ.

Based on this theorem, we then develop an algorithm for detecting the subtended angle of Ω, which is an intra-connected inlier subset of ellipse $C(\vec{\alpha})$:

1. Compute the polar angle θ_i of edge points $(x_i,\ y_i)$ in Ω.

2. Sort the polar angles θ_i in an increasing order.

3. Calculate φ_i, the incremental angle of the sorted polar angle θ_i, for each edge point in Ω.

4. Search for the pair of edge points $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ with $\varphi_k > \max(T_\varphi(\theta_{k-1}), T_\varphi(\theta_k))$

5. if the pair of edge points $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ is found, then the subtended angle of data set Ω, $\phi(\vec{\alpha}, \Omega) = 2\pi - \varphi_k$, and the edge points pair $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ are the endpoints of arc Ω; if the pair of edge points $(x_{k-1},\ y_{k-1})$ and $(x_k,\ y_k)$ is not found, then $\phi(\vec{\alpha}, \Omega) \geq 2\pi - \max_i T_\varphi(\theta_i)$, which indicates that no endpoints of this arc is found.

6.7 Analysis

In this section, we discuss the performance of the proposed connectivity-based multiple curve estimator in the presence of outliers and noise and demonstrate the advantages of using pixel connectivity in ellipse fitting. Comparisons between the algorithm and other curve-fitting methods are also presented.

6.7.1 Outliers

Outliers, especially if they are found in overwhelming numbers, is one of the major reasons for false detection in most curve-fitting algorithms. To evaluate the ability of our method to deal with outliers, we use the synthetic test image of Fig. 6.4(a), which comprises one partially occluded ellipse and several polygons. The edge image (Fig. 6.4(b)) is obtained by applying the Canny edge detector.

The inlier number of the detected ellipse in Fig. 6.4(c) is 117, while the total amount of edge data in Fig. 6.4(b) is 3155. The outliers, comprising 96.3% of the total edge data, greatly outnumber the inliers.

If pixel connectivity is taken into account, the task of detecting meaningful ellipses then becomes one of seeking intra-connected subsets that form valid elliptic arcs. The number of edge data in the intra-connected data subset containing the ellipse is 1447; resulting in the inlier ratio increasing sharply from 3.7% of the entire data set to 8.1% of this intra-connected data subset. The difficulty caused by the outliers is greatly reduced. Though they still number much more than the inliers. Fig. 6.4(c) shows that the ellipse is correctly extracted by our algorithm

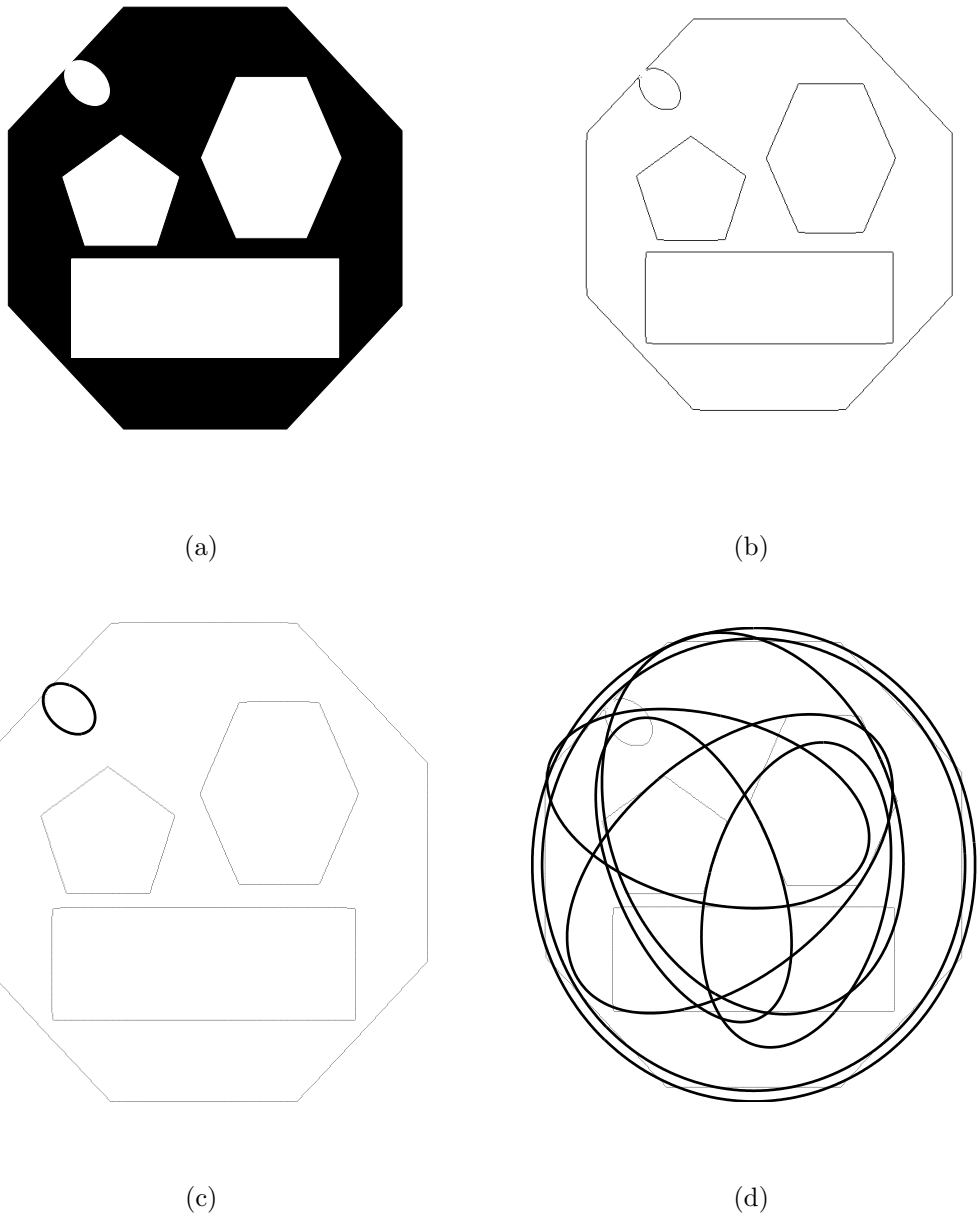
and no false detection is produced.

If pixel connectivity is not considered, a large number of false ellipses are detected. Fig. 6.4(d) shows only parts of ellipses with inliers more than that of the true ellipse. The extracted ellipses tend to touch as many sides of the polygons as possible due to the least-squares characteristic and the influence of all the edge data from the polygons.

Since the outlier ratio goes up to 96.3%, robust regression methods may fail with this example as they require outliers to number less than 50%. The opportunity of extracting the true ellipse by LS-based methods that start fitting from an initial guess is quite small [8], [7], [30] because the probability of randomly selecting an initial edge data subset located exactly on the true ellipse is less than 3.7%.

The HT method uses the the number of edge points lying on the ellipse whose parameters correspond to the cell mapped by these edge data. All seven ellipses in Figs. 6.4(d) may be extracted by the HT method. They have more inliers than the true one. These ellipses will be chosen ahead of the true ellipse. Furthermore, if we know *a priori* there is only one ellipse in the image, the HT is very unlikely to detect it.

The RHT randomly samples 5-tuple edge data from the image space and maps this 5-tuple into a cell in parameter space. The probability of choosing five edge points from the true ellipse is less than $0.037^5 = 6.9 \times 10^{-8}$, i.e., in a random selection of $10,000,000$ 5-tuple edge data only seven 5-tuples on average are from the true ellipse. The extraction of the true ellipse in this example is almost an impossible task for the RHT. It should also be noted that the number of edge data



(a) (b)

(c) (d)

Figure 6.4. Example 1: (a) original image of size 512×512; (b) edge data; (c) the true ellipse detected by the proposed method; (d) part of extracted ellipses if pixel connectivity is not considered, which have more inliers than the true ellipse.

from the largest false ellipse is 978, hence the probability of sampling 5-tuple edge data from this false ellipse is more than $8^5 = 32768$ times that of the true ellipse; thus the RHT is much more likely to extract the false model than the true model.

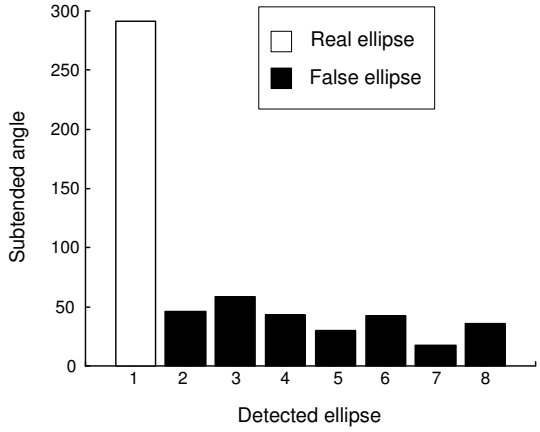
In contrast, our connectivity-based multiple curve estimator does not involve an initial guess and start ellipse fitting with the full intra-connected data subset. The result of every experiment with our approach is identical and only the true ellipse in the test image is successfully extracted.

6.7.2 Criterion for determining ellipse validity

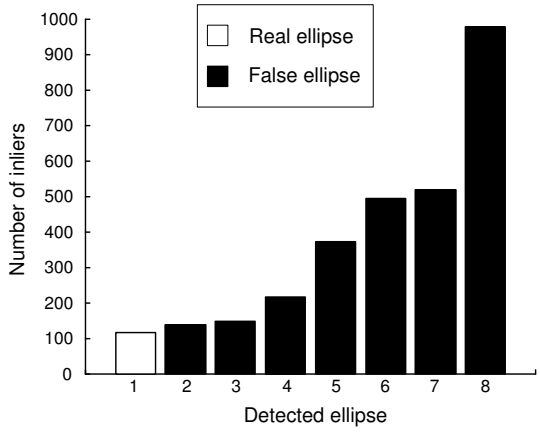
The criterion for evaluating the validity of extracted ellipse is of great importance for curve fitting algorithms. It plays a key role in preventing false detections. The validity criterion frequently used in HT-based methods is the cell score in parameter space, i.e., the total number of inliers of the evaluated ellipses. However, this criterion is not independent of the ellipse size and is thus not reliable.

The subtended angle of the largest elliptic arc is employed in our proposed estimator to evaluate ellipse validity. Obviously it is effective in preventing the detection of false ellipses that are large or touch the sides of polygons. It is also not influenced by the uneven distribution of edge data.

Fig. 6.5 compares the performance of the two validity criteria in evaluating the ellipses in Fig 6.4(c) and (d). The graphs present the criterion values for each of the ellipses. Fig. 6.5(a) shows that the subtended angle of the largest elliptic arc of the true ellipse is greater than 250° while those of the false ellipses (2-8) are less than 50°. Hence the true ellipse is easily identified. From Fig. 6.5(b), it is



(a)



(b)

Figure 6.5. Comparison of different validity criteria: (a) angle subtended by the largest arc; (b) total number of inliers. Ellipse 1 is the true ellipse detected in Fig. 6.4(c), ellipses 2-8 are those in Fig. 6.4(d) with increasing inlier number.

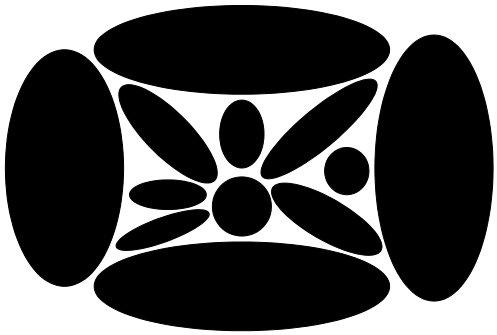
clear that the number of inliers of all seven false ellipses is more than that of true ellipse, which indicates that the number of inliers is ineffective in identifying the true ellipses.

6.7.3 Multiple unconnected ellipses

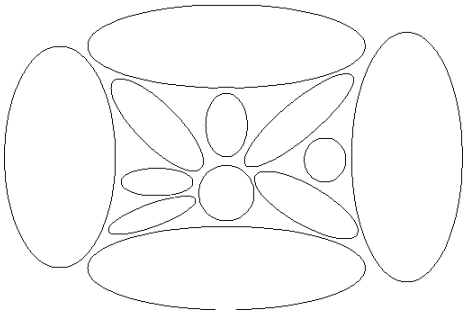
In some applications, there are multiple ellipses in the image that do not touch any other curves. The synthetic image of Fig. 6.6(a), which contains 12 un-connected ellipses, is one such example. The total amount of edge data in Fig. 6.6(b) is 4096.

McLaughlin [33] has employed various HT methods into multiple ellipse fitting. A comparing study of various HT methods (the RHT, the SHT, the PHT and the HT using geometric symmetric) is presented in [33]. The percentage of correctly extracted ellipses falls for all four methods when the number of ellipses in an image increases. The estimation correctness of all methods are less than 30% when there are 8 ellipses in the image. The conclusion is not surprising because the outlier ratio of each ellipse increases when more ellipses appear in the image.

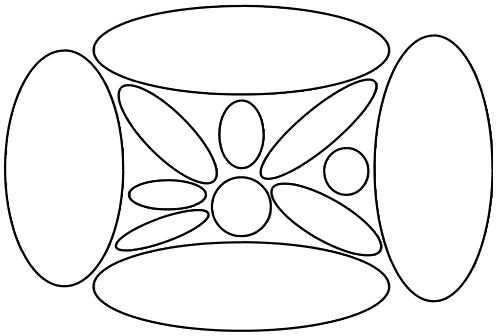
This situation will also occur in those robust regression methods and LS-based methods that start estimation from a randomly selected data subset and thus depend crucially on the quality of initial guess. In order to remove as many outliers as possible from the selected data set, the window size could not be too large. From our experiments, we observed that the shape and direction of the estimated ellipse may be severely deviated from the correct value (due to edge digitization error) when the angle subtended by the selected elliptic arc for estimation is too small. It indicates that the precision of the estimated model parameters is better if the



(a)



(b)



(c)

Figure 6.6. Example 2: (a) original image of size 512×512; (b) edge data; (c) the ellipses detected by the proposed method.

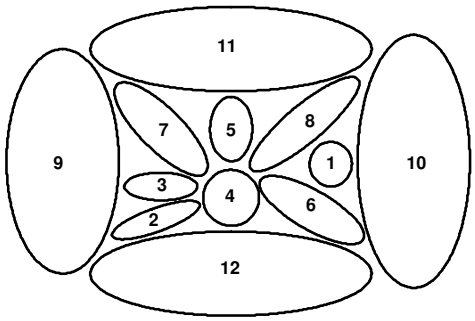


Figure 6.7. Indexed ellipses in Fig. 6.6 in increasing inlier order

subtended angle of the elliptic arc for fitting is larger. Therefore accurate ellipse fitting requires the window size to be as large as possible such that more elliptic arcs are included in this window for ellipse estimation. These two requirements for outlier problem and correctness of estimation are contradictory.

Our connectivity-based multiple curve estimator, however, avoids this kind of undesired detections. Using pixel connectivity, the entire data set is first separated into intra-connected data subsets. Ellipse estimation is always limited in the intra-connected data subsets; consequently, the problem of extracting multiple unconnected ellipses with our method is simplified to several subproblems of fitting a single ellipse in an intra-connected data subset, which is its inlier subset. The parameters of ellipses can easily be estimated by LS estimation. Fig. 6.6(c) demonstrates the result of extracting all 12 ellipses with our estimator.

The inlier ratio, $R(i)$, is the ratio of the number of inliers to the number of edge data in the entire set. The connectivity-based inlier ratio, $R_c(i)$, is the ratio of the number of inliers to the number of data in an intra-connected data subset.

Table 6.1. Inlier Number and Inlier Ratio of Ellipses.

Ellipse	Inlier number $N(i)$	Inlier ratio $R(i)$	Inlier ratio $R_c(i)$
1	136	3.3%	100%
2	170	4.2%	100%
3	171	4.2%	100%
4	176	4.3%	100%
5	198	4.8%	100%
6	256	6.2%	100%
7	260	6.3%	100%
8	278	6.8%	100%
9	552	13.6%	100%
10	608	14.8%	100%
11	645	15.7%	100%
12	646	15.8%	100%

Table 6.1 displays the number of inliers $N(i)$, and the two inlier ratios $R(i)$ and $R_c(i)$, for the 12 ellipses indexed in Fig. 6.7. Without considering pixel connectivity, the largest inlier ratio of all 12 ellipses is below 20%. In contrast, the inlier ratio of each ellipse in the corresponding intra-connected subsets rises to 100%, which makes this ellipse estimation problem very simple.

6.7.4 Occluded ellipses

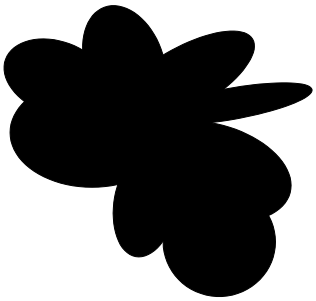
The occurrence of multiple occluded ellipses is a special form of the outlier problem. For every ellipse, the inliers of other occluded ellipses are outliers. The test image of Fig. 6.8(a), which contains 8 occluded ellipses, is used to demonstrate the effectiveness of the proposed algorithm in such a situation. The edge image obtained with the Canny edge detector is shown in Fig. 6.8(b).

Fig. 6.8(c) shows that all 8 occluded incomplete ellipses are accurately located with our algorithm. Compared with our method, the performance of four HT-based methods (the SHT, the RHT, the PHT and the HT using geometric symmetric) would be very poor, with percentage of correctness below 30% [33].

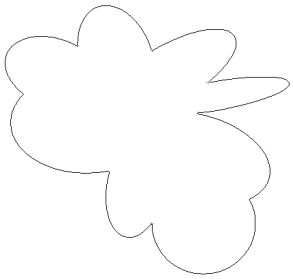
6.7.5 Noise

Noise is another important cause of unsuccessful ellipse detection. Since our algorithm is based on LS estimation, it inherits the robustness of the LS method against Gaussian noise. Therefore we focus here on salt-and-pepper noise, which severely distorts the detection in HT-based algorithms due to the high outlier ratios.

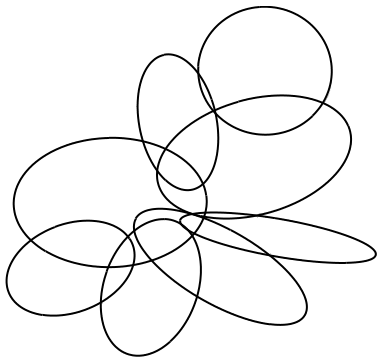
Fig. 6.9(a) shows the edge image of the third example corrupted by 5% salt-and-



(a)



(b)



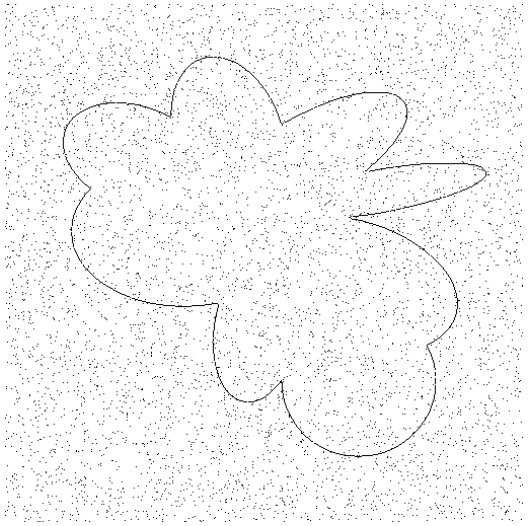
(c)

Figure 6.8. Example 3: (a) original image of size 512×512; (b) edge data; (c) occluded ellipses extracted by the proposed method.

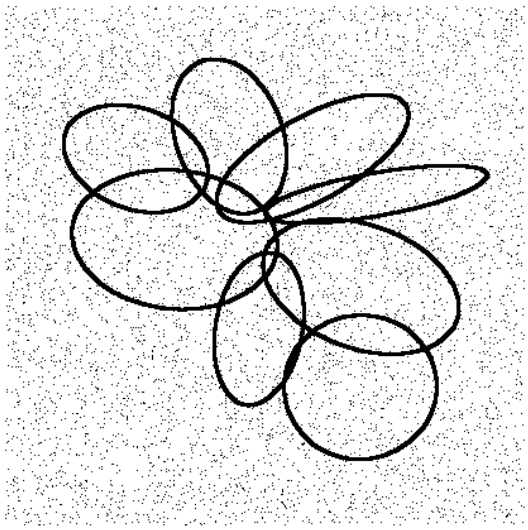
pepper noise. In this noisy image, none of the inlier percentages (the number of inliers as a percentage of the total number of edge points used in the fitting) of the 8 true ellipses exceeds 4%, and the sum of the inlier percentages of all the 8 ellipses is less than 25%. Our algorithm manages to extract all 8 true ellipses without any false detection (Fig. 6.9(b)) despite an outlier percentage of 96%. Furthermore, we note that for LS-based methods that commence fitting from an initial guess, the probability of poor initialization is more than 75%, which will very likely result in their failure. The probability of sampling 5-tuple edge points from a single model of all 8 true ellipses is below 8.2×10^{-7}. Such a low probability indicates that the RHT is very likely to fail in this example.

McLaughlin [33] pointed out that the percentages of correctness for four HT-based methods (the SHT, the RHT, the PHT and the HT using geometric symmetric) degrade rapidly as noise levels increased. His experiments show that the percentages of correctness for these four HT-based methods decrease to 0 when the percentage of salt and pepper noise in edge data image increases to 4%, i.e., these four methods cannot extract even one true ellipse. Our connectivity-based multiple curve estimator, in contrast, extracts all 8 occluded ellipses without any false detections from the noisy edge data corrupted by 5% salt-and-pepper noise.

It is clear that our algorithm deals very effectively with salt and pepper noise, the reason being that scattered noisy edge data are grouped into small intra-connected subsets that contain very few points. These subsets are rejected during ellipse fitting as there are too few points to form a valid elliptic arc.



(a)



(b)

Figure 6.9. (a) Edge data of Fig. 6.8(b) corrupted by 5% salt-and-pepper noise; (b) multiple occluded ellipses extracted by the proposed method.

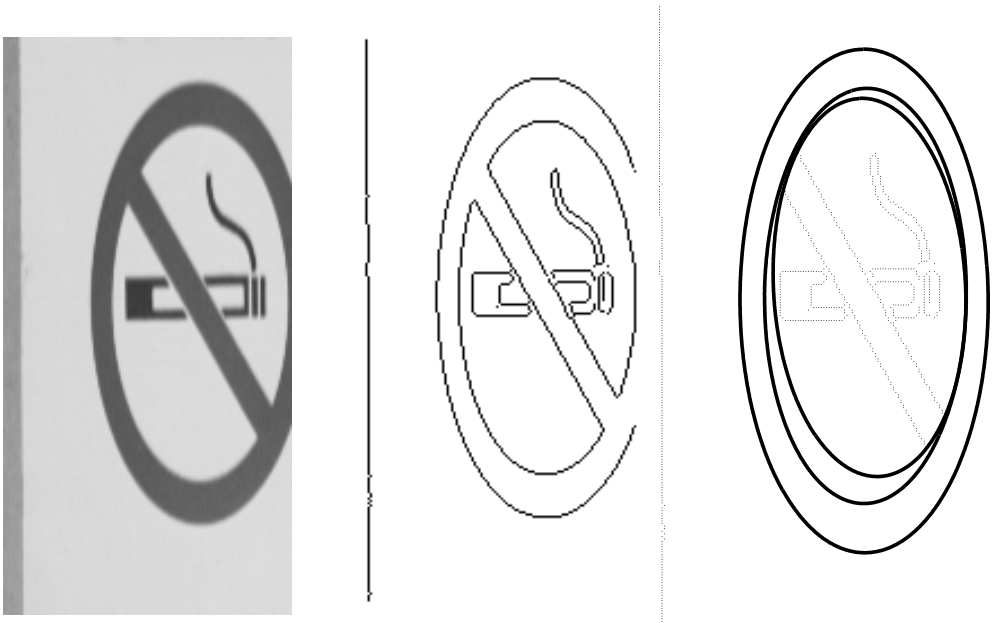
6.7.6 Computational Complexity

The classification of all edge points into intra-connected subsets before ellipse estimation requires the detection of connectivity between each pair of edge data. The computational complexity for connectivity detection ranges from $O(n)$ to the worst case of $O(n^2)$. The computation burden for outlier search depends on the number of data in the intra-connected subset and the inlier percentage of the valid models. The worst case is that all data in the image are connected, then the complexity for iterative weighted LS methods is $O(n^2)$. In a word, the computational complexity with our method is no more than $O(n^2)$. It is much better than the computational complexity of the HT method for ellipse detection, which is $O(n^5)$.

6.8 Experimental Results

The proposed connectivity-based multiple model estimator has been implemented for the reconstruction of occluded and touching ellipses from real binary edge images. A variety of images is used to test the performance and versatility of the algorithm. In the experiments, the edge images are obtained by applying the Canny detector.

Fig. 6.10(a) shows a real image obtained by viewing the No-smoking sign at an angle. The edge image obtained by Canny detector is shown in Fig. 6.10(b). Fig. 6.10(c) shows the estimation results of meaningful ellipses. Here the connectivity threshold is $T_c = 5$, the error threshold $T_\rho = 0.03$ and the subtended angle threshold $T_\phi = 100^\circ$. All three real ellipses in this image are successfully extracted



(a) (b) (c)

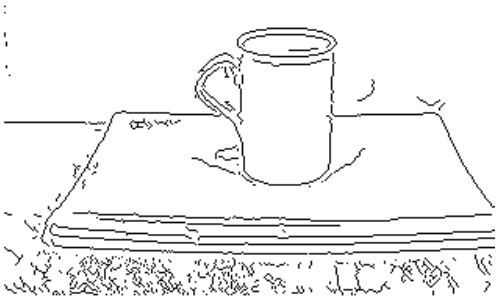
Figure 6.10. (a) Original image of size 128×256; (b) edges detected by Canny detector; (c) extracted ellipses.

without any false detections.

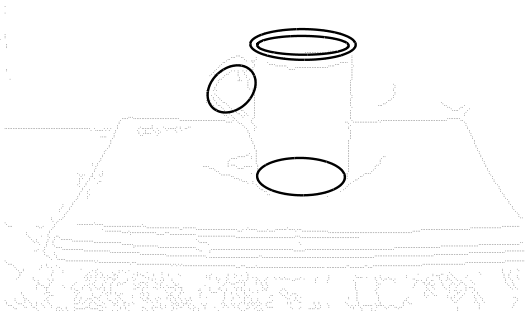
Fig. 6.11(a) shows a real image containing a cup and a book. Fig. 6.11(b) is the edge image. Fig. 6.11(c) shows the desired extraction of ellipses achieved by applying the proposed estimator. With the connectivity threshold $T_c = 2.5$, the error threshold $T_\rho = 0.1$ and the subtended angle threshold $T_\phi = 180°$, the ellipses generated by the boundary of the lip, bottom and handle of the cup are successfully extracted.

Fig. 6.12 shows different ellipse fitting results with different threshold selections. Fig. 6.12(a) presents extraction results with $T_N = 0$, i.e., there is no requirement for the minimum number of inliers in a valid elliptic arc. Hence up to eight small el-

(a)

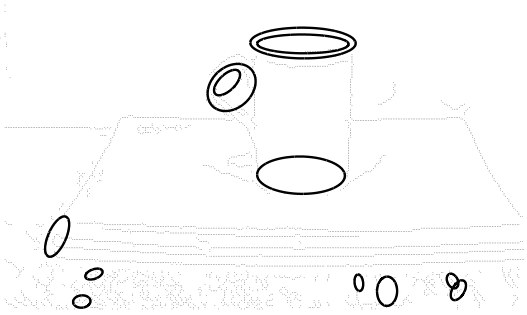


(b)

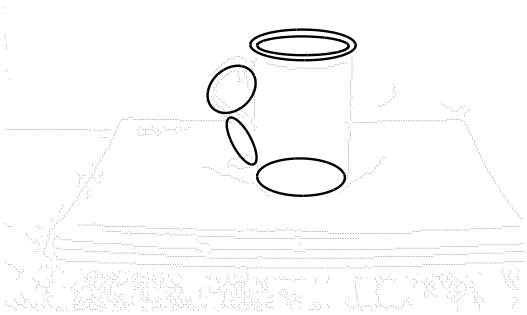


(c)

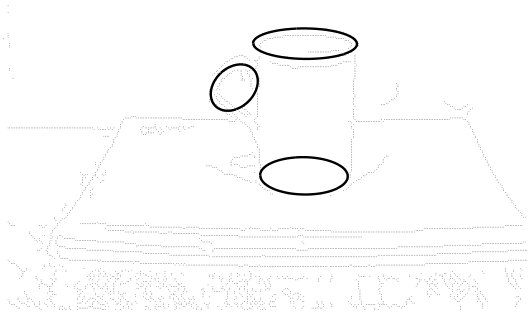
Figure 6.11. (a) Original image of size 512×480; (b) edges detected by Canny detector; (c) extracted ellipses.



(a)



(b)



(c)

Figure 6.12. (a) Extracted ellipses with $T_N = 0$; (b) extracted ellipses with $T_\phi = 0°$; (c) extracted ellipses with $T_r = 0.02$.

lipses formed by small groups of data are identified as valid models. In Fig. 6.12(b), the threshold for the angle subtended by a valid elliptic arc, T_ϕ, is set to 0°, which leads to the wrong detection of the ellipse under the cup handle. The error threshold T_r is raised to 0.02 in Fig. 6.12(c). The consequence is that the inner one of two valid elliptic arcs at the cup mouth in Fig. 6.12(c) is ignored because the inliers of the inner valid elliptic arc do not satisfy the Inlier Criterion (Criterion 3.1) with a threshold $T_r = 0.02$. Fig. 6.12 illustrates the fact that the selection of thresholds is very important to the correctness of ellipse fitting results. Inappropriate thresholds may lead to the false ellipse detection.

In the two examples, the outlier percentage is greater than 50% for the ellipses that intersect other curves. Therefore robust regression methods will not work well. The presence of a variety of curves in these examples will cause the HT and its variants to generate false estimations formed by the edges of straight lines and other curves. Our proposed algorithm, however, successfully extracts real ellipses while avoiding false detection by searching for valid elliptic arcs.

6.9 Conclusion

In this chapter, we apply the connectivity-based multiple curve estimator for fitting multiple touching or occluded ellipse with the Direct LS estimator, appropriately selected error and features of valid elliptic arcs. An efficient method to search for the endpoints of elliptic arcs and the angles subtended by the arcs is proposed.

Analysis of algorithm performance and experimental results show the following

major advantages of this multiple-model estimator for ellipse fitting: (1) it can effectively deal with outliers more than 50% of all edge data; (2) it can extract an *a priori* unknown number of meaningful ellipses in the edge image; and (3) it can handle fitting of multiple occluded or touching ellipses; (4) it can prevent false ellipses from being extracted; (5) it reduces the search field for model fitting using pixel connectivity, thus reduce computational complexity significantly.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we have developed a new general framework for multiple parametric model fitting. It allows us to build an accurate parametric model that fit curve edge data without requiring to a lucky initial estimate, and to extract the desired inliers of the curve even though there are more than 50% outliers of this curve in the original edge data set. It neither has a limit to breakdown point nor requires assumptions about the *a priori* distribution information of both inliers and outliers. The only necessary assumption that there are enough inliers constructing the underlying curve arcs guarantees the success of curve fitting with the iterative multiple model estimation framework.

The proposed framework has several novel features and compares favorably with various parametric model fitting algorithms proposed previously. The most significant advantage of this approach is that it does not require an accurate initial

local curve model estimate. Due to the global outlier search style of the region trimming scheme, the iterative multiple models estimation approach avoids the troublesome problem of making an accurate initial estimate because it starts inlier detection from the initial optimal global model based on the entire original edge data set.

The region trimming scheme is implemented by the iterative weighted LS approach. Thus it has the advantage of high accuracy in estimating parameters provided by the least squares technique if enough genuine inliers can be found. The region trimming scheme seeks the global least qualified edge data with respect to the optimal model estimation of all current available edge data, which is reasonably treated as the current most qualified outliers. The remaining edge data are used for further iterative model fitting. It is due to the strategy of expelling only those data with largest deviation that prevents underlying inliers from being eliminated even when the current model is seriously corrupted by outliers.

This characteristic assures the region trimming scheme of being robust against a high percentage of outliers and occluded multiple models. In this way, our proposed approach may begin with an initial global estimate that may not necessarily correspond to any underlying curve but give a global optimal curve model describing the entire edge data. The undesired global initial estimate then gradually converges to the local desired solution by consistently eliminating current detected outliers. As a consequence, it is robust against more than 50% outliers or occluded curves. The iterative multiple models estimation approach is in fact a global-to-local search strategy. In contrast, most curve fitting techniques are based on the

region growing scheme, which starts from a local initial estimate made from a local inlier edge data subset and then extends to other edge data.

Another significant thrust of our framework is that it is able to extract all curves when there is an *a priori* unknown number of models in the image. While applying the region trimming scheme to a subregion, the most favorable underlying curve will be first extracted. Then all its inliers in this subregion are excluded from the following model estimation. In this way, the current most favorable curves with respect to the updated edge data set are fitted sequentially until no further possible curves can be extracted. These extracted models will then be tested whether they are true or pseudo curves with the model verification method.

By introducing the intra-connectivity feature of curve arcs, we then develop a general framework for multiple curve fitting, the connectivity-based multiple curve estimator. In this approach, the curve fitting turns to be a task of searching for meaningful arcs, which are actually intra-connected inlier subsets.

The performance of applying the connectivity-based multiple curve estimator to multiple circles and ellipses fitting demonstrated the following advantages compared with most parametric model-fitting algorithms in the literature:

- It effectively avoids false curve detection because only valid arcs are used to extract real curves.
- It does not depend on the quality of the initial guess.
- It is able to handle more than 50% outliers, which is a cause of failure in many methods, e.g., the RHT and robust regression.

- It can successfully extract touching or occluded curves due to the ability of our estimator to deal with a high percentage of outliers.

- It is remarkably robust against salt-and-pepper noise because the use of pixel connectivity effectively excludes the majority of separated distributed noisy data from model estimation.

7.2 Future Work

We have demonstrated that the general framework proposed in this thesis is a robust regression estimator that can handle various types of noise. It is still reliable when there is an arbitrarily high percentage of outliers but no *a priori* information about noise distribution and number of models are available.

The error function and region trimming schemes are very important components of our framework, which offer a convenient way to extract the most qualified outliers with respect to current estimated model. Since the error function in the application of fitting circles and ellipses is defined as the error of edge data to the estimated model, the inliers of underlying model may have a large error with respect to the current extracted model when there is still a high percentage of outliers in current data set. Hence in each iteration we can only eliminate the most qualified outlier subset with the largest error. Otherwise we may mistakenly delete most of inliers before we eliminate outliers. The convergence speed of this method is therefore not satisfactory. A challenge is to achieve a better error function such that more outliers are correctly detected and removed at each iteration of the region trimming

scheme.

The image pre-partitioning strategy used in this framework is to split the entire image into sub-images according to some *a priori* application dependent information for the implementation convenience. A good heuristic image pre-partitioning strategy that separates a complicated image into several simplifier sub-images will greatly improve the speed and quality of model estimation.

Our connectivity-based multiple curve estimator cannot deal with virtual curve models supported by sparsely distributed edge points due to it being based on connectivity. It would be interesting and important to develop other appropriate features of the significant inlier subset for curves comprised of unconnected edge data such that false detection of this kind of curves can be prevented.

Due to the iterative style of searching for any underlying meaningful curve arcs in the entire edge data, the speed of connectivity-based multiple curve estimator is not satisfactory when the number of edge data is very large. Therefore fast outlier search strategies need to be developed for the applications of fitting multiple models in complicated images, especially those Possibilistic-based strategies and Monte Carlo approach.

Appendix A

Endpoints of Circular Arcs

In order to prove Theorem 5.1, we have to first introduce some lemmas.

Lemma A.1. *Let vertices A, B and C form a triangle as shown in Fig. A.1. Let $a = \overline{BC}$, $b = \overline{CA}$, $c = \overline{AB}$ and $\theta(a, b, c) = \angle BAC$. Suppose b and c are fixed, $|b - c| < T$, $0 < a \leq T$. Then θ is maximized when $a = T$, i.e.,*

$$\theta(T, b, c) = \max_{0<a\leq T} \theta(a, b, c). \tag{A.1}$$

Proof. From elementary geometry, we have

$$\cos\theta = \frac{b^2 + c^2 - a^2}{2bc}. \tag{A.2}$$

By differentiating both sides with respect to a, we get

$$-\sin\theta \frac{d\theta}{da} = \frac{-2a}{2bc}. \tag{A.3}$$

Since $\sin\theta > 0$ for $0 < \theta < \pi$, $a > 0$, $b > 0$ and $c > 0$, we have

$$\frac{d\theta}{da} = \frac{a}{bc\sin\theta} > 0. \tag{A.4}$$

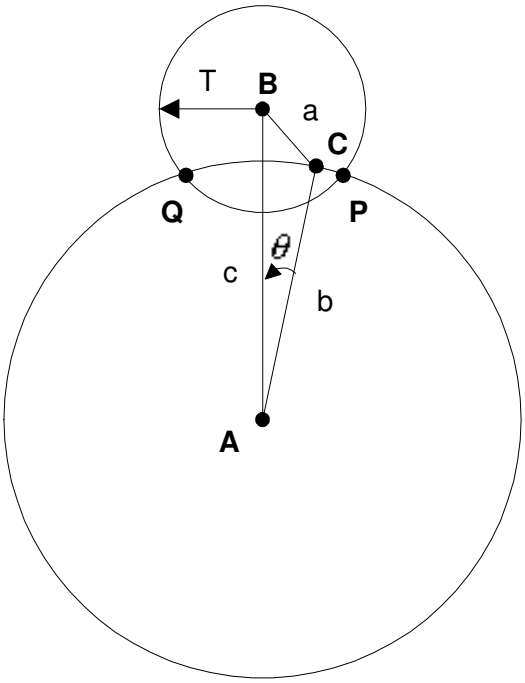


Figure A.1. Triangle ABC with fixed sides b and c.

This indicates that θ increases when a increases. Since $0 < a \leq T$, θ is maximized when $a = T$. The condition $|b - c| < T$ ensures the existence of a triangle with $a = T$.

Fig. A.1 shows that the vertex C can only move on the arc $\overset{\frown}{PQ}$. Obviously, $\angle BAC$ is maximized when C moves to point P or Q where $a = \overline{BC} = T$. □

Lemma A.2. *Let vertices A, B and C form a triangle as shown in Fig. A.2. Let* $a = \overline{BC}$, $b = \overline{CA}$, $c = \overline{AB}$ *and* $\theta = \angle BAC$. *Suppose b is fixed, and* $0 < a \leq T < r_1 \leq c \leq b \leq r_2$. *Then*

$$\max_{a,c} \theta(a, b, c) = \begin{cases} \theta(T, b, \sqrt{b^2 - T^2}), & \text{if } \sqrt{T^2 + r_1^2} \leq b \leq r_2 \\ \theta(T, b, r_1), & \text{if } r_1 \leq b < \sqrt{T^2 + r_1^2} \end{cases}, \tag{A.5}$$

Proof. In order to form a triangle, $c > b - T$ is a necessary condition since

$0 < a \leq T$. From Lemma A.1, we obtain $a = T$ as the solution of $\max_a \theta(a, b, c)$, i.e., $\max_{a,c} \theta(a, b, c) = \max_c \theta(T, b, c)$. Then

$$\cos\theta = \frac{b^2 + c^2 - T^2}{2bc}. \tag{A.6}$$

By differentiating both sides with respect to c, we have

$$\frac{d\theta}{dc} = \frac{b^2 - T^2 - c^2}{2bc^2 \sin\theta}. \tag{A.7}$$

We now consider:

(1) If $b \geq \sqrt{T^2 + r_1^2}$ and $c \geq r_1$, then

$$\frac{d\theta}{dc} = 0 \implies c = \sqrt{b^2 - T^2}. \tag{A.8}$$

The maximum value of θ is

$$\max_{a,c} \theta(a, b, c) = \theta(T, b, \sqrt{b^2 - T^2}) = \arctan \frac{T}{\sqrt{b^2 - T^2}}. \tag{A.9}$$

This is easily seen in Fig. A.2(a). When vertices A and B are known, vertex C must locate within the shaded area. Obviously, θ is the maximum value when $\overline{AC}$ is tangent to the circle centered at B.

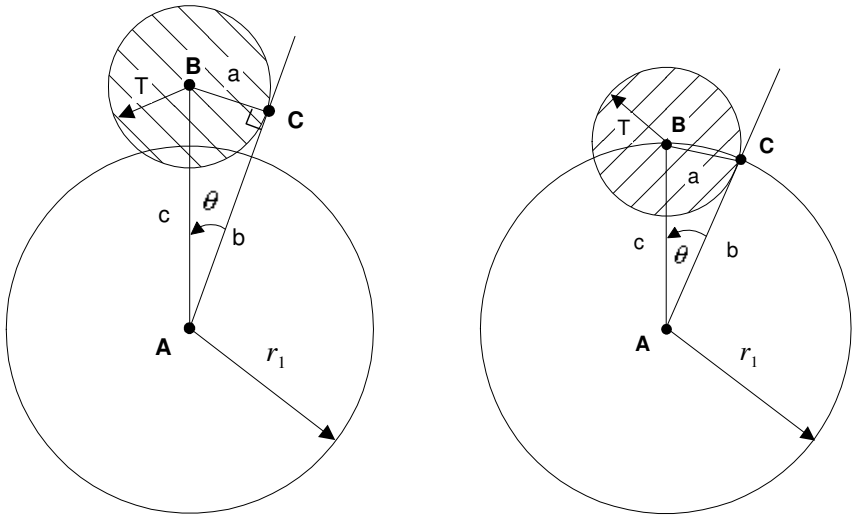
(2) If $b < \sqrt{T^2 + r_1^2} < T + r_1$ and $c \geq r_1$, then

$$\frac{d\theta}{dc} = \frac{b^2 - T^2 - c^2}{2bc^2 \sin\theta} < 0. \tag{A.10}$$

Therefore θ increases when c decreases. Since $c \geq r_1$,

$$\max_{a \leq T, c \geq r_1} \theta(a, b, c) = \theta(T, b, r_1) = \arccos \frac{b^2 + r_1^2 - T^2}{2br_1}. \tag{A.11}$$

This is also presented in a geometrical way in Fig. A.2(b). Vertex C is restricted in the shaded area. Obviously, when C moves to point P or Q, the maximum angle θ is achieved. □



(a) (b)

Figure A.2. (a)Triangle ABC with fixed side $b \geq \sqrt{T^2 + r_1^2}$; (b) triangle ABC with fixed side $b < \sqrt{T^2 + r_1^2}$.

Lemma A.3. *Let vertices A, B and C form a triangle. Let $a = \overline{BC}$, $b = \overline{CA}$, $c = \overline{AB}$ and $\theta = \angle BAC$. Suppose $0 < a \leq T$, $r_1 \leq b \leq r_2$, $r_1 \leq c \leq r_2$ and $0 < T < r_1$. Then*

$$\max_{a,b,c} \theta(a, b, c) = \theta(T, r_1, r_1) = \arccos \frac{2r_1^2 - T^2}{2r_1^2}. \tag{A.12}$$

Proof. Without loss of generality, we assume that $b \geq c$. Since $a < c$, then $\theta(a, b, c) < \frac{\pi}{2}$. Two cases are discussed in the following.

(1) If $\sqrt{T^2 + r_1^2} \leq b \leq r_2$, by Lemma A.2, we have

$$\begin{aligned} \max_{a,b,c} \theta(a, b, c) &= \max_{b} \theta(T, b, \sqrt{b^2 - T^2}) \\ &= \max_{b} \arctan \frac{T}{\sqrt{b^2 - T^2}}. \end{aligned} \tag{A.13}$$

The function arctan(·) is an monotonically increasing function over the interval $(0,\ \frac{\pi}{2})$. Hence θ is maximized when $b = b_{min} = \sqrt{r_1^2 + T^2}$, i.e.,

$$\begin{aligned}\max_{a,b,c} \theta(a,b,c) &= \max_{b} \theta(T, b, \sqrt{b^2 - T^2}) \\ &= \theta(T, \sqrt{r_1^2 + T^2}, r_1) \\ &= \arctan \frac{T}{r_1}. \end{aligned} \tag{A.14}$$

If $r_1 \leq b < \sqrt{T^2 + r_1^2}$, by Lemma A.2, we have

$$\begin{aligned}\max_{a,b,c} \theta(a,b,c) &= \max_{b} \theta(T, b, r_1) \\ &= \max_{b} \arccos \frac{b^2 + r_1^2 - T^2}{2br_1}. \end{aligned} \tag{A.15}$$

Therefore, $\max\limits_{a,b,c} \theta(a,b,c)$ becomes the function of b, i.e.,

$$\cos\theta = \frac{b^2 + r_1^2 - T^2}{2br_1}. \tag{A.16}$$

By differentiating both sides with respect to b,

$$\frac{d\theta}{db} = \frac{r_1^2 - b^2 - T^2}{2b^2 r_1 \sin\theta}. \tag{A.17}$$

Since $\sin\theta > 0$ for $0 < \theta < \pi$, and $b \geq r_1 > 0$, we have

$$\frac{d\theta}{db} = \frac{r_1^2 - b^2 - T^2}{2b^2 r_1 \sin\theta} < 0, \tag{A.18}$$

which indicates that θ increases when b decreases. Then,

$$\begin{aligned}\max_{a,b,c} \theta(a,b,c) &= \max_{b} \theta(T, b, r_1) \\ &= \theta(T, r_1, r_1) \\ &= \arccos \frac{2r_1^2 - T^2}{2r_1^2}. \end{aligned} \tag{A.19}$$

With the conclusions in case (1) and (2), we obtain

$$\max_{a,b,c} \theta(a,b,c) = \max(\theta(T, r_1, r_1),\ \theta(T, \sqrt{r_1^2 + T^2}, r_1)). \tag{A.20}$$

Let $\theta_1 = \theta(T, \sqrt{r_1^2 + T^2}, r_1)$ and $\theta_2 = \theta(T, r_1, r_1)$. Hence $\tan\theta_1 = \frac{T}{r_1}$ and $\tan\theta_2 = \frac{T\sqrt{4r_1^2 - T^2}}{2r_1^2 - T^2}$. Since function $\tan(\cdot)$ is monotonically increasing over interval $(0,\ \frac{\pi}{2})$, we compare the tangent function of both θ_1 and θ_2,

$$\begin{aligned}
\tan\theta_1 - \tan\theta_2 &= \frac{T}{r_1} - \frac{T\sqrt{4r_1^2 - T^2}}{2r_1^2 - T^2} \\
&= \frac{T}{r_1(2r_1^2 - T^2)}(2r_1^2 - T^2 - r_1\sqrt{4r_1^2 - T^2}) \\
&= \frac{T}{r_1(2r_1^2 - T^2)}[(\sqrt{4r_1^2 - T^2} - \frac{r_1}{2})^2 - \frac{9}{4}r_1^2].
\end{aligned} \tag{A.21}$$

Since $0 < T < r_1$, then $2r_1^2 - T^2 > r_1^2 > 0$. We then have

$$\sqrt{4r_1^2 - T^2} - \frac{r_1}{2} < \sqrt{4r_1^2} - \frac{r_1}{2} = \frac{3}{2}r_1. \tag{A.22}$$

This leads to

$$(\sqrt{4r_1^2 - T^2} - \frac{r_1}{2})^2 - \frac{9}{4}r_1^2 < (\frac{3}{2}r_1)^2 - \frac{9}{4}r_1^2 = 0. \tag{A.23}$$

Hence,

$$\tan\theta_1 - \tan\theta_2 < 0 \Longrightarrow \theta_1 < \theta_2 \tag{A.24}$$

Therefore, we obtain

$$\begin{aligned}
\max_{a,b,c} \theta(a,b,c) &= \max(\theta_1,\ \theta_2) = \theta_2 \\
&= \arccos\frac{2r_1^2 - T^2}{2r_1^2}.
\end{aligned} \tag{A.25}$$

Obviously, maximum θ exists in the triangle with sides T, r_1 and r_1. □

Lemma A.4. *Let two points P_1 and P_2 be inliers of circle $C(\vec{\alpha})$ cantered at O as shown in Fig. A.3. Let $r_1 = (1 - T_r)r$. We assume $T_c \ll (1 - T_r)r$.*

(1) If P_1 and P_2 are connected, then the angle $\angle P_1OP_2$ is limited, i.e.,

$$d(p_1, p_2) < T_c \implies \angle P_1OP_2 = |\theta_2 - \theta_1| \leq T_\varphi, \tag{A.26}$$

where

$$T_\varphi = \arccos \frac{2r_1^2 - T_c^2}{2r_1^2}, \tag{A.27}$$

and θ_1, θ_2 can be achieved from Eq. (5.20).

(2) If φ is larger than T_φ, these two points are not connected, i.e.,

$$\angle P_1OP_2 = |\theta_2 - \theta_1| > T_\varphi \implies d(p_1, p_2) \geq T_c. \tag{A.28}$$

Proof. (1) Since the two points P_1 and P_2 are inliers of circle $C(\vec{\alpha})$ centered at O, we have

$$r_1 \leq \overline{OP_1} \leq (1 + T_r)r, \tag{A.29}$$

$$r_1 \leq \overline{OP_2} \leq (1 + T_r)r. \tag{A.30}$$

Since P_1 and P_2 are connected, $0 < \overline{P_1P_2} < T_c$. Obviously, points O, P_1 and P_2 form a triangle (Fig. A.3), where $\angle P_1OP_2$ is actually φ_2. By Lemma A.3,

$$\angle P_1OP_2 = |\theta_2 - \theta_1| < \arccos \frac{2r_1^2 - T_c^2}{2r_1^2}. \tag{A.31}$$

(2) Suppose two points P_1 and P_2 are connected, then $|\theta_2 - \theta_1| < T_\varphi$. This is contradictory to the condition. Therefore the points P_1 and P_2 are not connected.

□

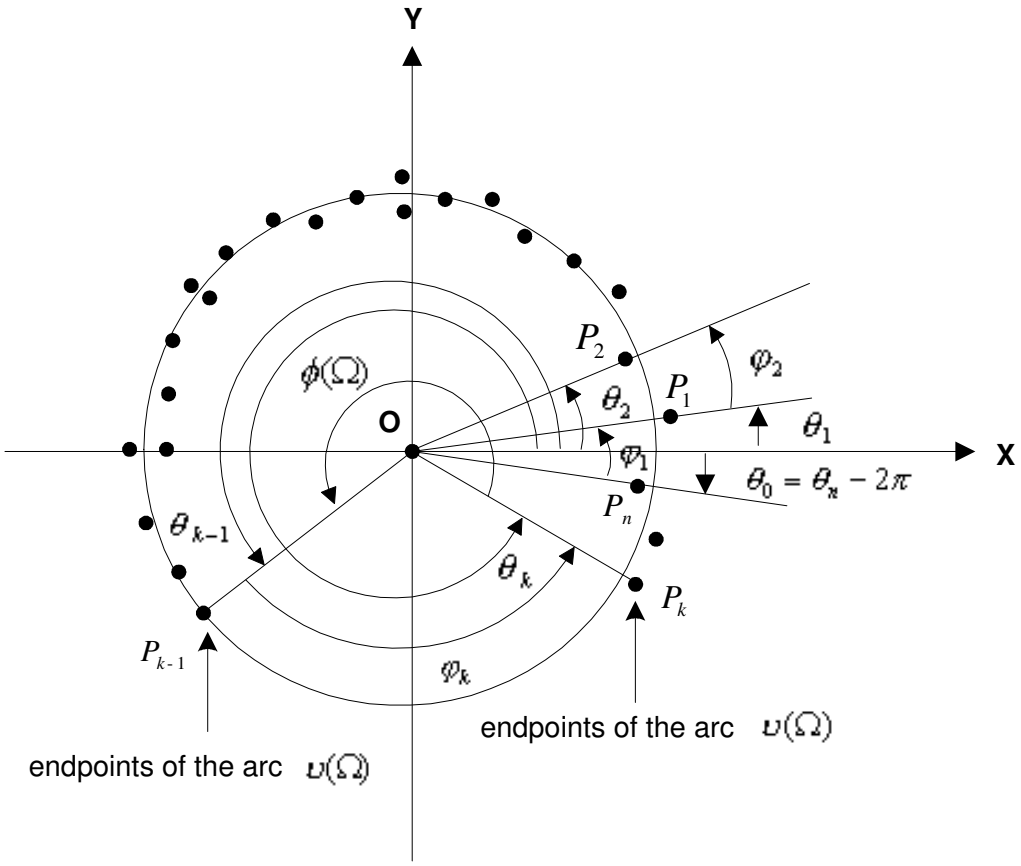


Figure A.3. Polar angles θ_i, incremental angles φ_i, the endpoints of the arc $\upsilon(\Omega)$ and its subtended angle $\phi(\Omega)$ (black points represent edge points with (θ_i,φ_i)).

Lemma A.5. *Suppose data set Ω is an intra-connected inlier subset of circle $C(\vec{\alpha})$. Let φ_i be the incremental angles of sorted polar angles θ_i. Let υ_i be the arc of circle $C(\vec{\alpha})$ subtending the angle φ_i in a counterclockwise direction from polar angle θ_{i-1} to θ_i.*

(1) All edge points are the endpoints of arcs $\upsilon_i,\ i = 1, \cdots, n$.

(2)

$$\varphi_i > 0, \tag{A.32}$$

$$\sum_{i=1}^{n} \varphi_i = 2\pi. \tag{A.33}$$

(3) The collection of circular arcs $\upsilon_i,\ i = 1, \cdots, n$, is a partition of detected circle.

Proof. The relationship among the polar angles θ_i of the edge points, incremental angles φ_i and the subtended angle $\phi(\Omega)$ of the detected arc is illustrated in Fig. A.3.

(1) Suppose there is a point with polar angle θ_k that is not an endpoint of arcs υ_i. Assume it is located within the neighbor of arc $\upsilon_i, i \neq 1$, which subtends the incremental angle $\varphi_i = \theta_i - \theta_{i-1}$. Therefore the polar angle θ_k is in the interval $(\theta_{i-1},\ \theta_i)$, i.e., $\theta_{i-1} < \theta_k < \theta_i$. This is contradictory to Eq. (5.24), in which all θ_i are sorted in increasing order.

If the point with polar angle θ_k is located within the neighbor of arc υ_1 in a counterclockwise direction from polar angle θ_0 to θ_1, then the polar angle θ_k is actually in the interval $(\theta_n,\ 2\pi) \bigcup [0,\ \theta_1)$, i.e., $\theta_k < \theta_1$ or $\theta_k > \theta_n$. This is contradictory to the Eq. (5.24), in which $\theta_1 = \min \theta_i$ and $\theta_n = \max \theta_i$.

Hence all edge points with polar angle θ_i are endpoints of arc υ_i subtending the incremental angles φ_i.

(2)

$$\theta_i > \theta_{i-1} \implies \varphi_i = \theta_i - \theta_{i-1} > 0,\ i = 2, \cdots, n, \tag{A.34}$$

$$\theta_n < 2\pi \implies \theta_0 = \theta_n - 2\pi < 0 \implies \varphi_1 = \theta_1 - \theta_0 > 0, \tag{A.35}$$

$$\sum_{i=1}^{n} \varphi_i = \sum_{i=1}^{n} (\theta_i - \theta_{i-1}) = \theta_n - \theta_0 = 2\pi. \tag{A.36}$$

(3) It is easy to see that all of arcs υ_i subtending the angles φ_i do not overlap each other because $\varphi_i = \theta_i - \theta_{i-1}$ and $0 \leq \theta_i < \theta_j < 2\pi$ for $i < j$. By Eq. (A.33), the collection of arcs υ_i is obviously a partition of circle $C(\vec{\alpha})$. □

With Lemmas A.4 and A.5, we can now present the proof of Theorem 5.1.

Proof of Theorem 5.1. Let $Z(\theta_i)$ denote the subset of edge points p_k with the polar angle θ_i, i.e., $Z(\theta_i) = \{p_k | \theta(p_k) = \theta_i\}$, because there may be more than one edge data having the same polar angle.

(1) Suppose there are two incremental angles, φ_i and φ_j, which are larger than T_φ, i.e., $\varphi_i = \theta_i - \theta_{i-1} > T_\varphi$ and $\varphi_j = \theta_j - \theta_{j-1} > T_\varphi$. By Lemma A.4, we have the conclusion that there is no point in $Z(\theta_i)$ that is directly connected to points in $Z(\theta_{i-1})$. By the same reasoning, there is no point in $Z(\theta_j)$ that is directly connected to points in $Z(\theta_{j-1})$. Without loss of generality, we assume that

$$0 < \theta_{i-1} < \theta_i < \theta_{j-1} < \theta_j. \tag{A.37}$$

Let $\Omega_1 = \{p_k | \theta(p_k) \leq \theta_{i-1} \text{ or } \theta(p_k) \geq \theta_j\}$ and $\Omega_2 = \{p_k | \theta_i \leq \theta(p_k) \leq \theta_{j-1}\}$. Obviously, Ω_1 and Ω_2 are not connected because there is no connection path from Ω_1 to Ω_2 through $Z(\theta_{i-1})$ to $Z(\theta_i)$ or $Z(\theta_j)$ to $Z(\theta_{j-1})$. Hence $\Omega = \Omega_1 \cup \Omega_2$ is not an intra-connected subset. This is contradictory to the condition that Ω is

an intra-connected inlier subset of circle $C(\vec{\alpha})$. Therefore, there is at most one incremental angle, φ_k, such that $\varphi_k = \max_i \varphi_i > T_\varphi$.

If the arc $\upsilon(\Omega)$ is actually the complete circle, then there are no endpoints in this arc. If the arc $\upsilon(\Omega)$ is not a closed circle, its endpoints are edge points like A and C in Fig. 5.1(b) such that no point in Ω with polar angle larger than theirs is directly connected to them, or edge points like B and D in Fig. 5.1(b) such that no point in Ω with polar angle less than theirs is directly connected to them.

By Lemma A.5, the collection of circular arcs $\upsilon_i,\ i = 1, \cdots, n$, is a partition of the detected circle. If the arc $\upsilon(\Omega)$ is open, there is an arc υ_k subtending φ_k, whose endpoints, $Z(\theta_{k-1})$ and $Z(\theta_k)$, are endpoints of $\upsilon(\Omega)$. The arc υ_k subtending the angle φ_k does not belong to the arc $\upsilon(\Omega)$ (Fig. A.3).

All other arcs belong to the arc $\upsilon(\Omega)$, i.e. $\upsilon_i \subset \upsilon(\Omega)$ for $i \neq k$. Furthermore, there is at least a pair of endpoints from two endpoint sets of those arcs that are directly connected, i.e.,

$$\min_{p \in Z(\theta_{i-1}),\, p' \in Z(\theta_i)} d(p,\, p') < T_c, \quad \text{for } i \neq k. \tag{A.38}$$

By Lemma A.4, inequality (A.38) indicates that $0 \leq \varphi_i \leq T_\varphi, \quad \text{for } i \neq k$.

If there is one and only one incremental angle φ_k such that $\varphi_k > T_\varphi$, by Lemma A.4, points from $Z(\theta_{k-1})$ and $Z(\theta_k)$ are nor directly connected. Obviously points of $Z(\theta_{k-1})$ and $Z(\theta_k)$, P_{k-1} and P_k, are endpoints of $\upsilon(\Omega)$, which is illustrated in Fig. A.3.

The arc $\upsilon(\Omega)$ consists of all arcs subtending incremental angles except υ_k, i.e.,

$$\upsilon(\Omega) = \bigcup_{i \neq k} \upsilon_i. \tag{A.39}$$

Therefore the angle subtended by the arc $\upsilon(\Omega)$ shown in Fig. A.3 is

$$\phi(\vec{\alpha}, \Omega) = \sum_{i \neq k} \varphi_i = 2\pi - \varphi_k. \tag{A.40}$$

(2) If all incremental angles φ_i are less than T_φ, there are two possibilities.

The first possibility is that the arc $\upsilon(\Omega)$ is a closed circular arc. There are no endpoints in $\upsilon(\Omega)$. Therefore the angle subtended by $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = 2\pi > 2\pi - T_\varphi. \tag{A.41}$$

The second possibility is that the arc υ_k, whose end points are endpoints of $\upsilon(\Omega)$, subtends an angle $\varphi_k \leq T_\varphi$. Hence, the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = \sum_{i \neq k} \varphi_i = 2\pi - \varphi_k \geq 2\pi - T_\varphi. \tag{A.42}$$

□

Appendix B

Endpoints of Elliptic Arcs

In order to prove Theorem 6.1, we have to first introduce some lemmas in the following.

Lemma B.1. *Let two points P_1 and P_2 be inliers of ellipse $C(\vec{\alpha})$ centered at O as shown in Fig. B.1. Let $\rho_1(\theta) = (1 - T_\rho)\rho(\theta)$. We assume $T_c \ll \rho_1(\theta)$.*

(1) If P_1 and P_2 are connected, then the angle $\angle P_1OP_2$ is limited, i.e.,

$$d(p_1, p_2) < T_c \implies \angle P_1OP_2 = |\theta_2 - \theta_1| \leq T_\varphi, \tag{B.1}$$

where

$$T_\varphi = \max(T_\varphi(\theta_1), T_\varphi(\theta_2)), \tag{B.2}$$

$$T_\varphi(\theta_i) = \arccos \frac{2\rho_1(\theta_i)^2 - T_c^2}{2\rho_1(\theta_i)^2} \tag{B.3}$$

and θ_1, θ_2 can be achieved from Eq. (6.5).

(2) If φ is larger than T_φ, these two points are not connected, i.e.,

$$\angle P_1OP_2 = |\theta_2 - \theta_1| > T_\varphi \implies d(p_1, p_2) \geq T_c. \tag{B.4}$$

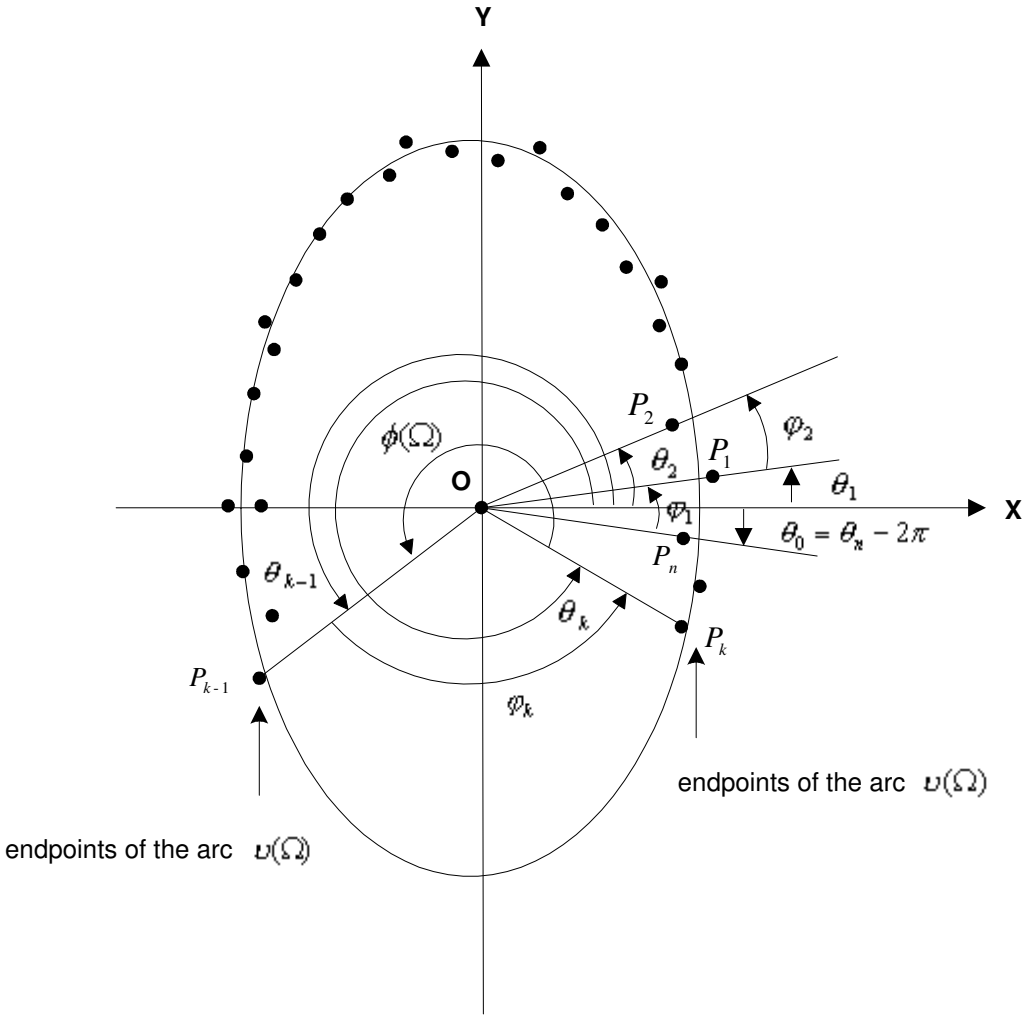


Figure B.1. Polar angles θ_i, incremental angles φ_i, the endpoints of the arc $\upsilon(\Omega)$ and its subtended angle $\phi(\Omega)$ (black points represent edge points with (θ_i,φ_i)).

Proof. (1) Since two points P_1 and P_2 are inliers of the ellipse $C(\vec{\alpha})$ centered at O, we have

$$\rho_1(\theta_1) \leq \overline{OP_1} \leq (1+T_\rho)\rho(\theta_1), \tag{B.5}$$

$$\rho_1(\theta_2) \leq \overline{OP_2} \leq (1+T_\rho)\rho(\theta_2). \tag{B.6}$$

We can summarize Eq. (B.5) and (B.6) as

$$\min(\rho_1(\theta_1), \rho_1(\theta_2)) \leq \overline{OP_1}, \overline{OP_2} \leq \max((1+T_\rho)\rho(\theta_1), (1+T_\rho)\rho(\theta_2)), \tag{B.7}$$

Since function $\arccos(\cdot)$ is monotonically decreasing, we have

$$\rho_1(\theta_1) < \rho_1(\theta_2) \implies T_\varphi(\theta_1) > T_\varphi(\theta_2). \tag{B.8}$$

This indicates that $\min(\rho_1(\theta_1), \rho_1(\theta_2))$ corresponds to $\max(T_\varphi(\theta_1), T_\varphi(\theta_2))$.

Since P_1 and P_2 are connected, $0 < \overline{P_1P_2} < T_c$. Obviously, points O, P_1 and P_2 form a triangle (Fig. 6.5), where $\angle P_1OP_2$ is actually φ_2. Let $r_1 = \min(\rho_1(\theta_1), \rho_1(\theta_2))$, then with (B.8), we obtain

$$\begin{aligned} \arccos\frac{2r_1^2 - T_c^2}{2r_1^2} &= \max(T_\varphi(\theta_1), T_\varphi(\theta_2)) \\ &= T_\varphi. \end{aligned} \tag{B.9}$$

By Lemma A.3,

$$\angle P_1OP_2 = |\theta_2 - \theta_1| < \arccos\frac{2r_1^2 - T_c^2}{2r_1^2} = T_\varphi. \tag{B.10}$$

(2) Suppose two points P_1 and P_2 are connected, then $|\theta_2 - \theta_1| < T_\varphi$. This is contradictory to the condition. Therefore the points P_1 and P_2 are not connected.

□

Lemma B.2. *Suppose data set Ω is an intra-connected inlier subset of ellipse $C(\vec{\alpha})$. Let φ_i be the incremental angles of sorted polar angles θ_i. Let υ_i be the arc of ellipse $C(\vec{\alpha})$ subtending the angle φ_i in a counterclockwise direction from polar angle θ_{i-1} to θ_i.*

(1) All edge points are the endpoints of arcs $\upsilon_i,\ i = 1, \cdots, n$.

(2)

$$\varphi_i > 0, \tag{B.11}$$

$$\sum_{i=1}^{n} \varphi_i = 2\pi. \tag{B.12}$$

(3) The collection of elliptic arcs $\upsilon_i,\ i = 1, \cdots, n$, is a partition of the detected ellipse.

Proof. The relationship among the polar angles θ_i of the edge points, incremental angles φ_i and the subtended angle $\phi(\Omega)$ by the detected ellipse is illustrated in Fig. B.1.

(1) Suppose there is a point with polar angle θ_k that is not an endpoint of arc υ_i. Assume it is located within the neighborhood of arc $\upsilon_i, i \neq 1$, which subtends the incremental angle $\varphi_i = \theta_i - \theta_{i-1}$. Therefore the polar angle θ_k is in the interval $(\theta_{i-1},\ \theta_i)$, i.e., $\theta_{i-1} < \theta_k < \theta_i$. This is contradictory to Eq. (6.49), in which all θ_i are sorted in increasing order.

If the point with polar angle θ_k is located within the neighborhood of arc υ_1 in a counterclockwise direction from polar angle θ_0 to θ_1, then the polar angle θ_k is actually in the interval $(\theta_n,\ 2\pi) \bigcup [0,\ \theta_1)$, i.e., $\theta_k < \theta_1$ or $\theta_k > \theta_n$. This is contradictory to Eq. (6.49), in which $\theta_1 = \min \theta_i$ and $\theta_n = \max \theta_i$.

Hence all edge points with polar angle θ_i are endpoints of arc υ_i subtending to incremental angle φ_i.

(2)

$$\theta_i > \theta_{i-1} \implies \varphi_i = \theta_i - \theta_{i-1} > 0,\ i = 2, \cdots, n, \tag{B.13}$$

$$\theta_n < 2\pi \implies \theta_0 = \theta_n - 2\pi < 0 \implies \varphi_1 = \theta_1 - \theta_0 > 0, \tag{B.14}$$

$$\sum_{i=1}^{n} \varphi_i = \sum_{i=1}^{n} (\theta_i - \theta_{i-1}) = \theta_n - \theta_0 = 2\pi. \tag{B.15}$$

(3) It is easy to see that all the arcs υ_i subtending the angles φ_i do not overlap with each other because $\varphi_i = \theta_i - \theta_{i-1}$ and $0 \leq \theta_i < \theta_j < 2\pi$ for $i < j$. By Eq. (B.12), the collection of arcs υ_i is obviously a partition of ellipse $C(\vec{\alpha})$. □

With Lemmas B.1 and B.2, we can now present the proof of Theorem 6.1.

Proof of Theorem 6.1. Let $Z(\theta_i)$ denote the subset of edge points p_k with the polar angle θ_i, i.e., $Z(\theta_i) = \{p_k | \theta(p_k) = \theta_i\}$, because there may be more than one edge data having the same polar angle.

(1) Suppose there are two incremental angles, φ_i and φ_j, that are larger than T_φ, i.e., $\varphi_i = \theta_i - \theta_{i-1} > \max(T_\varphi(\theta_{i-1}), T_\varphi(\theta_i))$ and $\varphi_j = \theta_j - \theta_{j-1} > \max(T_\varphi(\theta_{j-1}), T_\varphi(\theta_j))$. By Lemma B.1, we have the conclusion that there is no point in $Z(\theta_i)$ that is directly connected to points in $Z(\theta_{i-1})$. By the same reasoning, there is no point in $Z(\theta_j)$ that is directly connected to points in $Z(\theta_{j-1})$. Without loss of generality, we assume that

$$0 < \theta_{i-1} < \theta_i < \theta_{j-1} < \theta_j. \tag{B.16}$$

Let $\Omega_1 = \{p_k | \theta(p_k) \leq \theta_{i-1} \text{ or } \theta(p_k) \geq \theta_j\}$ and $\Omega_2 = \{p_k | \theta_i \leq \theta(p_k) \leq \theta_{j-1}\}$. Obviously, Ω_1 and Ω_2 are not connected because there is no connection path from

Ω_1 to Ω_2 through $Z(\theta_{i-1})$ to $Z(\theta_i)$ or $Z(\theta_j)$ to $Z(\theta_{j-1})$. Hence $\Omega = \Omega_1 \cup \Omega_2$ is not an intra-connected subset. This is contradictory to the condition that Ω is an intra-connected inlier subset of circle $C(\vec{\alpha})$. Therefore, there is at most one incremental angle, φ_k, such that $\varphi_k > \max(T_\varphi(\theta_{k-1}), T_\varphi(\theta_k))$.

If the arc $\upsilon(\Omega)$ is actually the complete ellipse, then there are no endpoints in this arc. If the arc $\upsilon(\Omega)$ is not a closed circle, some of its endpoints may be like A and C in Fig. 6.3(a). All of points directly connected to them have smaller polar angle. Other edge points may be like B and D in Fig. 6.3(a). All of point directly connected to them have larger polar angles.

By Lemma B.2, the collection of elliptic arcs $\upsilon_i,\ i = 1, \cdots, n$, is a partition of the detected ellipse. If the arc $\upsilon(\Omega)$ is open, there is an arc υ_k subtending φ_k, whose endpoints, $Z(\theta_{k-1})$ and $Z(\theta_k)$, are endpoints of $\upsilon(\Omega)$. The arc υ_k subtending the angle φ_k does not belong to the arc $\upsilon(\Omega)$ (Fig. B.1).

All other arcs belong to the arc $\upsilon(\Omega)$, i.e. $\upsilon_i \subset \upsilon(\Omega)$ for $i \neq k$. Furthermore, there is at least a pair of endpoints from two endpoint sets of those arcs that are directly connected, i.e.

$$\min_{p \in Z(\theta_{i-1}),\, p' \in Z(\theta_i)} d(p,\ p^{'}) < T_c, \quad \text{for } i \neq k. \tag{B.17}$$

By Lemma B.1, inequality (B.17) indicates that $0 \leq \varphi_i \leq \max(T_\varphi(\theta_{i-1}), T_\varphi(\theta_i))$, for $i \neq k$.

If there is one and only one incremental angle φ_k such that $\varphi_k > \max(T_\varphi(\theta_{k-1}), T_\varphi(\theta_k))$, by Lemma B.1, points from $Z(\theta_{k-1})$ and $Z(\theta_k)$ are not directly connected. Obviously points of $Z(\theta_{k-1})$ and $Z(\theta_k)$, P_{k-1} and P_k, are endpoints of $\upsilon(\Omega)$, which is illustrated in Fig. B.1.

The arc $\upsilon(\Omega)$ consists of all arcs subtending incremental angles except υ_k, i.e.,

$$\upsilon(\Omega) = \bigcup_{i \neq k} \upsilon_i. \tag{B.18}$$

Therefore the angle subtended by the arc $\upsilon(\Omega)$ shown in Fig. B.1 is

$$\phi(\vec{\alpha}, \Omega) = \sum_{i \neq k} \varphi_i = 2\pi - \varphi_k. \tag{B.19}$$

(2) If all incremental angles φ_i are less than $\max(T_\varphi(\theta_{i-1}), T_\varphi(\theta_i))$, there are two possibilities.

The first possibility is that the arc $\upsilon(\Omega)$ is a closed elliptic arc. There are no endpoints in $\upsilon(\Omega)$. Therefore the angle subtended by $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = 2\pi > 2\pi - \max(T_\varphi(\theta_k)). \tag{B.20}$$

The second possibility is that the arc υ_k, whose end points are endpoints of $\upsilon(\Omega)$, subtends an angle $\varphi_k \leq \max(T_\varphi(\theta_{k-1}), T_\varphi(\theta_k))$. Hence, the angle subtended by the arc $\upsilon(\Omega)$ is

$$\phi(\vec{\alpha}, \Omega) = \sum_{i \neq k} \varphi_i = 2\pi - \varphi_k \geq 2\pi - \max(T_\varphi(\theta_i)). \tag{B.21}$$

□

Bibliography

[1] A. Beaton and J. Tukey. The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. *Technometrics*, 16:147–185, 1974.

[2] D. Ben-Tzvi, V. F. Leavers, and M. B. Sandler. A dynamic combinatorial hough transform. In *Proc. 5th Int. Conf. Image Analysis and Processing*, pages 152–159, Positano, Italy, September 1989.

[3] D. Ben-Tzvi and M. B. Sandler. A combinatorial hough transform. *Pattern Recognition Letters*, 11(3):167–174, 1990.

[4] P. J. Besl and R. C. Jain. Segmentation through variable-order surface fitting. *IEEE Trans. Pattern Anal. Machine Intell.*, 10(2), March 1988.

[5] F. L. Bookstein. Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, (19):56–71, 1979.

[6] K. L. Boyer, J. Mirza, and G. Ganguly. The robust sequential estimator: A general approach and its application to surface organization in range data. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(10):987–1001, 1994.

[7] D. Chen. A data-driven intermediate level feature extraction algorithm. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(7):749–758, July 1989.

[8] G. Danuser and M. Stricker. Parametric model fitting: From inlier characterization to outlier detection. *IEEE Trans. Pattern Anal. Machine Intell.*, 20(2):263–280, March 1998.

[9] R. N. Dave. Generalized fuzzy c-shells clusterig and detection of circular and elliptic boundaries. *Pattern Recognition*, 25(7):713–722, 1992.

[10] R. N. Dave and K. Bhaswan. Adaptive fuzzy c-shells clusterig and detection of ellipese. *IEEE Trans. Neural Networks.*, 3(5):643–662, September 1992.

[11] E. R. Davies. *Machine Vision: Theory, Algorithms, Practicalities*. Academic Press Inc., 1990.

[12] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973.

[13] M. A. Fischler and O. Firschein. Parallel guessing: A strategy for high-speed computation. *Pattern Recognition*, 20(2):257–263, 1987.

[14] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Trans. Pattern Anal. Machine Intell.*, 21(5):476–480, 1999.

[15] W. Grimson. *Object Recognition by Computer: The role of Geometric Constraints*. MIT Press, 1990.

[16] F. Hampel, E.Ronchetti, P. Rousseeuw, and W. Stahel. *Robust statistics: An Approach Based on Influence Functions*. John Wiley & Sons, New York, 1986.

[17] T. Hastie and W. Stuetzle. Principal curve. *J. American Statistical Assoc.*, 84(406):502–516, June 1989.

[18] S. Heiler. Robust estimates in linear regression-a simulation approach. *Computational Statistics*, pages 115–136, 1981.

[19] P. Huber. *Robust Statistics*. John Wiley & Sons, New York, 1981.

[20] A. Iannio and S. D. Shapiro. A survey of the hough transform and its extensions for curve detection. In *Proc. IEEE Comput. Soc. Conf. Pattern Recognition and Image Processing*, pages 32–38, 1978.

[21] J. Illingworth and J. Kittler. A survey of the hough transform. *Computer Vision, Graphics, and Image Processing*, 44:87–116, 1988.

[22] L. A. Jaeckel. Estimating regression coefficients by minimizing the dispersion of residuals. *Ann. Math. Stat.*, 43:1449–1458, 1972.

[23] H. Kalviainen, P. Hirvonen, L. Xu, and E. Oja. Probabilistic and non-probabilistic hough transforms: overview and comparisons. *Image and Vision Computing*, 13(4):239–252, 1995.

[24] H. Kalviainen, L. Xu, and E. Oja. Recent versions of the hough transform and the randomized hough transform: Overview and comparisons. Research Report 37, Department of Information Technology, Lappeenranta University of Technology, Finland, 1993.

[25] B. Kegl, A. Krzyzak, T. Linder, and K.Zeger. Learning and design of principal curves. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(3):281–297, March 2000.

[26] Heung-Soo Kim and Jong-Hwan Kim. A two-step circle detection algorithm from the intersecting chords. *Pattern Recognition Letters*, 22:787–798, 2001.

[27] N. Kiryati, Y. Eldar, and A. M. Bruckstein. A probabilistic hough transform. *Pattern Recognition*, 24(4):303–316, 1991.

[28] P. Kultanen, L. Xu, and E. Oja. Randomized hough transform (rht). In *Proc. 10th Int. Conf. Patt. Recogn.*, pages 631–635, Atlantic City, NJ, 1990.

[29] V. F. Leavers, D. Ben-Tzvi, and M. B. Sandler. A dynamic combinatorial hough transform for straight lines and circles. In *Proc. 5th Alvey Vision Conf.*, pages 163–168, Reading, UK, September 1989.

[30] A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation of range images as the search for geometric parametric models. *Int'l J. Computer Vision*, 14:253–277, 1995.

[31] P. Liang. A new transform for curve detection. In *Proc. 3rd Int. Conf. Comput. Vision.*, pages 748–751, Osaka, Japan, December 1990.

[32] P. Liang. A new and efficient transform for curve detection. *J. Robotic Syst.*, 8(6):841–847, 1991.

[33] R. A. McLaughlin. Randomized hough transform: improved ellipse detection with comparison. *Pattern Recognition Letters*, 19:299–305, 1998.

[34] G. McLean and D. Kotturi. Vanishing point detection by line clustering. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(11):1,090–1,095, November 1995.

[35] P. Meer, D. Mintz, and A. Rosenfeld. Robust regression methods for computer vision: A review. *Int'l J. Computer Vision*, 6(1):59–70, 1991.

[36] J. Mirza and K. L. Boyer. Performance evaluation of a class of m-estimators for surface parameter estimation in noisy range data. *IEEE Trans. Robotics Automat.*, 9(1):75–85, 1993.

[37] R. Mohan and R. Nevatia. Perceptual organization for scene segmentation and description. *IEEE Trans. Pattern Anal. Machine Intell.*, 14(6):616–634, June 1992.

[38] J. Princen, J. Illingworth, and J. Kittler. A hierarchical approach to line extraction. In *Proc. IEEE Computer Vision and Pstt. Recogn. Conf.*, pages 92–97, San Diego, CA, June 1989.

[39] Y. Qiao. Block synergetic neural network and its usage in pattern recognition. Master's thesis, Shanghai Jiao Tong University, 1997.

[40] Y. Qiao and S. H. Ong. Connectivity-based multiple-circle fitting. *Pattern Recognition*, 37(4):755–765, April 2004.

[41] P. J. Rousseeuw. Least median of squares regression. *J. American Statistical Assoc.*, 79:871–880, 1984.

[42] D. Ruppert and J. Carrol. Trimmed least squares estimation in the linear model. *J. American Statistical Assoc.*, 75(372):828–838, 1980.

[43] B. G. Schunck. Robust computational vision. In *Proc. Int'l Workshop Robust Computer Vision*, pages 1–18, 1990.

[44] L. G. Shapiro and G. C. Stockman. *Computer Vision*. Prentice Hall, New Jersey, 2001.

[45] C. Stewart. Minpran: A new robust estimator for computer vision. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(10):925–938, October 1995.

[46] C. Taylor and D. Kriegman. Structure and motion from line segments in multiple images. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(11):1,021–1,032, November 1995.

[47] J. R. Taylor. *An Introduction to Error Analysis*. University Science Books, Sausalito, 1997.

[48] L. Xu and E. Oja. Randomized hough transform (rht): Basic mechanisms, algorithms, and computational complexities. *CVGIP: Image Understanding*, 57(2):131–154, 1993.

[49] L. Xu, E. Oja, and P. Kultanen. A new curve detection method: Randomized hough transform (rht). *Pattern Recognition Letters*, 11(5):331–338, 1990.

[50] H. Yan. Fuzzy curve-tracing algorithm. *IEEE Trans. Systems, Man and Cybernetics, Part B*, 31(5):768–780, October 2001.