ELLIPTIC ARC VECTORIZATION FOR 3D PIE CHART RECOGNITION

Weihua Huang, Chew Lim Tan and Wee Kheng Leow

SOC, National University of Singapore
3 Science Drive 2, Singapore 117543

ABSTRACT

In this paper, we present a novel approach to vectorize elliptic arcs for 3D pie chart recognition and data extraction. As a preprocessing step, a set of straight line segments are obtained first. By estimating the parameters based on the line segments, we are able to generate elliptic arc candidates. Curve fitting is then performed to reject false positives and further refine the parameters of each elliptic arc. The vectorized elliptic arcs are the key components in the 3D pie charts that allow us to determine the chart type and to extract data values embedded. Experiments using a group of testing images show that the proposed method works well and is computationally efficient.

1. INTRODUCTION

In recent years, document image analysis and recognition techniques have been applied in the area of chart image recognition. For example, Shi et al proposed a foreground-background separation model for driver schedule chart recognition [1]. Zhou and Tan proposed two different approaches for chart type recognition [2-3], namely Hough transformation and learning-based approach. However, then is no much work reported on high-level chart interpretation and data extraction. Recently, we are working on the development of a chart image recognition and understanding system (CIRUS). The aim of this system includes automatic chart location in document pages, automatic chart type determination and extraction of embedded data in the chart images. Some of our progress and results were reported in [4].

In our approach, the recognition of chart images heavily relies on the result of vectorization process (also known as raster-to-vector conversion). Previously, we have successfully vectorized straight lines and circular arcs based on some well established methods [5-7], allowing us to realize type recognition and data extraction to several types of charts, including 2D and 3D bar chart, 2D pie chart and 2D line chart. However, it is a more difficult task to vectorize elliptic arcs that are key components in 3D pie charts. Through literature review, we did not find any traditional technique claimed to perform this task very well. In this paper, we introduce a novel approach for elliptic arc vectorization. There are two major steps in the proposed method. The first step is the estimation of parameters required to specify an elliptic arc. And the second step is to perform curve fitting to further determine and refine the parameters. After the vectors of the elliptic arcs are obtained, we can remove the perspective distortion to extract precise data values from the 3D pie charts.

The remaining sections of the paper will explain the proposed method in details. Section 2 talks about some preprocessing steps. Section 3 introduces about the elliptic arc vectorization process. Section 4 illustrates the 3D pie chart recognition. Section 5 presents experimental results together with some discussions. Finally, section 6 gives a conclusion and some future work.

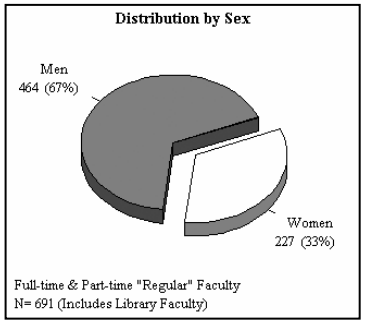
2. PREPROCESSING STEPS

The input image to the CIRUS system contains both graphical objects and textual information. Thus the first preprocessing step is text/graphics separation. To achieve this, we construct 8-connected components and then apply a series of filters that examines various features of the connected components, as suggested in [8]. In this step, noise removal is also performed by removing connected components with very small sizes.

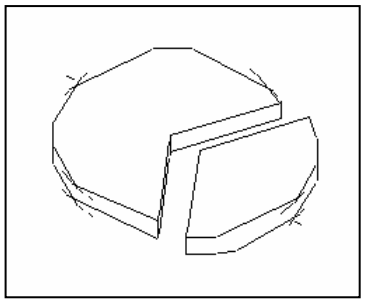
After text and noise are both removed from the input image, edge detection is performed to identify the edge pixels. In a typical chart image, edge detection is relatively easier due to the monotonic distribution of colors or grayscale levels in the graphical components. The edge pixels are all marked black, thus the resulting image becomes binary, which is usually required by the vectorization process.

Directly vectorizing elliptic arcs from the pixels in the binary image is both difficult and computationally expensive. A better approach is to vectorize the content of the image into a set of straight line segments first, as the last preprocessing step. These straight line segments become the basis for the gradual recovery of elliptical

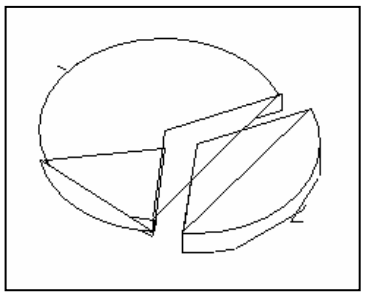
arcs. A sample image and the result of straight line vectorization are shown in Figure 1(a) and 1(b).



(a) Original image



(b) Straight line segments



(c) Elliptic arcs recovered

Figure 1. Example of elliptic arc vectorization.

3. ELLIPTIC ARC VECTORIZATION

After straight line vectorization, the edge of an elliptic arc becomes a set of line segments, and most of the time these line segments join together to form a polyline. In a 3D pie chart, the basic shape is a pie which consists of an elliptic arc and two straight line segments connecting the endpoints of the arc and the center of the arc, as can be seen in Figure 1(a). This suggests that we can estimate the parameters in the vector of an elliptic arc, namely:

- The center of the arc (x_0, y_0).
- The semi-major axis a.
- The semi-minor axis b.
- The starting point (x_s, y_s).
- The end point (x_e, y_e).

The center of the arc is estimated by examining the common endpoint shared by a pair of straight line segments. We can denote the two line segments as l_i and l_j, the common endpoint as (x_0, y_0) which is the candidate center for the arc, the other two endpoints are (x_1, y_1) and (x_2, y_2) which are the candidates of the starting point and the end point. Then we have the following group of equations:

$$\begin{cases} \dfrac{(x_1 - x_0)^2}{a^2} + \dfrac{(y_1 - y_0)^2}{b^2} = 1 & (1) \\ \dfrac{(x_2 - x_0)^2}{a^2} + \dfrac{(y_2 - y_0)^2}{b^2} = 1 & (2) \end{cases}$$

Let $C_1 = (x_1 - x_0)^2$, $C_2 = (y_1 - y_0)^2$, $C_3 = (x_2 - x_0)^2$ and $C_4 = (y_2 - y_0)^2$, we can calculate the value of a and b from (1) and (2):

$$a = \sqrt{\frac{C_2 C_3 - C_1 C_4}{C_2 - C_4}} \text{ and } b = \sqrt{\frac{C_2 C_3 - C_1 C_4}{C_3 - C_1}} \qquad (3)$$

For each pair of straight line segments sharing common endpoint, a set of parameters is estimated. The next step is to further check if a candidate is an elliptic arc based on these parameters. The idea is to use curve fitting. As we have mentioned, the original edge of the elliptic arc has become a polyline. Furthermore, this polyline starts from (x_1, y_1) and ends at (x_2, y_2), and all the vertices along the polyline lie on the elliptic arc. Finding a polyline connecting two points is not a difficult task. To test whether a vertex (x, y) is on the elliptic arc, we use the following function to test whether it fits into the curve with estimated parameters:

$$\left| \frac{(x - x_0)^2}{a^2} + \frac{(y - y_0)^2}{b^2} - 1 \right| \le \varepsilon \qquad (4)$$

where ε is a small constant to allow certain error in the testing. If function (4) is satisfied, then the vertex (x, y) is considered as lying on the elliptic arc.

The testing is done to all the vertices on the polyline, if any of the vertices failed the test, then the candidate arc is rejected. Otherwise, the candidate arc is accepted and store into a list. Figure 1(c) shows an example of the

resulting elliptic arc list. Note that for each arc in the figure, there is an extra line segment joining the starting point and the end point. This is because we have to use chords to represent the elliptical arcs since there is no existing function for drawing elliptical arcs directly in the software used.

4. 3D PIE CHART RECOGNITION

As we have mentioned previously, elliptic arcs are key components in a 3D pie chart. By vectorizing all the elliptic arcs and construct 3D pie shapes, we are able to tell if a given image is a 3D pie chart. After that, we are able to extract data embedded in the 3d pie chart. To guarantee the accuracy of the data extracted, one necessary action is to remove the effect of perspective distortion.

4.1. Chart type determination

It is not enough to determine if a given image is a 3D pie chart by just looking at the elliptic arcs. There are more constraints to be considered, such as:

- Each elliptic arc should belong to a pie shape, which consists of the arc and two line segments connecting the endpoints of the arc with the center of the arc.
- Among all the pie shapes, we should be able to find a subset of shapes with similar semi-major axis and semi-minor axis. If we denote the angle difference between the two line segments in a pie shape as the angle of the pie, then the summation of the angles of all pies in the subset should be close to 2π.
- There should not be x-y axes in the image. Also there should not be many other shapes in the image.

By considering these constraints, type determination can be achieved more reliably since most false positives can be eliminated.

4.2. Removing perspective distortion

In the case of 3D pie chart, perspective distortion results from the 3D effect by transforming the original circle into an ellipse. The side effect of perspective distortion is that the angle in an elliptic arc does not reflect the true data value. To remove this side effect, we adopt a projection-based approach. In this approach, a point on the ellipse is projected onto a circle sharing the same center with the ellipse and whose radius is equal to the semi-major axis of the ellipse. If we do this to both endpoints of an elliptic arc, then we can calculate the new angle difference between the line segments in the arc which reflect the true data value more accurately. An example of perspective distortion removal is shown in Figure 2.

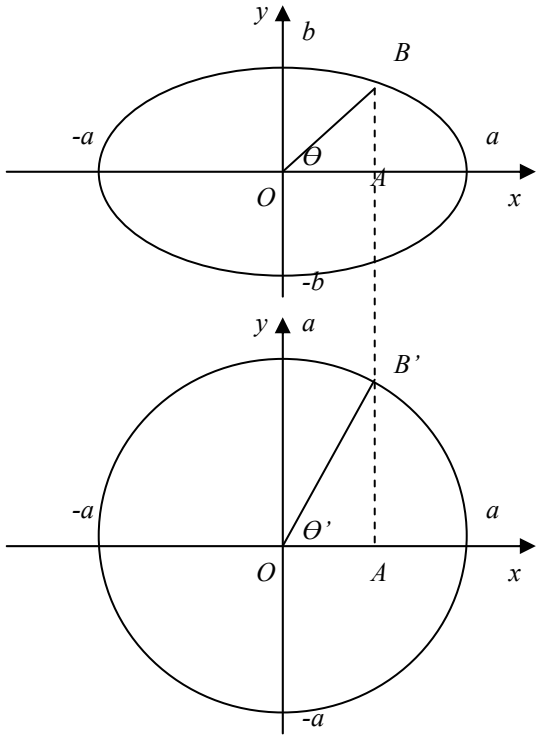


Figure 2. Illustration of perspective distortion removal.

In the example shown in Figure 2, O is the common center of the ellipse and the circle, point B on the ellipse is projected to point B' on the circle. Line segment BA and B'A are both perpendicular to the x-axis.

Based on our projection-based approach, once we know the angle Θ ($\angle$BOA) in the ellipse, the corresponding angle Θ' ($\angle$B'OA) can be calculated. It can be seen that $\tan\theta = \frac{AB}{OA}$ and $\tan\theta' = \frac{AB'}{OA}$, thus we can get $\frac{\tan\theta'}{AB'} = \frac{\tan\theta}{AB}$, which means:

$$\tan\theta' = \frac{AB'}{OA} = \frac{\sqrt{a^2 - OA^2}}{OA}, \qquad (5)$$

where a and OA are known.

Formula (5) is applied to both endpoints of an elliptic arc and then the new angle difference is calculated. In a pie chart, the data are percentages. Thus after the calculation of new angles for all the pie shapes in the chart, we can easily recover the percentages by dividing each angle by 2π.

5. EXPERIMENTAL RESULTS

The testing image set contains 10 3D pie chart images that are collected from the internet. The original data percentages are all known, so that we are able to evaluate the accuracies of the data extracted by the system.

Elliptic arc vectorization is performed after the preprocessing steps. The vectorization results are manually evaluated. The results show that all the elliptic arcs with their center known were successfully vectorized and used to construct pie shapes. One problem here is that the vectorization process relies on the estimation of the center of the arc, thus those elliptic arcs whose centers are occluded by other pies cannot be handled. However this does not affect the performance of the type determination and data extraction process, since the centers of all the major pie shapes should appear in a chart image.

The next task is to test the system's ability of extracting the embedded data from the images. To evaluate the effectiveness of perspective distortion removal, we performed data extraction in two rounds. In the first round, we directly calculated the angles of the pies and converted them into percentages without perspective distortion removal. In the second round, the perspective distortion removal was carried out, and the percentages were calculated again based on the new angles. We can define the error rate as the difference between the extracted percentage and the true percentage, and the average error rate as the summation of all error rates divided by the total number of data entries. The average error rate in the first round is 3.2%. On the other hand, the average error rate in the second round is only 0.6%, showing significant improvements in the data accuracies. An example is given in Figure 3 and the comparison of data accuracies in the two rounds is shown in Table 1.

6. CONCLUSION

In this paper, we presented a novel approach for elliptic arc vectorization. In the preprocessing stages, a set of straight line segments are obtained. These straight line segments are used for gradual elliptic arc recovery. There are two major steps in the proposed method: parameter estimation and curve fitting. In the first step, the Cartesian equation of an ellipse is used to estimate the semi-major axis and semi-minor axis given a candidate center. In the second step, curve fitting is done to the set of vertices belonging to the polyline surrounding the candidate center, to obtain the vector form of an elliptic arc and to refine the parameters. Experiment results show that the elliptic arcs can be accurately vectorized. After removing perspective distortion, precise data extraction from 3D pie charts can also be achieved.

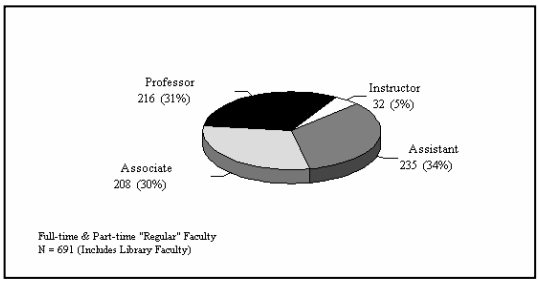


Figure 3. An example of perspective distortion.

Table 1. Improving data accuracy using perspective distortion removal (PDR) for the image in Figure 3.

Data No.	Original percentage	Without PDR	With PDR
1	5%	4%	4%
2	34%	25%	35%
3	30%	32%	31%
4	31%	39%	30%

7. REFERENCES

[1] G. Shi, C. Luo, K. Wang, W. Pan and Q. Wang, "Driver Schedule Chart Recognition Based on Background-Foreground Separation Model", *GREC 2003*, pg 113-120.

[2] Y. P. Zhou and C. L. Tan, "Hough technique for bar charts detection and recognition in document images", *International Conference on Image Processing, ICIP 2000*, page 494-497, 2000.

[3] Y. P. Zhou and C. L. Tan, "Learning-based scientific chart recognition", *4th IAPR International Workshop on Graphics Recognition, GREC2001*, page 482-492, 2001.

[4] W. H. Huang, C. L. Tan and W. K. Leow, "Model based chart image recognition", *International Workshop on Graphics Recognition, GREC2003*, 30-31 July 2003, Barcelona, Spain.

[5] W. Liu and D. Dori, "Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, vol. 21, pg. 202-215.

[6] D. Dori and W. Liu, "Incremental Arc Segmentation Algorithm and Its Evaluation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, vol. 20, pg. 424-431.

[7] J. Song, F. Su, J. Chen, C. Tai and S. Cai, "Line Net Global Vectorization: an Algorithm and Its Performance Evaluation", *CVPR 2000*, pp. 383-388.

[8] K. Tombre, S. Tabbone, L, Pelissier, B. Lamiroy and P. Dosch, "Text/Graphics Separation Revisited", *DAS 2002*, pg 200-211.