Detecting Ellipses via Bounding Boxes

CHUN-MING CHANG*
Department of Information and Design, Asia University, Taiwan

ABSTRACT

A novel algorithm for ellipse detection based on bounding boxes is proposed in this paper. The bounding box is used in many different contexts in computer graphics as a complexity limiting device. For a perfect ellipse, the bounding box can be located by scanning an image and the corresponding geometric features can then be derived. The concept of bounding box is similar to that of the geometric symmetry approach. Instead of finding edges, we utilize a preprocessed binary image as the input to our algorithm. The search space of parameters is also condensed to a small area. Simulation results show the effectiveness of our approach.

Key words: ellipse, bounding box, Hough transform, major axis, minor axis, axis-aligned bounding box.

1. INTRODUCTION

An ellipse, which is the perspective projection of a circle, is a common component in many applications of pattern recognition and computer vision (Fitzgibbon, Pilu, & Fisher, 1999; Hartley & Zisserman, 2000). Detecting the parameters of ellipses can assist us for higher-level processing tasks, such as camera calibration. A standard method of detecting ellipses is the Hough transform (Aguado, Montiel, & Nixon, 1995; Ho & Chen, 1995; McLaughlin, 1996; Ser & Siu, 1994; Yip, Tam, & Leung, 1992). The Hough transform extracts geometrical parameters of an image and maps them into the Hough domain. An ellipse has five parameters so it requires a 5-dimensional parameter space. This method is computationally expensive and requires intensive memory.

Several variations of the Hough transform have been proposed during last decades. Most methods utilized gradient information to reduce the dimension of the parameter space to accelerate the processing speed. McLaughlin (1996) proposed an algorithm for ellipse detection using a randomized Hough transform. Yip et al. (1992) decomposed the 5-dimensional parameter space into several subspaces of fewer dimensions via parallel edge points. Aguado et al. (1995) extracted ellipses by relating local geometric features of a parametric representation to the local features in an image. A multistage Hough transform approach using the polar and pole definition of ellipses has been proposed by Yoo and Sethi (1993). Ho and Chen (1995) utilized a global geometric symmetry method to locate the centers of ellipses and circles and then used the accumulative concept of the Hough transform to extract all ellipses and circles in an image.

Some non-Hough transform methods were also proposed for detecting ellipses. Dave (1992) used the fuzzy c-shells to detect circles and ellipses. Lee, Lu, and Tsai (1990) developed an algorithm to detect ellipses based on the moment

* E-mail: cmchang@asia.edu.tw

preserving principle. Using the geometric properties of an ellipse, Wu and Wang (1993) proposed a simple method to detect ellipses.

In this paper, instead of using the time-consuming Hough transform method, a fast novel algorithm for ellipse detection is proposed (Chang & Hsiao, 2005). First, bounding boxes are formed to locate ellipses in an image. The center of an ellipse is then obtained. Then the boundary of an ellipse is traced to find a vertex of an ellipse to identify the major axis. These bounding boxes allow us to focus only on smaller areas to speed up processing. Finally, the minor axis can be located in the normal direction of the major axis. The parameters of an ellipse can be computed without much effort. The limitation of this algorithm is that it can only be applied to locate perfect ellipses.

2. BACKGROUND

An ellipse can be obtained as a section cut from a right circular cone by a plane. It is the locus in the plane of all points the sum of whose distances from two fixed points is constant. The analytic equation for a conic in arbitrary position is as follows (Protter, M. H. & Protter, P. E., 1998):

$$A'x^2 + B'xy + C'y^2 + D'x + E'y + 1 = 0 \tag{1}$$

where *A'*, *B'*, and *C'* are not all zeroes.

If the center of an ellipse is translated to the origin of the coordinate system and there is no rotation, the equation of the ellipse can be simplified to

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{2}$$

with $a > b$. a is called the *major axis* and b the *minor axis*.

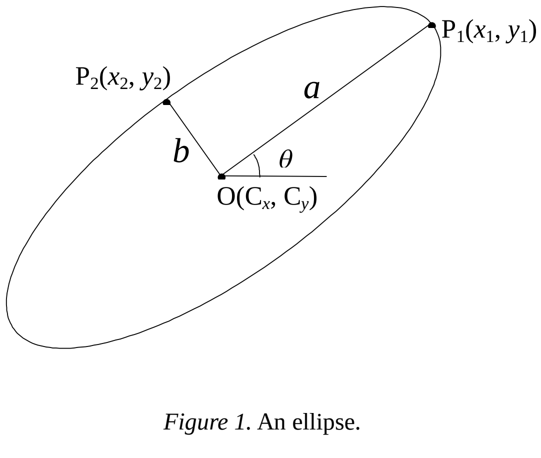

Figure 1. An ellipse.

There are five unknown parameters for an ellipse that we would like to find out: (C_x, C_y) for the center, (a, b) for major and minor axis, and θ for the orientation, as shown in Figure 1.

To locate the ellipses in a picture, the bounding boxes are constructed. The concept of bounding volume has been utilized in computer graphics fields for collision detection. The definitions of bounding boxes are given below for clarity of discussion.

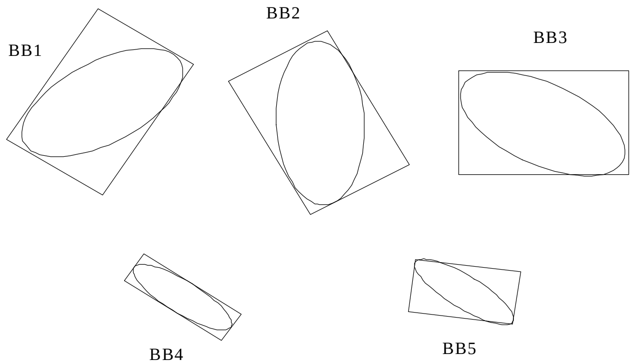

Figure 2. Bounding boxes.

Definition 1: A *bounding box* (*BB*) is a minimal rectangle that can cover an ellipse. It can have an arbitrary orientation.

Figure 2 shows some bounding boxes along different orientations. Notice that *BB1*, *BB2*, and *BB3* have the same elliptical shapes and corresponding bounding boxes, but their orientations are different. However, *BB4* and *BB5* have the same shapes and orientations inside but the shapes of the *BB*s are quite different.

Definition 2: An *oriented bounding box* (*OBB*) (Moller & Haines, 1999) is a box whose faces have normals that are all pairwise orthogonal.

All the bounding boxes (BB1 ~ BB5) in Figure 2 are OBBs.

Definition 3: An *axis-aligned bounding box* (*AABB*) (Moller & Haines, 1999) is a box whose faces have normals that coincide with the standard basis axes.

An AABB is a special case of the OBB. The BB3 in Figure 2 is an example of AABBs. The AABB can be obtained easily and computed efficiently in the applications. It will be used throughout this paper.

Theorem 1: Let E be an ellipse. If two horizontal scan lines (L_1, L_2) and two vertical scan lines (L_3, L_4) are tangents to the ellipse E at (P_1, P_2) and (P_3, P_4), respectively, then line segments $\overline{P_1P_2}$ and $\overline{P_3P_4}$ will intersect at the center (O) of E.

Proof: It was proved in (Yin & Chen, 1994) that all mid-points of the line segments at which the horizontal scan lines intersect with E lie on the same straight line. P_1 and P_2, which are the intersections of two horizontal lines tangents to E, can be considered as the boundary points of line segment $\overline{P_1P_2}$. Therefore, the line $\overleftrightarrow{P_1P_2}$ is referred to as the symmetric vertical axis. Similarly, the line $\overleftrightarrow{P_3P_4}$ is referred to as the symmetric horizontal axis. According to (Ho & Chen, 1995), the cross-point of the symmetric vertical axis and the symmetric horizontal axis is the center of *E*. ■

Theorem 2: The center of an ellipse is also the center of the corresponding bounding box.
Proof: Without loss of generality, we suppose that the center point of *E* is located at (0, 0). Let P_5 and P_6 be the intersections of lines L_1, L_2 and the *y*-axis; P_7 and P_8 be the intersections of lines L_3, L_4 and the *x*-axis, respectively. This is shown in Figure 3.

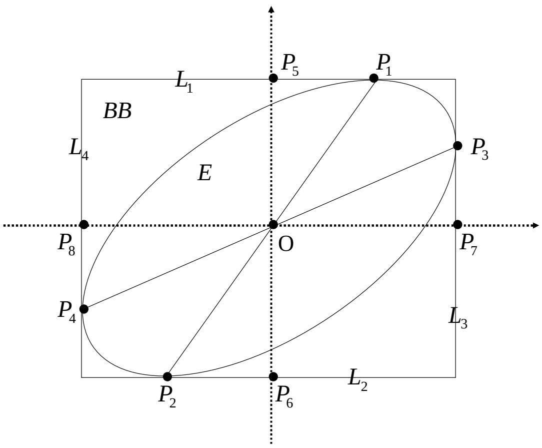

Figure 3. An ellipse E and its corresponding *BB.*

Following Theorem 1, all line segments passing through the center of *E* will be bisected by each other. We can obtain that

$$\overline{P_1O} = \overline{P_2O}\,.$$

Lines $\overleftrightarrow{P_1P_2}$ and $\overleftrightarrow{P_3P_4}$ cross at *O*, therefore,

$$\angle P_1OP_5 = \angle P_2OP_6\,.$$

Since L_1 and L_2 are parallel and cut by a transversal $\overleftrightarrow{P_1P_2}$, alternate interior angles are congruent, that is,

$$\angle P_5P_1O = \angle P_6P_2O\,.$$

According to the ASA Congruence Principle (Meyer, 1999), the triangles P_1OP_5 and P_2OP_6 are congruent. We have

$$\overline{OP_5} = \overline{OP_6}\,.$$

We can also prove, in the same way, that

$$\overline{OP_7} = \overline{OP_8} \,.$$

Thus, O is the center of both E and its corresponding BB. ■

3. PROPOSED APPROACH

In order to find the five unknown parameters, a method based on the bounding box is proposed. Our approach consists of two phases: bounding box locating and parameter estimation. First, the whole image is scanned to locate the bounding box for each ellipse. Each bounding box with an ellipse inside is identified and the center of the ellipse can be obtained. Then, for each bounding box, the boundary of the ellipse is traced to find the location of the major axis and its normal which is the corresponding minor axis. Finally, the orientation for each ellipse is then computed.

3.1 Bounding Box Locating

A bounding box is a minimal rectangle that can cover an ellipse. It can be any rectangle along an arbitrary axis as long as the ellipse is totally inside the boundary. For practical applications, the boundaries of a *BB* are usually set aligned with the directions of the coordinate system.

Before applying the proposed method to an image, the image is preprocessed to become a binary image. An image is scanned horizontally to locate four values ("top," "bottom," "left," and "right") of a bounding box. Let "0" denote a background pixel and "1" be a foreground pixel. There are four situations during the scanning: "0→0", "0→1", "1→0", and "1→1". Two of them are the most important and will be discussed in detail as follows.

a. 0→1:

This case means that a boundary from background to foreground is reached during the scanning. If this is the first time to reach an ellipse, the "top" and "left" positions are recorded for this ellipse. The "top" position is kept for each ellipse while the "left" position will be adjusted by selecting the smallest one in the subsequent scan lines. The boundaries of previous scan lines are also recorded so that the distinct ellipses can be distinguished if the differences between previous boundaries and current boundaries are greater than a preset threshold.

b. 1→0:

This case means that a boundary from foreground to background is reached during the scanning. The length of the connected foreground pixels in this scan line can then be obtained. If the length is smaller than a threshold, it will be considered as noise or fluctuation and discarded. Otherwise, the "right" position is set and will be adjusted by selecting the biggest one in the subsequent scan lines. The next scan line is also checked to see if the bottom of each ellipse is reached and the "bottom" position will then be set.

After the above processing, the number of ellipses in the image and the positions of the bounding boxes can be obtained. The pseudo code for this phase is illustrated as follows.

```
scan from top to bottom
    scan from left to right
        0→1:
            new ellipse:
                set boundary values;
            already used:
                decide which ellipse it belongs to;
                modify boundary values;
        0→0:
            skip;
        1→0:
            decide which ellipse it belongs to;
            if no. of connected pixels > threshold
                modify boundary values;
            else
                throw away;
            end if
        1→1:
            increase no. of connected pixels;
    end of scan
end of scan
```

3.2 Parameter Estimation

In this phase, we estimate the parameters of an ellipse, based on the information we already obtained. From Theorem 2, we know that the center of an ellipse is coincident to the center of the corresponding bounding box. Let *etop*, *ebottom*, *eleft*, and *eright* denote "top," "bottom," "left," and "right" positions of an ellipse, respectively. The center position (C_x, C_y) of an ellipse can be obtained as

$$C_x = \frac{eleft + eright}{2}; \tag{3}$$

$$C_y = \frac{etop + ebottom}{2}. \tag{4}$$

Then, the boundary of the ellipse within the bounding box is traced to find the position which maximizes the distance to the center position. This distance is taken to be the major axis. In order to reduce the search time, we estimate the rough orientation of the ellipse first. This can be done by finding the position of the foreground in the first scan line. If the number of pixels on the right-hand side is greater than that on the left-hand side, the tracing will be in a counterclockwise direction. Otherwise, the tracing will be in a clockwise direction. If the numbers are the same on both sides, either side can be used for tracing. Figure 4 illustrates these three phenomena.

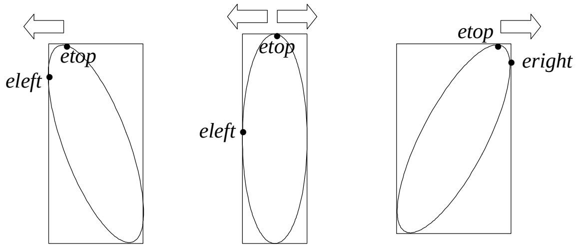

Figure 4. Possible directions of starting scan.

Once the searching direction is decided, the tracing procedure can be performed. The starting position is the first foreground pixel of the first scan line. During the tracing, the position which has the maximal distance to the center position is recorded. The operation is terminated when *eleft* (*eright*) is reached for tracing in a counterclockwise direction (in a clockwise direction).

Because of limited pixel resolutions of the output devices (such as display, scanner, etc.), some finite sizes of square pixels are formed around the high contrast edges. This is a well known defect in computer graphics, called jaggy. The edge information is forced into the horizontal and vertical edges of the pixels (Watt, 2000). In the first phase, the scan lines are from left to right, and top to bottom. The real boundary of a bounding box could not be identified correctly. Besides, some short boundary pixels might be considered as noise and discarded. As shown in Figure 5, the real boundary pixel is position “3”; however, position “1” is found in previous phase. In order to find the real maximal positions, some adjustments need to be made.

The searching path for “top” and “bottom” position (“left” and “right” position) is first along the horizontal (vertical) direction, and then along the vertical (horizontal) direction. If the found maximal position is on the boundary of the bounding box (*etop*), the searching will be along the horizontal line to find out the total number of foreground pixels. The selected maximal position is the middle point of the foreground pixels, even though the distance to the center position is not the maximum. For example, Figure 5 shows a portion of a bounding box. The thick line is the boundary of the bounding box. In previous scan, position “1” is recorded as the maximal position. Since it is on the boundary, we keep searching consecutive pixels until the background is reached. There are five foreground pixels and the middle one (position “2”) is selected.

Next, the searching procedure switches to the vertical direction. Since the “top” position needs to be decided, the searching path crosses over the top boundary of the bounding box to see if there is any foreground pixel connected to the current position (position “2” in this case). The searching priority is as follows:

upper (1), upper left (2), and upper right (3). If it does, the new position will be selected. This procedure is iterated until there is no more foreground pixel connected upwards. The final position with the extreme value can then be located.

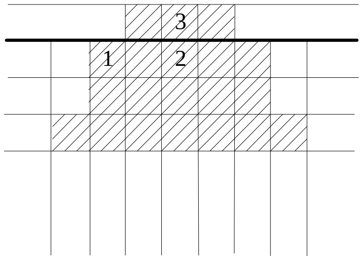

Figure 5. Upper portion of an ellipse.

Now, the position with maximal value has been found and therefore, the major axis can be computed. To compute the minor axis, it is done in a different way. Let O (C_x, C_y) be the center of the ellipse and P_1 (x_1, y_1) and P_2 (x_2, y_2) be the intersections of the ellipse with the major axis and minor axis, respectively. This is shown in Figure 1. Since lines $\overleftrightarrow{P_1O}$ and $\overleftrightarrow{P_2O}$ are perpendicular to each other, their slopes are the negative reciprocals of each other. Let line $\overleftrightarrow{P_1O}$ has slope m_1 and line $\overleftrightarrow{P_2O}$ has slope m_2, then we have

$$m_2 = -\frac{1}{m_1},$$

that is,

$$\frac{y_2 - C_y}{x_2 - C_x} = -\frac{x_1 - C_x}{y_1 - C_y}.$$

Therefore,

$$y_2 = C_y - (x_2 - C_x)(\frac{x_1 - C_x}{y_1 - C_y}). \tag{5}$$

We increase x_2 and compute y_2 to find the pixel, (x_2, y_2), which is on the boundary of the ellipse. Finally, the orientation of an ellipse can be computed as

$$\theta = \tan^{-1}(\frac{y_1 - C_y}{x_1 - C_x}). \tag{6}$$

The pseudo code for this phase is shown as follows.

for each ellipse
  compute the center position;
  find rough orientation of each ellipse;
  trace the boundary to find the maximal length to the center;
  adjust the maximal values on the boundary to obtain the major axis;

find the position of the minor axis along the normal direction of the major axis;
compute the orientation;
end for

4. EXPERIMENTAL RESULTS

In this section, some experimental results are presented to show the novelty of the proposed algorithm. The images of interest have the size of 512×512. Figure 6 shows a preprocessed image. It contains four different sizes of ellipses with distinct orientations. In the first phase, the image is scanned to locate the bounding box for each ellipse. Four bounding boxes are found in this image, as shown in Figure 7. Notice that the ellipse on the left-hand side has some pixels that are not inside the bounding box. This will not affect our results because they are not on the major axis or the minor axis.

For those ellipses whose orientations are horizontal or vertical, adjustments are applied to obtain real positions for the major axes and the minor axes. Figure 8 is a subimage of the bottom ellipse in Figure 6 (the upper portion of the ellipse). The orientation of the major axis is in the vertical direction and two pixels on the top are excluded from the bounding box, because of the short consecutive pixels. At the beginning, the maximal distance to the center was found at (317, 268). This is the first foreground pixel of the second scan line in Figure 8. After adjustments, the real position was decided at (316, 273), which is the second foreground pixel of the first scan line. Note that the origin is at the upper left corner.

After applying our algorithm to the image, the final results regarding the major axis and the minor axis are illustrated in Figure 9. The parameters of each ellipse are shown in Table 1.

Figure 6. A preprocessed image.

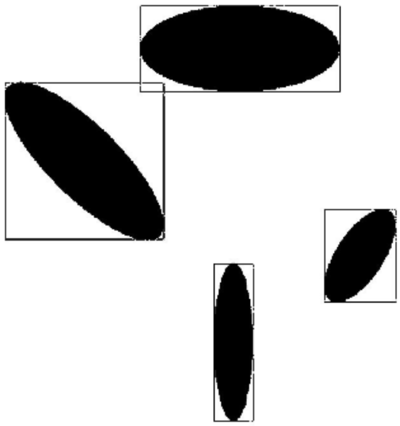

Figure 7. The bounding box for each ellipse.

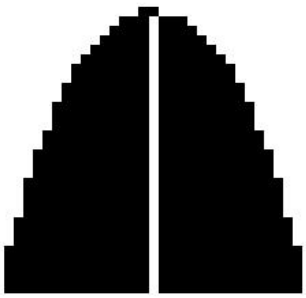

Figure 8. A subimage of the bottom ellipse in Figure 6.

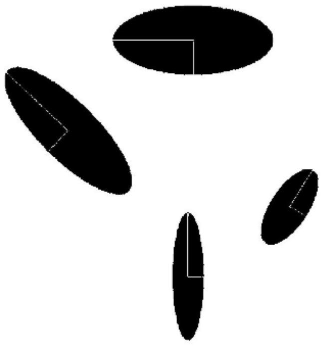

Figure 9. Major axes and minor axes.

Table 1. *Parameters of ellipses*

Ellipse	Center	Major Axis	Minor Axis	Orientation*
1	(80,280.5)	109.5	47	1.57
2	(204,110)	115.3	39.6	0.82
3	(308,412)	58.5	26.1	2.56
4	(403,273.5)	87.5	20.5	-0.0057

Note. * Unit in radian.

5. CONCLUSIONS

A fast algorithm for ellipse detection based on the use of bounding boxes is proposed in this paper. A bounding box for each ellipse is first identified to compute the center of an ellipse. The tracing area for the major axis is confined to a small corner. The minor axis can then be obtained along the normal direction of the major axis. The limitation of this algorithm is that it can only be applied to locate perfect ellipses. Simulation results show the effectiveness of our algorithm. Without the time-consuming procedure to search the parameter space for the Hough transform methods, our approach is fast and space saving.

REFERENCES

Aguado, A. S., Montiel, M. E. & Nixon, M. S. (1995). Ellipse detection via gradient direction in the Hough transform. *Proceedings of 5th IEEE International Conference on Image Processing and its Applications*, Edinburgh, UK, 375-378.

Chang, C. M. & Hsiao, H. W. (2005). A novel ellipse detector based on bounding frames. *Proceedings of 18th IPPR Conference on Computer Vision, Graphics, and Image Processing, Taipei, Taiwan*.

Dave, R. N. (1992). Generalized fuzzy c-shells clustering and detection of circular and elliptical boundaries. *Pattern Recognition*, *25*, 713-721.

Fitzgibbon, A. W., Pilu, M. & Fisher, R. B. (1999). Direct least squares fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *21*(5), 476-480.

Hartley, R. & Zisserman, A. (2000). *Multiple view geometry in computer vision*. Cambridge, UK: Cambridge University Press.

Ho, C. T. & Chen, L. H. (1995). A fast ellipse/circle detector using geometry symmetry. *Pattern Recognition*, *28*(1), 117-124.

Lee, R., Lu, P. C. & Tsai, W. H. (1990). Moment-preserving detection of elliptical shapes in gray-scale images. *Pattern Recognition Letters*, 11, 405-414.

McLaughlin, R. A. (1996). Randomized Hough transform: Better ellipse detection. *Proceedings of IEEE TENCON – Digital Signal Processing Applications*, Perth, Australia, 409-414.

Meyer, W. (1999). *Geometry and its applications*. New York, USA: Academic Press.

Moller, T. & Haines, E. (1999). *Real-time rendering*. Natick, MA, USA: A. K. Peters, Ltd.

Protter, M. H. & Protter, P. E. (1998). *Calculus with analytic geometry*. Boston, MA, USA: Jones and Bartlett Publishers.

Ser, P. & Siu, W. (1994). Novel 2-D Hough planes for the detection of ellipses. *Proceedings of International Symposium on Speech, Image Processing, and Neural Networks*, Hong Kong, 527-530.

Watt, A. (2000). *3D computer graphics.* (3rd ed.). Harlow, England: Pearson Education Limited.

Wu, W. Y. & Wang, M. J. (1993). Elliptical object detection by using its geometric properties. *Pattern Recognition*, *26*, 1499-1509.

Yin, P. Y. & Chen, L. H. (1994). A new method for ellipses detection using symmetry. *Journal of Electronic Imaging*, *3*, 20-29.

Yip, K. K. R., Tam, K. S. P., & Leung, N. K. D. (1992). Modification of Hough transform for circles and ellipses detection using a 2-dimensional array. *Pattern Recognition*, *25*, 1007-1022.

Yoo, J. H. & Sethi, I. K. (1993). An ellipse detection method from the polar and pole definition of conics. *Pattern Recognition, 26*, 307-315.

Chun-Ming Chang received the B.S. degree from National Cheng Kung University, Taiwan, in 1985 and the M.S. degree from National Tsing Hua University, Taiwan, in 1987, both in electrical engineering. He received the Ph.D. degree in electrical and computer engineering from University of Florida in 1997.

From 1998 to 2002, Dr. Chang served for two communication companies as a senior technical staff member and a senior software engineer, respectively. He joined the faculty of Asia University in Taiwan in 2002. His research interests include computer vision/image processing, video compression, computer networks and robotics.