ELLIPSE DETECTION USING FREEMAN CHAIN CODE

Shammi Sharma¹, Mandeep Singh Saini², Miss Palvee³

¹M.Tech Student, IITT Pojewal,
²Ass. Professor and Dean (R&D), ECE Department, IITT College of Engineering, Pojewal,
³Assistant Professor, EIE Department, IITT College Of Engineering, Pojewal

Abstract: This paper presents an algorithm to detect and accurately locate ellipse objects in digital images. The algorithm consists of two steps. First the edge pixels are extracted using any edge detection technique. In the second step, Freeman Chain code is applied to detect ellipse from the digital images. The code is done in MATLAB R2010a.

Keywords: Edge detection; Freeman Chain Code; Ellipse Detection.

Introduction

A human being can find out a particular object like Ellipse or Circle from various objects by observing their shape, color, texture and feature and after that we can calculate properties of shapes like perimeter, area etc. For this, use edge detection algorithm. Edge detection is the first and very important step in shapes detection. In this paper we use edge detection techniques for edge detection like Sobel, Prewitt, Robert cross operators and canny edge detection algorithm. It is a multi-step detector which performs smoothing and filtering, non-maxima suppression, followed by a connected-component analysis stage to detect “true” edges, while suppressing “false” non edge filter responses. After finding edges in image we apply Freeman chain code for the ellipse detection. The flow-chat is shown in figure 1 to detect ellipse from images using Freeman Chain Code [1].

A. Edge detection refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene.

There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories:

- **Gradient:** The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.
- **Laplacian:** The Laplacian method searches for zero crossings in the second derivative of the image to find edges.

Edge Detection Techniques: To find the edges from the image there are various techniques are used as shown below. We can use any one technique.

1. **Sobel Operator:** The operator consists of a pair of 3×3 convolution kernels as shown in Figure 1. One kernel is simply the other rotated by 90°.

   \[
   \begin{array}{ccc}
   -1 & 0 & +1 \\
   -2 & 0 & +2 \\
   -1 & 0 & +1 \\
   \end{array}
   \]

   \[G_x\]

   \[
   \begin{array}{ccc}
   +1 & +2 & +1 \\
   0 & 0 & 0 \\
   -1 & -2 & -1 \\
   \end{array}
   \]

   \[G_y\]

2. **Robert’s cross operator:** The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point.

   The operator consists of a pair of 2×2 convolution kernels as shown in Figure. One kernel is simply the other rotated by 90°. This is very similar to the Sobel operator.

   \[
   \begin{array}{cc}
   +1 & 0 \\
   0 & -1 \\
   \end{array}
   \]

   \[G_x\]

   \[
   \begin{array}{cc}
   0 & +1 \\
   -1 & 0 \\
   \end{array}
   \]

   \[G_y\]
3. **Prewitt’s operator**: The convolution Kernels for Prewitt’s Operator are:

\[
\begin{array}{ccc}
1 & 1 & 1 \\
0 & 0 & 0 \\
-1 & -1 & -1 \\
\end{array}
\]

\[
\begin{array}{ccc}
1 & 0 & 1 \\
-1 & 0 & 1 \\
-1 & 0 & 1 \\
\end{array}
\]

\(G_x\) \hspace{1cm} \(G_y\)

4. **Canny Edge Detection Technique**:

Canny edge detection is one of the basic algorithms used in shape recognition. The algorithm uses a multi-stage process to detect a wide range of edges in images. The stages are:

- Image smoothing. This is done to reduce the noise in the image.
- Calculating edge strength and edge direction.
- Directional non-maximum suppression to obtain thin edges across the image.
- Invoking threshold with hysteresis to obtain only the valid edges in an image.

A block diagram of the canny edge detection algorithm is shown in Figure 2. The input to the detector can be either a color image or a grayscale image. The output is an image containing only those edges that have been detected.

![Figure 2: Block Diagram of Canny Edge Detection Algorithm](image)

Canny Edge Detection approach is based on three basic objectives[8]:

1. **Low error rate.** All edge should be found, and there should be no spurious response. That is, the edges detected must be as close as possible to the true edges.
2. **Edge points should be well localized.** The distance between a point marked as an edge by the detector and the center of the true edge should be minimum.
3. **Third Criteria is to have only one response to a single edge.**

![Figure 1: Algorithm for Ellipse detection in Digital Images is](image)

**A. Freeman Chain Code for Ellipse Detection:**

Chain code is a list of codes ranging from 0 to 7 in clockwise direction[5][6]. These codes represent the direction of the next pixel connected in 3x3 window, as shown in table 1. The coordinates of the next pixel is calculated based on the addition and subtraction of columns and row by 1, depending on the value of chain code. Corresponding to the code in table 1, the next pixel position can be obtained by using table 2. For example, if a current pixel is located at coordinate (5,5), the coordinate of the next pixel based on chain code is given by table 2. The disadvantage is that we have to scan all the eight neighboring pixel while ellipse detection.

**Table 1. Chain code**

<table>
<thead>
<tr>
<th>Column-1</th>
<th>Column</th>
<th>Column+1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row-1</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Row</td>
<td>4</td>
<td>Current pixel</td>
</tr>
<tr>
<td>Row+1</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 2. Pixel position

<table>
<thead>
<tr>
<th>Code</th>
<th>Next Row</th>
<th>Next Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

At each pixel we determine the position of next pixel and so on the outline of the whole object can be obtained.

B. Results: Input Image:

D. REFERENCES:


Author:

Mandeep Singh Saini received his B.Tech ECE from Nagpur University (Maharashtra) during 2001. They obtained his M.Tech ECE from GNDEC Ludhiana (Punjab). His field of interest is Digital Signal Processing, Wireless Communication & VLSI Design. In future he wants to work in High Performance VLSI Architectures. He published papers in various aspects in Signal Processing. He has authored more than 12 research papers in national, international conferences and reputed journals.

Shammi Sharma received her B.Tech ECE Degree from KC College of Engineering and IT, Nawanshahr, Punjab. They obtain M.Tech in ECE from IITT Pojewal. Her field of interest is Digital Signal Processing, Digital Image Processing &Wireless Comm. In future she wants to work in Digital Image Processing. She has authored 02 research papers in reputed journals.