Hough transform algorithm for real-time pattern recognition using an artificial retina camera

Xin Lin and Kazunori Otobe
National Agriculture Research Center, 3-1-1 Kannondai, Tsukuba, 305-8666 Japan
xlin@narc.affrc.go.jp

Abstract: An artificial retina camera (ARC) is employed for real-time preprocessing of images. And the algorithm of Hough transform is advanced for detecting the biology-images with approximate circle edge-information in the two-dimension space. This method also works in parallel for processing multiple input and partial input patterns.

©2001 Optical Society of America

OCIS codes: (100.0100) Image processing, (070.5010) Pattern recognition and feature extraction

References and links

1. Introduction

The Hough transform (HT) [1] is an effective technique for detecting and finding the images within noise. Specifically, the straight-line detection case has been ingeniously exploited in several applications. However, the HT techniques to recognize other shapes except for the straight-line are quite complex because the points in the input space will be mapped to hypersurfaces and thus, the Hough space will become high dimensional. So that the calculation and memory cost is very high. A number of improved methods have been proposed to detect arbitrary shapes [2,3].

In fact, in an image, the pertinent information is very often contained in the shape of its boundary. The crude encoding of the boundary is sufficient for image recognition, i.e. an image may be initially encoded as an edge-image. In this paper, we describe an algorithm of modificatory HT for detecting the biology-images using its edge-information only in the two-dimension (no hypersurfaces). On the other hand, an artificial retina camera (ARC) is employed for edge-extraction processing in real-time. This method can work to process multiple and partial input patterns [4].

2. Principles of the method

2.1 Modificatory Hough transform for detecting circle-edge image

The straight-line HT maps point to sinusoids as shown in Fig.1(b). If there a peak point in Hough space and it is the cumulative value of all the sinusoids, then a straight-line can be
defined in the input space, as shown in Fig. 1(a). We can see that the Hough space only is a simple two-dimension (2-D) space.

In order to reduce the dimensionality in the Hough space when detecting a circle object, we describe an algorithm for detecting circle objects from an image that has been transformed into such an edge representation, and then, with the help of the concept of straight-line HT to reach detection circle patterns only in 2-D space. The principles of this method are described as follows.

Any image of edge representation can be considered to be composed small, tangential, straight-line segments. So we can try detecting a circle-edge image by the straight-line HT. Each of the short tangential-line segments in the input space maps to a point in the Hough space, with a value proportional to the length of the tangential-line segment. There are a large number of short tangential-line in the input edge-image, and these segments are mapped to different small peaks in the Hough space. If the input image has a continuous edge curve (such as a circle), then this pattern in the Hough space will also be a continuous curve (after thresholding): 

\[ \rho = H(\alpha, \ldots, \alpha, \theta), \text{ where } \alpha, \ldots, \alpha \text{ are the parameters of the curve.} \]

Circles can be approximated by straight-line segments that are tangential to the circle at various points. If the center of the circle is at the origin of the input image, then the peaks in the Hough space that arise due to these tangential-line segments can be described by

\[ \rho = H(\alpha, \ldots, \alpha, \theta) = H_1(r, \theta) = r, \]

where \( r \) is the radius. In this case, the Hough space only is a simple horizontal line. When the circular object is not centered, the locus of peak-points in the Hough space given by

\[ \rho = \begin{cases} r + t \cos(\theta - \alpha) & r + t \cos(\theta - \alpha) \geq 0 \\ -r + t \cos(\theta - \alpha) & r + t \cos(\theta + \pi - \alpha) < 0 \end{cases}, \]

where \( t = x_0^2 + y_0^2, \alpha = \tan^{-1}(y_0/x_0) \) are the polar coordinates, and \((x_0, y_0)\) is origin location. The periodic of \( \rho \) is \( 2\pi \). To conveniently discussion, we define the space that described by Eq. (2) is “HT-I space”. We can see that the pattern formed in HT-I is a cosine curve, which is unsymmetrical for the abscissa (\( \theta \) axis). The amount of translation will determine the radius \( r \) of the circle. So that when the radius of the circle is not known, we can translate this unsymmetrical cosine curve into a symmetrical type, and then the value of translation is the radius of this circle. We define this space that is obtained after translation is “HT-II space”, it can be described by

\[ \rho' = \rho \pm r. \]

Performing an inverse HT to the HT-II space [i.e. Eq. (3)], we will obtain a peak at the center of the detected circle. Thus, our algorithm involves three steps: (1) calculating the straight-line HT to the circle-edge image for forming the HT-I space, (2) translating the HT-I space for into the HT-II space, and (3) performing an inverse HT to the HT-II space.

2.2 Processing principle of ARC
Real-time recognition and understanding of images is one of important information processing technologies for real world. However, a lot of present image processing systems separate image sensing and image processing, i.e. images are sensed by a camera and processed (edge extraction etc.) with the help of a computer. So the performance speed will be limited by slow camera frame rates and low transmission rates between the camera and the computer.

In this work, we consider to use an ARC (Mitsubishi PCA6010AG10-20A), which can simultaneously sense and process images on chip [5,6]. Figure 2 shows a schematic structure of the ARC, including a artificial retina (AR) chip of 128×128 pixels and a 16-bit microprocessor with a clock speed of 10MHz. The AR chip is a combination of a 2-D array of variable-sensitivity photodetector (VSPD) cells for parallel optics computation, a random access scanner to control all the VSPDs’ sensitivity, and a multiplexer for image output. Major image-processing functions of the AR chip include image acquisition, edge extraction, and 1-D projection. The roles of the microprocessor are providing the sensitivity control signal for the AR chip, and processing the output data from the AR chip. Examples of image processing by the ARC are shown in Fig. 3.

![Fig. 2. Schematic structure of the ARC.](image)

3. Experiment and results

3.1 Experiment I: detecting a singular circle

In this experiment, a cabbage is used for the input image. The ARC is employed to simultaneously sense input image and extract its edge information, as shown in Fig. 4. Figure 4(a) is original input image with the size of 128×96 pixels. The results of image processing are shown in Fig. 4(b) and 4(c). From Fig. 4(c), the edge of the cabbage’s core is the parts of two circles (internal and external circle).

![Fig. 4. Input image (one cabbage): (a) normal image, (b) binarization, and (c) edge extraction.](image)
In our experiment, a sampling interval of 1 was used for $\rho$ and $\theta$ for all HT spaces. Each of the short tangential-line segments is length of 6 pixels. So these line segments will map to small peaks of 6 in the HT-I space and form a pattern in HT-I space. We chose a threshold of 6, which equal to the length of the tangential-line, then the pattern in HT-I space as shown in Fig. 5(a) after thresholding. They are parts of the cosine curve corresponds to Eq. (2). When there are some noises that the length is longer than the tangential-line segments in the input image, we can use the straight-line HT to detect these straight-lines/noises in first, and then use our method for forming the HT-I space.

Translating the pattern in HT-I by Eq. (3) into two perfect cosine curves (HT-II space) as shown in Fig. 5(b), we can obtain the radius of two circles. Furthermore, performing an inverse HT to this HT-II space, the centers of two circles are obtained at two cumulative points as shown in Fig. 5(c). Figure 5(d) shows the extractive circles. Their radiuses are 24 and 30 pixels, and centers at (1, 0) and (1, -1), respectively.

![Fig. 5. Experiment results: (a) in HT-I space, (b) in HT-II space, (c) in inverse HT space, and (d) extractive circles.](image)

3.2 Experiment II: detecting multiple circles

This method also works in parallel for detecting multiple input patterns. We use eight cabbages with size $128 \times 80$ pixels as the input image as shown in Fig. 6(a). The method of image processing is same as the experiment I. The results of image processing are shown in Fig. 6(b) and 6(c).

![Fig. 6. Input image (multiple cabbages): (a) normal image, (b) binarization, and (c) edge extraction.](image)
In this experiment, we use the straight-line HT to extract four lines and noises in Fig. 6(c) before the circle detection. Then we chose a threshold of 5 to form the pattern in HT-I space as shown in Fig. 7(a). Mapping the patterns in the HT-I space into the perfect cosine curves (HT-II), the radiuses of six circles are obtained, as shown in Fig. 7(b). The centers of six circles are obtained by the inverse HT as shown in Fig. 7(c). Figure 7(d) shows the extractive circles with respective radius and position. The cabbage 1 and 8 in Fig. 6(c) cannot be detected because their edge patterns are not a circle so that as noises are filtered.

![Fig. 7](image-url)  
Fig. 7 Experiment results: (a) in HT-I space, (b) in HT-II space, (c) in inverse HT space, and (d) extractive circles.

### 3.3 Results

To demonstrate the reliability of the proposed method for biology-image detection, we have also detected four cabbages with different radius, respectively. The results are shown in Fig. 8. As is seen from this figure, there is good linear relation between the measurement diameters and the estimation diameters by using the proposed method.

![Fig. 8](image-url)  
Fig. 8. Relation between the measurement values and estimation values
4. Conclusion

We have described a HT algorithm for real-time pattern detection using an ARC. In this method, the ARC is employed to simultaneously sense and process images on chip. The algorithm of the straight-line HT is advanced for detecting circle objects from an image that has been transformed into such an edge representation in the 2-D space. This method also works in parallel for multiple inputs and for partial input objects.

On the other hand, if we substitute other analytic curves such as the ellipse or the parabola etc. into Eq. (1), this method will can be used for detecting more type shapes. Further study should be conducted.