Expert Systems With Applications 51 (2016) 26–33

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Curvature aided Hough transform for circle detection

Zhenjie Yao[a,b,*], Weidong Yi[b]

[a] Center of Excellence for mHealth and Smart Healthcare, China Mobile Research Institute, Beijing 100053, China
[b] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Keywords:
Curvature
Hough transform
Circle detection
Statistic deviation

ABSTRACT

Conventional Hough based circle detection methods are robust, but for computers in last century, it is to slow and memory demanding. With the rapid development of computer hardware, Hough transform is acceptable now. Improvement on Hough based circle detection is valuable. In this paper, we present a novel curvature aided Hough transform for circle detection (CACD) algorithm, which estimates the circle radius from curvature. Curvature pre-estimation is capable to avoid both accumulating operations of all the points and interruption between different scales, which result in faster and more precise circle detection. Compared to the conventional Hough-based algorithm for circle detection, the algorithm is more practical and less time consuming. Its time taking is about 1/8 of that of conventional algorithm. Test results on traffic sign images shown that The CACD gets an AUC (Area Under Curve) of 0.9125. The CACD is capable to detect circles of different radius in complex scene.

1. Introduction

Circle detection is an important issue in computer vision and pattern recognition. Circle is one of the most frequently occurring shapes in nature. It constructs celestial body in macro-scope; cell/atom in microscopy; even a lot of fruit in everyday life. More than that, many man-made objects are designed in circular, such as traffic signs, warning signs, basketball and football. These circular objects contain rich information. In numerous computer vision applications, we need to find out the information in the circular objects. One of the examples is traffic sign understanding or camera calibration. Circle detection is the first step before further understanding the information contained in the circular objects.

Most circle detection algorithms are based on Hough transform. Standard Hough transform circle detection needs huge memory and take a long time for calculation (Duda & Hart, 1972). Researchers tried to improve it. Davies took gradient orientation into consideration, calculated Hough transform accumulator on gradient orientation only Davies (1988). Kimme et al. recognized circle from single 2-D accumulator (Kimme, Ballard, & Sklansky, 1975). Yuen, Princen, Illingworth, and Kittler (1990) and Atherton and Kerbyson (1999) speed up the calculation by Hough transform filter. The improved works need prior knowledge such as radius range; otherwise, the calculation will be time consuming and need a lot of memory. The detection performance depends on whether the prior knowledge about the circles is accurate. Moreover, they will fail to detect some circles when there are multiple circles with different radius in complex scene, because circles with small radius will be submersed by big ones.

On the other hand, researchers proposed detection method so called Randomized Circle Detection (RCD). The essential part of this work is: randomly select 3 points for circle fitting, then check if there is a circle by examining whether there are enough edge points on the fitted circle (Chen & Chung, 2001; Xu, Oja, & Kultanen, 1990). There are also some improvements on this work. Jiang randomly selected 4 points and check every 3 points in them (Jiang, 2010). Yu and Wei selected the 3 points from the same curve, and a shrinking the edge collecting range was adopted (Dan & Wei, 2009). Huang and Li directly fit a circle from points with the similar curvature (Huang & Li, 2007). De Marco et al. also take curvature into consideration to improve RCD (De Marco, Cazzato, Leo, & Distante, 2015). Gonzalez adopt hash map to speed up the search procedure of the RCD algorithm (Gonzalez, 2015). This kind of algorithms is highly randomized. Suppose we randomly select 3 points from an image with complex texture, the possibility of the points are from the same circle is small. We have to suffer failure again and again until the 3 selected points are from the same circle. It takes a long time to find a circle successfully. Even worse, the randomness of RCD causes unstable detection results. Another kind of circle detection method is arc-based. Kim et al. use a least-squares for arc segment fitting

* Correspondig author at: Center of Excellence for mHealth and Smart Healthcare, China Mobile Research Institute, Beijing 100053, China. Tel.: +86 15801696688x33499.

E-mail addresses: yaozhenjie@gmail.com (Z. Yao), weidong@gucas.ac.cn (W. Yi).

http://dx.doi.org/10.1016/j.eswa.2015.12.019

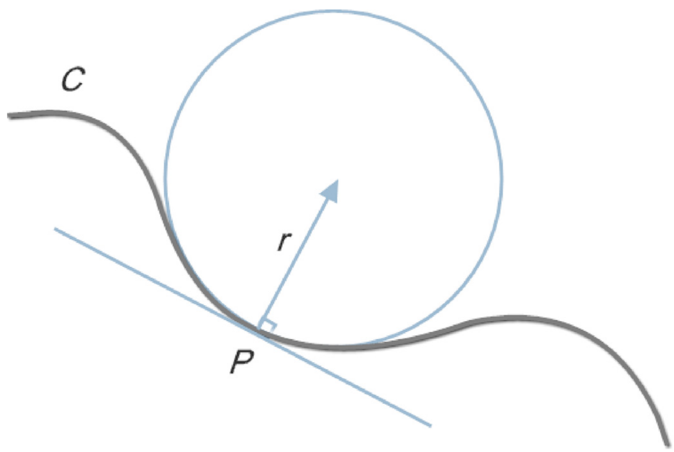

Fig. 1. Geometric illustration of curvature radius.

to detect circles (Kim, Haseyame, & Kitajima, 2002). Akinlar and Topal give a very fast arc-based circle detection method, Helmholtz principle is adopted to eliminate false detection (Akinlar & Topal, 2013).

This paper presents the curvature aided Hough transform for circle detection(CACD), which finds out the circle center and radius by adaptively estimating curvature radius. The CACD is capable to detect circles of different radius in complex scene. Compared to conventional circle detection methods, it is more practical and less time consuming.

The remaining of the paper is organized as follows. Section 2 describes the curvature aided Hough transforms for circle detection step by step. Experimental results are presented and discussed in Section 3. Finally, Section 4 concludes the paper.

2. Curvature aided Hough transform for circle detection

In geometry, curvature measures change rate of curve direction. In other words, it measures how deviant a curve is from straight line. As shown in Fig. 1, the inverse of curvature is curvature radius, which is the radius of osculating circle. If the point is on the circle, its curvature radius should be the radius of the circle. According to this property, we propose the curvature aided Hough transform for circle detection (CACD) algorithm. The remaining of the section describes the algorithm step by step.

2.1. Edge detection

First of all, the edge of the image is extracted. As other algorithms, the CACD algorithm works on the edge of the image, so edge detection is necessary before further processing. Canny edge is the choice. The main steps of Canny edge detection including: image smoothing, gradient calculation, non-maximum suppression, tracing edge by two thresholds. More details about Canny edge detection can be found in Canny (1986). After edge detection, we get a binary edge image.

2.2. Curve extracting

After edge detection, we extract curve from edge detection result by connectivity analyzing. If endpoints of any two curves are close, we fill the gap between them and merge them into one curve. Here the merging is allowed if the gap size is no more than 1 pixel. Generally speaking, curve composed by few points is generated by noise or some unimportant details, they ought to be abandoned. Here curve composed by less than 10 points are abandoned. Finally, extracted curves have the following format:

$$L_i = \begin{bmatrix} x_0 & x_1 & \dots & x_n \\ y_0 & y_1 & \dots & y_n \end{bmatrix}, \tag{1}$$

which indicates the point coordinates one by one, line i is composed by n points, whose coordinates are (x_0, y_0), (x_1, y_1),..., (x_n, y_n), respectively.

2.3. Curvature radius estimation

When we get the curves, we can calculate curvature point by point for each curve. As literature (He & Yung, 2004), for each curve, Gaussian smooth is necessary. Smoothing not only suppresses spikes on the curve, but also calculates the curvature in sub-pixel precision. The curvature is calculated by

$$c(u,\sigma) = \frac{\dot{X}(u,\sigma)\ddot{Y}(u,\sigma) - \ddot{X}(u,\sigma)\dot{Y}(u,\sigma)}{(\dot{X}(u,\sigma)^2 + \dot{Y}(u,\sigma)^2)^{1.5}}, \tag{2}$$

where $g(u, \sigma)$ is 2-D Gaussian window with u as the center position and σ as the deviation. $\dot{X}(u,\sigma) = x(u) * \dot{g}(u,\sigma)$, $\ddot{X}(u,\sigma) = x(u) * \ddot{g}(u,\sigma)$, $\dot{Y}(u,\sigma) = y(u) * \dot{g}(u,\sigma)$, $\ddot{Y}(u,\sigma) = y(u) * \ddot{g}(u,\sigma)$, $\dot{g}(u,\sigma)$ and $\ddot{g}(u,\sigma)$ are the 1st and 2nd derivative of $g(u, \sigma)$, respectively. And $f*g$ indicate the convolution of f and g.

We need to get the inverse of curvature as radius estimation. It is worth to say that if curvature is 0, we cannot find its inverse. If the curvature is very small, its inverse will be very large, the radius will be much larger than the image size, and they are not taken into consideration. So are the pixels with very large curvature, the corresponding radius is very small, which always generated by noise. They are abandoned, too. Finally, the radius is estimated by

$$r(u,\sigma) = \begin{cases} \frac{1}{c(u,\sigma)} & \frac{2}{3m} < c(u,\sigma) < 1 \\ 0 & otherwise \end{cases}, \tag{3}$$

in which m is the max size of image.

Curvature radius estimation of a circle curve is shown in Fig. 2, we can see that the true radius of the circle is 21 pixels, the radius estimation of each pixel is in the range of [15,28]. The estimation result of curvature radius is not precise. It is just a coarse approximation. However, as shown in remaining of the paper, such approximation is enough for circle detection.

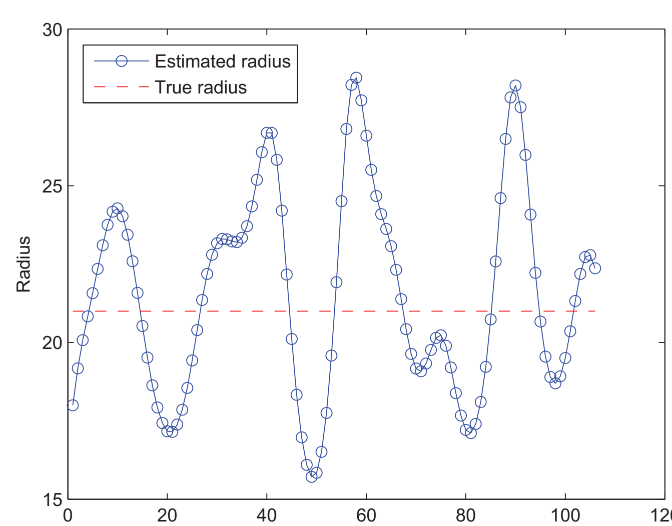

Fig. 2. Curvature radius estimation on part of a circle.