Pattern Recognition 54 (2016) 218–228

Contents lists available at ScienceDirect

Pattern Recognition

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

A fast and robust circle detection method using isosceles triangles sampling

Hanqing Zhang, Krister Wiklund, Magnus Andersson *

Department of Physics, Umeå University, Linneaus Vaeg 9, SE-901 87 Umeå, Sweden

ARTICLE INFO

Article history:
Received 7 September 2015
Received in revised form
4 November 2015
Accepted 3 December 2015
Available online 23 December 2015

Keywords:
Circle detection
Randomized algorithm
Sampling strategy
Isosceles triangles

ABSTRACT

Circle detection using randomized sampling has been developed in recent years to reduce computational intensity. However, randomized sampling is sensitive to noise that can lead to reduced accuracy and false-positive candidates. To improve on the robustness of randomized circle detection under noisy conditions this paper presents a new methodology for circle detection based upon randomized isosceles triangles sampling. It is shown that the geometrical property of isosceles triangles provides a robust criterion to find relevant edge pixels which, in turn, offers an efficient means to estimate the centers and radii of circles. For best efficiency, the estimated results given by the sampling from individual connected components of the edge map were analyzed using a simple clustering approach. To further improve on the accuracy we applied a two-step refinement process using chords and linear error compensation with gradient information of the edge pixels. Extensive experiments using both synthetic and real images have been performed. The results are compared to leading state-of-the-art algorithms and it is shown that the proposed methodology has a number of advantages: it is efficient in finding circles with a low number of iterations, it has high rejection rate of false-positive circle candidates, and it has high robustness against noise. All this makes it adaptive and useful in many vision applications.

1. Introduction

Detection of circular objects is a fundamental feature extraction task in pattern recognition, which has continuously been developed to achieve better computational performance and accuracy [1,2]. Several applications within the fields of; computer vision, physics, and biology, relies on detection of circular patterns, e.g., iris detection [3], cell counting [4], cell shape identification [5], and tethered particle tracking [6]. Apart from the execution time and accuracy, the major challenges when detecting circles are the presence of noise, low contrast, distortion and blurring boundaries, which are common in real images. These are either due to the settings of the image sensor or bad light conditions resulting in excessive amount of irrelevant textures, incomplete and distorted circle contours, and an added number of false-positives. Therefore, there is a need of circle detection methods with robustness against noise in images since these are handled rather poorly in conventional circle detection algorithms. The motivation and aim of this work is hence to improve the detection of circular shapes in the presence of noises.

The Hough transform (HT) was invented in the 50s and is a technique that can isolate features of a particular shape, e.g., circles in an image [7,8]. In the circular Hough transform (CHT), an accumulator array is created for mapping the extracted edge pixels in the image to circle parameters in a 3-dimensional (Hough) space. This technique reaches a relatively high degree of accuracy but at the cost of the huge demand of memory space for the accumulator, and high computational complexity due to the time-consuming voting procedure. In addition, the indiscriminative voting procedure is also susceptible to noise for complex backgrounds.

Over decades, modifications of the CHT algorithm have been made to improve the accuracy and execution time. One strategy is to use geometrical constraints, where three commonly used constraints are; gradient-, chord-, and geometry-symmetry-based. The gradient-based approach [9] was proposed to accelerate the voting process. Ideally, the gradient of edge points on the circumference of a circle points at the center of the circle. This simple geometrical property can be used to reduce the dimension of a 3D accumulator to a 2D accumulator according to Yip et al. [10]; Rad et al. included both gradient and symmetric constraints for the selection of circles [11]; Jia et al. used a gradient based method to determine the circle center [12]; and Pan et al. calculated the gradient to build center projection lines in parameter space to locate the maximum intersecting points [13]. However, it is

* Corresponding author. Tel.: +46 90 786 6336.
E-mail address: magnus.andersson@umu.se (M. Andersson).

http://dx.doi.org/10.1016/j.patcog.2015.12.004
0031-3203/

reported that using gradients is more sensitive to noise compared to using the edge points [14]. The chords-based methods find the centers at which perpendicular bisectors of the chords intersect. Better robustness against noise and distinctive probability distribution in the voting space can be found according to refs. [15,16]. However, the computational complexities of calculating the intersecting point become a major drawback for such a method. The global-symmetry of a circle can also be used to locate center points accurately and efficiently [17]. However, this property is based on the assumption that circles are ideal and complete, which is not very sufficient in handling occlusion and noisy conditions. Global symmetry are often used in combination with other properties to enhance the accuracy and efficiency [18].

Besides using the geometrical constraints, the enumerative voting process of the traditional Hough transform has been modified using randomized sampling for efficiency. The Randomized Hough transform (RHT) replaces the fundamental voting procedure for HT-based methods with random sampling of three edge points, and apply converging mapping as well as dynamic storage [19]. Based on a randomized sampling strategy several refined methods have been developed. Chen and Chung proposed a Randomized circle detection (RCD) algorithm using four edge points sampling [20]. According to their experimental results, the performance of RCD outperforms RHT in low to modest noise levels. By using four sampling points instead of three, the sampling strategy in RCD generates considerably less number of circle candidates in the validation process, which reduce a large amount of computational overhead. Inspired by the four-points sampling strategy, with a fourth point as validity check, multiple-evidence-based sampling strategy is proposed to further reduce the computational overhead in GRCD-R and GLRCD-R [21].

Improved circle detection work-flow, using classification of edge candidates or generation of line segments, has also been shown to reduce the computation time and efficiently find circle candidates. The EDCircles algorithm [2] achieves parameter-free, real-time performance using line segments generated by the Edge Drawing Parameter Free (EDPF) algorithm [22]. The edge segments were also applied in [23] along with a top-down scheme to enhance efficiency. Isophote curvature analysis was applied in [24] as a classification of edge pixels that improved the robustness and had limited dependency from the edge extractor. All these methods obtain decent performance and show certain robustness against noise.

In this paper, we present a randomized multi-circle detection algorithm based on a novel sampling process using geometrical constraints for searching valid circles. Our proposed method replaces the conventional three or four edge pixel sampling strategies with identification of isosceles triangles (ITs) using gradient information of the edge points. The geometrical properties of ITs have several advantages. First of all, ITs can be detected from any pair of edge points on a circle, resulting in high probability for finding true circles. Secondly, the geometrical constrains of ITs are capable of suppressing the false-positives from background noise and unrelated textures. Thirdly, the calculation of the ITs constraints has low computational complexity for the sampling process. In support of the special property of ITs features, the algorithm first collects pixels in connected components formed by the rule of connectivity of edges pixels, then applies a sampling process with clustering in parameter space using ITs features. Based on the collected ITs, circle candidates are calculated and refined with chords and a linear error compensation with inliers verified using gradients. In the end, each candidate circles are verified with a sector based validation process. We apply our method using synthetic images as well as real images to demonstrate its efficiency and robustness in the presence of noise.

2. Background

We briefly introduce the fundamental sampling strategies used in many state-of-the-art randomized circle detection algorithms, where two commonly used strategies are the three-points circle sampling [17,18] and the four-points circle sampling [19,20,24]. We start by introducing the basics sampling strategies and their properties.

2.1. Basic circle sampling strategy

The three-points circle sampling strategy randomly takes three edge pixels at each iteration to represent a circle. Suppose the coordinates of edges are (x_1,y_1), (x_2,y_2) and (x_3,y_3). The center a and b and the radius r can be obtained by [20],

$$a=\frac{\begin{vmatrix} x_2^2+y_2^2-(x_1^2+y_1^2) & 2(y_2-y_1) \\ x_3^2+y_3^2-(x_1^2+y_1^2) & 2(y_3-y_1) \end{vmatrix}}{4((x_2-x_1)(y_3-y_1)-(x_3-x_1)(y_2-y_1))} \tag{1}$$

$$b=\frac{\begin{vmatrix} 2(x_2-x_1) & x_2^2+y_2^2-(x_1^2+y_1^2) \\ 2(x_3-x_1) & x_3^2+y_3^2-(x_1^2+y_1^2) \end{vmatrix}}{4((x_2-x_1)(y_3-y_1)-(x_3-x_1)(y_2-y_1))} \tag{2}$$

$$r=\sqrt{(x_i-a)^2+(y_i-b)^2}, \text{ for any } i=1,2,3. \tag{3}$$

In practice, the calculation of $4((x_2-x_1)(y_3-y_1)-(x_3-x_1)(y_2-y_1))$ is evaluated before deriving a and b in order to avoid division by zero where three edge points are collinear. This is also the only way of rejecting false candidates in three-points sampling.

In four-points circle sampling, a fourth edge pixel (x_4, y_4) is selected each iteration. This edge pixel is used to estimate a distance that determines whether it is located close enough to a circle. The distance is calculated as follows,

$$d=\left|\sqrt{(x_4-a)^2+(y_4-b)^2}-r\right| \tag{4}$$

If the absolute value d satisfies a threshold T_r which defines the valid distance to form a circle, $d \le T_r$, a circle based on a, b and r can be chosen as a candidate. In practice, each edge points are separated with a minimum distance to avoid false positives under extreme cases during randomized sampling. The validation of this fourth points is reported to be able to discard 95% of bad sampled candidates, which lead to a huge boost in performance [21]. However, it is not able to validate true circles when the noise level is high and the edge map is filled with randomly distributed noises. These above mentioned sampling approaches are based on the assumption that a circle exists among sampled edge points without further checking of any other evidence. It is therefore susceptible to noise and can lead to false detection of circles.

2.2. Solutions to problems

To overcome the above mentioned shortcomings, additional information of each edge point is necessary. Properties regarding the quality and validity of sampled edge points are often used for checking the edge point candidates. Such improvements include gradient information, symmetry of circles and chords. On one hand, these geometrical properties help identify true circles in a more robust and accurate way. On the other hand, each of these geometrical properties has its own weakness: the gradient estimation suffers from poor accuracy in the presence of noise; using symmetric properties restrict the number of valid edge points especially when distortion or occlusion occurs in an image and therefore not every point on a circle can contribute to finding the circle using the symmetric property; Chords-based methods have