AN ADAPTIVE FITTING APPROACH FOR THE VISUAL DETECTION AND COUNTING OF SMALL CIRCULAR OBJECTS IN MANUFACTURING APPLICATIONS

Ying Gu, Liyuan Li, Fen Fang, Mark Rice, Jamie Ng, Wei Xiong, and Joo-Hwee Lim

Institute for Infocomm Research, A*STAR, Singapore
Email: {guy, lyli, fangf, mdrice, jamie, wxiong, joohwee}@i2r.a-star.edu.sg

ABSTRACT

Detecting, localizing and counting small circular objects in machine parts is an important task in many applications for manufacturing. Existing methods of circle detection face difficulties due to the high-curvature and limited edge points of circles. As a result, in this paper we propose a novel two-stage circle detection method, which integrates bottom-up coarse detection and top-down circle fitting. First, a circle detector combining low-level feature descriptors and a linear SVM is developed. This is used to scan an input image in a sliding window mode to detect small circles with coarse estimates of locations and scales. Next, a hierarchical Bayesian model performs a top-down adaptive circle fitting, with the ability to achieve a maximum a posteriori probability to fit circles to local image features. The evaluation of our approach with manufacturing images has demonstrated to be efficient in detecting small circles in machine parts.

Index Terms— Small circle detection, coarse detection, circle fitting, SVM, hierarchical Bayesian model

1. INTRODUCTION

The problem of detecting circular features in images is an established research topic in computer vision [1], and of importance to industrial applications, such as the automated inspection of manufactured products [2, 3, 4, 5]. For example, manufacturing companies may produce products containing circular objects that need to be counted and checked (*e.g.* for security, assembly, defect detection, etc.), by using related algorithms to recognize their location and precise scale. Methods for object counting can vary from manual processes, to the use of image processing to detect related objects [6, 7]. These may entail working at a high speed, or require a high degree of accuracy.

Existing circle detection algorithms can be clustered into three categories: (1) methods based on Circular Hough Transform (CHT) [8] and shape matching [9, 10]; (2) stochastic techniques such as Random Sample Consensus (RANSAC) [11], simulated annealing [12] and genetic algorithms [13], and (3) more recently cited methods using Line Segment Approximation (LSA) [14, 15].

Circular Hough Transform (CHT) is a popular approach for circle detection, extending from the well-known technique of Hough Transform (HT) [8], and variants such as Probabilistic Hough Transform [16], Randomized Hough Transform [17] and Fuzzy Hough Transform [18]. CHT accumulates votes from edge points in a parameter space, extracting significant peaks to represent a circle. However, this approach is known to face difficulties in detecting a large number of small circles in noisy images due to the limited number of edge points available.

To compensate for this, instead of depending on a time-consuming accumulation operation, it has been proposed to randomly pick four edge points (three to generate a hypothesis circle, and one for testing) [11]. This approach has proved to be more efficient than HT-based methods. However, it introduces unnecessary processing when applied to an image of small circles with a noisy background.

More recently, a new group of circle detection algorithms have been proposed [14, 15]. These methods apply a linear line segment approximation to circle detection with false detection control, making the implementation efficient and more accurate. However, while these approaches are able to obtain good results, in the absence of performing low-curvature arc segmentation along a circle perimeter, they often fail to detect small circles (see Section 3).

On the other hand, dominate deep learning approaches like Faster R-CNN have been identified to perform poorly for small object detection [19, 20]. This is due to the difficulty of region proposal networks to localize small objects accurately.

Inspired by the success of object detection based on local feature descriptors (*e.g.* HOG) and a powerful classifiers (*e.g.* SVM) [21], we propose a novel method by integrating bottom-up detection and top-down circle fitting to obtain the precise position and scale of small circles in an image. This consists of a coarse detector to detect small circles in a sliding window. The detector employs SIFT (Scale-Invariant Feature Transform) as a sketch pattern descriptor and POAG (Pyramid of Average Gray-levels) as a graph pattern descriptor. A linear SVM classifier is used to train for accurate detection.

This research is supported by the Agency for Science, Technology and Research, under the AME Programmatic Funding Scheme (Project#A18A2b0046).

By scanning an image with a sliding window, we are able to generate a rough estimate of the location and scale of embedded circles. Based on the detection box, a hierarchical Bayesian model (HBM) is then applied to perform top-down circle fitting for the precise location and scale of detected circles. As a result, we believe this iterative solution is efficient for detecting small circles in related images.

The remaining paper is organized as follows. In Section 2, we describe the proposed method for small circle detection. In Section 3, experimental results on real-world images from the manufacturing industry are presented. Finally, the conclusion is outlined in Section 4.

2. OUR APPROACH

The proposed approach consists of two stages. In the first stage, an efficient detector is applied to perform multi-scale scanning on an input image in a sliding window mode. In the second stage, a hierarchical Bayesian model is applied to fit a circle to local image features, to obtain the precise location and scale of small circular objects. The technical details are described in the following sections.

2.1. Bottom-up circle detection

Following the conventional scheme of vision-based object detection with a sliding window, our detector first extracts the local visual features and applies a machine learning classifier to detect the objects-of-interest.

We employ two low-level feature descriptors - *SIFT as a sketch feature descriptor*, and *POAG as a graph pattern descriptor*. The SIFT descriptor exploits gradients, captures the features of contours and edges around a circular object, while the POAG descriptor provides a supplementary feature representation to encode the global and local brightness patterns for small circular objects.

A pyramid structure is configured to extract local graph patterns, which is composed of five layers. In the n-th layer, the window is divided into $n \times n$ blocks. At the top layer, the feature is represented as the average brightness of the window (*i.e.* g). At the n-th layer, the feature is represented as a vector of n^2 elements, with each element value the difference of the average brightness of the block, and that of the window (*i.e.* $g_{ij}^n - g$). The features of each layer are concatenated to form a feature vector of 55 dimensions. Concatenating the SIFT and POAG descriptors forms a low-level feature representation of both sketch and graph patterns from the detection window. To be efficient for real-time detection, a linear SVM is trained on the low-level features for small circular object detection.

For an input image, multi-scale detection is performed in a sliding window mode, with the stride being 1/4 of the window scale at each scale level. To achieve real-time performance, the SIFT descriptor is computed from integral images of gradient orientations, and the POAG is computed from integral images of brightness. The detections are clustered with non-maximum suppression.

2.2. Top-down circle fitting

The bottom-up detection produces a set of detected boxes which provide rough estimates of locations and scales of potential small circular objects. To obtain precise and reliable detection, a hierarchical Bayesian model (HBM) is used to find a circle under the priors of location and scale according to the detected box.

A circle is expressed as the mathematical formulation:

$$\begin{cases} x_t = x_c + r\cos t \\ y_t = y_c + r\sin t \end{cases} \tag{1}$$

where $\mathbf{x}_t = (x_t, y_t)$ represents the points along the circle perimeter, $\mathbf{x}_c = (x_c, y_c)$ is the center point of the circle, and r is the radius. Fitting a circle in an image requires finding a sequence of edge points that satisfy the circle model (1).

From a detected box, we obtain the box center $\mathbf{x}_b$ and size s. We can use $\mathbf{x}_b$ and $r_s = \rho s$ to represent the initial estimated location and scale of a circle, which are employed as priors to anchor the search of a circle in a neighborhood. Applying this prior constraint on a circle fitting can be formulated as a maximum a posteriori problem:

$$\begin{aligned}(\mathbf{x}_c^*, r^*) &= \max_{(\mathbf{x}_c, r)} P(C(\mathbf{x}_c, r)|\mathbf{I}) \\ &= \max_{(\mathbf{x}_c, r)} P(\mathbf{I}|C(\mathbf{x}_c, r))P(\mathbf{x}_c|\mathbf{x}_b, \sigma_c)P(r|r_s, \sigma_r), \end{aligned} \tag{2}$$

where $C(\mathbf{x}_c, r)$ represents a circle centered at $\mathbf{x}_c$ with radius r, and the first term $P(\mathbf{I}|C(\mathbf{x}_c, r))$ is the likelihood of finding a circle centered at $\mathbf{x}_c$ with radius r in image $\mathbf{I}$. The second term $P(\mathbf{x}_c|\mathbf{x}_b, \sigma_c)$ is the prior of the circle centered at $\mathbf{x}_c$ given the initial estimated center location at $\mathbf{x}_b$ with potential spatial variation σ_c, and the last term $P(r|r_s, \sigma_r)$ is the prior of the radius r given the initial estimate r_s with potential variation σ_r. The problem formulation can be represented as a HBM, where hyper-parameters are $(\mathbf{x}_b, \sigma_c, r_s, \sigma_r)$, and the prior parameters are $(\mathbf{x}_c, r)$.

For the task of searching for a circle within a local neighborhood, the prior probabilities are defined as:

$$P(\mathbf{x}_c|\mathbf{x}_b, \sigma_c) = \exp(-\|\mathbf{x}_c - \mathbf{x}_b\|^2/(2\sigma_c^2)) \tag{3}$$

$$P(r|r_s, \sigma_r) = \exp(-(r - r_s)^2/(2\sigma_r^2)). \tag{4}$$

Here, we use Gaussian functions to control variations of the circle parameters (*i.e.*, $\mathbf{x}_c$ and r). The priors provide top-down constraints on fitting a circle to the local image features. The likelihood provides the bottom-up evidence of a circle appearing in a local image. By presuming that $\{\mathbf{x}_t\}_{(t=1)}^T$ are densely sampled points along a circle contour, the likelihood of a circle in an image given the center at $\mathbf{x}_c$ with radius r is defined as:

$$P(\mathbf{I}|C(\mathbf{x}_c, r)) = \frac{1}{T}\sum_{t=1}^{T} P(\mathbf{x}_t|C(\mathbf{x}_c, r)), \tag{5}$$

where $P(\mathbf{x}_t|C(\mathbf{x}_c,r))$ represent the probability of $\mathbf{x}_t$ being a point on the circle centered at $\mathbf{x}_c$ with radius r.

Three kinds of low-level local evidence are employed to evaluate the matching of a circle point and an edge pixel in an image: the significance of gradient magnitude, the consistency of orientations, and the least spatial displacement. Hence, the local evidence on a point $\mathbf{x}_t$ of a circle is defined as:

$$P(\mathbf{x}_t|C(\mathbf{x}_c,r)) = \max_{\mathbf{x}\in N(\mathbf{x}_t)} P_e(\mathbf{x})P_o(\mathbf{x},\mathbf{x}_t)P_d(\mathbf{x},\mathbf{x}_t), \quad (6)$$

where $N(\mathbf{x}_t)$ represents a small local neighborhood of $\mathbf{x}_t$. $P_e(\mathbf{x})$ with the probability of $\mathbf{x}$ being an edge point (which depends on the strength of the gradient magnitude). Using a sigmoid function, this is defined as:

$$P_e(\mathbf{x}) = [1+\exp(-\beta(\|\mathbf{g}_x\| - g_0))]^{-1}, \quad (7)$$

where $\mathbf{g}_x = (g_x, g_y)$ is the gradient vector at point $\mathbf{x}$, g_0 and β are the parameters of the sigmoid function, which are selected empirically ($\beta = 0.2$ and $g_0 = 15$). To filter out image noise, $P_e(\mathbf{x})$ is set to 0 when $\mathbf{g}_x < 8$. If $\mathbf{x}$ is a point on the edge of a circle in an image, the gradient orientation at a given point should be perpendicular to the tangent vector at the closest point $\mathbf{x}_t$ on the circle. Let $\mathbf{g}_t = (g_{tx}, g_{ty})$ be the tangent vector at point $\mathbf{x}_t$. The probability of point $\mathbf{x}$ being on the edge orientation of a circle is given as:

$$P_o(\mathbf{x},\mathbf{x}_t) = (1-|\cos\theta|)^2 = (1-|\mathbf{g}_x\mathbf{g}_t|/(\|\mathbf{g}_x\|\|\mathbf{g}_t\|))^2. \quad (8)$$

To weight the local evidence from the pixel closer to $\mathbf{x}_t$, the distance to the radiation line from the circle center $\mathbf{x}_c$ to $\mathbf{x}_t$ is evaluated. A straight line connecting the circle center $\mathbf{x}_c$ and $\mathbf{x}_t$ can be described as $ax+by+c=0$ according to the two points $\mathbf{x}_c$ and $\mathbf{x}_t$, so that the distance from $\mathbf{x}$ to the line can be computed as $d_p = |ax+by+c|/\sqrt{(a^2+b^2)}$. Now, the probability of the spatial displacement of the local supporting can be defined as:

$$P_d(\mathbf{x},\mathbf{x}_t) = (e^{-d_p^2/\sigma_d^2})(e^{-\|\mathbf{x}-\mathbf{x}_t\|^2/(2\sigma_s^2)}), \quad (9)$$

which is a combination of 1D Gaussian and 2D Gaussian functions.

Let $\partial P(C(\mathbf{x}_c,r)|\mathbf{I})/(\partial\mathbf{x}_c) = 0$ and $\partial P(C(\mathbf{x}_c,r)|\mathbf{I})/\partial r = 0$. Combining the priors and likelihood functions, we can obtain an iterative solution for the circle model $\mathbf{x}_c$ and r as:

$$\mathbf{x}_c^{n+1} = \frac{\frac{1}{\sigma_s^2}\frac{1}{P_I^n}\frac{1}{T}\sum_{t=1}^{T} P_t^n(\mathbf{x}_t' - \Delta\mathbf{x}_t)}{\frac{1}{\sigma_s^2}+\frac{1}{\sigma_c^2}} + \frac{\frac{1}{\sigma_c^2}\mathbf{x}_b}{\frac{1}{\sigma_s^2}+\frac{1}{\sigma_c^2}} \quad (10)$$

$$r^{n+1} = \frac{\frac{1}{\sigma_s^2}\frac{1}{P_I^n}\frac{1}{T}\sum_{t=1}^{T} P_t^n((x_t'-x_c^n)\cos t + (y_t'-y_c^n)\sin t)}{\frac{1}{\sigma_s^2}+\frac{1}{\sigma_r^2}} + \frac{\frac{1}{\sigma_r^2}r_s}{\frac{1}{\sigma_s^2}+\frac{1}{\sigma_r^2}} \quad (11)$$

where $P_I^n = P(\mathbf{I}|C(\mathbf{x}_c^n, r^n))$ as (5), $P_t^n = P(\mathbf{x}_t|C(\mathbf{x}_c^n, r^n))$ as (6), $\mathbf{x}_t' = (x_t', y_t') \in N(\mathbf{x}_t)$ is selected according to (6), $\Delta\mathbf{x}_t = (r\cos t, r\sin t)$ from (1), and the center of the detection box $\mathbf{x}_b$.

In most cases, the procedure is terminated in a few iterative steps. We target at small circular objects, where the number of points along a circle perimeter are limited. Therefore, the iterative computing for top-down circle fitting is very efficient.

Finally, the HBM-based top-down circle fitting produces a precise description of the circle (*i.e.* $\mathbf{x}$ and r), as well as the posterior probability for matching the circle model to the local image features (*i.e.*, $P(C(\mathbf{x}_c,r)|\mathbf{I})$). If the posterior probability is less than a preset threshold, the detection is rejected. The remaining detections are then clustered into groups according to their size $2r$, from which we obtain the number of circular objects and their categories in an image.

3. EVALUATION

We evaluated our method on real-world images (756×1008 pixels) of machine parts provided by a local manufacturing company. Specifically, 96 images of steel breadboards for buildings and industrial electronic circuits were used in the experiment, containing over 5,000 circles. Most of the images contained small circles with a diameter of 20 to 40 pixels, and a few images contained relatively large diameters of 40 to 60 pixels. The images were captured with variations in viewing angle and lighting conditions, and some images contained scratches on the metallic breadboard surface.

For performance comparisons, we compared our method with a recent version of the Circular Hough Transform (CHT) method from OpenCV 4.0.0, released in 2018 [22], and the Line Segment Approximation (LSA) method [15], with the source code available at [23]. All testing was completed on a PC with an Intel 3.2 GHz dual processor, and 4GB of RAM.

For each circle, the ground truth was represented as a center point and a radius. For each detection, we obtained the center and radius of each circle. We then computed the ratio between the intersection and the union of two circle areas. If the ratio was larger than 0.5, it was accepted as a True Positive (TP). Otherwise, it was assigned as a False Positive (FP). Ground truth circles with no overlapping detection were marked as False Negatives (FN). The precision (P), recall (R), and F1-score were computed as $F_1 = 2\frac{P\cdot R}{P+R}$, $P = \frac{TP}{TP+FP}$, $R = \frac{TP}{TP+FN}$. The average performance time was computed per image.

A few examples of the image results are shown in Figure 1. Overall, our method outperformed both the CHT and LSA methods, especially for small circle detection (see Table 1).

In particular, the CHT method performed poorly when the parameters of the circle scale were not preset. This method generated many large false circles, due to the votes from mul-

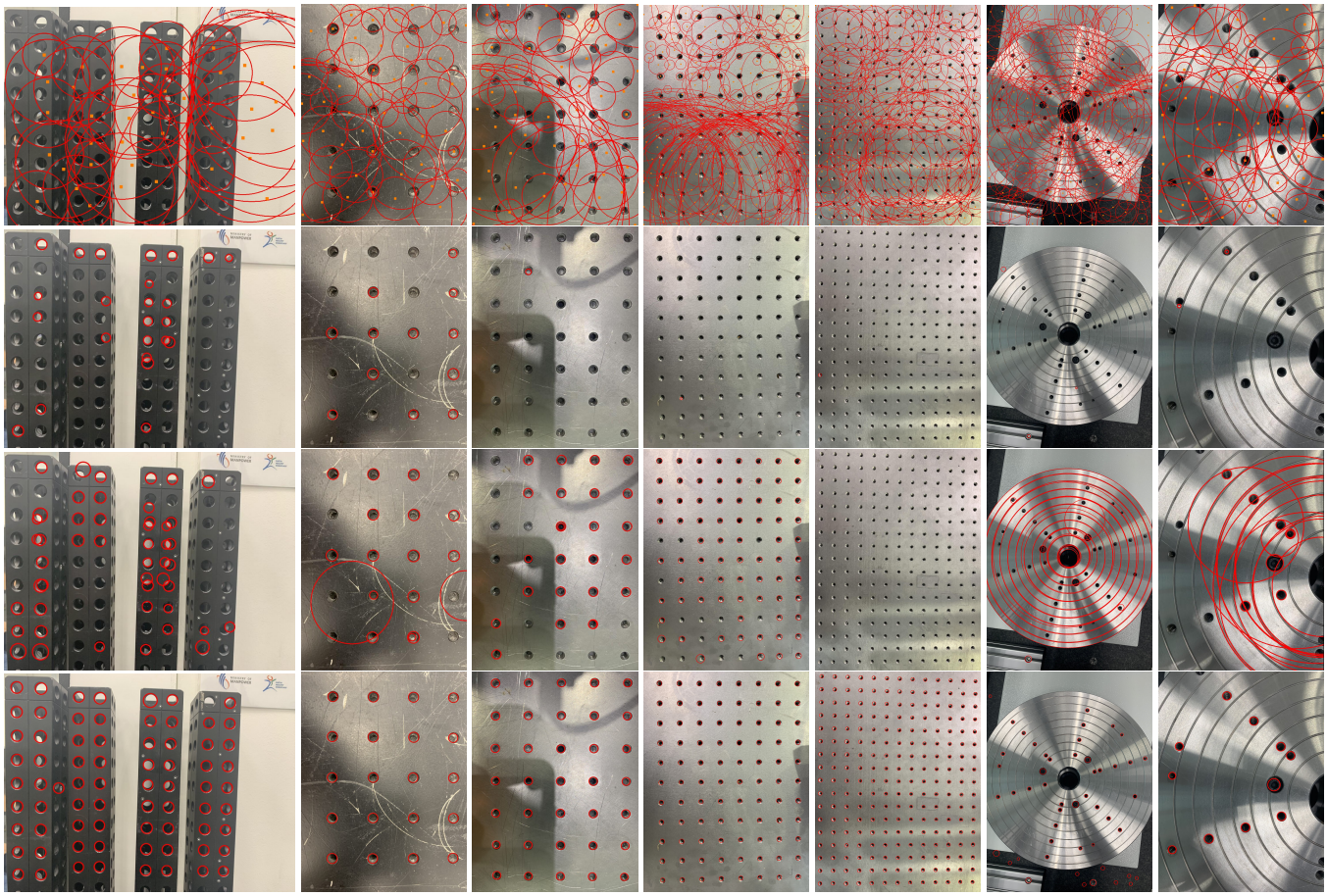

Fig. 1. Comparison of CHT without setting parameters (first row), CHT with setting parameters (second row), LSA (third row) and our method (fourth row) on sample images.

tiple objects being much larger than those from a single small circle. CHT also performed poorly in detecting small circles. Note, that we only show the evaluation results of CHT *with* setting parameters, given the unconstrained results for *without* setting parameters (see Figure 1).

Alternatively, the LSA method performed better than CHT, especially for circles with diameters over 40 pixels. Prior work has shown that LSA can get satisfactory results from industrial PCB images [15, 23], which we identified when circles are larger than around 70 pixels in diameter. In comparison, our results indicate that LSA missed a lot of circles smaller than 30 pixels in diameter. This we believe was due to the circles being too small for the line segmentation to detect.

Based on these results, we estimate that for images containing circular objects that are not too close to an image boundary, our method holds promise in detection accuracy. Moreover, we observe that our algorithm can obtain faster and more accurate results than the tested CHT and LSA methods for small circle detection.

Table 1. Method performance comparisons.

Method	Precision	Recall	F1 score	Average time (sec)
CHT	0.91	0.23	0.37	8.03
LSA	0.96	0.67	0.79	3.12
Our	0.97	0.94	0.95	0.84

4. CONCLUSION

In this paper, we propose a novel method for circular object detection, which combines bottom-up coarse detection and top-down circle fitting. We designed a SVM-based approach for circle detection to obtain coarse estimates of the location and scale of small circles. A model-driven method based on a hierarchical Bayesian model is used to get precise positions and scales of small circles. Evaluation on manufacturing images demonstrate the advantage of the proposed method on circle detection, localization and counting. For future work, we aim to test our algorithm on larger datasets, with the intention of commercial deployment.

5. REFERENCES

[1] E.R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*, Academic Press, 1990.

[2] V. Allada and S. Anand, "Efficient vertex detection algorithms using the hough transform," *International Journal of Advanced Manufacturing Technology*, vol. 11, no. 6, 1996.

[3] D. M. Tsai, "Locating overlapping industrial parts for robotic assembly," *International Journal of Advanced Manufacturing Technology*, vol. 12, no. 4, 1996.

[4] S. J. Huang and C. C. Lin, "A three-dimensional non-contact measurement system," *International Journal of Advanced Manufacturing Technology*, vol. 13, no. 6, 1997.

[5] F.C. Luciano and M.C.J. Roberto, *Shape Classification and Analysis: Theory and Practice*, CRC Press, 2001.

[6] M. F. Ghazali, L.K. Wong, and J. See, "Automatic detection and counting of circular and rectangular steel bars," *9th International Conference on Robotic, Vision, Signal Processing and Power Applications*, 2016.

[7] M. Baygin, M. Karakose, A. Sarimaden, and E. Akin, "An image processing based object counting approach for machine vision application," *ICAIE*, 2017.

[8] H. Yuen, J. Princen, J. Illingworth, and J. Kittler, "Comparative study of hough transform methods for circle finding," *Image Vision Comput.*, vol. 8, no. 1, 1990.

[9] J. Iivarinen, M. Peura, J. Sarela, and A. Visa, "Comparison of combined shape descriptors for irregular objects," *Proc. 8th British Machine Vision Conf.*, 1997.

[10] G. Jones, J. Princen, J. Illingworth, and J. Kittler, "Robust estimation of shape parameters," *Proc.British Machine Vision Conf.*, 1990.

[11] M. Fischer and R. Bolles, "Random sample consensus: A paradigm to model fitting with applications to image analysis and automated cartography," *CACM*, vol. 24, no. 6, 1981.

[12] G G. Bongiovanni and P. Crescenzi, "Parallel simulated annealing for shape detection," *Computer Vision and Image Understanding*, vol. 61, no. 1, 1995.

[13] G. Roth and M.D. Levine, "Geometric primitive extraction using a genetic algorithm," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, no. 9, 1994.

[14] Truc Le and Ye Duan, "Circle detection on images by line segment and circle completeness," *IEEE International Conference on Image Processing*, 2016.

[15] C.S. Lu, S.Y. Xia, W.M. Huang, M. Shao, and Y. Fu, "Circle detection by arc-support line segments," *IEEE International Conference on Image Processing*, 2017.

[16] D. Shaked, O. Yaron, and N. Kiryati, "Deriving stopping rules for the probabilistic hough transform by sequential analysis," *Comput. Vision Image Understanding*, vol. 63, 1996.

[17] L. Xu, E. Oja, and P. Kultanen, "A new curve detection method: Randomized hough transform(rht)," *Pattern Recognition Lett.*, vol. 11, no. 5, 1990.

[18] J.H. Han, L.T. Koczy, and T. Poston, "Fuzzy hough transform," *Proc. 2nd Int. Conf. on Fuzzy Systems*, 1993.

[19] C. Eggert, S. Brehm, A. Winschel, D. Zecha, and R. Lienhart, "A closer look: Small object detection in faster r-cnn," *ICME*, 2017.

[20] B. Singh and L. Davis, "An analysis of scale invariance in object detection snip," *CVPR*, 2018.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *CVPR*, 2005.

[22] "Hough circle transform," `https://docs.opencv.org/4.0.0/d4/d70/tutorial_hough_circle.html/`, 2018.

[23] C.S. Lu, S.Y. Xia, W.M. Huang, M. Shao, and Y. Fu, "Circle detection by arc-support line segments," `https://github.com/AlanLuSun/Circle-detection/`, 2017.