Fast Method for Rectangle Detection

Cheng Wang

Software engineering department, Hainan College of Software Technology, Hainan 571400, China

naturavo@163.com

Keywords: Digital image, computer vision, rectangle detection, geometrical center

Abstract: In this paper, we introduce a fast method for rectangle detection. For any object has closed curve in a digital image, it is easy to calculate the geometrical center of the object. Hence conveniently find its width and length, area, diagonal line length, etc. If the closed curve is really a rectangle, these values must satisfy specific equations, and these equations can be used to detect rectangle. Compared with other detecting methods, our method only need to process border points of the object, hence has higher detection speed.

1. Introduction

Object recognition is an important research aspect in pattern recognition field, and rectangle object is a common geometric shape for recognition of two dimensional shapes. Rectangle detection has a wide range of applications, such as car label recognition, two-dimensional bar code detection, etc.

A common method is using Hough transform to detect rectangle [1, 2]. Hough transform is an effective method to detect geometric objects in digital image, even in the presence of noise and missing data. However, highly computational cost is the drawback of this method. Other works on rectangle detection lies in various regions with specific usage. Jung, C.R. [3] proposed a rectangle detection method based on windowed Hough transform. In this method, every pixel of the image is scanned, and a sliding window is used to compute the Hough Transform of small regions of the image. Peaks of the Hough image (which correspond to line segments) are then extracted, and a rectangle is detected when four extracted peaks satisfy certain geometric conditions. J. Bazin [4] proposed a rectangle detection method in catadioptric images which is adopt in robotic systems equipped with catadioptric cameras. Zhang Cong-peng and Wei Xue-guang [5] proposed rectangle method detection based on Harris corner, but Harris corner is also highly computational cost. In this paper, we proposed a rectangle detection method based on rectangle properties, this method only need calculate border points of an object to determine whether an object is really a rectangle, hence faster than other methods.

2. The geometrical properties of rectangle

For a rectangle boundary, it is easy to calculate its centroid, and its centroid must be its geometrical center. If a rectangle has length l and width w, the minimal distance of the centroid to boundary is w/2, as in Fig. 1. If point $\mathrm{o}(\mathrm{x}_0, \mathrm{y}_0)$ is the calculated geometrical center, point $\mathrm{a}(\mathrm{x}_\mathrm{a}, \mathrm{y}_\mathrm{a})$ is the point which has the minimal distance to center point o, then line oa has slope $k_a = \frac{\mathrm{y}_\mathrm{a}-\mathrm{y}_0}{\mathrm{x}_\mathrm{a}-\mathrm{x}_0}$. If line ob is the line which perpendicular to line oa, point $\mathrm{b}(\mathrm{x}_\mathrm{b}, \mathrm{y}_\mathrm{b})$ is the crossing point of line ob and the border, let line ob's slope $k_b = \frac{\mathrm{y}_\mathrm{b}-\mathrm{y}_0}{\mathrm{x}_\mathrm{b}-\mathrm{x}_0}$, we have equation:

$$k_a * k_b = -1, \text{ or } k_a * k_b + 1 = 0 \qquad (1)$$

If the length of oa is d_a and the length of ob is d_b, then the length of oD must be $\sqrt{\mathrm{d}_\mathrm{a}^2 + \mathrm{d}_\mathrm{b}^2}$, point D must be one of the points on the border which have the maximal distance to the centroid of this rectangle. If the diagonal line length of the rectangle is $2d_m$, we have equation:

$$d_m = \sqrt{d_a^2 + d_b^2} \qquad (2)$$

For any closed curve, its enclosed area is easy to calculate. If this area is S and this object is a rectangle, then we have equation:

$$\mathrm{S} = 4 * \mathrm{d_a} * \mathrm{d_b} \qquad (3)$$

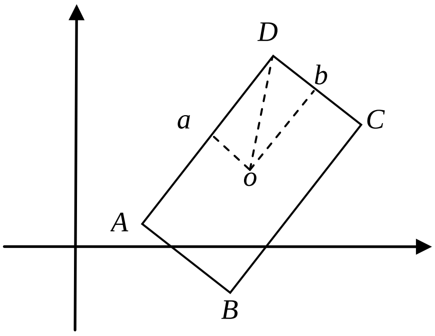


Fig. 1 geometrical properties of rectangle

3. Detection algorithm

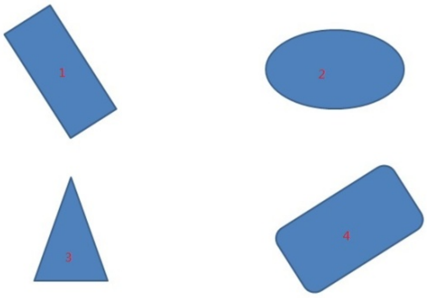
To detect rectangle, we can assume each closed curve is a rectangle, and then compute its centroid, find its minimal distance to its centroid and corresponding border point a, calculate the length of oa, ob, oD and area S. For perfect condition, if these values satisfy equation (2) and equation (3), then this closed curve must be a rectangle, otherwise, it is not. Considering the zigzag effect of the rectangle border in digital image, we can calculate the ratio $\mathrm{r_s} = \mathrm{abs}((\mathrm{S} - \mathrm{d_a} * \mathrm{d_b})/S)$ and the ratio $\mathrm{r_l} = \mathrm{abs}((\mathrm{d_m} - \sqrt{\mathrm{d_a^2} + \mathrm{d_b^2}})/\mathrm{d_m})$, if the ratio $\mathrm{r_s}$ and $\mathrm{r_l}$ below certain thresholds, we believe the assumed rectangle is exactly a rectangle, otherwise, it is not.

So, the rectangle detection algorithm can be described as follow:

1) Image preprocessing. Denoise, binarize, transform the image to a binary image, and extract the borders of the binary image.
2) Assume each object that has closed curve is a rectangle, calculate and remember coordinates of the centroid point *o* of each object.
3) For each closed curve, calculate the distance of each border pixel to the centroid; find the minimal distance d_a and the maximal distance d_m, remember the coordinates of the point *a* which has the minimal distance, and the point *D* which has the maximal distance.
4) For each border point *p*, calculate the slope $\frac{y_p - y_0}{x_p - x_0}$ of line *op*, and calculate the value of $\mathrm{abs}(\frac{y_p - y_0}{x_p - x_0} * \frac{y_a - y_0}{x_a - x_0} + 1)$, find the minimal value, remember corresponding border point *b*, and the distance d_b of point *b* and point *o*.
5) Calculate the enclosed area *S* of the boundary.
   Let $r_s = abs((S - d_a * d_b)/S)$, if r_s larger than predetermined value, the boundary is not a rectangle.
6) Let $r_l = abs((d_m - \sqrt{d_a^2 + d_b^2})/d_m)$, if r_l smaller than predetermined value, this boundary is a rectangle, otherwise, it is not.

4. The experimental results and analysis

Using the algorithm describe in section 3, we can quickly determine whether an object is a rectangle. Fig. 2 is an image drawn by MSpaint, table 1 listed the MATLAB detection data for its four objects. Fig. 3 is a practical image, and table 2 listed the MATLAB detection data of the rectangle object (the white paper) in the image.



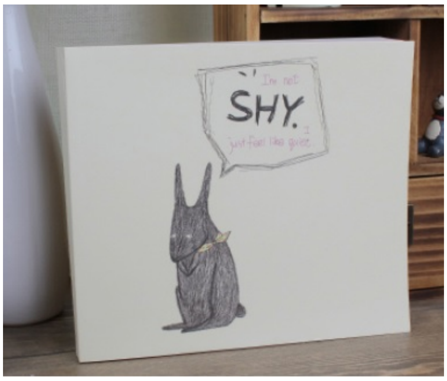


Fig. 2 Hand drawn image   Fig. 3 Practical image

Table 1 Detection data of Fig. 2

object	d_a	d_b	S	d_m	r_s	r_l
1	34.05	77.58	10895.6	84.73	0.0302	0.00014
2	49.2	86.78	13779.4	87.23	0.2394	0.14364
3	27.43	50.32	6284.4	81.76	0.1216	0.29906
4	46.38	86.49	16255.5	93.61	0.0139	0.04844

Table 2 Detection data of rectangle in Fig. 3

object	d_a	d_b	S	d_m	r_s	r_l
paper	124.61	150.08	77989	200.46	0.0408	0.0269

As the data of the hand drawn image and the practical image shown, for hand drawn image which has low noise, we can take small threshold to detect rectangle. In Fig. 2, with r_s threshold value 0.04 and r_l threshold value 0.001, the first object can be detected correctly; for the round corner rectangle object 3, with r_s threshold value 0.02 and r_l threshold value 0.05, it can be detected as a rectangle, if we don't want to treat object 3 as rectangle, just set r_l threshold smaller than 0.01. In Fig. 3, set r_s threshold value larger than 0.04 and r_l threshold value larger than 0.03, the paper object in the image can be detected correctly. In practical image, detection efficiency will also depend on the preprocessing algorithm. We tested with 3000 rectangles randomly generated by OpenCV function cvRectangle, our algorithm can detect all these rectangles correctly with r_s threshold value 0.05 and r_l threshold value 0.002.

5. Summary

For any object has closed curve in an image, it is easy to calculate the centroid of the object. If the object is a rectangle, we can find its geometrical center, and then calculated the minimal distance of the border point to the center, this distance must be half of the rectangle's width *w*. For the line of the minimal border point to centroid, its perpendicular line must be half of the rectangle's length *l*, and the maximal distance of border point to centroid must equals to $\sqrt{w/2^2 + l/2^2}$, and the rectangle's area must equals to *l*w*. For rectangle detection, we only need calculate these values of each closed curve. If the number of pixels of the boundary of a closed curve is n, then the time complexity of our algorithm is $O(n^2)$. The detection algorithm is sensitive to noise.

References

[1]. Illingworth, J, Kittler J. A survey of the Hough transform. Computer Vision, Graphics, &Image Processing，Vol.44 (1988) No. 1, p. 87-116

[2]. Yuanxin Zhu, Carragher, B., Potter, C.S. Fast detection of generic biological particles in cryo-EM images through efficient Hough transforms. 2002 IEEE International Symposium on Biomedical Imaging (Cat. No.02EX608). USA, July 2002, p 231-236

[3]. Jung, C.R., Schramm, R., Rectangle detection based on a windowed Hough transform. 17th Brazilian Symposium on Computer Graphics and Image Processing. Brazil, 2004, p. 113-120.

[4]. Bazin Jean-C., Inso Kweon; Demonceaux, C. et al. Rectangle extraction in catadioptric images. 2007 11th IEEE International Conference on Computer Vision. Oct. 2007, p. 2563-2569.

[5]. Zhang Cong-peng, Wei Xue-guang. Rectangle detection based on Harris corner. Optics and Precision Engineering. Vol. 22(2014) No. 8, p. 2259-2266