Corner Detection Algorithm Based on Edge Contour in Automatic Loading Positioning

Jiwu Wang
School of Mechanical and Electronic Engineering, Beijing Jiaotong University
Beijing, Haidian District, China

Junwei Fu
School of Mechanical and Electronic Engineering, Beijing Jiaotong University
Beijing, Haidian District, China
E-mail: jwwang@bjtu.edu.cn, 20125999@bjtu.edu.cn
www.bjtu.edu.cn

Abstract

Corner detection is a basic task in the field of image processing and computer vision. Aiming at the problem that the existing corner detection algorithms based on curvature scale space (CSS) can not obtain the required corners accurately, a corner detection algorithm based on angle is proposed. Firstly, the appropriate threshold is set by CSS algorithm to extract the candidate corners, and then the angle value of candidate corners is calculated at long-distance pixels, and the threshold is used to filter to obtain the final required corners. Experimental results show that the algorithm can accurately extract the required corners.

Keywords: Corner detection; CSS; IPAN; Angle calculation; Corner positioning.

1. Introduction

As a part of the supply chain, logistics takes warehousing as the center to promote production and keep pace with each other; Logistics is to meet the needs of customers and realize the process of relevant goods from origin to consumption at the lowest cost through transportation, storage and distribution. As an important link in the logistics process, the efficiency and automation of loading and unloading process greatly affect the production cost in the transportation process. Therefore, the realization of full automation in the process of loading and unloading is of great significance to improve the efficiency of logistics transportation and reduce the transportation cost. In the process of loading, the accuracy of vehicle parking pose measurement is a very important link to realize automatic production. Corner detection is an indispensable step in the process of vehicle pose measurement through visual technology. Corner is an important feature in two-dimensional image. Corner recognition provides important information for many applications in vision based fields, such as target detection, motion detection , scene analysis. They point out the existence and location of objects, so as to narrow the scope of searching objects and make it easier to interpret images. Therefore, the corner detection is also particularly important. In the process of vehicle loading and positioning, the accuracy of corner detection is an important problem that must be solved, which has a vital impact on the calculation of vehicle posture.

2. Corner detection algorithm

Corner detection is a basic subject in the field of machine vision and computer vision. At present, there is no precise mathematical definition of corner. The following three kinds of points are usually called corner points: the first is the intersection of more than two edges, the second is the point where the brightness change in all directions on the image is large enough, and the third is the point of maximum curvature on the edge curve . The existing corner detection algorithms are mainly divided into three categories: model-based method, gray-scale method and contour based method. The corner detection algorithm based on contour has the advantage of small positioning error. Before corner detection of the image, edge detection is carried out first, and then corners are detected on the detected contour rather than on the whole image. By identifying the significant points on the contour curve (generally selected as the maximum curvature points). The key of contour based corner detection algorithm is to construct an efficient corner response function.

Mokhtarian et al. proposed a corner detection algorithm based on CSS, which is also a landmark algorithm in this field. This algorithm has good robustness, and introduced a tracking method in the detection process to ensure that corner detection is not affected by noise. Subsequently, more scholars proposed corner detection algorithm based on CSS, and Zhong et al. proposed DCSs algorithm, which reduces the amount of calculation and improves the efficiency of the algorithm. Zhang [6] and others proposed a robust multi-scale curvature product detection algorithm by analyzing the curvature behavior of multi-scale space. The corner detection algorithm based on CSS technology usually faces two problems: ① the algorithm is sensitive to the local change of curve and noise, which may lead to poor detection effect. ② Select appropriate Gaussian smoothing parameters. In the process of practical application, there are more problems. Although the concept of angle seems to be very clear literally, there is no generally accepted mathematical definition at present.

2.1. *CSS algorithm*

CSS (curvature scale space) corner detection algorithm is one of the most classic corner detection algorithms based on contour. CSS technology is suitable for extracting a continuous scale of curvature features from the input contour. In the detection process, Canny edge detection algorithm is used to extract the edge contour. Then parameterize the contour curve C:

$$C(u) = (x(u), y(u))$$

An evolved version Cσ of C can then be computed. Cσ is defined by:

$$C_\sigma = (X(u,\sigma), Y(u,\sigma))$$

where

$$X(u,\sigma) = x(u) \otimes g(u,\sigma)$$

$$Y(u,\sigma) = y(u) \otimes g(u,\sigma)$$

Then calculate the curvature through the curvature formula, which is defined as:

$$K(u,\sigma) = \frac{X_u(u,\sigma)Y_{uu}(u,\sigma) - X_{uu}(u,\sigma)Y_u(u,\sigma)}{[X_u(u,\sigma)^2 + Y_u(u,\sigma)^2]^{1.5}}$$

where

$$X_u(u,\sigma) = x(u) \otimes g_u(u,\sigma)$$

$$X_{uu}(u,\sigma) = x(u) \otimes g_{uu}(u,\sigma)$$

$$Y_u(u,\sigma) = y(u) \otimes g_u(u,\sigma)$$

$$Y_{uu}(u,\sigma) = y(u) \otimes g_{uu}(u,\sigma)$$

$\otimes$ is the convolution operator and $g(u,\sigma)$ denotes a Gaussian of width σ .

2.2 . IPAN algorithm

IPAN (image and pattern analysis group) is a very famous corner detection algorithm, which is based on the physical angle of points on the curve. The algorithm draws a triangle on the curve, in which one point is a candidate corner and the other two points are points with the same distance from the candidate corner. As shown in Figure 2.1, {P1, P2, P3... PN-1, PN} is a group of continuous points on the curve. The algorithm determines the final corner in two steps. Firstly, the triangle angle formula is used to calculate the angle α Compared with the threshold value, PI in the figure is used as the candidate point, pi-k and PI + k are used as the other two points of the triangle. If the triangle meets the following formula, this point is used as the candidate corner point.

$$d_{\min}^2 \le |P_i - P_{i+k}| \le d_{\max}^2$$

$$d_{\min}^2 \le |P_i - P_{i-k}| \le d_{\max}^2$$

$$\alpha \le \alpha_{\max}$$

where, $|P_i - P_{i+k}| = |a| = a$ and $|P_i - P_{i+k}| = |b| = b$ are the distance P_i to P_{i+k} an P_i to P_{i-k} ,the angle can be calculated by cosine formula α,Although the default value of $d_{\min}^2$, $d_{\max}^2$ and is given $\alpha_{\max}$ in this way, the value of the equally important K is not given. Similarly, only one K value can not well estimate the angle of Pi point.

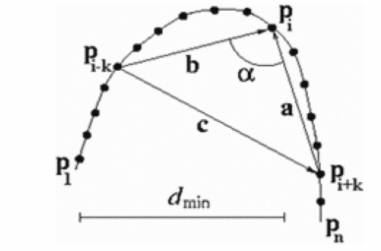

Fig2.1 IPAN algorithm

3. Paper Algorithm

The algorithm steps are as follows:

Step1:Use Canny operator to detect the edge of the image, and use Gaussian kernel function to smooth the curve.

Step2: Extract the contour of the edge detected in step 1, fill the contour gap, and find the T-shaped corner of the contour

Step3:Calculate the curvature value of edge points on a large scale, filter false corners and false corners with appropriate threshold, and get all candidate corners.

Step4:Improve the positioning by tracking corners in a small scale to obtain the final corners under the CSS algorithm

Step5:The final corner extracted by CSS algorithm is taken as the candidate corner, and the Ipan algorithm is used to take a larger value for the threshold K, so as to eliminate the corner generated by local noise to obtain the final corner.

3.1. *Canny edge detector*

This method uses Canny edge detector to extract image edges.Canny proposed three criteria for the evaluation of edge detection operators:

1. High detection rate: Edge operators should only respond to edges. The detection operator does not miss any edges and should not mark non-edges as edges.
2. Position accurately: The distance between the detected edge and the actual edge should be as small as possible.
3. Clear response: only one response for each edge, only one point.

3.2. *Obtaining candidate corners*

After using the Canny algorithm to extract edge contours, gaps may occur at some contiguous edges, and the CSS method may not be able to find T-shaped corners due to contour gaps (Figure 3.1). When the edge extraction method reaches the end of the contour, it performs two checks:

1. If the endpoint is nearly connected to another endpoint, fill the gap and continue the extraction.
2. If the endpoint is nearly connected to an edge contour. but not to another endpoint, mark this point as a T-junction corner.

In the process of extracting corners with CCS algorithm, these corners are defined as the local maximum of the absolute value of curvature. On a very fine scale, there are many such maxima due to the noise on the digital contour. With the increase of scale, the noise is smoothed, leaving only the maximum value corresponding to the real edge angle. CSS corner detection method finds corners at these local maxima. With the change of the edge contour after smoothing, the actual position of the corner will also change. If the detection is implemented in a large range, there will be a great deviation in the positioning of diagonal points. CSS algorithm overcomes this problem by tracking corners to ensure that corner detection is not affected by noise. Then take the smaller one σ value and check the same corner at the low scale. Therefore, the position at the corner may be updated until the scale is very low and the operation is very limited. This improves the positioning effect and the calculation cost is also very low, because it is lower than σ_{high} The curvature value of high scale does not need to be calculated at each contour point, but only in a small area near the detected corner.

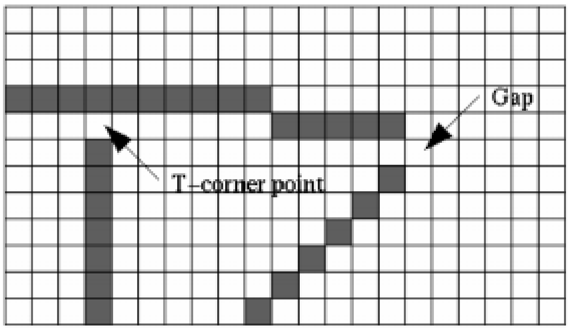

Fig 3.1 The two cases of gaps in the edge contours.

3.3. *Final corner determination*

As shown in Figure 3.2, among the candidate corners obtained by the CSS corner detection algorithm, because the algorithm is sensitive to local changes and noise on edge curves, and because of environmental factors, it is impossible to extract only the edge contours that we need according to the threshold setting during edge extraction. So only a few of the candidate corners extracted by the CSS algorithm are what we need. Analysis of a large number of images shows that the required corner points are concentrated on a large contiguous edge curve. Based on the inspiration of IPAN algorithm, a larger threshold K is selected to remove some corner points and get the final required corner points.

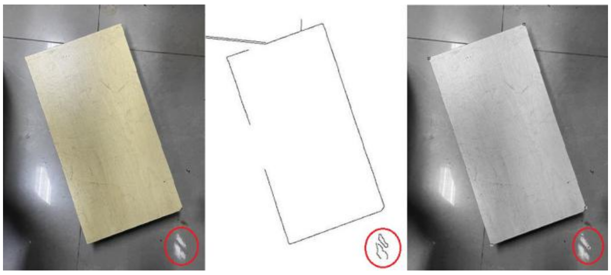

Fig 3.2 Image of CSS algorithm corner detection

4. Experimental Results

This algorithm will detect different images, and compare the effect of CSS algorithm, so as to intuitively understand the function of the algorithm.

(First column left is original picture,Fist column right is edge picture;Second column left is CSS,Second column right is the algorithm results in this paper)

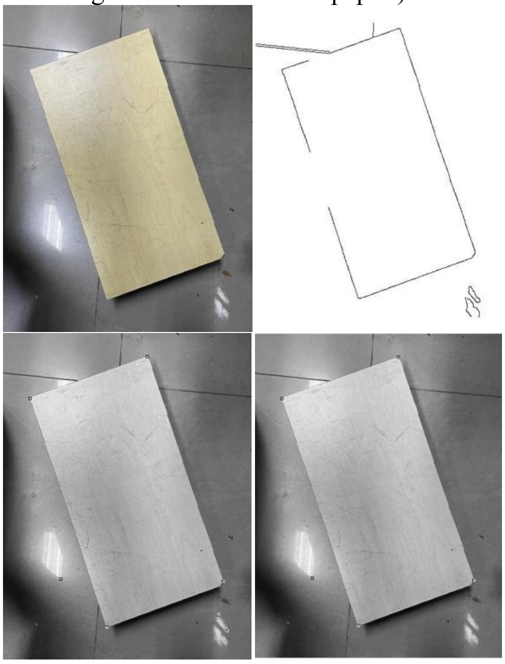

Fig 4.1

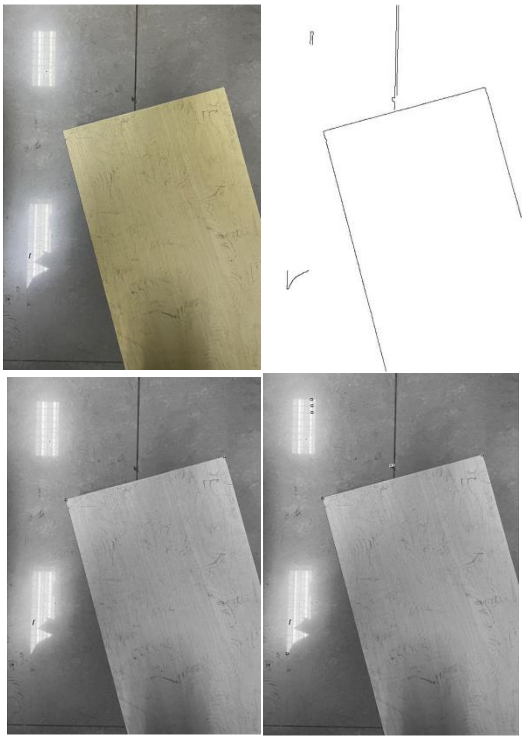

Fig 4.2

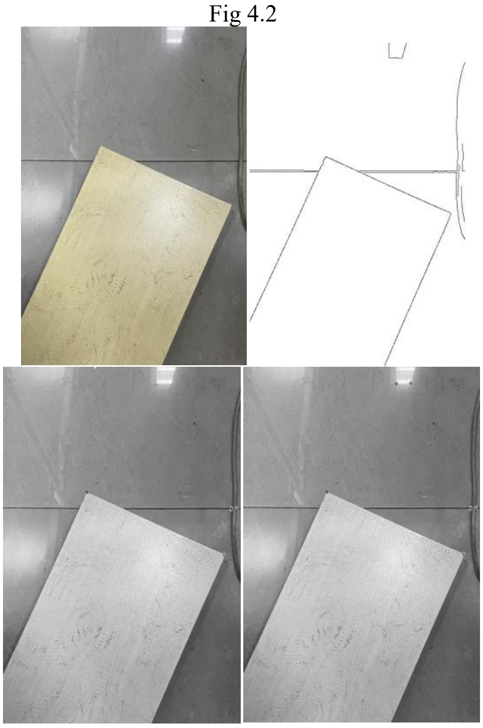

Fig 4.3

5. Conclusion

This paper introduces an angle-based corner detection algorithm. Firstly, the candidate corner points are obtained by using the CSS corner detection algorithm, and the corner points generated by the image edge are proposed to ensure that the required corner points are in the set of candidate corner points and minimize the unnecessary corner points. Then, the angle of each candidate corner point is measured at a high threshold through the angle threshold. Corners that do not meet the requirements are eliminated. The experimental results show that the algorithm can extract the desired corner points accurately. However, it does not have good reliability in complex background. Further research on more reliable corner detection algorithm in complex background and its application in practice will be carried out in the future.

Acknowledgements

This work has been supported by my mentor and thanks for his help in the research process.

References

1. Lebeda K, Hadfield S, Matas J, et al. " Texture-independent long-term tracking using virtual corners," IEEE Transactions on Image Processing,359-371, Vol25, No 1,2016.
2. Peng Lu, Yu Liang; Shuwei Chen, "Abnormal Behavior Detection of Video Groups Based on Corner Kinetic Energy"[J], Journal of Zhenzhou University(Engineering Edition). Vol.36, No 3,20-24, 2015.
3. Farzin Mokhtarian, Riku Suomela, " Roubust Image Corner Detection Through Curvature Scale Space, "IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 20,No 12,1998.
4. D. Chetverikov. A simple and efficient algorithm for detection of high curvature points in planar curves.2756:746-753,2003.
5. Rafi Md. Najmus Sadat, Zinat Sayeeda, et al."A Corner Detection Method Using Angle Accumlation," ICCIT2011,22-24.

Authors Introduction

Dr. Jiwu Wang

He is an associate professor, Beijing Jiaotong University. His research interests are Intelligent Robot, Machine Vision, and Image Processing.

Mr. Junwei Fu

He is a postgraduate in Beijing Jiaotong University. He research interest is visual positioning and image processing.