

Real Time Object Pose Estimation by Two-Step Crossing Line Fitting

¹Minglei Tong, ²Shudong Chen

¹School of Electronics and Information, Shanghai University of Electric Power, Shanghai 201300, China

²Institute of Microelectronics of Chinese Academy of Sciences, Shanghai, China

¹E-mail: tongminglei@gmail.com

Received: 21 May 2014 /Accepted: 31 July 2014 /Published: 31 August 2014

Abstract: Real-time industrial vision system for object detection and pose estimation is a promising area yet posing a challenge for high processing efficiency. This paper presents a fast object detection and pose estimation method which captures the specific but common visual pattern as contained in many objects – the two-line cross. A two-step grid based scheme is designed, being able to fast detect the crossing line on the objects and thus identifying the object location and pose. Superior efficiency – 4 milliseconds per frame on a laptop with 2.53 HZ is reported for real image data, without any parallelization or hardware acceleration. Our method outperforms the state-of-the-art line detection method significantly, and has been applied in embedded inspection platform for pipeline object pose estimation. Copyright © 2014 IFSA Publishing, S. L.

Keywords: Object pose estimation, Monocular images, Crossing line fitting.

1. Introduction

Traditionally, visual inspection and quality control are performed by human experts like [1]. Recently, real-time machine vision system for object detection has become more and more prevalent due to its efficiency, non-stop and tolerance to harsh working environments such as nuclear and chemical industry [2, 6-10]. There are various machine vision approaches to object detection. Among them, feature point based methods are widely used because they can utilize versatile features which are robust to scale, rotation, and perspective variance [3]. Its shortcoming is the expensive computational overload due to the feature point detection and descriptor generation on image patches. Template matching is another line of important approaches due to its simplicity and feasibility in real application. It is usually good at handling low-textured objects and

more efficient than point based methods. However, it requires a traverse in the searching space of location, rotation and scale. One state-of-the-art in this category is the Dominant Orientation Template (DOT) method [4], where a similarity measure is proposed by using orientation of strong gradients. As a general object detection method, it can achieve typically 100 milliseconds per frame on a workstation, if not performing Streaming SIMD Extensions (SEE) acceleration or parallel.

In an industrial pipeline, the vision inspection process is highly time-constrained or computationally intensive, which directly impacts the productivity. Object detection or pose estimation, as a fundamental step in many applications, can serve as a pre-step for further processing including defects inspection, character recognition and robot arm manipulation etc. Such practical demands for high efficiency motivates this paper and we build our system based on the two

observations which holds in many occasions: 1) The inspected parts (object) bear one or more crossing lines; 2) The overhead-mounted camera can be readily focused on one area of line crossing – See Fig. 1 for illustration. One typical application scenario is using camera to guide the robot arms placing components on printed circuits, painting surfaces.

Contribution: 1) This paper makes the important observation that line cross is a pervasive and discriminative visual pattern that covers a large number of object categories for pose estimation; 2) Significant speedup has been achieved by the proposed method.

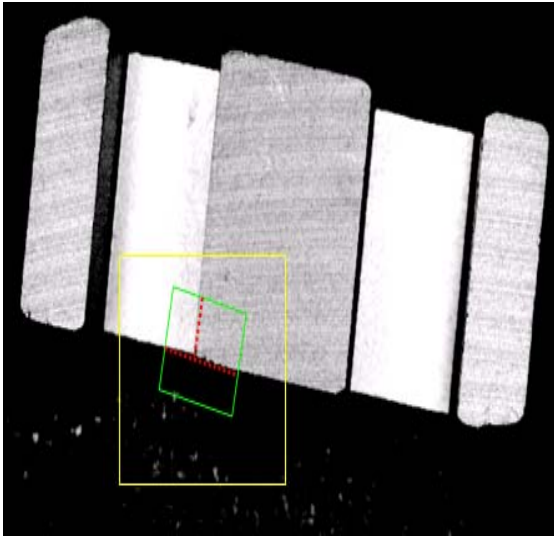


Fig. 1. Practical industry scenario: the overhead mounted camera can tightly cover the area of interest (marked by the yellow rectangle) where only two crossing line (denoted by the red lines) exists.

2. Line Feature Detection

Framework: First canny edge detector is adopted to obtain the edge map. This edge extraction step is similar with the widely used line detection methods like Hough Transform (HT). Then, the idea is to assign each edge pixel to one of the two cross lines, such that least square fitting can be performed on two groups of pixels respectively to find the pose and position of the cross. To achieve efficiency, our scheme is based on the philosophy ‘divide-and-conquer’ and ‘local-global’ two stage fitting. The method divides the edge map to grids and estimates the local fitting edge orientation in each grid via least square fitting.

The motivation of this step is based on two observations:

1) compared with the global pair-wise voting scheme in the HT method, grouping the edge pixels reduces the problem complexity thus bringing about possible speedup and robustness;

2) There are two crossing lines in the image area of interest, such that the grids can be assigned to three labels: i) One line, ii) The other line, and iii) Noises.

Thus we convert the edge pixel classification problem to a coarser granularity—assigning the grids into two lines. Note that for the grids covering the line crossing area, it can be identified as the noise since

1) Its fitting orientation is much less relevant to any of the two line directions that can be identified by the method as shown in below;

2) The local line fitting residue will be much higher due to no single dominant orientation in the grid.

3. Histogram of Orientation

Having obtained each grid’s local fitting orientation, we build a histogram, and the range of its x-axis is degree $[0,180]$, which is evenly divided by 36 intervals. The y-axis shows the accumulated frequency of the grid dominant orientations falling within the intervals, shown in Fig. 2. Smoothing between two neighboring intervals is performed as a standard process to avoid boundary effects in frequency accumulation. Then two peaks (i.e. line directions) θ_1 and θ_2 are found by finding the local maximums in the histogram. An optional post step can be used to verify whether the degree difference between the two peaks is close to the true intersection angle if this information is prior known. Now each grid can be labeled 1) as one of the two peaks according to the distance from the grid fitting direction to the peaks; or 2) as noises due to big in-grid fitting variance or its fitting direction is close to neither of the two peaks. As a result, the edge pixels take the same label as the grid they belonged to.

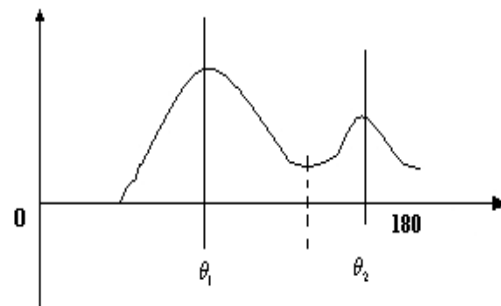
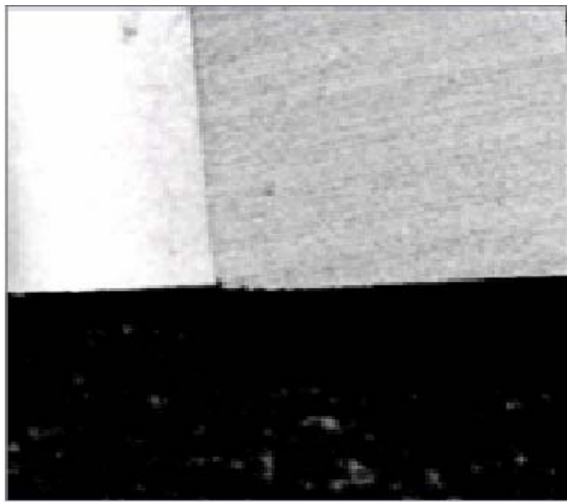
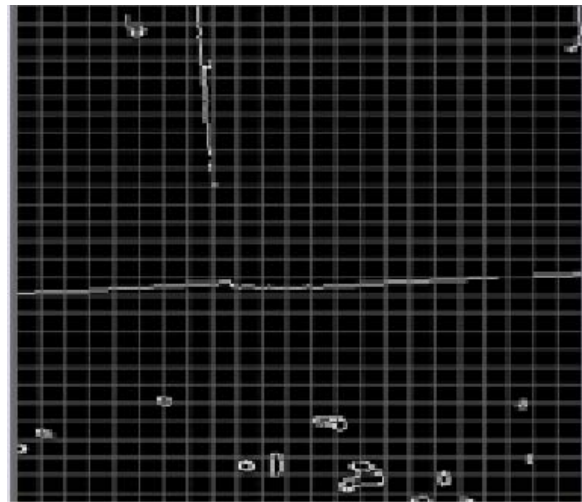


Fig. 2. Histogram of Orientation for the local fitting.

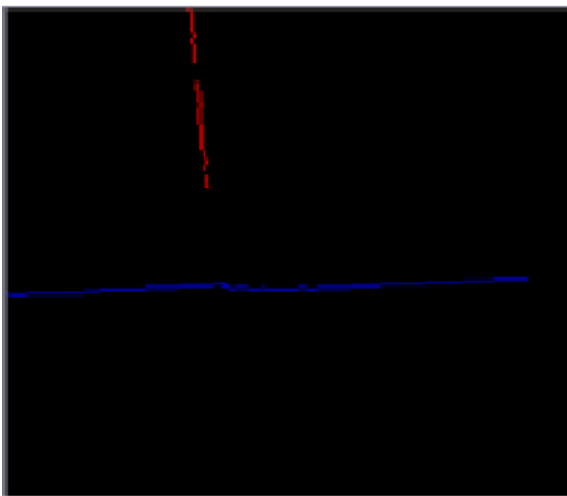
Having grouped edge pixels to each line respectively via the first stage local fitting at the grid scale as mentioned above, in the second stage, an image-scale global least square fitting is performed to fit the two crossing lines. Fig. 3 illustrates the main steps of our method.



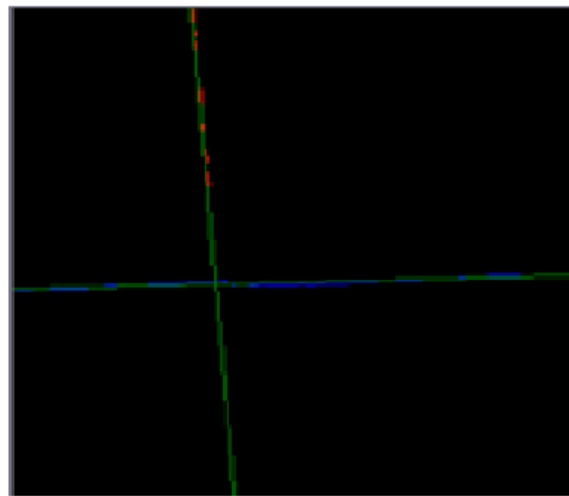
(a) Original input image



(b) Edge points detected by canny detector



(c) Classification for edge pixels, edges in the identified 'noisy' grid are removed, different colors distinguish two lines



(d) Fitted line directions by performing least square fitting

Fig. 3. Step by step illustration of the proposed method.

In terms of least square fitting used in this paper, given a set of edge points $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, for numerical stability, we adopt the symmetric parametric representation for line equation: $ax + by = 1$. And the parameters can be obtained by the following equations:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \frac{1}{\sum x_i^2 \sum y_i^2 - (\sum x_i y_i)^2} \begin{pmatrix} \sum y_i^2 & -\sum x_i y_i \\ -\sum x_i y_i & \sum x_i^2 \end{pmatrix} \begin{pmatrix} \sum x_i \\ \sum y_i \end{pmatrix} \quad (1)$$

4. Experimental Results

We compared the proposed method with the state-of-the-art line detection algorithm proposed by Gioi et al [5] – LSD (Line Segment Detector), which has been widely regarded as one of the most fast line

detection approaches. The testing dataset consists of 1000 images with size 200*200 collected from real industrial pipeline environment and the object in the images have various appearances – while with the common pattern – line crossings. The time overhead is evaluated by the overall mean and standard deviation in millisecond.

The testing environment is on a laptop with a 2.53 HZ CPU and the approaches have been both implemented using C++. In our running speed evaluation, LSD costs on average 16.47 ms on an image of size 200*200, and our approach costs 4.08 ms, while resulting in similar or even better results. Fig. 4 shows the detecting results on two soft disks.

Fig. 4 (a) gives an original input image in which crossing lines separate 3 blocks of regions with different gray values. This is a very challenging sample. Fig. 4 (b) shows the detecting result of edge

points with the method of Canny detector. There are still many noises in the preliminary results including disordered isolated points and holes. The aim is to detect complete crossing lines from the noisy image. Fig. 4(c) gives the result of the incomplete crossing edges (the assembly of red and blue points) excluding the noisy points and holes. Fig. 4(d) shows the final results by using least squared fitting. With the proposed method the crossing lines can be detected fast and accurate.

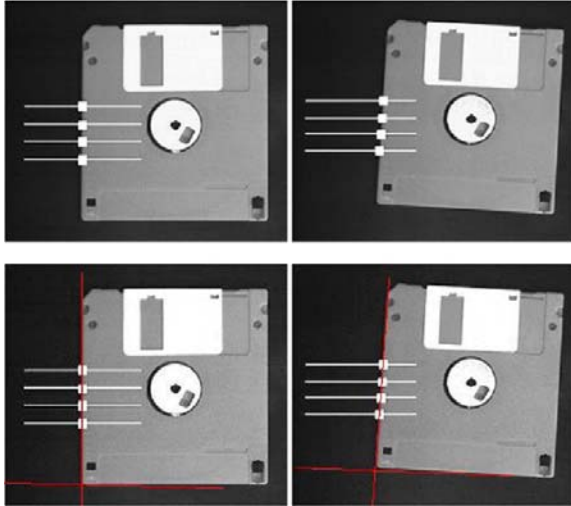


Fig. 4. Experimental results on image samples of soft disk.

Table 1 shows the performance results. From Table 1 we compared the proposed methods with the way of LSD. Our method can achieve a mean running time of 4.08 ms/frame with a standard deviation of 0.202 while the LSD method will cost 16.47 ms/frame. The successful rate in our proposed methods is 99.4 % while in LSD, the successful rate is only 92 %. Some exemplary results are displayed in Fig. 5. There are six typical examples to prove the accuracy of our method. In Fig. 5 the detected lines are green in every image. For each image test, the success is defined as the detected pose error is less than a given threshold (3 degrees) compared with the ground truth as labeled by human on the raw image.

Table 1. Performance evaluation over 1000 image samples with various appearances for the proposed method and the LSD algorithm proposed in [5].

Method	Mean runtime in millisecond (Standard deviation)	Success rate
Our method	4.08 (0.202)	99.4%
LSD	16.47 (0.927)	92.8%

5. Conclusion

In this paper, A method that obeys the two principles is proposed: i) 'divide-and-conquer',

ii) 'first fit locally, and then fit globally' towards the specific but important problem of crossing line detection in real-time industrial vision system. Object detection or pose estimation, as a fundamental step in many applications, can serve as a pre-step for further processing including defects inspection, character recognition and robot arm manipulation etc. We build our system based on the two observations which holds in many occasions: 1) the inspected parts (object) bear one or more crossing lines; 2) the overhead-mounted camera can be readily focused on one area of line crossing. One typical application scenario is using camera to guide the robot arms placing components on printed circuits, painting surfaces.

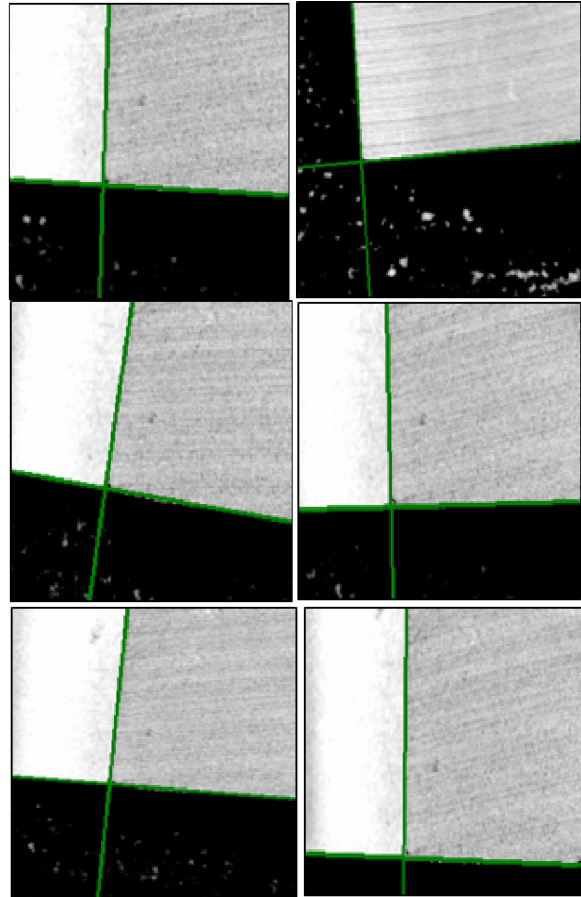


Fig. 5. Experimental results on real image samples.

Acknowledgements

This paper is supported by the National Nature Science Foundation of China (No. 61105016), the Innovation Program of Shanghai Municipal Education Commission (Grant No. 12YZ139),

References

- [1]. A. Mital, M. Govindaraju, B. Subramani, A comparison between manual and hybrid methods in

- parts inspection, *Integrated Manufacturing Systems*, Vol. 9, Issue 6, 1998, pp. 344-349.
- [2]. M. Ozuysal, P. Fua, V. Lepetit, Fast keypoint recognition in ten lines of code, in *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA, 2007, pp. 1-8.
- [3]. E. Rosten, T. Drummond, Machine learning for high-speed corner detection, in *Proceedings of the 9th European Conference on Computer Vision*, Graz, Austria, 2006, pp. 430-443.
- [4]. S. Hinterstoisser, V. Lepetit, S. Ilic, et al, Dominant orientation templates for real-time detection of texture-less objects, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, USA, 2010, pp. 473-475.
- [5]. R. Gioi, J. Jakubowicz, J. Morel, et al, LSD: A fast line segment detector with a false detection control, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 32, Issue 4, 2010, pp. 722-732.
- [6]. C. Akinlar, C. Topal, EDLines: real-time line segment detection by edge drawing (ED), in *Proceedings of the International Conference on Image Processing (ICIP)*, Brussels, 11-14 September 2011, 2837-2840.
- [7]. P. Franti, E. I. Ageenko, H. Kalviainen, S. Kukkonen, Compression of line drawing images using Hough transform for exploiting global dependencies, in *Proceedings of the International Conference on Information Sciences*, 1998, pp. 433-436.
- [8]. Yefeng Zheng, Huiping Li, David Doermann, A parallel-line detection algorithm based on HMM decoding, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, Issue 5, May 2005, pp. 777-792.
- [9]. Florence Tupin, Henri Maître, Jean-François Mangin, Jean-Marie Nicolas, and Eugène Pechersky, Detection of linear features in SAR images: application to the road network extraction, *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 36, Issue 2, 1998, pp. 434-453.
- [10]. F. O'Gorman, M. B. Clowes, Finding picture edges through collinearity of feature points, *IEEE Transactions on Computers*, Vol. 25, Issue 4, April 1976, pp. 449-456.

2014 Copyright ©, International Frequency Sensor Association (IFSA) Publishing, S. L. All rights reserved.
(<http://www.sensorsportal.com>)

Promoted by IFSA

Status of the CMOS Image Sensors Industry Report up to 2017

The report describes in detail each application in terms of market size, competitive analysis, technical requirements, technology trends and business drivers.

Order online:

http://www.sensorsportal.com/HTML/CMOS_Image_Sensors.htm