A Real-time Automatic Level Bar Calibration Based on Canny Edge Detection and Weighted Least Squares Method*

Wei Sheng[1], Qing-Guo Wang[2] and Shanan Zhu[1]
[1]College of Electrical Engineering, Zhejiang University, Hangzhou, China
[2]Institute for Intelligent Systems, the University of Johannesburg, South Africa
E-mail: Saintway@qq.com

Abstract—**This paper presents a real-time automatic calibration method for the level bar based on the Canny edge detection and the weighted least squares. In order to reduce the effects of changes in the light field on detection accuracy, the Canny edge detection algorithm for image pre-processing is introduced. The weighted least squares is employed to model the parallel reference lines and the binary search algorithm is used to search the bubble position, so as to achieve high-precision calibration of the level bar. The proposed method has been implemented in a real production line and works well.**

Keywords—**image processing; real-time detection; level bar; Canny edge detection; weighted least squares**

I. INTRODUCTION

The level bar is a common tool for measuring small angle, and it is often used in the construction, machinery and instrument manufacturing industry. Accordance to different working principles, the level bar is divided into bubble level, electronic level and laser level. The bubble level is the most widely used traditional level bar, which determines the horizontality of a plane by the bubble deviation. The electronic level includes inductive level and capacitive level: the inductive level determines whether a plane is level by the voltage change of the induction coil, while the capacitive level by the capacitance of interstices on both ends. The laser level leads the beam of the laser emission to telescope tube, making it beam through the collimation axis, if the beam isn't blocked, the plane is level. The level bar in this paper is one kind of bubble level.

Before leaving the factory, the level bar needs to be calibrated, namely to put the level bar on the standard horizontal plane, to find the bubble deviation from the center of the level bar to check the measurement accuracy of the product. In China, the detection is mainly depended on the human eye judgment which is slow, erroneous and instable. Therefore, there is an urgent need for a novel method capable of accurate calibration of the level bar to improve the precision of the product.

*Research supported by Dongfang Precision Science and Technology Co., Ltd.

Wei Sheng is with College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, China (e-mail: Saintway@qq.com).

Qing-Guo Wang is with Institute for Intelligent Systems, the University of Johannesburg, South Africa.

Shanan Zhu is with College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, China (corresponding author to provide e-mail: zsa@zju.edu.cn).

The bubble in the level bars formed from the remaining air by filling the cylindrical cavity 4/5 green oily liquid. A calibration algorithm based on optimal thresholding was proposed by Wei Yu [1-2]. In this paper, we present a realization of automatic and real-time detection of level bar based on Canny edge detection and weighted least squares method. In order to reduce the effects of light changes on detection accuracy, we introduce a method to adaptively set the thresholds of the Canny Edge Detection. A faster contour tracing algorithm based on binary search is employed. Experimental results show that this solution is feasible.

The bubble of the level bars sealed in a glass column, and it will reflect strong light and disturb the identification of bubble edge and the whole image recognizing process if we direct light the bubble or light it in a wrong angle. Considering that the bubble and the glass column are transparent, the background is required to be flat and uniform. Since the upper surface of the bubble of the level bar is convex, if taking the image from the top of the level bar, there is always a mirror reflection effect, so that the quality of the image deteriorates and will impact the effect of subsequent detection algorithm. So we decide to take the image from the side of the level bar. A CCD camera is used as an image sensor in this application.

II. IMAGE PRE-PROCESSING AND IMAGE SEGMENTATION

Before proceeding with target identification and location, we need to pre-process the collected images as to obtain the desired input image for the target recognition. Pre-processing is an important procedure, which directly influences the algorithm detection accuracy. The basic pre-processing contains image graying, image intensification and image filtering. Since the image shoot by the CCD camera is a grayscale of two million pixels, only the image filtering is needed in pre-processing. Image segmentation is the process which divides the image into several specific areas, and these areas have different qualities and can be handled in the following processes.

A. Gaussian Smoothing Filter

Since the image acquisition in industrial environment will be subject to external environmental factors such as electromagnetic interference, dust and so on, the collected images will contain some noise components. Canny proposed a new edge detection method [4], which is optimal to detect step edges affected by the white noise. He used the functional derivation method to deduce that the first derivative of Gaussian function is the best approximation of the optimal edge detection operator. Since convolution and differential are the operations which satisfy the associative law, it is possible to use two-dimensional Gaussian function to smooth the image first:

$$Gauss(x, y) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{1}$$

where σ is the Gaussian filter parameter to control the degree of smoothing.

For a filter of a small σ, the scale of the separation event is small, so that the precision of the edge positioning is high, but the SNR is low. For a large σ, the answer is opposite. In the collected image, the parallel reference lines are about 15 pixels wide. It can be obtained by trial and error that the side of the mask of the Gaussian smoothing filter is about half of the line width so that the effect of the edge extraction is the best. Therefore, in this application, the side of the mask is taken as 7 pixels, and the standard deviation is $\sigma = 1.55$.

B. Calculation of Gradient Magnitude and Direction

Canny edge detection algorithm is based on three basic goals: (1) low error rate: all edges should be found and no pseudo repulse exists, which means these edges contain as much real edges as possible. (2) peripheral points are well located: the edge which is already located must be as close as possible to the real edge, which means the distance between the peripheral point(marked by the detector) and the real edge point should be shortest. (3) single edge response: the detector should feedback one point when the real edge is detected, and the local maximum number around the real edge should be minimal, it also means we should not detect more than one edge pixel when there exists single edge point. Canny operator uses a first-order differential operator to calculate the gradient magnitude G and the gradient direction θ at each point of the smoothed image. Here, a finite difference operator is chosen as a first-order differential operator. And the partial derivative $G_x(x, y)$ and $G_y(x, y)$at the point (x, y) are as follows:

$$G_x(x, y) = f(x + 1, y) - f(x - 1, y) \tag{2}$$

$$G_y(x, y) = f(x, y + 1) - f(x, y - 1) \tag{3}$$

where f is the smoothed image.

The gradient magnitude and the gradient direction at the point (x, y) are as follows:

$$G(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)} \tag{4}$$

$$\theta(x, y) = arctan\frac{G_y(x,y)}{G_x(x,y)} \tag{5}$$

C. Non-maxima Suppression

In order to accurately locate the edge, we must refine the ridge in the image gradient magnitude G, and retain only the local maximum of the magnitude, namely the non-maxima suppression. Non-maximum suppression (NMS) can be regarded as a local maximum value of the search problem, and is a part of many computer vision algorithms. Canny operator performs the interpolation along the gradient direction θ (x, y) and within the field of 3 × 3 whose center is at the point (x, y) in the gradient magnitude image G. If the gradient magnitude G at the point (x, y) is greater than the two adjacent interpolations in the direction of θ (x, y), then the point (x, y) is marked as candidate edge point, otherwise marked as non-edge point. Thereby the candidate edge image N is obtained.

D. Detection and Connection of the Edges

Canny operator uses the dual threshold method to get the final edges from the candidate edge points by detections and connections. First set the high threshold T_h and the low threshold T_l. The choice of high and low threshold determines the number of edge points. Too few edge points means that a part of the real edge lost, leading to decline the continuity of the edge. Too many edge points will introduce noise and affect the testing result. Then scan the image and detect any point (x, y) which are marked as candidate edge point in the candidate edge image N. If the gradient magnitude G at the point (x, y) is greater than the high threshold T_h, the point is determined as the edge point; if the gradient magnitude G at the point (x, y) is less than the low threshold T_l, the point is definitely not the edge point. As for any point whose gradient magnitude is between the two thresholds, it is considered as a suspected edge point and is needed to be further determined by the connectivity of the edge. If any adjacent point of this point is an edge point, this point is also determined as an edge point, otherwise as a non-edge point.

In this application, the high threshold T_h is set adaptively, so that the number of the strong edge points, namely whose gradient magnitude is larger than the high threshold T_h, is 5% of the total. And set $T_l = T_h \times 40\%$. Repeated tests showed that the proposed method with the adaptive thresholds can extract the edge information of the image effectively and has strong robustness against optical instability, see Figure. 1 and 2.

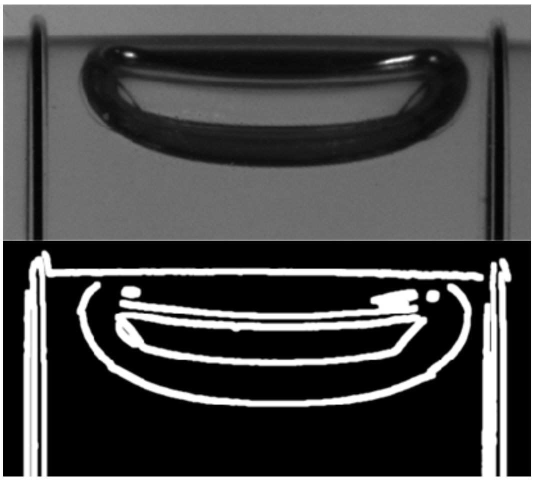

Figure 1. The Result of Image Segmentation (Insufficient Light;SlantLens;$T_h = 91.4$,$T_l = 36.5$)

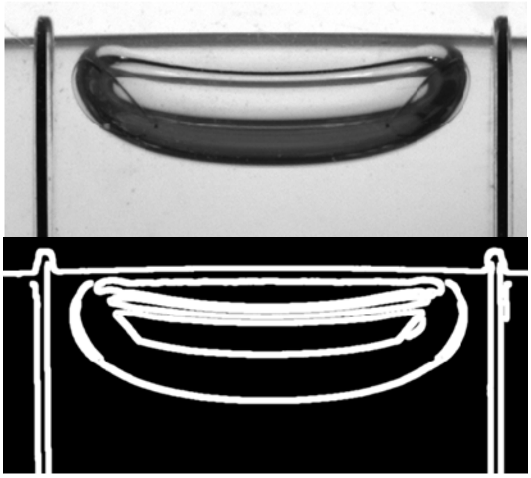

Figure 2. The Result of Image Segmentation(Adequate Light; Direct FacingLens;T_h = 201.3, T_l = 80.5)

III. THE IDENTIFICATION OF THE PARALLEL REFERENCE LINES AND THE MEASUREMENT OF THE BUBBLE POSITION

To determine the horizontality of the level bar's bubble, it requires accurate positioning of the bubble and the two parallel reference lines after the image segmentation. First we need to identify the two parallel reference lines from the image.

A. Identify and Locate the Parallel Reference Lines

To identify and locate the parallel reference lines, we can use the Hough transform. The least squares method can also be used to get the line equations of the two parallel reference lines. It detects the known point of the total line, which is a global detection method. When the known data points focus on the interference points and noise, it can suppress the interference and the noise very well. At the same time, it can also synthesize a number of known data points. However, the accuracy of Hough transform is not easy to control. When fitting a straight line with high accuracy, you cannot use Hough transform. Hough transform is the output parameters of collinear point linear equation, when we need to get the line, we still need further treatment.

Hough transform is a relatively rather fast method to look for lines in a binary image [5-6]. In this application of the Hough transform, the parameterized form of the line equation $\rho = x\cos\theta + y\sin\theta$ is used. The range of the horizontal axis θ of the accumulator plane is [-45 °, + 45 °], and its resolution is taken as 0.1 °. The range of the vertical axis ρ is [0, 1000] pixels and its resolution is taken as 1 pixel.

Another approach is to use the least squares method to fit the line equations of the parallel reference lines. Least squares method is a mathematical optimization techniques.

Let the line equation of the left reference line be $\widehat{\rho_n} = x\cos\hat{\theta} + y\sin\hat{\theta}$ and let the line equation of the right reference line be $\widehat{\rho_m} = x\cos\hat{\theta} + y\sin\hat{\theta}$. Identification and localization of the parallel reference lines can be expressed as the following minimum problem:

$$\begin{pmatrix}\widehat{\rho_n}\\ \widehat{\rho_m}\\ \hat{\theta}\end{pmatrix} = \underset{(\rho_n\rho_m\theta)^T}{arg\,min} \quad \|\boldsymbol{e_n}\rho_n + \boldsymbol{e_m}\rho_m - \boldsymbol{x}\cos\theta - \boldsymbol{y}\sin\theta\|^2 \quad (6)$$

where $\boldsymbol{e_n} = [\overbrace{1 \;\cdots\; 1}^{\mathrm{n}} \;\; \overbrace{0 \;\cdots\; 0}^{\mathrm{m}}]^{\mathrm{T}}$; $\boldsymbol{e_m} = [\overbrace{0 \;\cdots\; 0}^{\mathrm{n}} \;\; \overbrace{1 \;\cdots\; 1}^{\mathrm{m}}]^{\mathrm{T}}$; n is the number of points on the left reference line to be collected; m the number of points on the right reference line to be collected; $\boldsymbol{x}$ a column vector consisted of the abscissas of the points on the two parallel reference lines; $\boldsymbol{y}$a column vector composed of the ordinates of the points on the two parallel reference lines.

Let $J(\rho_n, \rho_m, \theta) = \|\boldsymbol{e_n}\rho_n + \boldsymbol{e_m}\rho_m - \boldsymbol{x}\cos\theta - \boldsymbol{y}\sin\theta\|^2$, and by making$\partial J/\partial\rho_n = 0$, $\partial J/\partial\rho_m = 0$, $\partial J/\partial\theta = 0$, we obtain:

$$\widehat{\rho_n} = \frac{\boldsymbol{e_n^T x}\cos\hat{\theta} + \boldsymbol{e_n^T y}\sin\hat{\theta}}{n} \quad (7)$$

$$\widehat{\rho_m} = \frac{\boldsymbol{e_m^T x}\cos\hat{\theta} + \boldsymbol{e_m^T y}\sin\hat{\theta}}{m} \quad (8)$$

$$\hat{\theta} = \frac{1}{2}arctan\frac{2\left[\boldsymbol{y}^T\boldsymbol{x} - \frac{\boldsymbol{y}^T\boldsymbol{e_n}\boldsymbol{e_n^T}\boldsymbol{x}}{n} - \frac{\boldsymbol{y}^T\boldsymbol{e_m}\boldsymbol{e_m^T}\boldsymbol{x}}{m}\right]}{\left[(\boldsymbol{x}^T\boldsymbol{x} - \boldsymbol{y}^T\boldsymbol{y}) - \frac{\boldsymbol{x}^T\boldsymbol{e_n}\boldsymbol{e_n^T}\boldsymbol{x} - \boldsymbol{y}^T\boldsymbol{e_n}\boldsymbol{e_n^T}\boldsymbol{y}}{n} - \frac{\boldsymbol{x}^T\boldsymbol{e_m}\boldsymbol{e_m^T}\boldsymbol{x} - \boldsymbol{y}^T\boldsymbol{e_m}\boldsymbol{e_m^T}\boldsymbol{y}}{m}\right]} \quad (9)$$

In this application, a morphological contour tracking method is used, which search along the two parallel reference lines from the bottom up for all connected sample points. In this practical application, the used image is not the whole image captured by the CCD camera but a small area as shown in figure 1 and figure 2. There is a white area above the bubble, and between the white area and the bubble lies a black boundary. Searching along the two parallel reference lines from the bottom up can avoid the influence of this boundary so make this process more simple and easy. The least square method has short comings to equal the contribution for all data and does not take into account the existence of the final results of the abnormal points. In this application, there is a problem for determining the end of sampling, so the final results can not accurately reflect the actual situation. Here a weighted least squares method is proposed. The steps are as follows:

1) take 3 points from each two parallel reference lines, then calculate the parameters $\widehat{\rho_n}$, $\widehat{\rho_m}$, $\hat{\theta}$.

2) repeatedly take 1 point from each two parallel lines, and calculate the distances from the points to $\widehat{\rho_n} = x\cos\hat{\theta} + y\sin\hat{\theta}$, $\widehat{\rho_m} = x\cos\hat{\theta} + y\sin\hat{\theta}$ respectively. If the distance is greater than a threshold value (in this application, it is taken as 3 pixels), then discard the sample point. Otherwise, use the sample point to update the parameters $\widehat{\rho_n}$, $\widehat{\rho_m}$, $\hat{\theta}$.

Same robustness can be achieved, in this application, using the weighted least squares method described above compared to using the Hough transform. And the former is much more efficient on the computation time and occupies lesser memory storage than the latter.

B. Identify and Locate the Endpoints of the Bubble

After the identification and location of the left and right parallel reference lines, we need to identify and locate the endpoints of the bubble.

In this application, a contour tracing algorithm based on step halving is adopted.

Suppose the line equation of the left reference line, fitted out by the weighted least squares, to be $\rho_n = x\cos\theta + y\sin\theta$, and the line equation of the right reference line to be$\rho_m = x \cos\theta + y \sin\theta$, the distance from the center line to the origin is $\rho_o = (\rho_n + \rho_m)/2$.

The contour tracing algorithm to identify and locate the bubble left endpoint A is as follows:

1) take point A as A $(\rho_o/\cos\theta, 0)$, and take the step as $d = (\rho_m - \rho_n)/4$. In this case, the accuracy of the point A is $\pm 2d$.

2) along $\theta + 90°$ direction, taking the step as 1 pixel, move the point A step by step, looking for the edge point of the bubble. If there is an edge point on the moving locus of point A, then along the $\theta + 180°$ direction, move point A d pixels. Otherwise, return point A to the initial position, and along the θ direction, move point A d pixels. The step d is then reduced to the half. Repeat step 2) until the accuracy of point A, which is $2d$, less than 0.5 pixels.

The contour tracing algorithm to identify and locate the bubble right end point B is the same as for A.

Suppose the results of the algorithm is point A (x_A, y_A) and B (x_B, y_B). The distance from the tangent through the point A to the origin is $\rho_A = x_A \cos\theta + y_A \sin\theta$. And the distance from the tangent through the point B to the origin is $\rho_B = x_B \cos\theta + y_B \sin\theta$.

The deviationD_{bias} of the bubble from the center can be calculated by:

$$D_{bias} = (\rho_A + \rho_B)/2 - \rho_o \ \ pixels \tag{10}$$

$D_{bias} < 0$ indicates the bubble to deviate to the left; $D_{bias} > 0$ indicates the bubble to deviate to the right; $D_{bias} = 0$ represents the bubble to be at the center.

To obtain physical distance in millimeters, a calibration is needed to be done before the detection. This calibration can be done by getting the ratio γ of mm to the pixel size.

IV. Results

We applied our algorithm to the images acquired in the industrial environment. The following table shows two test results of those images:

TABLE I. MEASUREMENTS OF THE DEVIATIONS

Image	Features	Deviation		Deviating Direction
		pixel	*mm*	
Fig. 3	insufficient light	8	0.300	deviating to left
Fig. 4	slant lens	3.5	0.131	deviating to right

Picture Resolution is 1600 × 1200, γ = 0.0375

Figure 3. The Image Acquired (Insufficient Light)

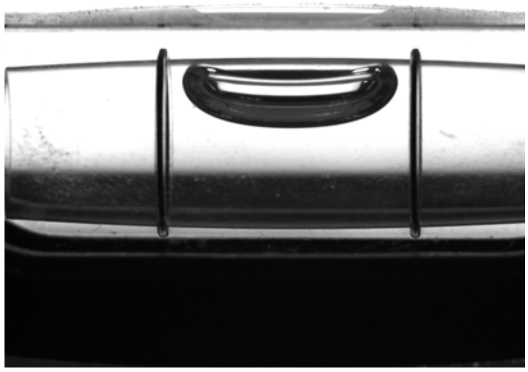

Figure 4. The Image Acquired (Slant Lens)

As can be seen from the actual test results, when dealing with poor quality images (such as insufficient light, slant camera lens, etc.), the algorithm still can get satisfactory results.

The interface is based on the MFC framework, as shown in Figure 5. We use the cvSmooth and cvCanny function in OpenCV v1.0 [11-13] to implement the first part of the work (Image Pre-Processing and Image Segmentation). The algorithm of the second part of the work (Identification of the Reference Lines and Measurement of the Bubble Position) is implemented in C language.

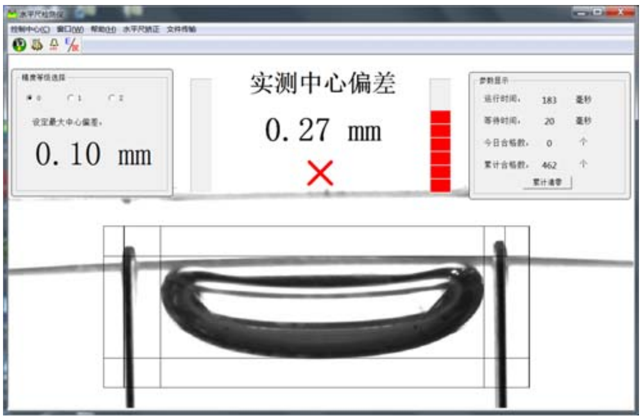

Figure 5. The Interface of the Software

V. CONCLUSIONS

The proposed detection method has been combined into a system software, and with a PC and video capture card, has been put into practical use. The results show that for even images with poor quality, the detection system is robust and can obtain high detection accuracy. The users are satisfied with these results.

REFERENCES

[1] Yu Wei and Shanan Zhu, "The Application of Image Processing in Level Bar Calibration System." [J] Application Research of Computers, 2004, 21 (12): 168-169,172. DOI: 10.3969 / j.issn.1001-3695. 2004. 12. 058.

[2] Yu Wei and Shanan Zhu, "The Application of Optimal Threshold Transform and Contour Tracking in Contour Detection." [J] Computer Engineering and Applications, 2004, 40 (24): 88-90. DOI: 10.3321 / j.issn: 1002-8331. 2004. 24. 027.

[3] Cuntang Wang, Enhui Yang, Jiangguo Zhang, Yingxing Ma, Bi Zhang. “The Structure and Measurement Principle Analysis for Omnibearing Electronic Gradienter.”[J] Chinese Journal of Scientific Instrument. 2006.27:183-185.

[4] J. Canny, “A computational approach to edge detection,” [J] IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (1986): 679-714.

[5] P. V. C. Hough, “Machine analysis of bubble chamber pictures,” [J] Proceedings of the International Conference on High Energy Accelerators and Instrumentation (pp. 554-556), 1959.

[6] R. O. Duda and P. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” [J] Communications of the Association for Computing Machinery 15 (1972): 11-15.

[7] F A Pellegrino. “Edge Detection Revisited”, [J]. IEEE Trans on System Man and Cybernetics. 2004, 34(3):1500- 1517.

[8] T Nguyen,D Ziou. “Contextual and Non-contextual Performance Evaluation of Edge Detectors”. [J]. Pattern Recognition Letters, 2000, 21(8):805- 816.

[9] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing (Second Edition), [M] Beijing: Electronic Industry Press, 2002, 7.

[10] Jinbao Wang, “The Application of Image Processing Technology in the Automatic Measurement of Level Bars.”[D] Zhejiang University.2007

[11] Gary Bradski, Adrian Kaebler, Learning OpenCV, [M] Beijing: Tsinghua University Press, 2009,10.

[12] Richard O. Duda, Peter E. Hart, David G. Stork, Pattern Classification (Second Edition), [M]Beijing: Mechanical Industry Press, 2003, 9.

[13] Gonzalez R C, Woods R E. Digital Image Processing (Third Edition)[M].Beijing Publishing House of Electronics Industry, 2011:445-465.