Research Article

Research on Subpixel Algorithm of Fixed-Point Tool Path Measurement

Xi Zhang , Zixie Guo , Xiangwei Liu , and Longjia Zhang

School of Mechanical and Automation, Shanghai University, Shanghai 201900, China

Correspondence should be addressed to Zixie Guo; guozixie18@shu.edu.cn

Received 9 July 2021; Accepted 24 August 2021; Published 3 September 2021

Academic Editor: Mario Versaci

Copyright © 2021 Xi Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Tool safety is an important part of machining and machine tool safety, and machine tool path image detection can effectively obtain the in-machine condition of a tool. To obtain an accurate image edge and improve image processing accuracy, a novel subpixel edge detection method is proposed in this study. The precontour is segmented by binarization, the second derivative in the neighborhood of the demand point is calculated, and the obtained value is sampled according to the specified rules for curve fitting. The point whose curve ordinate is 0 is the subpixel position. The experiment proves that an improved subpixel edge can be obtained. Results show that the proposed method can extract a satisfactory subpixel contour, which is more accurate and reliable than the edge results obtained by several current pixel-level operators, such as the Canny operator, and can be used in edge detection with high-accuracy requirements, such as the contour detection of online tools.

1. Introduction

With the continuous improvement of machining accuracy, especially precision machining, increasing tool chip parameters and optical imaging system precision is limited. Thus, under the same conditions, subpixel subdivision can improve measurement accuracy and reasonably control production cost. Therefore, examining a set of mature and stable subpixel contour detection algorithms is of considerable significance. This study conducts in-depth research on existing subpixel detection algorithms and proposes algorithm optimization for the environmental application of numerical control tools.

Image edge detection in computer image processing is a technology that was developed in the past but is continuously advancing. It plays a very important role in computer vision and image processing and is widely used in various fields such as image segmentation and pattern recognition [1, 2]. Feng et al. [3] proposed a saliency detection method based on the histogram contrast algorithm and wireless multimedia sensor network (WMSN) image to process animal images. In recent years, Zhang et al. [4] proposed an edge detection algorithm based on structured random forest,

which can make full use of the RGB-D image information of Kinect. With the development of industrial cameras and the improvement of image quality, the application of image edge detection technology in the industry became increasingly extensive.

Early commonly used pixel-level edge extraction methods are mainly classical edge extraction operator algorithms, such as the Sobel operator [5], Scharr operator [6], Laplacian operator [7–10], and gate operator. These algorithms demonstrate one-pixel accuracy; that is, they can determine which pixel the edge is in and the most accurate position, which traditional algorithms cannot judge. With the improvement of the accuracy requirements of target edge detection, the pixel-level edge obtained by common operators cannot meet the needs of precision measurement and visual calibration. Therefore, research on subpixel edge detection is essential. Current subpixel detection methods include the probability theory method, demodulation measurement method, and polynomial interpolation method [11–16].

Subpixel algorithms have been studied by many people. Wu proposed a subpixel edge detection algorithm based on Franklin's matrix, which was fast and accurate and

demonstrated strong noise immunity [17]. Xie et al. [18] proposed an improved subpixel edge detection algorithm combining coarse positioning with fine positioning. Kumar and Ratnam [19] used the moment invariant subpixel detection method to improve the detection accuracy. A tool wear measurement system based on machine vision was developed to solve the problems of manual operation and shutdown detection in actual production [20]. Wu proposed a laser image subpixel edge detection method based on Gabor filtering and mathematical morphology [21]. Ai et al. proposed a corner detection algorithm based on subpixel edges [22].

2. Materials and Methods

Image subpixel processing involves complex mathematical operations, which can considerably increase the running time of a program and affect the efficiency of the entire production and processing. In actual NC machining, the fixed-point tool diameter is typically used as a parameter. Fixed-point tool diameter measurement is employed to determine the diameter of a tool by measuring the radial distance at the fixed height of the tool.

In this situation, subpixel processing the entire image is not necessary, we only need to process the image to obtain the pixel-level contour of the image, then establish the ROI region within the specified line range, and use a specific subpixel algorithm for the ROI region to obtain the required subpixel points, which can take into account the improvement of calculation accuracy and algorithm efficiency at the same time.

2.1. Image Preprocessing. Image preprocessing is an important part of finding the image edge point, but images before being processed may be affected by uncertainty and/or inaccuracy, so sometimes we need a fuzzy preprocessing [23, 24]. Image preprocessing and ROI region positioning in advance can realize the fast positioning of the subpixel region. In this study, the reverse binarization operation is performed on the image to distinguish the image foreground and background before determining the ROI region. The calculation method is shown in equation (1). The advantage of this method is that because the contour envelope algorithm used in contour extraction will envelope the white region, the reverse binarization of the image can reduce the influence of image boundary on image contour extraction.

$$dst(x, y) = \begin{cases} 0, & \text{if } src(x, y) > \text{threshold}, \\ \text{Maxval}, & \text{otherwise.} \end{cases} \quad (1)$$

As the input image is a binary image and its value is only 0 and 255, in the image processing, the input image is binary $f(i, j)$ when scanning, and scanning is terminated in the following two cases: (1) $f(i, j-1) = 0$ and $f(i, j) = 255$, where $f(i, j)$ is the starting point of the outer boundary, and (2) $f(i, j) = 255$ and $f(i, j+1) = 0$, where $f(i, j)$ is the starting point of the hole boundary. This contour extraction algorithm comes from a paper published by Satoshi Suzuki [25].

At this time, the starting point is determined, and the contour boundary sequence can be obtained with the Freeman chain code [26]. However, the actual image may generate multiple contours owing to the influence of a backlight, but the cutter part occupies the dominant position in the image contour. Therefore, the longest contour sequence is screened so as to successfully separate the tool contour from the contour extracted from the picture.

After the image contour sequence is obtained, the next step involves finding the fixed point in the contour sequence to measure the position of the pixel-level contour points in the contour sequence. In this study, to find the one-dimensional sequence, a binary search algorithm is used to search for the contour points. First, the known array is sorted by the quick sort algorithm. Second, the binary search method is used for the search.

As the data structure of points belongs to user-defined data structures, the sorting rules must be defined in the quick sort algorithm. In this study, the ascending order of the Y coordinate and the ascending order of the X coordinate are chosen when the y coordinate is the same to sort the contour points. This method places the contour points on the same row of pixels arranged from left to right, which can provide a satisfactory basis for searching.

According to the search method described above, when the first contour point in the line where the measurement position is located is detected, all points with the same y coordinate as the point are taken out, and the two points with the smallest and largest X coordinate are taken as the two starting positions of the fixed-point tool path subpixel iteration; the region formed by extending 10 pixels to the left and right with these two points as the center is the ROI region where the subpixel points in the target row are located as shown in Figures 1 and 2.

If the gray value at the back of the falling-edge or rising-edge ROI is insufficient, then the ROI will be extended 5 pixels backward until the value is insufficient. If the gray value at the front of the rising-edge or falling-edge ROI is insufficient, then the ROI will be extended 5 pixels forward until the value is insufficient.

2.2. Single-Point Subpixel Algorithm. In the common image edge solving methods, finding the highest point of the first derivative and finding the zero-crossing point of the second derivative are the two common methods. As shown in Figure 3, the black coordinate points represent the derivative values obtained by solving the first derivative of the image with Canny operator. The blue curve is the result of interpolating these values. It can be seen that the maximum value of the interpolation curve is between the black coordinate points, which means that the actual maximum value is not on the coordinate points, and the actual edge of the image is also on the subpixel points. In this study, the second derivative is used to calculate the single-point subpixel.

Assume that the image function expression of the ROI region is $f(x)$, and the expression of the first derivative is

FIGURE 1: Grayscale map of the ROI region of fixed-point measurement (falling-edge).

FIGURE 2: Grayscale map of the ROI region of fixed-point measurement (rising-edge).

FIGURE 3: First-order derivative curve of gray value in the ROI region.

FIGURE 4: Second-derivative curve of gray value in the ROI region.

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}. \quad (2)$$

The image function expression is discretized, and the pixel interval $h=1$ is taken, and then the discretized second-derivative expression is as follows:

$$f''(x) = f'(x+1) - f'(x). \quad (3)$$

After expansion, the second-order expression is as follows:

$$f''(x) = f(x+2) - 2f(x+1) + f(x). \quad (4)$$

Thus, the convolution kernel of the second-order differential operator can be obtained as follows:

$$[1, -2, 1]. \quad (5)$$

After the image function of the one-dimensional ROI is convoluted with the convolution kernel of the second-order differential operator, the second-order difference value can be obtained, as shown in Figure 4.

It can be found from Figure 4 that the curve distribution near the highest point and the lowest point of the curve composed of second-order derivative values is similar to the quartic function curve. At the same time, it is noted that the zero-crossing point is not at the pixel point, but at the subpixel point, which indicates that the image edge is also at the subpixel point. In this study, we approximately replace the second-derivative curve of the gray value with the quartic curve.

Suppose that the expression of the quartic curve is

$$f(x) = ax^4 + bx^3 + cx^2 + dx + e, \quad (6)$$

where a, b, c, d , and e are constant and $a \neq 0$.

By solving the root of the above quartic equation, the zero-crossing point of the second derivative of gray value in the ROI region can be obtained.

The quartic curve zero-crossing problem can be transformed into the solution of the root of the quartic equation of one variable. Finding the root of a high-order equation is tedious; thus, the numerical solution can be obtained by the iterative method. In this study, the Durand-Kerner (D-K) algorithm [27] is selected to solve the problem. For the quaternary equation with one variable, the complex numbers P, Q, R , and S are assumed to be the roots of equation (7). After decomposition, the following equation can be obtained:

$$f(x) = (x-P)(x-Q)(x-R)(x-S). \quad (7)$$

For all of the X 's, we can factor out the values from this.

$$P = x - \frac{f(x)}{(x-Q)(x-R)(x-S)}. \quad (8)$$

The corresponding fixed-point iteration formula is as follows:

$$x = x_0 + \frac{f(x_0)}{(x_0-Q)(x_0-R)(x_0-S)}, \quad (9)$$

where $x_0 \neq Q, R$, and S .

If the approximate points of Q, R , and S are substituted, that is, $q \approx Q, p \approx P$, and $s \approx S$, then P remains a fixed point. Therefore, the general form of the iterative formula for finding the root can be expressed as follows:

$$x_{k+1} = x_k + \frac{f(x_k)}{(x_k-q)(x_k-r)(x_k-s)}. \quad (10)$$

Through iteration, the values of P, Q, R , and S can be obtained. In practical application, we only need the values that meet the requirements, so the values of P, Q, R , and S

need to be filtered according to the definition of zero-crossing point:

- (1) The zero-crossing point must be a real root, and its imaginary part is 0
- (2) The value of the root of the zero crossing should be within the axis interval of x where the maximum and minimum values are located

If the obtained values do not meet the above conditions, it is deemed that the calculation of subpixel edge points at this point has failed; if there is a root that meets the above conditions, take it as the coordinate of the subpixel edge point in the ROI area coordinate system and map it to the image coordinate system through the following equation:

$$x_n = x_o - l + x_s + x_r - 1, \quad (11)$$

where x_n is the value of the subpixel point under the global coordinate of the image, x_o is the value of the initial test pixel point under the global coordinate of the image, x_s is the value of the first fitting-point pair in the ROI, x_r is the value of the subpixel point under the ROI coordinate system, and l is the extended length of the pixel's value forward.

Through equation (6), we can find that at least five different points must be fitted to determine the constant value in the expression. In order to make the change trend of the fitting curve closer to the reality, first select the two points where the maximum and minimum values of the second-order difference value are located in the second-order difference ROI area and then expand from these two points; the two points near each of these two points are selected to form a group of six-point fitting-point pairs. However, in the actual calculation, only one point may exist between the maximum and minimum values. Therefore, the constant value solution methods can be divided into five- and six-point methods.

For five-point methods, assume that the position of the fitting point in the ROI is defined as the axis data of x -axis, and its corresponding second-order difference value is the axis data of the y -axis. It can be converted into the following matrix expression after being placed in the point pair:

$$\begin{bmatrix} x_1^4 & x_1^3 & x_1^2 & x_1 & 1 \\ x_2^4 & x_2^3 & x_2^2 & x_2 & 1 \\ x_3^4 & x_3^3 & x_3^2 & x_3 & 1 \\ x_4^4 & x_4^3 & x_4^2 & x_4 & 1 \\ x_5^4 & x_5^3 & x_5^2 & x_5 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}. \quad (12)$$

For convenience, the above equation can be simplified as $AX = B$. As the A matrix is a full-rank matrix, the matrix is invertible; thus, the expression can be solved by the following formula:

$$X = A^{-1}B, \quad (13)$$

where X is the constant value to be solved.

Similarly, the matrix of the six-point fitting method can be expressed by a matrix similar to the matrix of the five-point fitting method in the equation (12). For convenience, the matrix solution equation of the six-point fitting method can be expressed as $AX = B$. As the A matrix is not a square matrix, it cannot be solved simply in the form of an inverse matrix, but the numerical solution of the Eu matrix can be obtained with the least squares estimation method. Therefore, the value of Eu can be obtained as

$$X = (A^T A)^{-1} A^T B. \quad (14)$$

At this point, the expression of the quartic curve can be obtained using the five- or six-point fitting method.

2.3. Algorithm Framework. The whole subpixel algorithm is shown as Algorithm1.

Finally, the X coordinates of subpixel edge points in the original image x_n can be obtained by the above algorithm, combined with the fixed measurement position height, that is, the fixed Y coordinates y_m , and we can get subpixel point (x_n, y_m) .

3. Results and Discussion

To verify the correctness and accuracy of the algorithm, experimental analysis is conducted by using actual tool photos and comparing the Canny operator, binary algorithm, and subpixel algorithm [28] proposed by Rafael Grompone von Gioi.

3.1. Experimental Environment. The algorithm is tested in a laboratory environment and in Shanghai, Xuzhou, Suzhou, and other processing plants. The test results show that a Haas machine tool and tools for an actual shooting machine environment are used in the test.

A photo of the tools is presented in Figure 5.

The preprocessed image is presented in Figure 6.

3.2. Verification

3.2.1. Five-Point Method. Suppose that the 112th-row pixel of the tool image is measured at a fixed point, and one of the binary contour edge coordinates is obtained after binarization. The ROI area is obtained by extending 10 pixels forward and backward from this point, and the gray value of the ROI is as follows:

$$[248, 247, 246, 245, 243, 239, 225, 192, 138, 82, 44, 23, 14, 9, 7, 7, 7, 6, 6, 6]. \quad (15)$$

Input: image I_F , measure position's Y coordinate y_m

Output: Subpixel point's value under the global coordinate of the image x_n

Transform input image I_F into reverse binary image by equation (1)

Use Satoshi Suzuki's algorithm [25] and Freeman chain code [26] to find the longest Contour in image as Tool Contour

Sort all points in Tool contour, perform binary search for Tool contour to find a row whose y coordinate value is equal to y_m , and get two points where the maximum and minimum values of X coordinate are located

Expand the two points to get ROI area according to the rules mentioned in the previous theoretical part

Calculate ROI region second-order difference point set by equations (3)–(5)

According to the different number of points selected, use the five- or six-point fitting method to obtain the fitting curve expression and calculate five constants of quartic curve by equations (12), (13), or (14)

Calculate roots of quartic curve by equations (6)–(10)

Obtain the qualified root by judging the rationality of roots, and take it as the X coordinate of the zero-crossing point in the ROI area

Calculate x_n by equation (11)

ALGORITHM 1: Subpixel algorithm.

FIGURE 5: Flat bottom knife picture.

Thus, it can be determined that the gray change type is a falling-edge, and the value of its corresponding position after the second-order difference is as follows:

$$[0, 0, 0, -1, -2, -10, -19, -21, -2, 18, 17, 12, 4, 3, 2, 0, -1, 1, 0, 0]. \quad (16)$$

The maximum value of the second-order difference is 18 at the 10th position, the minimum value is -21 at the 8th position, and the number of intermediate points is 1. Therefore, a point is taken from both sides of the maximum

value to form a five-point sequence $[-19, -21, -2, 18, 17]$. Next, the fitting coordinate system is established according to the position and second-order difference value, and the five-point sequence is transformed into Table 1.

According to the sequence of fitting points, the coefficient matrix of the quartic curve can be obtained by solving the equations with the least squares method.

$$[-0.0833, -2.5, 27.583, -66, 22]^T. \quad (17)$$

Therefore, the quartic curve expression is as follows:

FIGURE 6: Preprocessed image.

TABLE 1: Coordinates of fitting points.

Point	$P1$	$P2$	$P3$	$P4$	$P5$
Coordinate	(1, -19)	(2, -21)	(3, -2)	(4, 18)	(5, 17)

$$f(x) = -0.00833x^4 - 2.5x^3 + 27.583x^2 - 66x + 22. \quad (18)$$

The position relationship between the fitting-point and quartic curve expression is shown in Figure 7.

$$[254, 253, 247, 238, 231, 226, 219, 199, 160, 107, 59, 30, 17, 12, 10, 9, 9, 8, 8, 7]. \quad (19)$$

And the value of its corresponding position after the second-order difference is as follows:

$$[0, -5, -3, 2, 2, -2, -13, -19, -14, 5, 19, 16, 8, 3, 1, 1, -1, 1, -1, 1]. \quad (20)$$

In the fitting coordinate system, the six-point sequence is transformed into Table 2.

According to the sequence of fitting points, the coefficient matrix of the quartic curve can be obtained by solving the equations with the least squares method.

$$[3.167, -17.505, -0.785, 2.551, -0.3125]^T. \quad (21)$$

The quartic curve expression is as follows:

$$f(x) = 3.167x^4 - 17.505x^3 - 0.785x^2 + 2.551x - 0.3125. \quad (22)$$

The D-K algorithm [26] is used to solve the root of the quartic equation. The numerical results of the iterative process are as shown in Table 3.

After 16 iterations, the numerical solution is obtained. According to the screening conditions of the zero-crossing point, the fourth root of the quartic equation solution is

By solving the matrix equation and according to the coordinate transformation formula, the subpixel coordinates of a single point can be updated to (227.087, 112).

3.2.2. Six-Point Method. In the early stage of the procedure, similar to the five-point method, the pixel is selected as [91, 110].

The ROI is obtained by extending 10 pixels forward and backward from this point. The gray value of the ROI is as follows:

selected as the position of the zero-crossing point. According to the coordinate transformation formula, the coordinates of a single-point subpixel can be updated to (89.793, 110), and its position is shown in Figure 8. On the right side is the binary edge point, and on the left side is the subpixel edge point calculated by the algorithm.

3.3. Comparison. In order to verify the accuracy of this algorithm, we also carried out many comparative experiments and selected some common image edge algorithms to compare with the algorithms in this paper. In the experiment, different algorithms are used to find the image edge points in the specified area of the same tool picture. Some experimental results are as follows.

A total of 16 pairs of points are selected for comparison, and the area is shown as Figure 9.

The image Y-axis is 195–210 pixels, and the left figure's X-axis is 220–233 pixels, and the right figure's X-axis is 779–791 pixels, respectively.

Binarization, the Canny operator, the subpixel algorithm proposed by Rafael Grompone von Gioi [28], and the algorithm in this study are used to process the region. The data obtained are shown in Table 4.

FIGURE 7: Fitting curve.

TABLE 2: Coordinates of fitting points.

Point	$P1$	$P2$	$P3$	$P4$	$P5$	$P6$
Coordinate	(1, -13)	(2, -19)	(3, -14)	(4, 5)	(5, 19)	(6, 16)

TABLE 3: Iterative process of root.

Number of iterations	P	Q	R	S
1	$-19.372 - 2.547i$	$1.512 + 5.553i$	$-8.754 + 6.415i$	$-3.407 - 4.349i$
2	$73.797 - 61.853i$	$1.290 + 6.074i$	$-1.691 - 7.623i$	$-2.230 - 4.324i$
3	$35.058 - 8.776i$	$-0.755 + 11.781i$	$-11.648 + 1.478i$	$-2.4134 - 5.867i$
...	—	—	—	—
15	$26.58 - 6.136e^{-91}i$	$6.537 - 1.473e^{-90}i$	$0.525 + 0i$	$3.461 - 2.286i$
16	$26.588 + 0i$	$6.537 + 0i$	$0.525 + 0i$	$3.461 + 0i$

TABLE 4: Comparative data.

Binarization	Canny	R-subpixel	Text subpixel
[229, 195]	[226, 195]	[226.243, 195]	[226.272, 195]
[229, 196]	[227, 196]	[225.884, 196]	[225.803, 196]
[227, 197]	[226, 197]	[225.334, 197]	[225.145, 197]
[226, 198]	[225, 198]	[224.87, 198]	[224.778, 198]
[226, 199]	[225, 199]	[224.521, 199]	[224.514, 199]
[226, 200]	[224, 200]	[224.313, 200]	[224.316, 200]
[226, 201]	[224, 201]	[224.228, 201]	[224.237, 201]
[226, 202]	[224, 202]	[224.243, 202]	[224.214, 202]
[226, 203]	[224, 203]	[224.418, 203]	[224.398, 203]
[226, 204]	[225, 204]	[224.746, 204]	[224.694, 204]
[227, 205]	[225, 205]	[225.231, 205]	[225.046, 205]
[229, 206]	[226, 206]	[225.949, 206]	[225.889, 206]
[229, 207]	[227, 207]	[226.686, 207]	[226.445, 207]
[229, 208]	[227, 208]	[227.187, 208]	[227.239, 208]
[229, 209]	[227, 209]	[227.33, 209]	[227.306, 209]
[229, 210]	[227, 210]	[227.362, 210]	[227.328, 210]
[783, 195]	[785, 195]	[784.971, 195]	[784.414, 195]
[783, 196]	[785, 196]	[785, 196]	[784.475, 196]
[783, 197]	[785, 187]	[785, 197]	[784.515, 197]
[783, 198]	[785, 198]	[785, 198]	[784.454, 198]

TABLE 4: Continued.

Binarization	Canny	R-subpixel	Text subpixel
[783, 199]	[785, 199]	[784.986, 199]	[784.487, 199]
[783, 200]	[785, 200]	[784.986, 200]	[784.453, 200]
[783, 201]	[785, 201]	[784.985, 201]	[784.515, 201]
[783, 202]	[785, 202]	[784.971, 202]	[784.389, 202]
[783, 203]	[785, 203]	[784.971, 203]	[785.042, 203]
[783, 204]	[785, 204]	[785, 204]	[784.47, 204]
[783, 205]	[785, 205]	[784.986, 205]	[784.505, 205]
[783, 206]	[785, 206]	[784.971, 206]	[784.51, 206]
[783, 207]	[785, 207]	[784.985, 207]	[784.504, 207]
[783, 208]	[785, 208]	[785, 208]	[784.504, 208]
[783, 209]	[785, 209]	[785, 209]	[784.504, 209]
[783, 210]	[785, 210]	[784.987, 210]	[784.454, 210]

FIGURE 8: Subpixel edge points.

FIGURE 9: The test figure.

Through the values of edge points obtained by different algorithms in Table 4, combined with the pixel coordinates in Figure 9, we can draw the following conclusions:

- (1) Binarization and Canny operator can only feedback pixel-level image edge point coordinates. Both this algorithm and R-algorithm can provide subpixel level edge point coordinates.
- (2) Combined with the pixel coordinates in Figure 9, we can find that the image edge effect obtained by binarization is the worst, and both the algorithm in this paper and R-algorithm can provide better image subpixel edge points.

4. Conclusions

The experimental data can prove that this algorithm can indeed improve the pixel-level edge points to subpixel level edge points. The subpixel points obtained by the proposed method demonstrate high accuracy and are tested in a tool machining environment, which can help increase the accuracy of tool diameter measurement.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was supported by National Natural Science Foundation of China (Grant no. 51205243).

References

- [1] W. Liu, Y. Zhang, and P. Gao, "A method of sub-pixel center extraction for laser strip combined with hierarchical processing," *Infrared And Laser Engineering*, vol. 46, no. 10, pp. 232–239, 2017.
- [2] L. Gao, Y. He, and B. Zeng, "Research on spot location algorithm of triangular ranging LiD-AR," *Chinese Journal of Sensors and Actuators*, vol. 29, no. 11, pp. 1692–1697, 2016.
- [3] W. Feng, J. Zhang, and C. Hu, "A novel saliency detection method for wild animal monitoring images with WMSN," *Journal of Sensors*, vol. 2018, Article ID 3238140, 11 pages, 2018.
- [4] H. Zhang, Z. Wen, and Y. Liu, "Edge detection from RGB-D image based on structured forests," *Journal of Sensors*, vol. 2016, Article ID 5328130, 10 pages, 2016.
- [5] K. Zhang, Y. Zhang, P. Wang, Y. Tian, and J. Yang, "An improved sobel edge algorithm and FPGA implementation," *Procedia Computer Science*, vol. 131, pp. 243–248, 2018.
- [6] H. Li and C. Ma, "Marine ship image edge detection based on improved Scharr algorithm," *Ship Electronic Engineering*, vol. 39, no. 3, pp. 103–106, 2019.
- [7] H. Xie, S. Wang, and X. Wu, "Image fusion algorithm based on laplacian operator with wavelet Transform," *Information Technology*, vol. 40, no. 11, pp. 114–117, 2016.
- [8] Z. Duan and Z. Shan, "Study on the image edge extracting algorithm for machine vision measuring system," *Modular Machine Tool & Automatic Manufacturing Technique*, no. 12, pp. 12–14, 2017.
- [9] S. Zhang, "Edge extraction algorithm based on quantum flexible representation," *Computer Engineering and Design*, vol. 39, no. 6, pp. 1697–1703, 2018.
- [10] K. Chen, "An algorithm for optimized edge detection based on image mask and Hit-miss transform," *Journal of Xi'an University of Technology*, vol. 34, no. 1, pp. 116–121, 2018.
- [11] P. Wu, H. Xu, and W. Song, "Sub-pixel edge detection algorithm based on nonlinear fourth-order image interpolation," *Journal of Harbin Engineering University*, vol. 2, pp. 243–247, 2016.
- [12] F. Da and H. Zhang, "Sub-pixel edge detection based on an improved moment," *Image and Vision Computing*, vol. 28, no. 12, pp. 1645–1658, 2010.
- [13] Y. Cai, L. Ye, and D. Sun, "Non-contact measurement of external thread based on improved zernike moment sub-pixel edge extraction algorithm," *Journal of Shanghai Jiaotong University*, vol. 48, no. 10, pp. 1468–1472, 2014.
- [14] W. Zhu and M. Liu, "Edge detection of small module gear sub-pixel image based on neighborhood characteristic analysis," *Chinese Journal of Scientific Instrument*, vol. 3, pp. 148–156, 2018.
- [15] A. Fabijanska, "Gaussian-based approach to sub-pixel detection of blurred and unsharp edges," in *Proceedings of the Computer Science and Information Systems*, pp. 641–650, Warsaw, Poland, September 2014.
- [16] D. Han, Y. Li, and Y. Wu, "Sub-pixel edge detection algorithm based on gaussian fitting," *Computer Applications and Software*, vol. 6, pp. 210–213, 2018.
- [17] W. Yiquan, "Sub pixel image edge detection algorithm based on Franklin moment," *Chinese Journal of Scientific Instrument*, vol. 40, no. 5, pp. 221–229, 2019.
- [18] X. Xie, S. Ge, M. Xie, F. Hu, and N. Jiang, "An improved industrial sub-pixel edge detection algorithm based on coarse and precise location," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 5, pp. 2061–2070, 2020.
- [19] B. M. Kumar and M. M. Ratnam, "Machine vision method for non-contact measurement of surface roughness of a rotating workpiece," *Sensor Review*, vol. 35, no. 1, pp. 10–19, 2015.
- [20] J. Zhou and J. Yu, "On-line measurement of tool wear in machining based on machine vision," *Journal of Shanghai Jiaotong University*, vol. 55, no. 6, 2021.
- [21] B. Wu, "Laser image sub-pixel edge detection based on Gabor filtering and mathematical morphology," *Laser Journal*, vol. 41, no. 11, pp. 110–114, 2020.
- [22] Y. Ai, M. Zhao, and Q. Zhang, "Corner detection based on sub-pixel edge checkerboard," *Journal of Xi'an University of Technology*, vol. 35, no. 3, pp. 333–337, 2019.
- [23] M. Versaci, S. Calcagno, and F. C. Morabito, "Fuzzy geometrical approach based on unit hyper-cubes for image contrast enhancement," in *Proceedings of the 2015 IEEE International Conference On Signal And Image Processing Applications (ICSIPA)*, pp. 488–493, Kuala Lumpur, Malaysia, October 2015.
- [24] M. Alruwaili, M. H. Siddiqi, and M. A. Javed, "A robust clustering algorithm using spatial fuzzy C-means for brain MR images," *Egyptian Informatics Journal*, vol. 21, no. 1, pp. 51–66, 2020.
- [25] S. Suzuki, "Topological structural analysis of digitized binary images by border following," *Computer Vision Graphics & Image Processing*, vol. 30, 1985.
- [26] Y. Liu, "Research on Freeman chain code compression algorithm," *Chinese Journal of Computers*, vol. 24, no. 12, pp. 1294–1298, 2001.
- [27] I. O. Kerner, "Ein Gesamtschrittverfahren zur Berechnung der Nullstellen von Polynomen," *Numerische Mathematik*, vol. 8, no. 3, pp. 290–294, 1966.
- [28] R. G. von Gioi and G. Randall, "A sub-pixel edge detector: an implementation of the canny/devernay algorithm," *Image Processing On Line*, vol. 7, pp. 286–310, 2017.