Research on O-ring Dimension Measurement Algorithm Based on Cubic Spline Interpolation

Hu Haibing * , **Xipeng Zheng, Jiajie Yin and Yueyan Wang**

Academy of Opto-Electric Technology, Special Display and Imaging Technology Innovation Center of Anhui Province, National Engineering Laboratory of Special Display Technology, State Key Laboratory of Advanced Display Technology, Collaborative Innovation Center of Advanced Display Technology, Anhui Key Laboratory of Advanced Imaging and Display Technology, Opto-Electric Display Industry Innovation Center, Anhui Province Key Laboratory of Measuring Theory and Precision Instrument, School of Instrument Science and Optoelectronics Engineering, Hefei University of Technology, Hefei 230009, China; 2018170033@mail.hfut.edu.cn (X.Z.); 2020110037@mail.hfut.edu.cn (J.Y.); 2020170101@mail.hfut.edu.cn (Y.W.)

* Correspondence: huhb@hfut.edu.cn; Tel.: +86-1811-097-2015

Abstract: Current O-ring dimension measurement algorithms based on machine vision are mainly whole-pixel level algorithms, which have the disadvantage of a low measurement accuracy. In order to improve the stability and accuracy of O-ring dimension measurement, a sub-pixel edge detection algorithm based on cubic spline interpolation is proposed for O-ring dimension measurement. After image pre-processing of the O-ring graphics, the whole-pixel-level O-ring edges are obtained by using a noise-resistant mathematical morphology method, and then the sub-pixel edge contours are obtained using a sub-pixel edge detection algorithm based on cubic spline interpolation. Finally, the edge curve is fitted with the least squares method to obtain its inner and outer diameter as well as the size of the wire diameter. The experimental data show that the algorithm has a mean square error of 4.8 μm for the outer diameter and 0.18 μm for the wire diameter. The outer diameter error is kept within ±100 μm and the wire diameter error can be kept within ±15 μm. Compared with the whole pixel algorithm, the measurement accuracy has been greatly improved.

Keywords: O-ring dimension measurement; morphology; cubic spline interpolation; sub-pixel edge detection

1. Introduction

The engine is the core component of a car, and the O-ring plays a very important role in the airtight structure of a car's engine; the quality of the seal directly affects the life and reliability of the engine's operation [1,2]. There are currently thousands of O-ring manufacturers in China, but the size of an O-ring still requires manual inspection; in the face of large-scale testing needs, this is not only time-consuming and labor-intensive, but also, due to contact measurement, the detection accuracy is not high and is susceptible to the effects of personal subjective factors [3]. Current O-ring dimension measurement algorithms are mainly based on algorithms at the whole pixel level, such as the Sobel operator, Canny operator, morphological algorithms, etc. However, an algorithm at the whole pixel level has the disadvantages of having poor image edge localization and low accuracy of size measurements [4].

To address these problems with whole-pixel algorithms, the authors of this paper design a sub-pixel O-ring dimension measurement algorithm based on cubic spline interpolation, and achieve sub-pixel edge detection of O-rings through algorithmic processing. Compared with the whole-pixel edge detection algorithm, it has the advantages of a high detection accuracy, good image edge localization and strong detection stability. It can effectively improve the accuracy of current O-ring dimension measurements.

2. Image Acquisition System Design

The principle of the machine vision-based O-ring inspection method is to obtain a clear image of the seal using an industrial camera, and then apply the technology of image processing to obtain the dimensions of the inner and outer diameters, as well as the wire diameter. Therefore, capturing a clear image of the seal is the first step in the implementation of the measurement system [5].

The structure of the image acquisition system is shown in Figure 1, which mainly consists of high-definition industrial camera, inspection platform, lighting system, etc. The O-ring is placed on the inspection platform, the front and back lighting LED lights are turned on, the seal is irradiated, and then a high-definition camera is used to get a clear image of the O-ring. In order to meet the accuracy requirements of O-ring dimension measurements, the camera has a resolution of 3664 × 2748 pixels. The lens is an M1614-MP2, which has focal length of 16 mm [6].
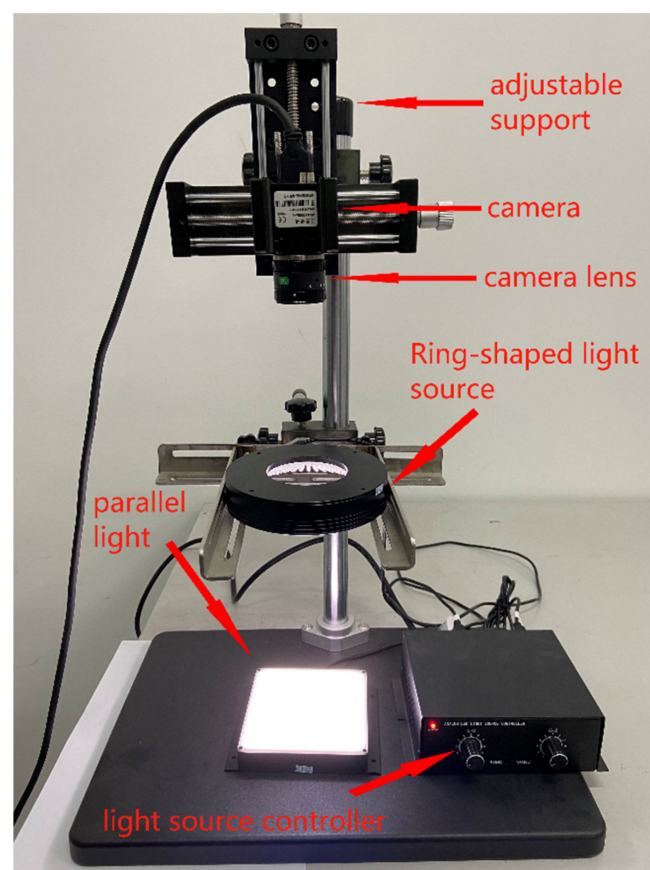
Figure 1. Physical structure of image acquisition system.

3. O-Ring Dimension Measurement Algorithm Design

The edge of an image is the most basic feature of an image, and the accuracy of the positioning accuracy of an image's edge points directly affects the results of dimensional measurements. Edge points are pixel points in the image where there is a step change or roof change in the grey scale of the surrounding pixels, i.e., where the derivative of the grey scale value is large or extremely large. Traditional edge detection algorithms run fast, but accuracy is hard to achieve. Therefore, a traditional algorithm can be used to coarsely locate the edge position of the image, and then the coarsely located edge points are interpolated by cubic spline interpolation to find the sub-pixel edge with the largest derivative of the grey value, which can enhance the edge pixel point from the pixel level to the sub-pixel level, thus greatly improving measurement accuracy. Finally, the outer diameter and wire diameter of the O-ring are obtained via least squares circle fitting [7].

3.1. Pixel-Level Edge Algorithm Based on Noise-Resistant Mathematical Morphology

A morphologically based whole-pixel edge detection algorithm that compared and analyzed through traditional edge detection algorithms was adopted. Morphology was described with the help of the language of set theory and was mainly applied in image processing, such as in image refinement, pixelation, edge detection, noise removal, etc. It is one of the most widely used techniques in image processing. The basic operations of morphology include erosion, expansion, open operations, and closed operations [8].

The corrosion operation on A using S is denoted as $A \ominus S = \{z | (S)_z \subseteq A\}$, corrosion can dissolve the boundary of an object. When selecting an appropriate size and shape of structural elements, it is possible to filter out all noise points that do not fully contain the structural elements. The expansion operation using S is denoted as $A \oplus s = \{z | (\hat{S})_z \cap A \neq \varphi\}$. In contrast to erosion, expansion expands the boundaries of objects, and is often used to bridge the same objects in an image that would otherwise be broken apart [9].

Both open and closed operations are compounded by corrosion and expansion; the open operation is corrosion followed by expansion, while the closed operation is expansion followed by corrosion. The open operation on A, using the structure element S, is denoted as $A \circ S$ and can be expressed as $A \circ S = (A \ominus S) \oplus S$. In general, the on operation makes the contour of the image smooth, breaks narrow connections and eliminates fine burrs, and the on-operation filters noise without significantly affecting the shape and contour of an object. The closed operation on A using structure element S is denoted as $A \bullet S = (A \oplus S) \ominus S$. The closed operation also smooths the contour, but in contrast to the open operation, it usually bridges narrow interruptions and fills small holes [10].

These four basic operations are the basis for morphological edge detection [11]. The basic idea is to select the appropriate elements to perform mathematical morphological operations on a target image, and subtract the result from the original image to obtain the edge of the image. Let the image be represented by A, S is a suitable structural element; first let A be corrupted by S and then let image A be the difference with it, then inner edge image D_1 is obtained.

$$D_1 = A - (A \ominus S) \tag{1}$$

Similarly, using the morphological expansion algorithm, we can obtain outer edge image D_2, which is expressed by:

$$D_2 = (A \oplus S) - A \tag{2}$$

Using the expansion and erosion algorithm, gradient edge image D is obtained.

$$D_3 = (A \oplus S) - (A \ominus S) \tag{3}$$

The three edge detection algorithms described above are composed simply by using a combination of erosion and expansion operations and do not achieve the desired level of accuracy for noise removal, especially Gaussian noise, or for edge localization. Through basic operations of mathematical morphology, it is known that the erosion and opened operations are good at suppressing peak noise in images, and the expansion and closed operations are good at suppressing trough noise in the image. Using this feature of mathematical morphology, traditional morphological edge detection algorithms can be improved [12].

Edge image D_4 is computed using a morphological edge detection algorithm that suppresses trough noise obtained using an expansion and closure operation, as shown in Equation (4), and edge image D_5 is computed using a morphological edge algorithm that suppresses peak noise obtained using an erosion and opening operation, as shown in Equation (5).

$$D_4 = (A \oplus S) - (A \bullet S) \tag{4}$$

$$D_5 = (A \circ S) - (A \ominus S) \tag{5}$$

The above morphological operations are combined for the suppression characteristics of different noises. The four morphological operations are combined and applied to edge detection to obtain a noise-resistant mathematical morphological edge detection algorithm, and can be expressed as Equation (6).

$$R = (A \circ S) \oplus S - (A \bullet S) \Theta S \tag{6}$$

Using mathematical morphology for image processing, in addition to the flexible combination of the four basic operations, different structural elements can also have an impact on edge detection [13]. Moreover, for the extraction of edges using expansion etching, a structural element will only significantly work on edges that are perpendicular to it. Therefore, we used a set of four-direction structure elements as morphological structure elements, as shown in Figure 2.

| (a) | (b) | (c) | (d) |
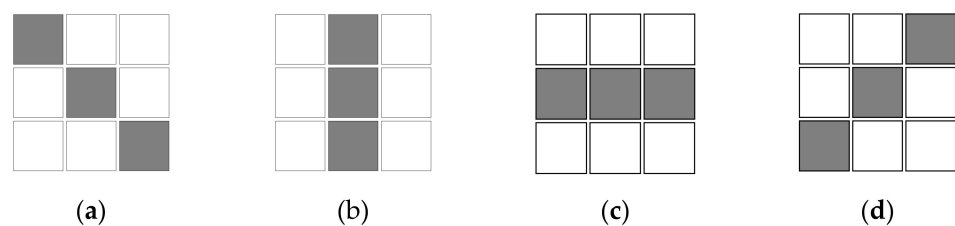
Figure 2. Schematic diagram of four-direction structural elements. (**a**) the unit matrix; (**b**) the matrix with 0 except the second column as 1; (**c**) the matrix with 0 except the second row as 1; (**d**) the sub-diagonal matrix as 1 and the rest as 0.

(a), (b), (c), and (d) are used as structural elements, respectively, in the noise-resistant mathematical morphology edge detection algorithm. After four structure elements are calculated, the four structures are summed with average weights to obtain the final processed image. The expression for this is Equation (7).

$$R = (R_1 + R_2 + R_3 + R_4)/4 \tag{7}$$

The Sobel operator, Canny operator and noise-resistant mathematical morphology edge detection algorithm were used to detect the edge of an O-ring, respectively, and the obtained results are shown in Figure 3. It can be seen that the obtained O-ring edge image using the algorithm in this paper is more delicate compared with other classical algorithms, the extraction of the edge is more accurate, and the details of the edge can be better retained. This vision measurement system uses a noise-resistant mathematical morphology edge detection algorithm for whole-pixel edge extraction of O-ring seals.
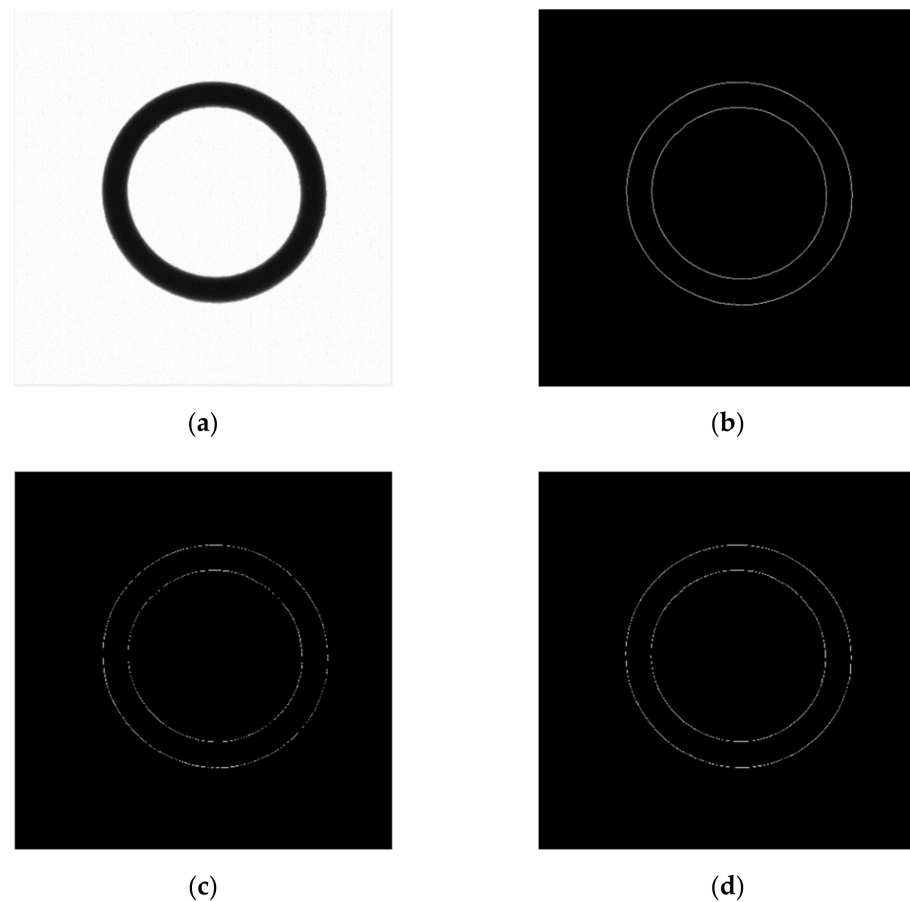
Figure 3. Comparison of noise-resistant mathematical morphological edge detection algorithm and whole-pixel edge detection algorithm. (**a**) Original image; (**b**) noise-resistant mathematical morphology edge detection results; (**c**) the Sobel operator edge detection result; (**d**) plot of the edge detection results of the Canny operator.

3.2. Sub-Pixel Level Edge Measurement Algorithm Based on Cubic Spline Interpolation

As vision measurement becomes more and more widely used in industrial production, the accuracy of measurements is becoming more and more important to people [14]. Positioning edge locations to pixel-level accuracy is also increasingly unsatisfactory. The sub-pixel edge detection algorithm is used to first coarsely localize the image edges to the pixel level using an integer pixel edge detection algorithm, and then the pixel, the basic unit, is subdivided by analyzing the grey values around the integer pixel so that the edges are localized to the sub-pixel level [15].

In order to accurately determine the interpolation nodes, we need to know the direction of the grey gradient of the whole-pixel edge points of an image [16]. A whole pixel point is used first and a 7×7 rectangular area point is drawn around it, as shown in Figure 4. In the figure, a, b, c and d represent one grey gradient direction, and the angle between two adjacent grey gradient directions is $45°$ [17].

The absolute value of the difference between the greyscale values of adjacent pixels in the four gradient directions a, b, c and d are summed, and the sum of the absolute values E_i is the maximum of the greyscale gradient direction of the pixels [7,18]. The calculation formula is as follows:

$$E_i = \sum_{j=1}^{7} \left| p_{j+1} - p_j \right|, (i = a, b, c, d) \tag{8}$$
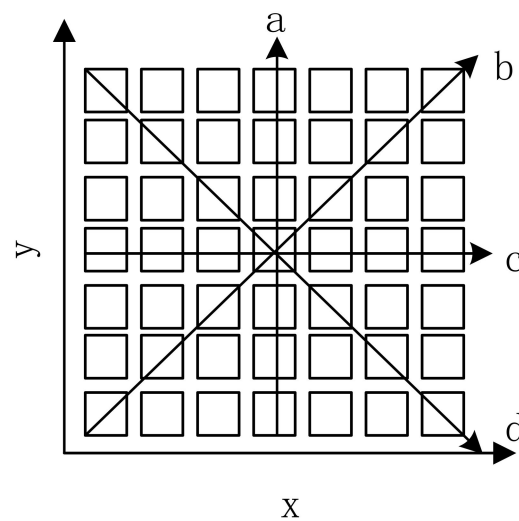
Figure 4. Schematic diagram of gradient direction.

When the gradient direction is obtained, the greyscale and coordinate values of the original image in the gradient direction are taken out and used in the following cubic spline interpolation [18].

Cubic spline interpolation is a set of nodes, $a = x_0 < x_1 < \cdots < x_N < x_{N+1} = b$ on $[a, b]$, each node (x_i) corresponding to a data point $(x_i, f(x_i))$, using these data points to construct a cubic spline function $S(x)$ and the function needs to satisfy the following three conditions:

(1) $S(x) \in C^2[a, b]$, i.e., $S(x)$,$S'(x)$,$S''(x)$ is continuous on $[a, b]$;
(2) $S(x)$ is a cubic polynomial over each small interval $[x_i, x_{i+1}]$;
(3) $S(x_i) = f(x_i) = y_i (i = 0, 1, \cdots, N+1)$

Let $M_j = S''(x_j), j = 0, 1, \cdots, N+1$, since $S(x)$ is a sectional cubic polynomial, $S''(x)$ is a primary polynomial over all of the subintervals $[x_{j-1}, x_j]$. Denote $h_j = x_j - x_{j-1}, j = 1, 2, \cdots, N+1$. Then on $[x_{j-1}, x_j]$ there is:

$$S''(x) = M_{j-1}\frac{x_j - x}{h_j} + M_j\frac{x - x_{j-1}}{h_j}, x \in [x_{j-1}, x_j] \tag{9}$$

where $h_j = x_j - x_{j-1}$.

Integrating Equation (9) twice over $[x_{j-1}, x_j]$ yields the expressions for $S(x)$ and $S'(x)$.

$$S'(x) = -\frac{M_{j-1}}{2h_j}(x_j - x)^2 + \frac{M_j}{2h_j}(x - x_{j-1})^2 + C_1 \tag{10}$$

$$S(x) = \frac{M_{j-1}}{6h_j}(x_j - x)^3 + \frac{M_j}{6h_j}(x - x_{j-1})^3 + C_1 x + C_2 \tag{11}$$

where C_1 and C_2 are constants, the interpolation condition is $S(x_{j-1}) = y_{j-1}$, and $S(x_j) = y_j$ can be used to find the integration constant as:

$$C_1 = \frac{h_j}{6}(M_{j-1} - M_j) + \frac{1}{h_j}(y_j - y_{j-1})$$

$$C_2 = \frac{h_j}{6}(x_j M_{j-1} - x_{j-1} M_j) + \frac{1}{h_j}(x_j y_{j-1} - x_{j-1} y_j)$$

Substituting the integration constants into Equation (11) and organizing it gives

$$S(x) = \frac{M_{j-1}}{6h_j}(x_j - x)^3 + \frac{M_j}{6h_j}(x - x_{j-1})^3 + \left(y_{j-1} - \frac{M_{j-1}h_j^2}{6}\right)\frac{x_j - x}{h_j} \\ + \left(y_j - \frac{M_j h_j^2}{6}\right)\frac{x - x_{j-1}}{h_j} \tag{12}$$

where $x_{j-1} \le x \le x_j, h_j = x_j - x_{j-1}, j = 1, 2, \cdots, N+1$.

Then, finding the derivative of $S(x)$ to obtain,

$$S'(x) = -\frac{M_{j-1}}{2h_j}(x_j - x)^2 + \frac{M_j}{2h_j}(x - x_{j-1})^2 + \frac{y_j - y_{j-1}}{h_j} - \frac{M_j - M_{j-1}}{6}h_j \tag{13}$$

With Equation (12) we can see that the key to finding $S(x)$ is to determine $M_j = S''(x), j = 1, 2, \cdots, N+1$.

Since node $x_j(j = 1, 2, \cdots, N)$ is continuously differentiable at $S'(x)$, then,

$$S'(x_j - 0) = S'(x_j + 0), j = 1, 2, \cdots, N \tag{14}$$

From Equation (14) we get:

$$\frac{h_j}{6}M_{j-1} + \frac{h_j + h_{j+1}}{3}M_j + \frac{h_{j+1}}{6}M_{j+1} = \frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}, j = 1, 2, \cdots, N \tag{15}$$

Then, the above equation can be reduced to:

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = d_j, j = 1, 2, \cdots, N \tag{16}$$

where $\mu_j = \frac{h_j}{h_j + h_{j+1}}, \lambda_j = 1 - \mu_j, d_j = \frac{6}{h_j + h_{j+1}}\left(\frac{y_{j+1} - y_j}{h_{j+1}} - \frac{y_j - y_{j-1}}{h_j}\right)$.

Equation (16) is a linear system of equations with respect to $M_0, M_1, \cdots, M_{N+1}$. To determine these $N + 2$ unknowns, two additional boundary conditions are required.

Since we do not know the values of the first and second order derivatives of $S(x)$ on boundaries x_0 and x_{N+1}, here we choose to use the natural boundary conditions, i.e., $S''(x_0) = S''(x_{N+1}) = 0, M_0 = M_{N+1} = 0$. Then M_j is written in the form of a matrix:

$$\begin{bmatrix} 2 & \lambda_1 & & & & & \\ \mu_2 & 2 & \lambda_2 & & & & \\ & \mu_3 & 2 & \lambda_3 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \mu_{N-2} & 2 & \lambda_{N-2} & \\ & & & & \mu_{N-1} & 2 & \lambda_{N-1} \\ & & & & & \mu_N & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ \vdots \\ M_{N-2} \\ M_{N-1} \\ M_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{N-2} \\ d_{N-1} \\ d_N \end{bmatrix} \tag{17}$$

With the coordinate values in the direction of the gradient as the independent variable, i.e., $x_j(j = 1, 2, \ldots, 7)$, as a function of the greyscale value of the corresponding pixel, i.e., $f(x_j) = p_j$, cubic spline interpolation is performed to obtain the cubic spline function $S(x)$ [19]. Then, the second order derivative of its third spline function $S''(x)$ is found.

Since the grey value in the gradient direction must be monotonically decreasing and the sub-pixel edge points are the points where the grey value changes most significantly in the gradient direction, the sub-pixel edge points must follow two principles [20]:

(1) $S''(x_0) = 0$ and the positive and negative signs on both sides of $S''(x_0)$ are not equal.
(2) If there are multiple inflection points for the second order derivative $S''(x)$, choose the minimal value point on the first order derivative.

Using the above theoretical calculation, the sub-pixel edge point coordinate $x_{j\max}$ in the direction of the gray gradient is obtained. The coordinates of the point consist of the integer part $x_{j\max}^{pixel}$ and the fractional part $x_{j\max}^{subpixel}$, using the following Equation (18) [21].

$$x_{j\max} = x_{j\max}^{pixel} + x_{j\max}^{subpixel} \tag{18}$$

where the integer part $x_{j\max}^{pixel}$ of $x_{j\max}$ is the pixel-level coordinate of the edge point, and the fractional part $x_{j\max}^{subpixel}$ of $x_{j\max}$ is the sub-pixel-level coordinate of the edge point. The coordinates of the corresponding pixel levels (x, y) in the greyscale map can be determined from the integer part of $x_{j\max}$. According to the direction of the gradient, the calculation formulas of sub-pixel coordinates $(x^{subpixel}, y^{subpixel})$ are as follows [22]:

If the direction of the grey gradient is the 'a' direction the expression is Equation (19):

$$\begin{cases} x^{subpixel} = x \\ y^{subpixel} = y + x_{j\max}^{subpixel} \end{cases} \tag{19}$$

If the direction of the grey gradient is the 'b, c' direction the expression is Equation (20):

$$\begin{cases} x^{subpixel} = x + x_{j\max}^{subpixel} \cos 45° \\ y^{subpixel} = y + x_{j\max}^{subpixel} \sin 45° \end{cases} \tag{20}$$

If the direction of the grey gradient is the 'd' direction the expression is Equation (21):

$$\begin{cases} x^{subpixel} = x + x_{j\max}^{subpixel} \\ y^{subpixel} = y \end{cases} \tag{21}$$

A flow chart of the whole algorithm is shown in Figure 5.

As the theoretical part is rather complex, an example of the calculation of the sub-pixel coordinates of an O-ring contour point is presented. Taking the coordinates of a point (136,528) of the O-ring contour point as an example, the grey scale values of the pixels in a 7×7 rectangular area centered on this point is shown in Figure 6.

According to the gradient direction formula (Equation (8)), we can calculate that $E_a = 25, E_b = 181, E_c = 185, E_d = 186$. Therefore, the gradient direction is direction d.

The third spline interpolation is carried out for each point in direction d. The image of the third spline function $S(x)$, the first order derivative $S'(x)$, and the second order derivative $S''(x)$ are shown in Figure 7. The calculation gives $x_{j\max} = 1.8159$. Finally, the coordinates of the sub-pixel edge point are (134.5769, 530.5769) according to the formula for sub-pixel coordinates.

As shown in Figure 8, a sub-pixel edge point extraction map of the O-ring is shown. Since the sub-pixel edge points cannot be represented in the pixel map, the coordinate points are plotted to visually reflect the sub-pixel edge extraction results [23].
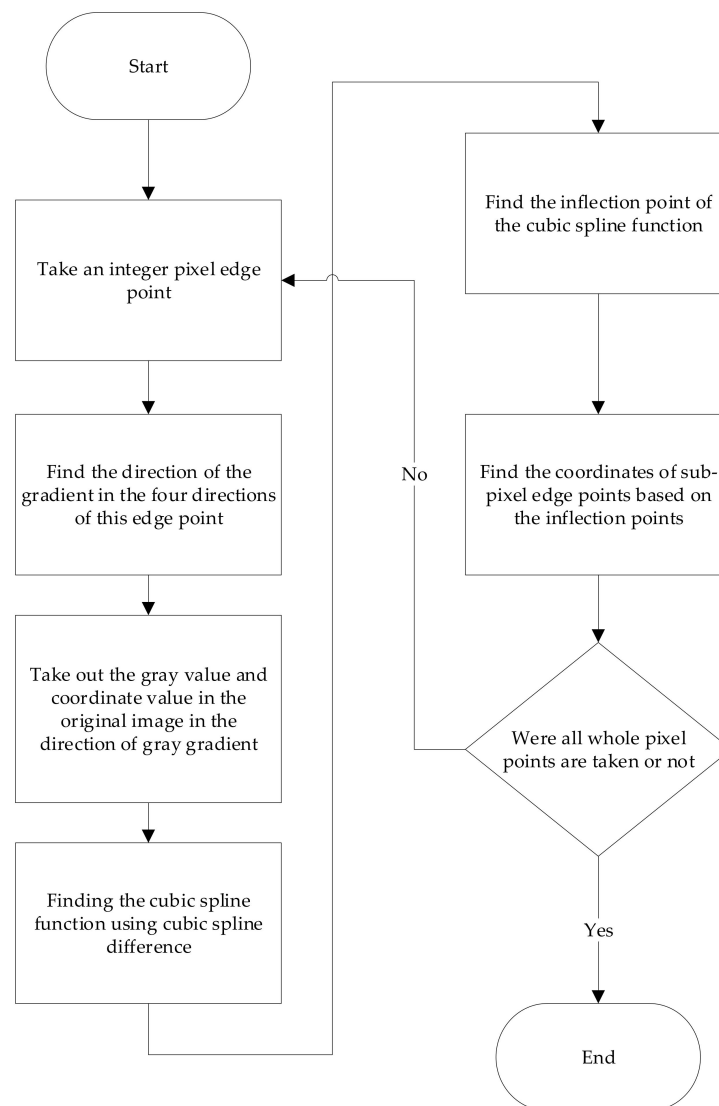
Figure 5. Algorithm flow.
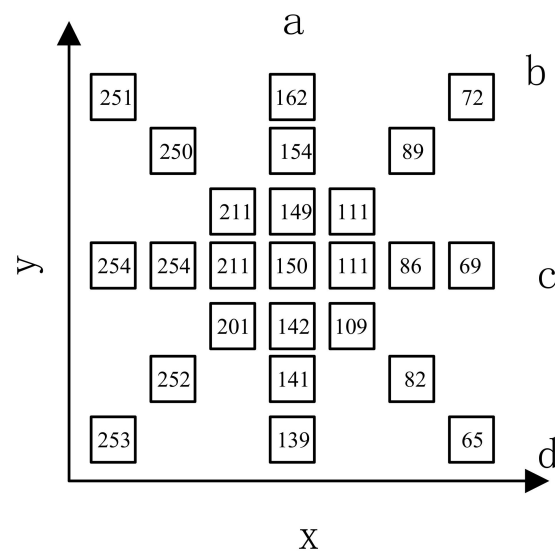
Figure 6. Grayscale values of pixels in each gradient direction.
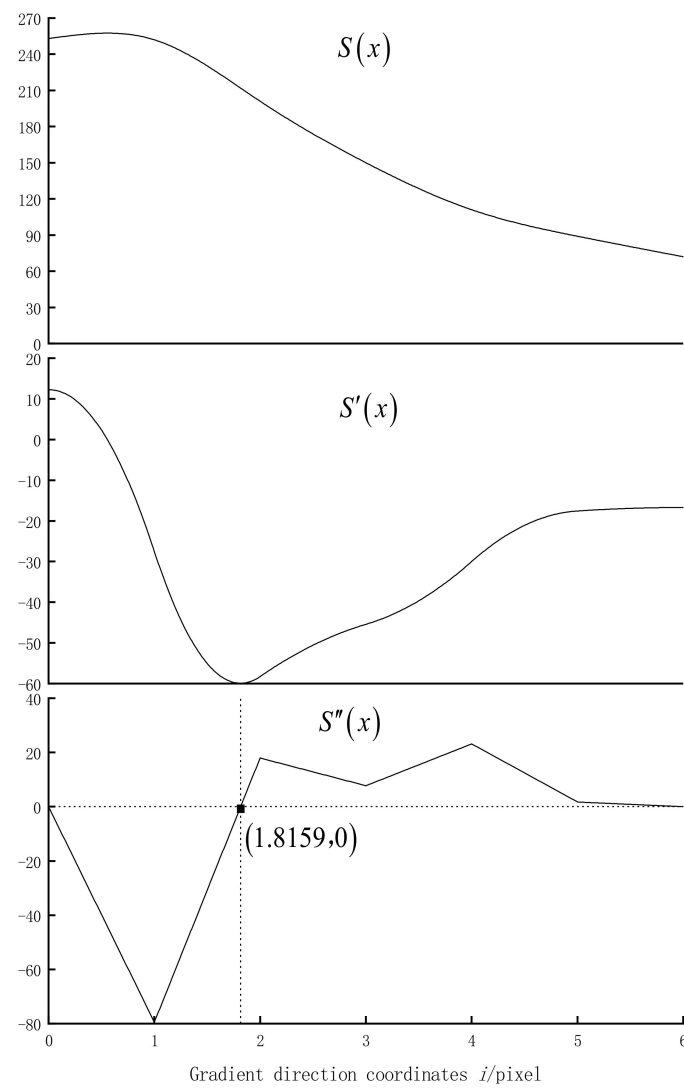
Figure 7. Image of a triple spline interpolation function and its first-order and second-order derivatives.
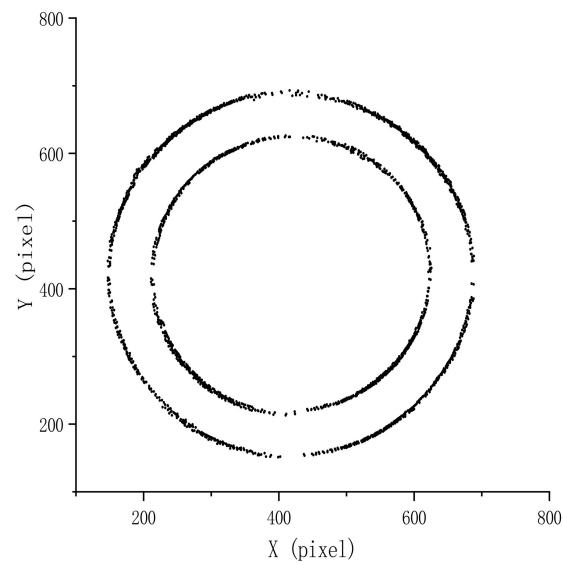
Figure 8. O-ring sub-pixel edge point.

3.3. Least Square Method for Circle Fitting

After obtaining its sub-pixel edge points, it is necessary to fit the sub-pixel edge points by least squares and then obtain the center point coordinates of the circle, as well as the radius diameter and other related parameters [24].

We know that the curve of a circle is given by Equation (22):

$$R^2 = (x - A)^2 + (y - B)^2 = x^2 - 2Ax + A^2 + y^2 - 2Bx + B^2 \tag{22}$$

Let $a = -2A, b = -2B, c = A^2 + B^2 - R^2$, then another expression for the equation of the circular curve can be obtained as Equation (23):

$$x^2 + y^2 + ax + by + c = 0 \tag{23}$$

The sub-pixel level edge coordinates of the O-ring are obtained by the cubic spline interpolation algorithm and are recorded as the sample $(X_i, Y_i), (i = 1, 2, 3, \ldots, N)$. Let the distance of the sample to the center of the circle be d_i. The expression is Equation (24):

$$d_i^2 = (X_i - A)^2 + (Y_i - B)^2 \tag{24}$$

The difference between the square of the distance from the sampling point (X_i, Y_i) to the fitted circle and the square of the radius of the fitted circle is Equation (25) [25].

$$\sigma_i = d_i^2 - R^2 = (X_i - A)^2 + (Y_i - B)^2 - R^2 = X_i^2 + Y_i^2 + aX_i + bY_i + c \tag{25}$$

Let $Q(a, b, c)$ be the sum of squares of σ_i, Equation (26),

$$Q(a, b, c) = \sum \sigma_i^2 = \sum \left(X_i^2 + Y_i^2 + aX_i + bY_i + c\right)^2 \tag{26}$$

So, it is only necessary to find a, b, c such that the value of $Q(a, b, c)$ is minimized. The final solution for a, b, c is [26],

$$a = \frac{HD - EG}{CG - D^2}, \quad b = \frac{HC - ED}{D^2 - GC},$$
$$c = \frac{\sum (X_i^2 + Y_i^2) + a \sum X_i + b \sum Y_i}{N}$$

Among them:

$$C = \left(N \sum X_i^2 - \sum X_i \sum X_i\right);$$
$$D = N \sum X_i Y_i - \sum X_i \sum Y_i;$$
$$E = N \sum X_i^3 + N \sum X_i Y_i^2 - \sum (X_i^2 + Y_i^2) \sum X_i;$$
$$G = N \sum Y_i^2 - \sum Y_i \sum Y_i;$$
$$H = N \sum Y_i^3 + N \sum Y_i X_i^2 - \sum (X_i^2 + Y_i^2) \sum Y_i;$$

The radius and the coordinates of the center point of the circle can be found by substituting the sample $(X_i, Y_i)(i = 1, 2, 3, \ldots, N)$ into the above least square method. As shown in Figure 9, Figure 9a shows the O-ring outer diameter circle fitting and Figure 9b shows the O-ring inner diameter circle fitting.

(**a**) (**b**)
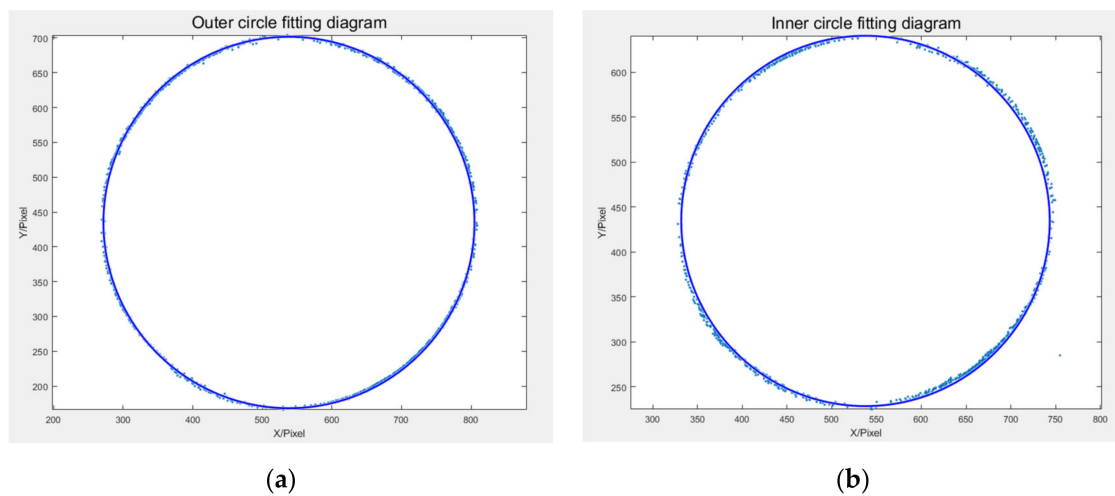
Figure 9. O-ring inner and outer circle fitting diagram. (**a**) Fitting of the O-ring outer diameter circle; (**b**) fitting of inner diameter circle of the O-ring.

4. Experiments and Discussion

4.1. Calibration Experiments

To get the actual size of the O-ring from the sub-pixel image, we have to know the actual size of each pixel point of the camera, so the pixel calibration must be done in advance [27].

We have selected a standard block with a diameter of 17 mm using a sub-pixel edge measurement algorithm based on third spline interpolation for 20 measurements to obtain the pixel diameter size of the fitted circle [28]. The image of standard gauge block is shown in Figure 10. Then its actual value is calculated as a ratio to the pixel length, here we call it K-value. k-value is calculated as follows:

$$K_{subpixel} = r/d_{pixel} \tag{27}$$

where r is the diameter of the block, d_{pixel} is the pixel diameter size of the block, and the $k_{subpixel}$ value is in $mm/pixel$.
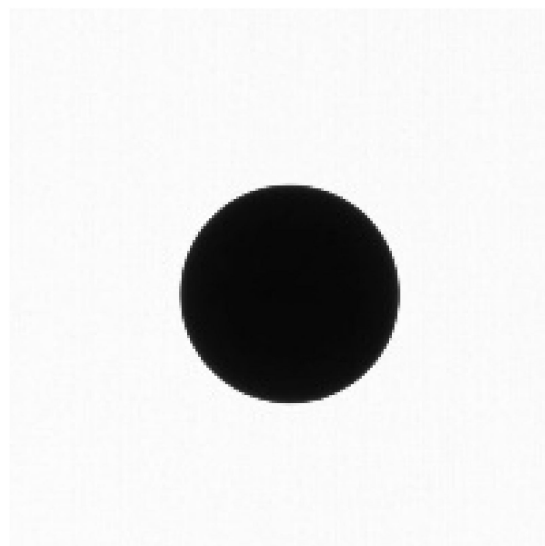
Figure 10. Images of the 17-mm standard gauge block.

The measurement results are shown in the data in Table 1, and we averaged the 20 values obtained so as to obtain a standard $k_{subpixel}$ value of 0.037623 $mm/pixel$.

Table 1. Table of sub-pixel values and corresponding K values for the 17-mm standard block.

No.	Sub-Pixel Values for 17 mm Standard Block Diameter (Pixel)	K Value (mm/Pixel)
1	451.9374	0.037616
2	451.956	0.037614
3	451.8846	0.03762
4	451.7348	0.037633
5	451.8928	0.03762
6	452.1574	0.037598
7	451.925	0.037617
8	451.5642	0.037647
9	451.9922	0.037611
10	451.8948	0.037619
11	451.702	0.037635
12	452.0722	0.037605
13	451.8054	0.037627
14	451.7554	0.037631
15	451.6664	0.037638
16	451.7822	0.037629
17	451.7796	0.037629
18	451.9564	0.037614
19	451.9132	0.037618
20	451.7258	0.037633

In order to compare with the results of the algorithm in this paper, the classical algorithms, the Sobel and Canny algorithms, were chosen here for a standard block with a diameter of 17 mm and also 20 measurements were carried out, consistent with the above principle, and obtained $k_{sobel} = 0.037202$ mm/pixel, $k_{canny} = 0.037305$ mm/pixel respectively.

4.2. Measurement Experiments

In order to evaluate the accuracy of the algorithm measurement, a qualified O-ring with an outside diameter of 20 mm and a wire size of 2.3 mm was selected and measured 20 times using the algorithm in this paper. Additionally, for comparison with the algorithms in this paper, the classical whole-pixel edge detection algorithms, the Sobel and Canny algorithms, were chosen here to also perform 20 measurements for calculation, and the results are shown in Table 2, all in mm.

4.3. Analysis of Results

In terms of outside diameter measurement, it can be seen from Table 2 that the mean square error of the three algorithms is calculated separately using the outside diameter of 20 mm as the true value, and the mean square error using the algorithm of this paper is 0.00480, while for the Sobel algorithm it is 0.02246 and for the Canny algorithm it is 0.00858. The comparison graph of the outside diameter measurement error in Figure 12a shows that the measurement error of the algorithm in this paper can be stable within 0.1 mm, while the measurement errors of both the Sobel and Canny algorithms exceed 0.1 mm. Drawing the comparison graph of the outside diameter measurement values of the three algorithms, as shown in Figure 11a, and observing the dispersion of the data, we can see that the algorithm in this paper also has good stability. Therefore, in terms of outside diameter measurement, the algorithm in this paper has better accuracy and stability than the classical algorithms Sobel and Canny for a whole pixel.

Table 2. Comparison table of measurement results of 20-mm O-ring.

No.	Algorithm of This Paper		Sobel Algorithm		Canny Algorithm	
	Outer Diameter	Wire Diameter	Outer Diameter	Wire Diameter	Outer Diameter	Wire Diameter
1	20.062	2.286	19.847	2.260	19.906	2.267
2	20.063	2.293	19.841	2.261	19.899	2.266
3	20.064	2.285	19.840	2.262	19.899	2.268
4	20.054	2.289	19.837	2.260	19.896	2.268
5	20.065	2.289	19.855	2.265	19.913	2.271
6	20.070	2.287	19.848	2.261	19.907	2.269
7	20.076	2.292	19.860	2.264	19.920	2.269
8	20.067	2.290	19.852	2.264	19.912	2.271
9	20.058	2.291	19.840	2.260	19.899	2.266
10	20.068	2.279	19.856	2.262	19.915	2.269
11	20.068	2.289	19.853	2.262	19.912	2.268
12	20.062	2.286	19.844	2.261	19.899	2.267
13	20.066	2.290	19.847	2.261	19.905	2.267
14	20.077	2.293	19.861	2.264	19.920	2.273
15	20.074	2.290	19.852	2.262	19.907	2.266
16	20.069	2.289	19.847	2.261	19.901	2.268
17	20.066	2.289	19.843	2.261	19.896	2.264
18	20.079	2.289	19.861	2.263	19.917	2.269
19	20.082	2.291	19.863	2.265	19.916	2.269
20	20.085	2.292	19.860	2.265	19.914	2.271
Average value	20.069	2.289	19.850	2.262	19.908	2.268
Mean Square Error	0.00480	0.000132	0.02246	0.00143	0.00858	0.00101
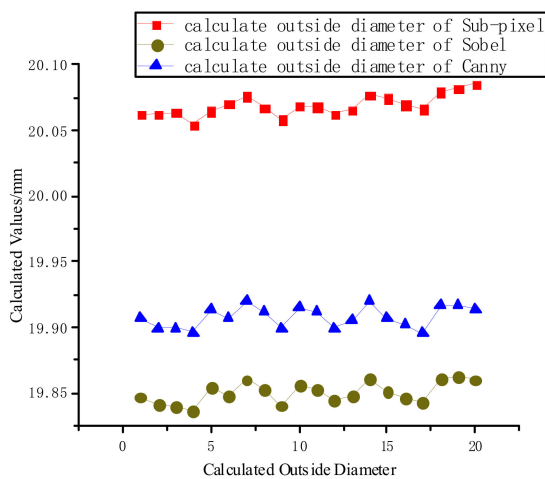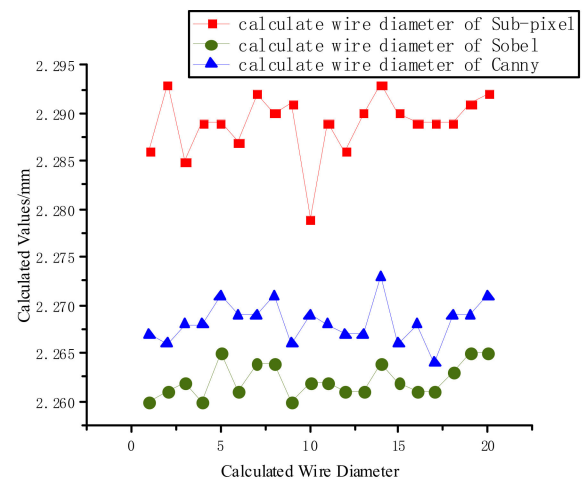
Figure 11. Graph of measurement results. (**a**) Comparison of outside diameter measurement values; (**b**) comparison of wire diameter measurement values.

In terms of wire diameter measurement, it can be seen from Table 2 that the mean square error of the three algorithms was calculated separately using the wire diameter of 2.4 mm as the true value, and the mean square error using the algorithm of this paper was 0.00018, while the Sobel algorithm was 0.00143 and the Canny algorithm was 0.00101. Analyzing the comparison graph of the line diameter measurement error in Figure 12b,

only the measurement error obtained using the algorithm of this paper exceeds −0.015 mm for −0.021 mm for the 10th sample, and the rest of the measurement errors can be stably maintained within −0.015 mm, while the measurement errors of the Sobel and Canny algorithms both exceed −0.02 mm. Drawing a comparison of the wire diameter measurements of the three algorithms, as shown in Figure 11b, and observing the dispersion of the data, we can see that the algorithm in this paper also has good stability. Therefore, the algorithm of this paper has better accuracy and stability than the classical whole-pixel algorithms, the Sobel and Canny algorithms, in terms of line diameter measurements.

(**a**) (**b**)
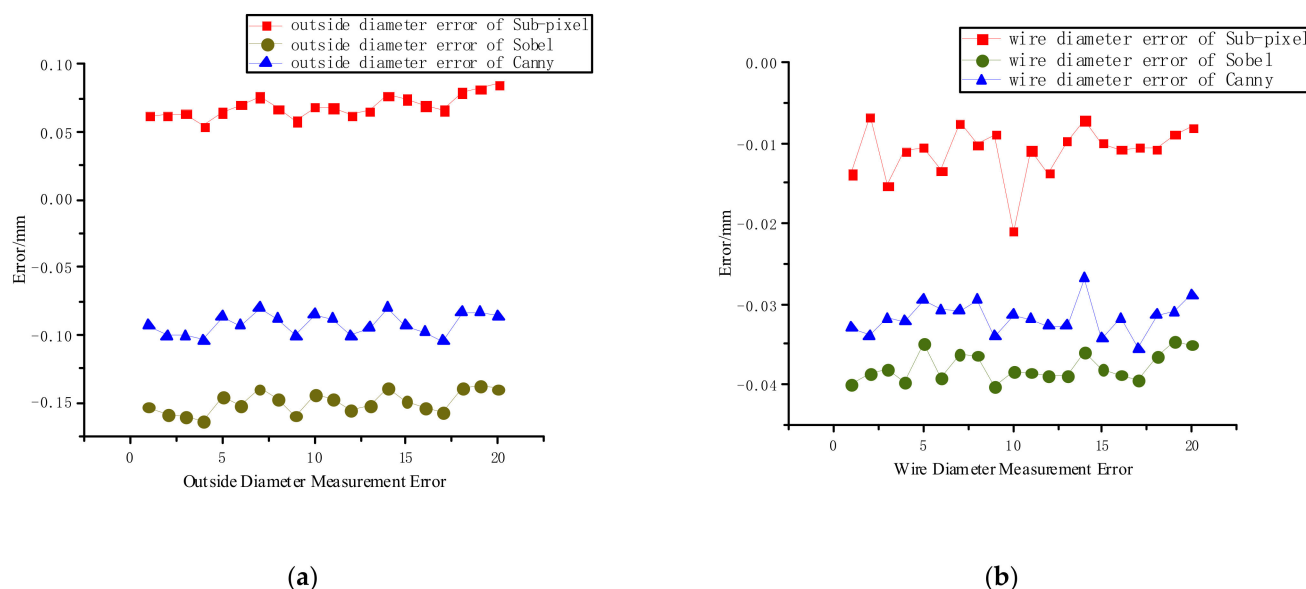
Figure 12. Error analysis chart. (**a**) Comparison chart of outside diameter measurement error; (**b**) comparison chart of wire diameter measurement error.

5. Conclusions

In this paper, we first used the noise-resistant mathematical morphology edge detection algorithm to coarsely locate the edges of an O-ring image, then we used cubic spline interpolation to precisely locate the O-ring edges at the sub-pixel level, and finally performed circle fitting by least squares to calculate the line diameter and diameter of the O-ring. The experimental data show that the algorithm has a mean square error of 0.00480 mm for the outer diameter and 0.00018 mm for the wire diameter. The outer diameter error is kept within ±0.1 mm and the wire diameter error can be kept within ±0.015 mm. By comparing with the whole-pixel edge detection algorithms (Sobel and Canny), the algorithm in this paper has a higher accuracy and good stability in the dimensional measurement of O-rings. This meets the design requirements of the algorithm.

Supplementary Materials: The following are available online at https://www.mdpi.com/article/10.3390/app11083716/s1. The supplementary file contains mainly algorithms written in MATLAB and experimental data. Algorithms: Algorithmic Program, Pictures used in the experiment: Experimental Pictures, Experimental results: Experimental Data.

Author Contributions: Conceptualization, X.Z. and H.H.; methodology, X.Z.; software, X.Z.; validation, X.Z., H.H. and J.Y.; formal analysis, X.Z.; investigation, X.Z.; resources, H.H.; data curation, Y.W.; writing—original draft preparation, X.Z.; writing—review and editing, X.Z.; visualization, X.Z.; supervision, H.H.; project administration, H.H.; funding acquisition, H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This paper was supported by the National Key Research and Development Program of China (2018YFB2003800).

Data Availability Statement: Data is contained within the article or Supplementary Materials.

Conflicts of Interest: The authors declare no conflict of interest.

References

1.　Davies, P.; Le Gac, P.Y.; Ciausu, V.; Gallois, H. Testing of nitrile rubber joints for a deep submergence vehicle. *Polym. Test.* **2020**, *90*, 106630. [CrossRef]
2.　Koemmling, A.; Jaunich, M.; Goral, M.; Wolff, D. Insights for lifetime predictions of O-ring seals from five-year long-term aging tests. *Polym. Degrad. Stab.* **2020**, *179*, 109278. [CrossRef]
3.　Mo, Y.-M.; Gong, Y.; Yang, Z.-G. Failure analysis on the O-ring of radial thrust bearing room of main pump in a nuclear power plant. *Eng. Fail. Anal.* **2020**, *115*, 104673. [CrossRef]
4.　Oskoei, M.A.; Hu, H. *A Survey on Edge Detection Methods*; University of Essex: Colchester, UK, 2010; p. 33.
5.　Zhang, H.; Meng, C.; Bai, X.; Li, Z. Rock-ring detection accuracy improvement in infrared satellite image with sub-pixel edge detection. *IET Image Process.* **2019**, *13*, 729–735. [CrossRef]
6.　Hu, H.; Zhang, B.; Xu, D.; Xia, G. Battery Surface and Edge Defect Inspection Based on Sub-Regional Gaussian and Moving Average Filter. *Appl. Sci.* **2019**, *9*, 3418. [CrossRef]
7.　Peng, G.; Zhang, Z.; Li, W. Computer vision algorithm for measurement and inspection of O-rings. *Measurement* **2016**, *94*, 828–836. [CrossRef]
8.　Liu, J.; Du, H.; Du, Q. Improvement of sub-pixel morphological anti-aliasing algorithm. *J. Comput. Appl.* **2017**, *37*, 2871–2874.
9.　Ezekiel, S.; Lang, M. *Edge Detection Using Minmax Measures*; ACTA Press: Calgary, AB, Canada, 2003.
10.　Zhang, H.; Jin, X. A Method for New Energy Electric Vehicle Charging Hole Detection and Location Based on Machine Vision. In Proceedings of the 2016 5th International Conference on Environment, Materials, Chemistry And Power Electronics, Zhengzhou, China, 11–12 April 2016; pp. 392–399.
11.　Jishen, S.; Edward, J.D. The analysis of morphological filters with multiple structuring elements. *Comput. Vis. Graph. Image Process.* **1990**, *50*, 308–328.
12.　Maragos, P. Differential morphology and image processing. *IEEE Trans. Image Process.* **1996**, *5*, 922–937. [CrossRef]
13.　Zheng, L.; Lawlor, B.; Katko, B.J.; McGuire, C.; Zanteson, J.; Eliasson, V. Image Processing and Edge Detection Techniques to Quantify Shock Wave Dynamics Experiments. *Exp. Tech.* **2020**. [CrossRef]
14.　Qiaoming, W.; Lei, W.; Hui, L.; Chun, L. Correction of Systematic Error in Sub-pixel Edge Location. In Proceedings of the 2020 IEEE 14th International Conference on Anti-Counterfeiting, Security, and Identification (ASID), Xiamen, China, 30 October–1 November 2020.
15.　Huang, C.; Jin, W.; Xu, Q.; Liu, Z.; Xu, Z. Sub-Pixel Edge Detection Algorithm Based on Canny-Zernike Moment Method. *J. Circuits Syst. Comput.* **2020**, *29*, 2050238. [CrossRef]
16.　Wang, G.; Xiao, L.; Jiang, Z.; Song, Y.; He, A. Sub-pixel multifractal method for image processing application. *Acta Opt. Sin.* **2006**, *26*, 1345–1349.
17.　Touqir, I.; Saleem, M.; Siddiqui, A.M. Novel wavelet based edge detection. *Wseas Trans. Signal Process.* **2006**, *2*, 1248–1253.
18.　Wang, G. A sub-pixel circle detection algorithm combined with improved RHT and fitting. *Multimed. Tools Appl.* **2020**, *79*, 29825–29843. [CrossRef]
19.　Feng, Y.; Yang, T.; Niu, Y. Subpixel Computer Vision Detection Based on Wavelet Transform. *IEEE Access* **2020**, *8*, 88273–88281. [CrossRef]
20.　Kanchanatripop, P.; Zhang, D. Adaptive Image Edge Extraction Based on Discrete Algorithm and Classical Canny Operator. *Symmetry* **2020**, *12*, 1749. [CrossRef]
21.　Chen, Y.-Q.; Zhu, X.-J.; Wang, Y.-H.; Guo, J.-J.; Chen, L.-Q. Subpixel edge detection of image for industrial thin sheet parts based on fuzzy clustering. *Modul. Mach. Tool Autom. Manuf. Tech.* **2014**, *10*, 66–68.
22.　Li, B.; Hu, H.-B.; Zhuang, T.-G. Contours embellishment using adaptive cubic B-spline in image segmentation. *J. Infrared Millim. Waves* **2001**, *20*, 401–405. [CrossRef]
23.　Xie, X.; Ge, S.; Xie, M.; Hu, F.; Jiang, N. An improved industrial sub-pixel edge detection algorithm based on coarse and precise location. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 2061–2070. [CrossRef]
24.　Di, Y.; Li, M.-Y.; Qiao, T.; Lu, N. Edge detection and mathematic fitting for corneal surface with Matlab software. *Int. J. Ophthalmol.* **2017**, *10*, 336–342.
25.　Gu, Y.; Lu, Y.; Yang, H.; Huo, J. Concentric Circle Detection Method Based on Minimum Enveloping Circle and Ellipse Fitting. In Proceedings of the 2019 IEEE 10th International Conference on Software Engineering And Service Science, Beijing, China, 18–20 October 2009; pp. 523–527.
26.　Tao, W.; Zhong, H.; Chen, X.; Selami, Y.; Zhao, H. A new fitting method for measurement of the curvature radius of a short arc with high precision. *Meas. Sci. Technol.* **2018**, *29*, 075014. [CrossRef]
27.　Hagara, M.; Kulla, P. Edge Detection with Sub-pixel Accuracy Based on Approximation of Edge with Erf Function. *Radioengineering* **2011**, *20*, 516–524.
28.　Chang, W.-T.; Su, C.-H.; Guo, D.-X.; Tang, G.-R.; Shiou, F.-J. Automated optical inspection for the runout tolerance of circular saw blades. *Int. J. Adv. Manuf. Technol.* **2013**, *66*, 565–582. [CrossRef]